<u>DESIGN OF A WEARABLE KNEE ANGLE SENSOR</u>

by

Venkata Naga Sai Chaitanya Neelamraju
A Thesis
Submitted to the
Graduate Faculty
of
George Mason University
in Partial Fulfillment of
The Requirements for the Degree
of
Master of Science
Electrical and Computer Engineering

Committee:

___________________________________ Dr. Lance Sherry, Thesis Director

___________________________________ Dr. Kris Gaj, Committee Member

___________________________________ Dr. Jens-Peter Kaps, Committee Member

___________________________________ Dr. Monson Hayes, Department Chair

___________________________________ Dr. Kenneth S. Ball, Dean, Volgenau School of Engineering

Date:_______________________________ Summer Semester 2018
George Mason University
Fairfax, VA

Design of a Wearable Knee Angle Sensor

A Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science at George Mason University

by

Venkata Naga Sai Chaitanya Neelamraju
Bachelor of Technology
Jawaharlal Nehru Technological University, 2016

Director: Lance Sherry, Associate Professor
Center for Air Transportation and Systems Research

Summer Semester 2018
George Mason University
Fairfax, VA

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to my advisor Dr. Lance Sherry and co-advisor Dr. Kris Gaj for their patience, motivation and guidance through the research and thesis documentation. I would also take this opportunity to thank people of Center for Air Transportation and Systems Research (CATSR) group for their immense help and sponsorship. Finally, I want to thank my parents and friends for their moral support and encouragement without which the thesis would not have been possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

DESIGN OF A WEARABLE KNEE ANGLE SENSOR

Venkata Naga Sai Chaitanya Neelamraju, M.S.

George Mason University, 2018

Thesis Director: Dr. Lance Sherry

This thesis describes the design, construction, and testing of a Wearable Knee Angle Sensor (WKAS) device that measures knee flexion angle designed for operation in various competitive sports. The requirement is to design a rugged wearable (i.e., low weight, low profile) sensor that can be worn in game situations in contact sports. The sensor can be used for performance assessment (e.g., leg motion while sprinting) and for injury prevention (e.g., leg motion leading to specific injuries). An analysis of alternate sensor options was conducted leading to selection of a Flex-Sensor. The WKAS was configured with a Flex Sensor mounted in a sensor-sleeve behind the knee that is attached to a knee sleeve. An Arduino was used to collect and process the data and stored it on an SD card. The data from the SD card can then be downloaded and analyzed in visualization software (simulation) developed in MATLAB. The accuracy of the sensor was assessed, and future improvements are proposed.

CHAPTER 1: INTRODUCTION

Competitive sports are a multi-billion-dollar industry. Athletes are required to perform at their peak levels while avoiding injury. Recent development in sensor technology use accelerometers and position sensors (e.g., GPS) to track player position on the court and field (e.g., Leser, Baca, Orgris_2011), players heart rate can be tracked (e.g., Castellano, Casamichana, 2010).

One area of performance that is not measured is leg motion defined by hip, knee, and ankle angles. Sprinting performance is directly determined by the timing and sequence of knee angle (e.g., Brown et.al, 2004; Majumdar & Robergs, 2011). Leg injuries are also a function of the timing and sequence of knee angle (e.g., Boden *et.al*., 2009). Over the past decade, there has been much work focused on developing wearable sensors to facilitate real-time, continuous human movement analysis in free-living conditions. These include designs using inertial movement unit (IMU) [Kun_2011], [Cooper_2009], ultrasonic sensors [Qi_2014, 2015], rigid electrogoniometers [Toffola_2012], [Riskowski_2010], soft sensors like flex [Wang_2011], fiber-optical [Stupar_2012], [Silva_2013], e-textile [Enokibori_2014], [Gibbs_2005], and liquid metal sensors [Menguc_2014], [Michaud_2015]. However, very few systems are (i) power efficient for sensing and processing of the acquired data, (ii) cost effective, (iii) safe and easy to use, (iv) providing flexible form factor to comply with highly dynamic, heterogeneous human body shapes, and (v) supporting the maximum +/- 5 degrees of estimation accuracy that was suggested by the American Medical Association for movement analysis in a clinical context [Qi_2014], [Zheng_2005].

The objective of this project is to design and test a rugged wearable (i.e., low weight, low profile) sensor that can be worn in game situations in contact sports. The sensor can be used for performance assessment (e.g., leg motion while sprinting) and for injury prevention (e.g., leg motion leading to specific injuries).

The reminder of thesis is organized as follows: Literature Review of WKAS is presented in chapter two followed by design of WKAS in chapter three. Results are in chapter four, Visualization tool is described in chapter five and finally Conclusion and Future work are provided in chapter six.

CHAPTER 2: LITERATURE REVIEW OF WEARABLE KNEE ANGLE SENSORS

Researchers have designed and evaluated alternate wearable knee sensors to enable joint angle estimation and motion capture capabilities in ambulatory settings. In this chapter, related work on body joint angle measurement using wearable sensors is reviewed. There are three functions in any knee joint angle measurement system: (1) Mounting of sensor on the leg, (2) sensors for knee angle measurement, and (3) collection and storage of the data Table 1.

Table 1: Summary of Sensor Systems

Wearable Knee Angle Sensor (WKAS)System	**Sensor used for Knee Angle Measurement**	**Sensor Mount**	**Processing and Data Storage**	**References**
Flex-sensor system	Flex-sensor	Knee sleeve for mounting two flex-sensor on back of knee.	Microcontroller and extended Kalman filter for processing, storage was not mentioned.	[Khayani_2011]
Flex-sensor system	Flex-sensor	Flex-sensor is inserted into pocket sleeve on elbow, wrist, fingers	Multivariant linear regression model for processing and data acquisition MP150 for storage.	[Wang_2011]
Potentiometer system	Potentiometer	Retractable string is used	Embedded system and	[Lee_2016]

		to mount it on leg.	small signal processing method for processing and PC, USB for storage	
Potentiometer system	Potentiometer	Knee brace used for mounting potentiometer.	Microcontroller for processing, cloud for storing.	[IDG_2016]
IMU system	IMU	Mounted on thigh and shank using Velcro	Kalman filter for processing.	[Kun_2011], [Cooper_2009], [Menguc_2014]
Ultrasonic system	Ultrasonic transmitter and receiver	Mounted on trunk and receivers on hip, knee, ankle, toe.	Microcontroller and Kalman filter for processing, PC for storage.	[Qi_2014], [Qi_2015]
Attachable Clothing system	Conductive fiber	Attached to a garment	Algorithm used for processing.	[Gibbs_2005]
Attachable Clothing system	Conductive polymer	Attached to side of knee on a garment.	Wheatstone bridge circuit for processing and data acquisition board for output	[Bergmann_2013]

2.1 Inertial Measurement Unit

2.1.1 Sensor

The IMU sensor consists of triple axis accelerometer, gyroscope, and magnetometer. The most prevalent form factor of wearable sensors is the IMU (e.g., Kun_2011, Cooper_2009).

2.1.2 Sensor Mounting

IMU's were mounted on thigh and on shank to measure the knee angles using Velcro.

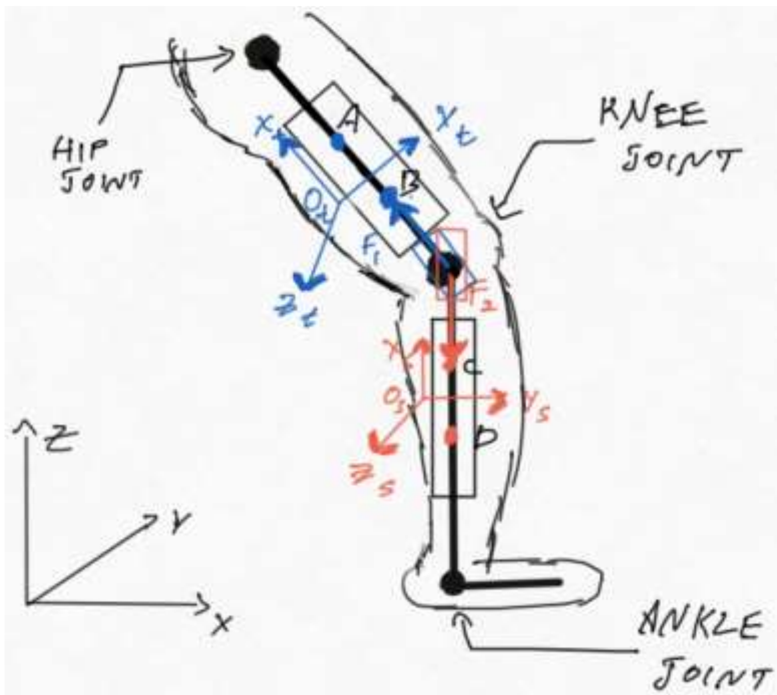

Figure 1: Installation of IMU sensor on limbs [Kun_2011].

2.1.3 Processing and Storage

This approach requires extensive signal processing such as Kalman filtering and consistent calibration of the gyroscope for the integration drift, which often requires additional sensing units [Menguc_2014]. The use of gyroscopes, additional sensing units for the calibration, and complex real time signal processing requires large power consumption. The difference of thigh and shank angular rate yields knee angle.

2.1.4 Performance

The method was validated experimentally by calculating knee angle from measurements taken from two IMUs placed on adjacent body segments. In contrast to many previous studies which have validated their approach during relatively slow activities or over short durations, the performance of the algorithm was evaluated during both walking and running over 5 minutes periods. Seven healthy subjects were tested at various speeds from 1 to 5 miles/hour. Errors were estimated by comparing the results against data obtained simultaneously from a 10 cameras motion tracking system (Qualysis). The average measurement error ranged from 0.7 degrees for slow walking (1 mph) to 3.4 degrees for running (5mph). The joint constraint used in the IMU analysis was derived from the Qualysis data [Cooper_2009].

2.2 Ultrasonic

2.2.1 Sensor

Qi *et al.* in 2015 developed a measurement system which is a combination of an ultrasound transmitter (mobile) and multiple receivers (anchors) whose positions are known that are positioned at the extremities of the joint can compute the distance between both sensors and convert the measurements into joint angles based on biomechanical modeling.

2.2.2 Sensor Mounting

Qi *et al.* in 2015 developed a measurement system based on a wearable wireless ultrasonic sensor network to track the lower extremity joint and trunk kinematics during a squat exercise with only one ultrasonic sensor attached to the trunk.

2.2.3 Processing and Storage

The extended Kalman filter is applied to estimate the displacements in the vertical and horizontal direction of the ultrasonic sensor, and then, the recorded displacements together with known joint constraints are used to estimate the joint angles of the trunk using the damped least-squares-based technique for the singularity avoidance problem of redundant systems. The data was recorded and stored in PC.

2.2.4 Performance

The performance of the proposed ultrasonic measurement system was validated against a camera-based tracking system on eight healthy subjects performing a planar squat exercise. Joint angles estimated from the ultrasonic system showed a root mean square error (RMSE) of $2.85^\circ \pm 0.57^\circ$ with the reference system. These results show that the proposed ultrasonic measurement system is useful for applications, such as rehabilitation and sports [Qi_2014, 2015].

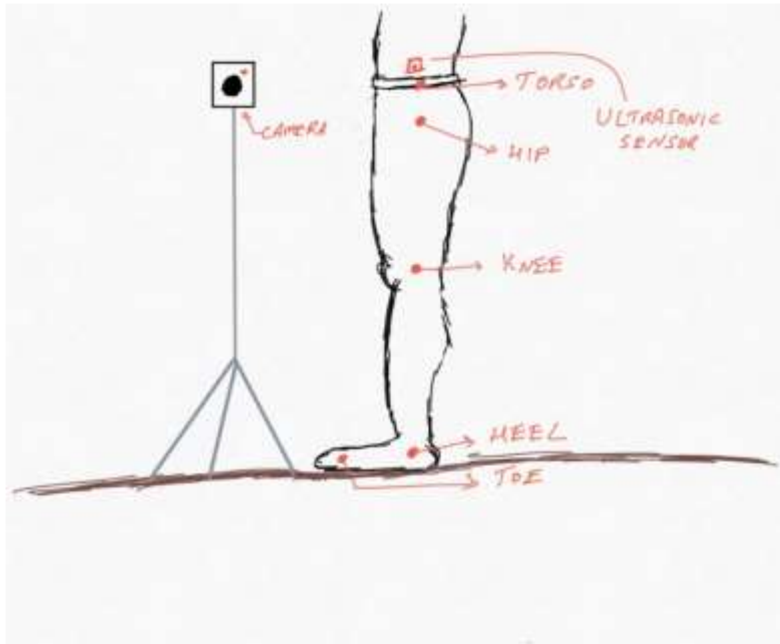

Figure 2: Test set-up for testing wireless unit with embedded ultrasonic sensor[Qi_2015].

This approach provides highly accurate estimation of the joint angles but requires continuous transmission and reception of wireless signals, which again requires large power consumption and line-of -sight is also a big problem for the ultrasonic sensors.

2.3 Attachable Clothing

2.3.1 Sensor

Gibbs and Harry developed a system for continuous day-to-day monitoring of body joint movement using conductive fiber sewed on a wearable and comfortable garment [Qi_2014]. As shown in Figure 3, there is one conductive fiber which is employed as a sensor. The goal in designing these wearable sensors is to make a tool that is eventually self-registering for subsequent uses after the first one-time calibration experiments.

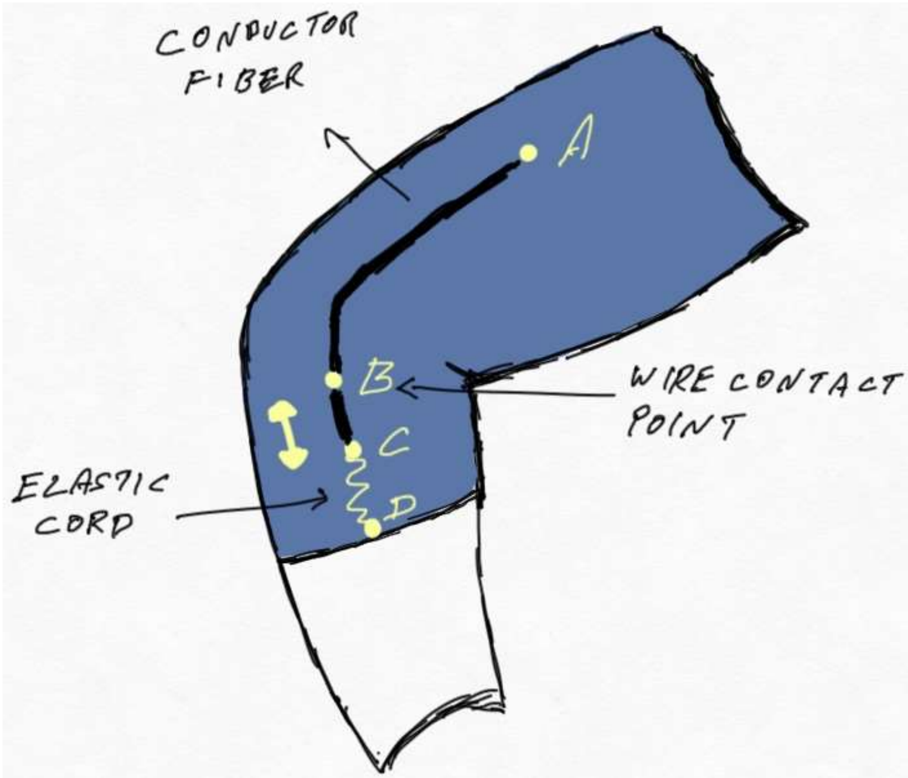

Figure 3: Sensor design Schematic [Gibbs_2005].

This means that no extra equipment is needed to register the sensors for every use. Furthermore, it is critical that any procedures that are needed for self-registration are uncomplicated, and capable to be performed by the patient without supervision [Gibbs_2005].

Bergmann *et al.* in [Bergmann_2013] also developed a system similar to attachable clothing sensor but with a different kind of material called conducting polymers. Figure 4 shows the conductive polymer which is integrated with garment. This type of sensor was made up of 20% conductive carbon nanotubes material and 80% non-conductive polyurethane. This ratio was selected as it showed highly electrical properties.

Figure 4: Polymer sensor fitted with connectors [Bergmann_2013].

2.3.2 Sensor Mounting

Gibbs *et al.* in 2005 permanently attached one end of fiber to the nonconductive fiber (point A). Along the conductive fiber, there is a wire contact point at B that is sewed into the fabric. The other end of the conductive fiber, point C, is kept in tension by a coupled elastic cord. It is attached to the isolated side of the joint, point D. The length of the elastic cord is changed if the joint moves. Since the length of conductive thread between points A and B changes as the joint rotates, the resistance, which is linearly related to length, is measured constantly across these two points A and B.

In [Bergmann_2013] the sensor was mounted on the side of the knee because when compared to frontal part, it showed less displacement artifact during movement. It was also deemed the best location to minimize contact between the sensor and other objects. This sensor was sewn into garment as shown in the Figure 5.

Figure 5: Polymer sensor attached to garment [Bergmann_2013].

2.3.3 Processing and Storage

A multi-thread sensor array design was presented by [Gibbs_2005]. An array of M sensors covered a single-axis joint as shown in Figure 6, each sensor thread is separated from the adjacent sensor thread by a known, constant distance, d. This multi-thread sensor array was employed to approximate a single-axis joint angle, θj, lower body joint angles. To develop a registration procedure, first calibration of each sensor thread individually was performed. Creating a denser sensor array in this way leads to more accurate estimates of sensor sensitivities, which in turn leads to more accurate estimates of θj. The registration algorithm takes place in real time as the sensor is in use. The only task that a patient needs to do for the first using of these sensors is to first "zero" the sensor output with the joint entirely extended in the 0° position, and then without restraint move the joint to obtain non-zero data. This non-zero data then allows the self-registration to take place.

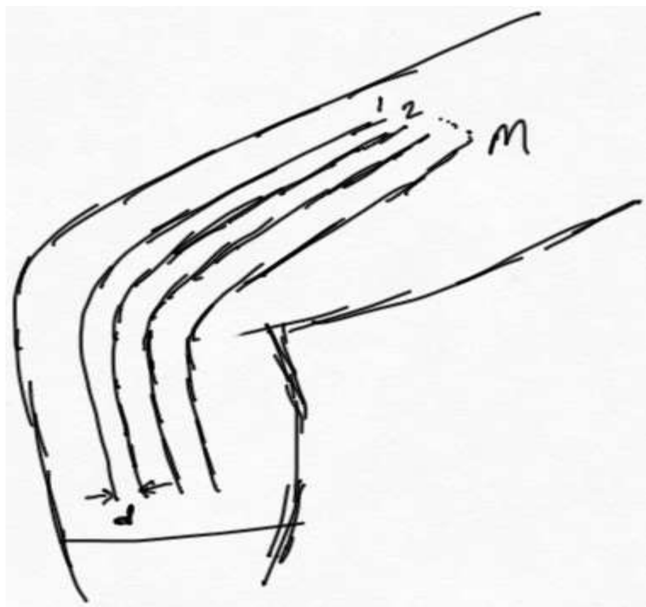

Figure 6: Array of sensors over knee joint in an equal distance [Gibbs_2005].

In [Bergmann_2013], the Wheatstone bridge circuit was used for processing along with data acquisition board (USB 6211 DAQ, National Instruments, USA) to collect the output. Simulink model was used to gather the data with sampling frequency of 50 Hz.

2.3.4 Performance

The pants sensing garment was first employed to calculate single-axis knee angle measurements. For the single-axis experiments, a rotary potentiometer firmly attached to the leg was utilized as a goniometer, and this was the standard for which to compare joint angles. In every trial, the potentiometer was "zeroed" with the leg in the complete extension position. The average root mean square (RMS) error between the pants sensor approximation and the potentiometer using the linear predictor was 5.4°.

Despite the success of this method in measuring the joint angle with very flexible form factors since they can bend and stretch in different directions, the authors [Gibbs_2005] mentioned several uncertainties in measuring the resistance across the conductive fiber which resulted in incorrect sensor output. For example, joint movement produces small changes in the fiber tension resulting in a change of fiber resistance which leads to incorrect sensor output, they produce a single dimensional output which is often inaccurate.

The performance study of polymer sensor was conducted using 10 volunteers to determine the relationship between the ACS system and a gold standard apparatus. The comparison yielded an average root mean square error of =1°, a mean absolute error of =3°, and coefficient of determination above (R2) 0.99 between the two systems. Despite having few results there were many obstructions faced like deformation of ACS during flexion and extension of knee. Moreover, the construction cost of polymer ACS is expensive because of its material.

2.4 Potentiometer

2.4.1 Sensor

The system in [IDG_2016] was developed using potentiometer to measure the knee flexion angle. Another system in [Lee_2016] also used potentiometer.

2.4.2 Sensor Mounting

The potentiometer in [IDG_2016] was mounted over the pivot point of the knee on the brace to measure the knee bend. The construction of the potentiometer pivot arm was through both bespoke 3D-printed parts and, of all things, Meccano. A lot of Velcro and some glue was used as a bit of weather-proofing.

Figure 7: Potentiometer setup [IDG_2016].

Lee *et al.* (2016) used retractable reel, a string to mount the potentiometer where the rotations of the reel are measured by potentiometer. Figure 8 shows how it is arranged.

Figure 8: Wearable sensor which utilizes retractable string [Lee_2016].

2.4.3 Processing and Storage

Microcontroller receives the input from the potentiometer, stores it on an SD card and sends data to the cloud via Wi-Fi in [IDG_2016].

The sensor prototype illustrated in Fig. 8 used an embedded system to collect the sensor data at approximately 50 Hz and analog-to-digital converter was used to convert the analog input voltage to digitized output. A small signal processing method was used to convert sensor data in terms of length to knee angle. USB was used to transmit the output of the embedded system to a personal computer for data storage.

2.4.4 Performance

IDG_2016 has good accuracy, but the main drawback is the set-up which is huge in size to carry it on the field for any player in any sport and delay in sending the data for storage.

The system in [Lee_2016] tested 9 subjects and an average root mean square error (RMSE) of 4.51° was achieved with the maximum of 6.34° and minimum of 3.19°. Despite the results, there was delay in response of potentiometer in this set up resulting different waveforms for extension and flexion of knee. When the leg was changing the phase from flexion to extension, the shaft of the potentiometer responded late causing different paths than usual.

2.5 Flex-sensor

2.5.1 Sensor

Wang *et al*. used two unidirectional flexible sensors placed back-to-back, and [Khayani_2011] also used two flex-sensors placed side-by-side.

2.5.2 Sensor Mounting

Wang *et al.* (2011) mounted these on elbow, wrist and on fingers. Two flex sensors placed back-to-back, inserted into expandable polyethylene terephthalate (PET) sleeving for insulation. Figure 9 shows in detail how it is done.

Figure 9: Flex-sensor [Wang_2011].

Khayani *et al.* (2011) used supportive cloth called knee sleeve to mount the two flex-sensors side-by-side. Figure 10 describes the placement of sensors along with the sleeve used.

Figure 10: Top and bottom view of supportive cloth [Khayani_2011].

2.5.3 Processing and Storage

The measurement of flexion/extension angles of various hinge joints was done by reliable electrogoniometer as a calibration in (Wang_2011). A multivariate linear regression model was used to combine measurements from the two sensors. The data acquisition device MP150 (Biopac Systems Inc.) was connected to collect the output with input impedance ≥1 M. The data were sampled at 200 Hz and acquired using AcqKnowledge 3.8.2 software (Biopac Systems Inc.).

Khayani *et al.* (2011) used microcontroller and extended Kalman filter (EKF) for processing the data but storage details were not mentioned.

2.5.4 Performance

The performance of the goniometer has been tested on a population of 21 healthy subjects performing flexion/extension of index finger, wrist, and elbow. The proposed device achieves the quality of joint angle measurements comparable to that of commercial electrogoniometers, while having a significantly higher durability-to-cost ratio (Wang_2011).

In (Khayani_2011), based on the experiments the correlation between the fused measurement and the output of the goniometer is 0.98, the range of error is 0.076° to 11.32°, and the average error is 6.92°. The average error in measuring the joint angle using Sensor 1 and Sensor 2, are 10.99° and 9.36°, respectively. However, when we fuse the two sensors using the EKF, the average error rate reduces to 6.92°. Due to the space limitation underneath the knee, mounting more than two sensors is not practical and comfortable.

Mounting the flex sensor on the top of the knee was also studied, but it prohibited the knee to be flexed/bent freely and hence was not practical.

CHAPTER 3: DESIGN OF WEARABLE KNEE ANGLE SENSOR (WKAS)

This chapter describes the design of a wearable knee angle sensors. After going through literature review, the only two sensors which were suitable for the desired outcome were potentiometer and flex-sensor. Section 3.1 describes the requirements. Section 3.2 describes a design using a potentiometer. Section 3.3 describes the design using a flex sensor.

3.1 Requirements for Wearable Knee Angle Sensor (WKAS)

The wearable knee angle sensors (WKAS) shall satisfy the following requirements:

1. The WKAS shall sense the knee angle at a minimum rate of 10Hz (i.e., 10 times per second).
2. The WKAS shall sense the angle with an accuracy of +/- 5 degrees
3. The WKAS shall sense the angular rate with an accuracy of +/- 1 degree per second
4. The WKAS shall sense and store the data on the athlete for an athletic event that lasts at least 60 minutes
5. The WKAS shall provide the means to download the data after an athletic event
6. The WKAS shall have a form factor that will not impinge on athlete’s motion
7. The WKAS shall not weigh more than 20 grams
8. The WKAS shall not injure the athlete or the opponent in the event of player-to-player collision or athlete to ground collision.

9. The WKAS shall remain in the correct location without athlete adjustment or curtailing athletic movement.

3.2 Construction of WKAS using Potentiometer

This design was called the “Potentiometer Design” because it measures the knee flexion angle using a variable resistor, also known as a potentiometer (pot). Most small potentiometers have a base with a rotating knob on top. It changes the resistance, increasing or decreasing the voltage dropped across the resistor when the knob is turned clockwise (CW) or counterclockwise (CCW).

Table 2: Requirements Mapping of WKAS-Potentiometer

Requirements	**Design**		
	Processing	**Storage**	**Mounting**
1. 10Hz	Microcontroller	-	-
2. +/-5 degrees	-	-	Knee brace
3. +/-1 deg/sec	-	-	Knee brace
4. Storage	-	SD-card	It won’t work
5. Download	-	SD-card	-
6. Form Factor	-	-	It won’t work
7. Weight< 20	-	-	Knee brace

8. No injury	-	-	It won’t work
9. Correct place	-	-	It won’t work

Potentiometer was inserted inside a knee brace and this knee brace was designed using 3D-printer. Downloaded a ready to use design of knee brace and made few changes in sizes (https://www.thingiverse.com/thing:1509833). Then by using software called CURA, knee brace was built with the help of 3D-printer. Printed out all the parts individually and then joined together to place it on knee with a Velcro belt. The following images are of 3D-printed knee brace.

Figure 11: Individual parts of 3D-printed knee brace.

Figure 11 shows all the individual parts of 3D-printed knee brace which can be joined together. Potentiometers were inserted into the hinge of knee brace. The image of potentiometer inserted into hinge of knee brace can be seen in Figure 12 followed by the image of complete assembled part of knee brace in Figure 13.

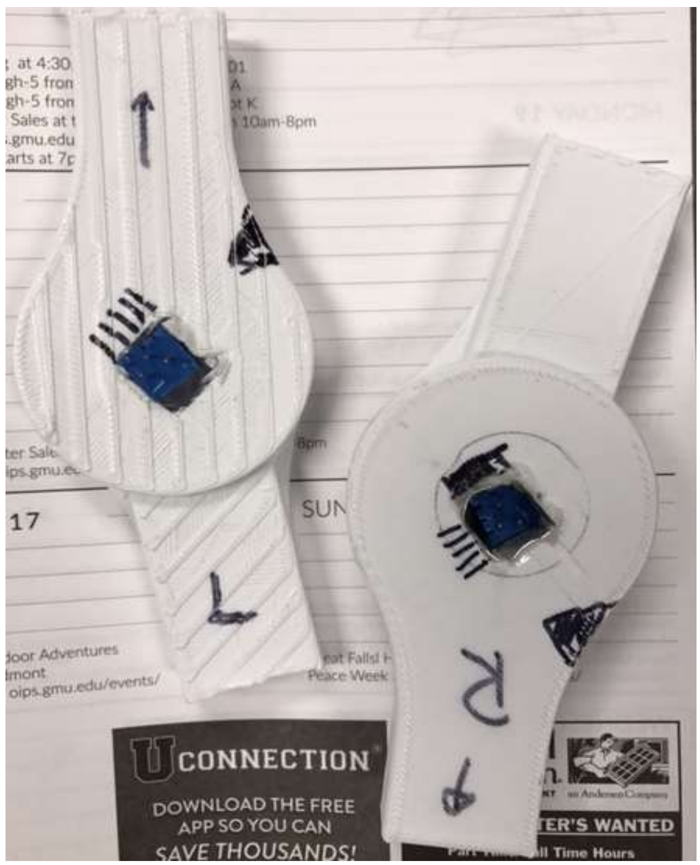

Figure 12: Potentiometer inserted inside a hinge of knee brace.

After assembling all the parts of 3D-printed knee brace and inserting the potentiometer into the hinges, it was attached to knee with a Velcro strap for testing. The main drawback of this set up was though it was light weight, but it was uncomfortable to use due to its size. Thus mounting option was abandoned.

Figure 13: Assembled 3D-printed knee brace.

3.3 Construction of WKAS using Flex-Sensor

In this section another approach of measuring knee joint angle is described. The flex-sensor has a positive and negative terminal and it is of very thin film. When it bends towards one direction, the value changes from 0 to 1023. These numbers are converted into angles by calibration.

Table 3: Requirements Mapping of WKAS-Flex-Sensor

Requirements	**Design**		
	Processing	**Storage**	**Mounting**
1. 10Hz	Microcontroller	-	-
2. +/-5 degrees	-	-	Knee sleeve
3. +/-1 deg/sec	-	-	Knee sleeve
4. Storage	-	SD-card	Knee sleeve
5. Download	-	SD-card	-
6. Form Factor	-	-	Knee sleeve
7. Weight<20	-	-	Knee sleeve
8. No injury	-	-	Knee sleeve
9. Correct place	-	-	Knee sleeve

Flex-sensor along with knee sleeve is used. A small pocket is stitched back side of the sleeve and a small 3D-printed cap to hold the flex-sensor in the pocket to avoid pressure on flex-sensor.

Figure 14: Knee sleeve with pockets.

These flex-sensors are attached to a microcontroller along with SD-card and battery. All the components attached to each other according to the circuit diagram which can be seen in Figure 15.

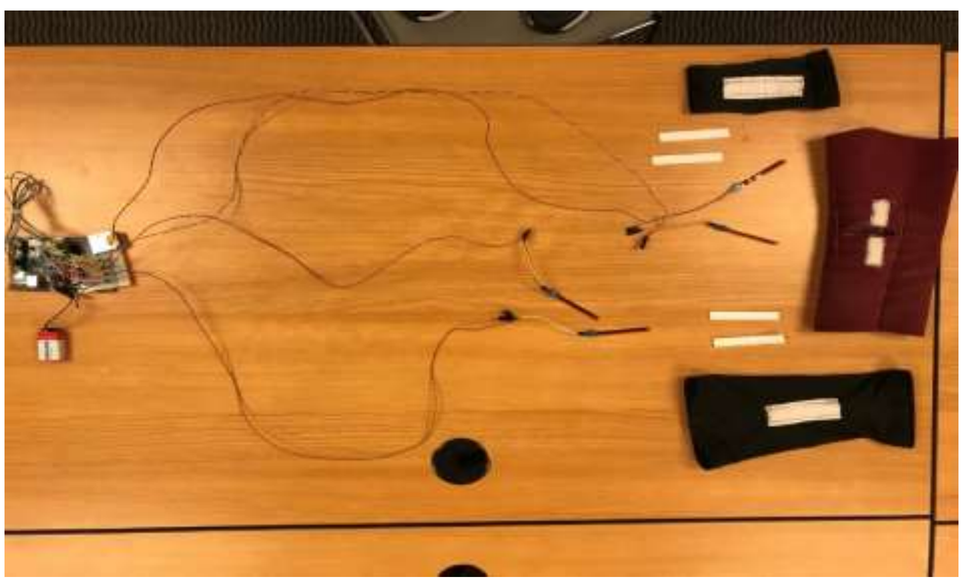

Figure 15: Flex-sensors attached to microcontroller along with knee and sensor sleeves.

The detailed block diagram of the microcontroller, flex-sensor and SD-card is shown in figure below.

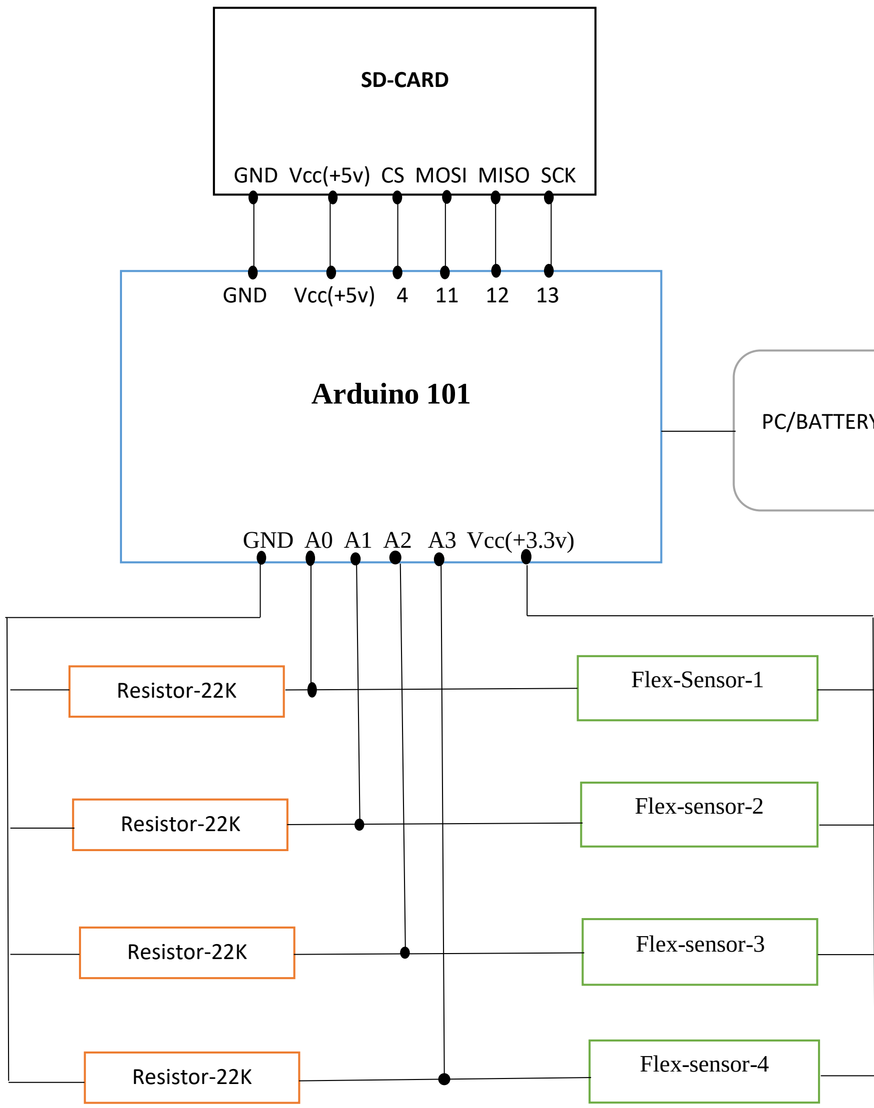

Figure 16: Circuit diagram of flex-sensor with Arduino and SD-card

In the above figure, the complete circuit diagram of flex-sensor attached to Arduino 101 microcontroller board along with SD-card and resistors of 22k ohms can be seen. The negative terminal of the flex-sensors is attached to one end of resistors of 22k ohms and positive terminal is attached to power of +3.3v pin. Then the connection between resistors of 22k ohms and flex-sensors is connected to A0, A1, A2, A3 analog input pins of the microcontroller respectively as shown in the Figure 12. The other end of resistors is connected to ground. The power supply to the microcontroller is given by 9V battery or by PC.

The microcontroller processes the data and is stored in SD-card. This SD-card uses +5v power supply to operate. Ground and power supply pins of SD-card are connected to ground and power supply of microcontroller. SD-card has other four pins namely Chip Select (CS), Master Out Slave Input (MOSI), Master Input Slave Output (MISO), Master Serial Clock (SCK) to be connected to microcontroller.

Chip Select (CS) pin is connected to any digital pin from 4 to 9 on microcontroller to write the data. The MOSI pin is connected to pin 11 on microcontroller which signifies master line sending data to peripherals and MISO pin is connected to pin 12 on microcontroller which signifies slave line sending data to master. SCK is connected to pin 13 where SCK is a serial clock which generates clock pulses to synchronize data transmission generated by master. The microcontroller used is 32-bit with 32MHZ clock speed along with 14 digital I/O pins, 6 analog input pins with input voltage of 7-12v and operating voltage of 3.3v (5v tolerant to I/O).

Thus, this design is easy, light weight, low-profile, feels comfortable wearing during game, inexpensive and accurate. The testing results are mentioned in the next chapter.

CHAPTER 4: RESULTS

After evaluating all the sensors and matching the requirements that were mentioned in section 3.1, flex-sensor was tested for wearable knee angle sensor. The following are all the results which are acquired during experimentation in different scenarios like calibration test, lab test and field test.

4.1 Calibration test

The flex-sensor is initially calibrated using a protractor. During the calibration three angles are observed 180, 145 and 90 degrees. It is tested for 10 times and the average values are taken into consideration and got the equation for slope as shown in Figure 17. The resistance in the sensor changes from 0 to 1023 and this is reflected into angles using equation.

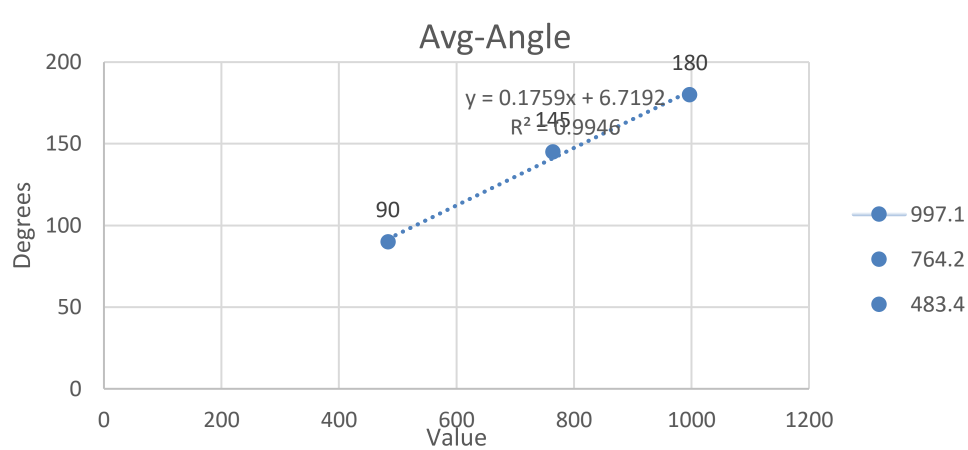

Figure 17: Graph for flex-sensor calibration.

4.2 Lab Test

4.2.1 Sitting and stretching

After the calibration is done, the flex-sensor was tested in lab for a small exercise of sitting in a chair and stretching out the leg and bring it back to normal position. Flex-sensor was placed on back of the knee inserted in knee sleeve pocket.

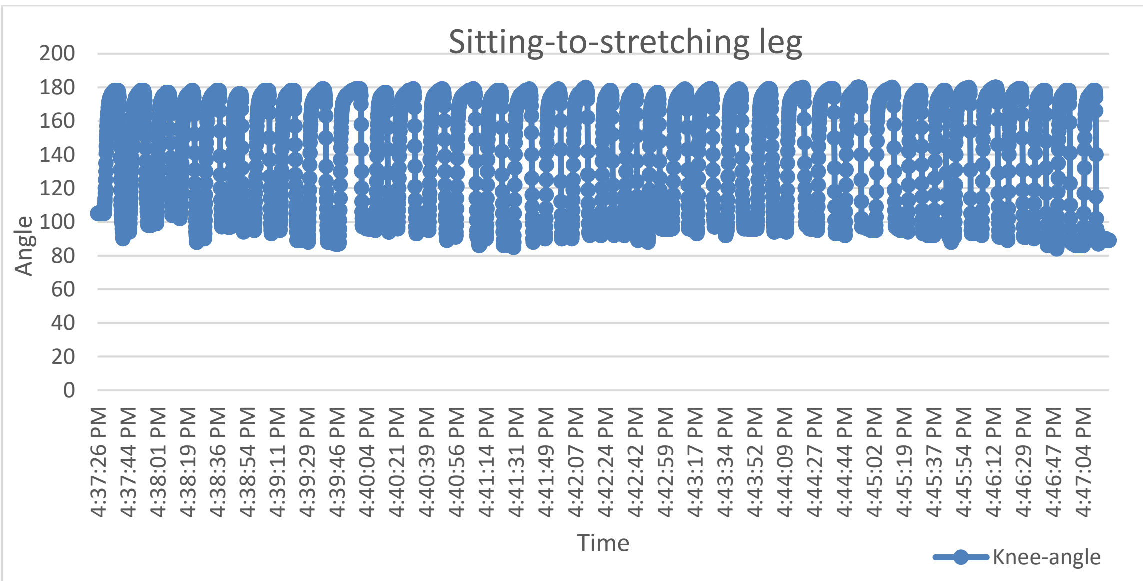

Figure 18: Lab test result of sitting straight and stretching leg and bringing it back.

In the graph x-axis is time and y-axis is angle and it can be seen that it starts off around 90 to 100 degrees which is a sitting position with leg bent and knee angle around 80 to 100 degrees in general and then stretches the leg out thus moves to 180 degrees and brings back the leg to normal position to ideally 90 degrees. Thus, the same pattern of moving from 90 to 180 and coming back to around 90 degrees can be seen in the Figure 18 and this is done for 10 minutes where time can be seen in the x-axis of the graph.

4.2.2 Single Leg Rise to 90 with Ankle Movement

Another lab test performed was standing straight and raising the leg to 90 degrees position and making it to stay for few seconds and then bring it back to standing position. Meanwhile, when the leg is in the air, moving ankle up and down to test the relation between knee angle an ankle angle was also performed where ankle sleeve and flex-sensor were placed on ankle for this experiment along with knee sleeve and flex-sensor. The result of that can be seen in the below Figure 19.

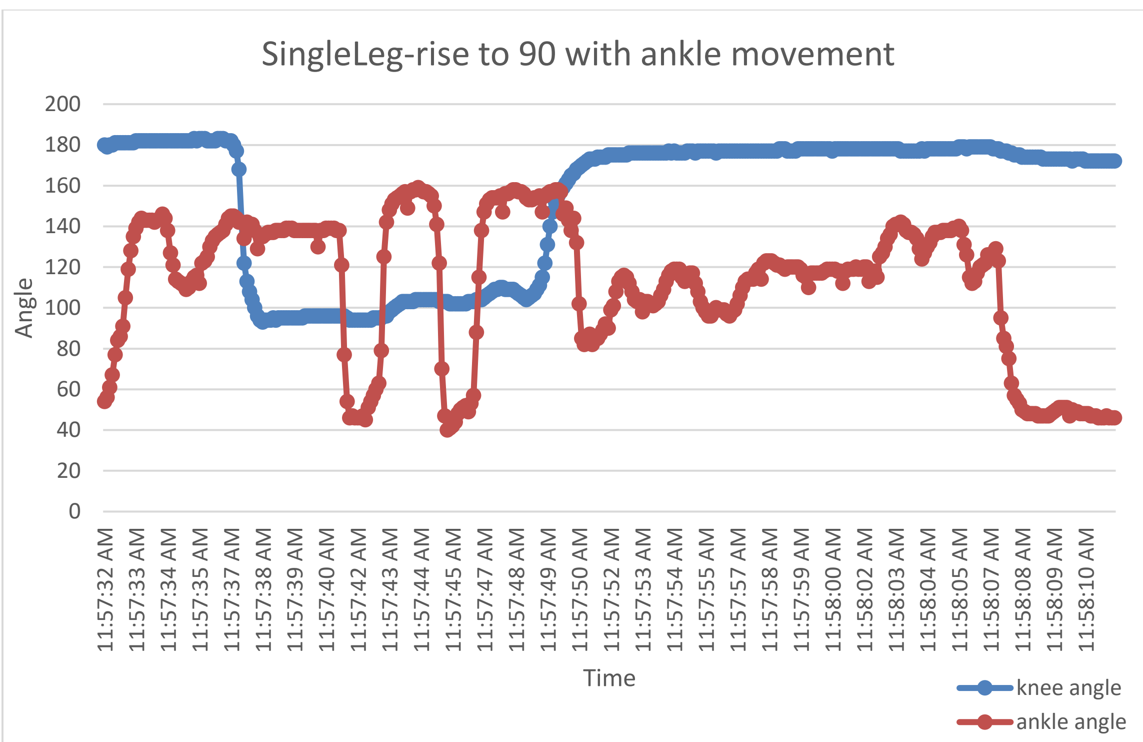

Figure 19: Lab test for single leg rise to 90 degrees with ankle movement.

4.3 Field Test

The flex-sensor passed the calibration test and the lab test with couple of experiments conducted inside the lab. Now, it was taken into the field where walking, jogging, spot jogging, jumping, hurdles and many more experiments were performed, and the results are discussed below.

4.3.1 Spot Jogging

One experiment was to do spot jogging with one flex-sensor on knee sleeve and another on ankle sleeve. The following graph shows the result of spot jogging with varying speed where sensors were attached to only one leg. The knee angle starts at 180 degrees as leg would be straight before start to jog and then when the spot jogging starts, knee angle moves from 180 degrees to 140 and comes back to around 170 since the leg won't be straight while jogging.

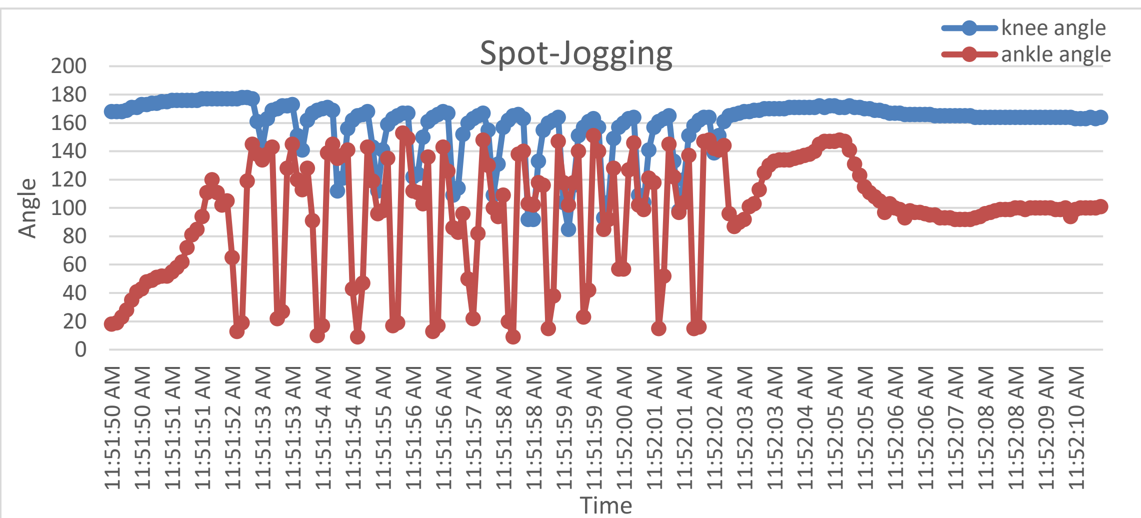

Figure 20: Field test of spot jogging.

When the spot jogging is done rigorously the knee is raised increasingly making the leg to bend more, thus angle drops down slowly to 120, then to 90 degrees and come back to 160-170 degrees. This on-filed experiment lasted about a minute which started off with a slow spot jogging for few seconds and increased the speed for few seconds and ended up with slow spot jogging. Thus, you can see the knee angle in the Figure 20 to move from 180 to 140 coming back to 170 and repeating this for couple of seconds and the moving from 165 to 120 and 165 to 90 when it is done with more speed and again goes back to 170 to 120 degrees when slowed down in the end of the spot jogging.

Ankle angle is also determined to check the relation between knee and ankle angle and ankle angle starts to move from around 100 degrees down to 40 and moves up to 140 degrees and moves back to 20-40 range and this repeats continuously till end. Whenever there is a high bend in the knee angle, that means the leg is in the air and ankle angle straightens up in that motion and when knee angle straightens with leg reaching ground then ankle bends to as low as 50 degrees on average and settles at 90 degrees at the end of the experiment since the ankle angle is 90 degrees when we stand straight.

4.3.2 Walking Up and Down the Stairs

Next did walking up and down the stairs again varying the speed from normal to high and back to normal speed. In this experiment flex-sensors are used for knee and ankle of right leg and only for ankle on left leg. The Figure 21 shows the result of the experiment walking up and down as this was done for around 1000 seconds which is around 16 minutes.

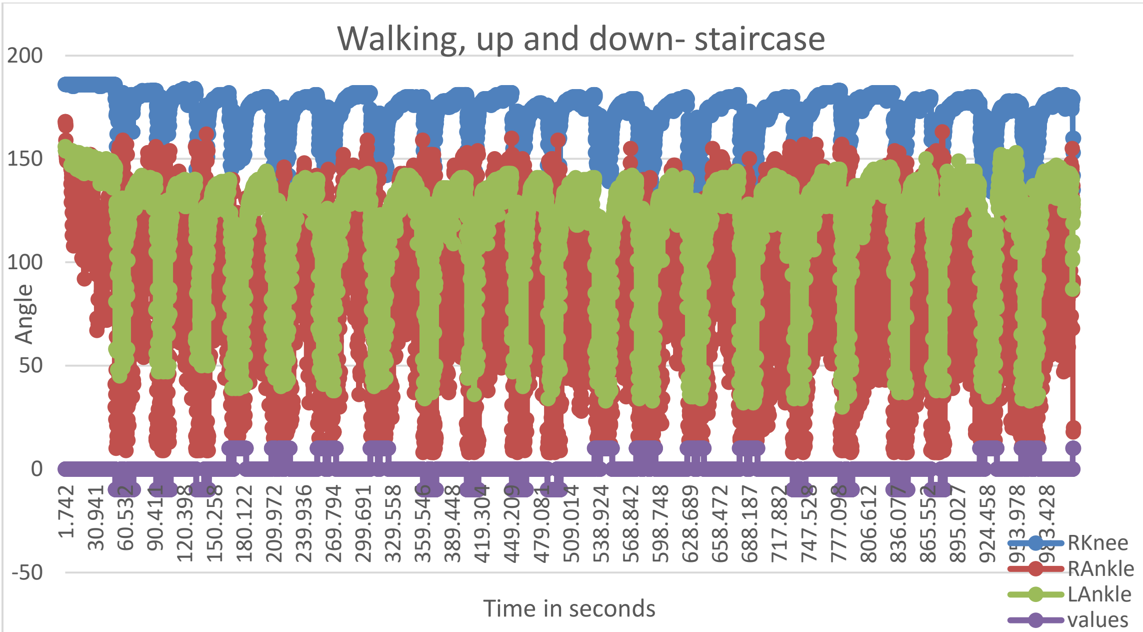

Figure 21: Field test of walking up and down stairs

The right knee angle in the blue color initially starts off at 180 degrees and as we walk it remains at 180 degrees. When goes down the stairs or up the stairs, the knee angle changes from 180 to 120 degrees and comes back to around 170 degrees. This repeat for 1000 seconds. The up stair and down stair walk is differentiated with a purple line on 0 at x-axis named values. The value starts off at 0 and goes to -1 if it is down stairs and goes to +1 if it is upstairs. The ankle angles of both left and right leg looks similar but there is once cycle delay since one leg moves after another. The right ankle angle goes down below 50 degrees describing the fact that though flex-sensors are accurate they tend to deteriorate in their performance due to a kink on flex-sensor because of more usage. Since the Knee angle walking for upstairs and downstairs cannot be seen clearly a chunk of data is plotted separately as shown below.

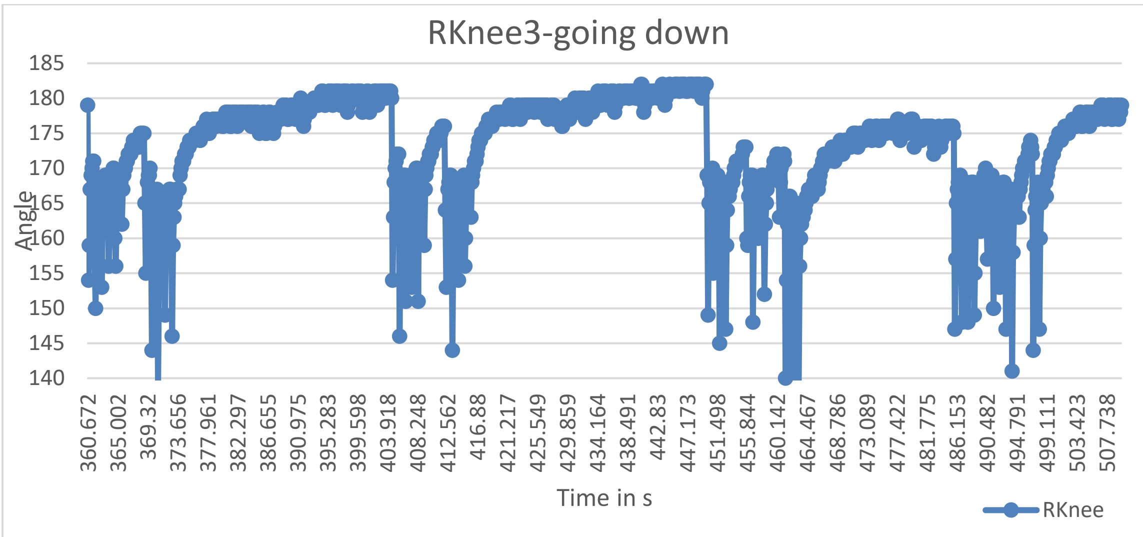

Figure 22: Part of walking downstairs

Figure 22 shows the right knee angle for walking downstairs around 150 seconds. The horizontal blue lines indicate walking part on plain surface that's why the knee angle goes back to 180 degrees and the rest is walking downstairs by varying the speed. Here the knee angle moves from 170 to 140 and back to 170 degrees in a repetitive manner. Whereas, for walking upstairs the knee angle moves from 170 to 130 and back to 170 degrees since knee bends a bit more when compared to walking downstairs. Figure 23 shows a part of walking upstairs with flex-sensor on back of the knee and this part is of around 210 seconds. Though walking upstairs and downstairs look in a similar pattern, there is slightly more bend in the knee angle when walking upstairs since it requires a bit more effort.

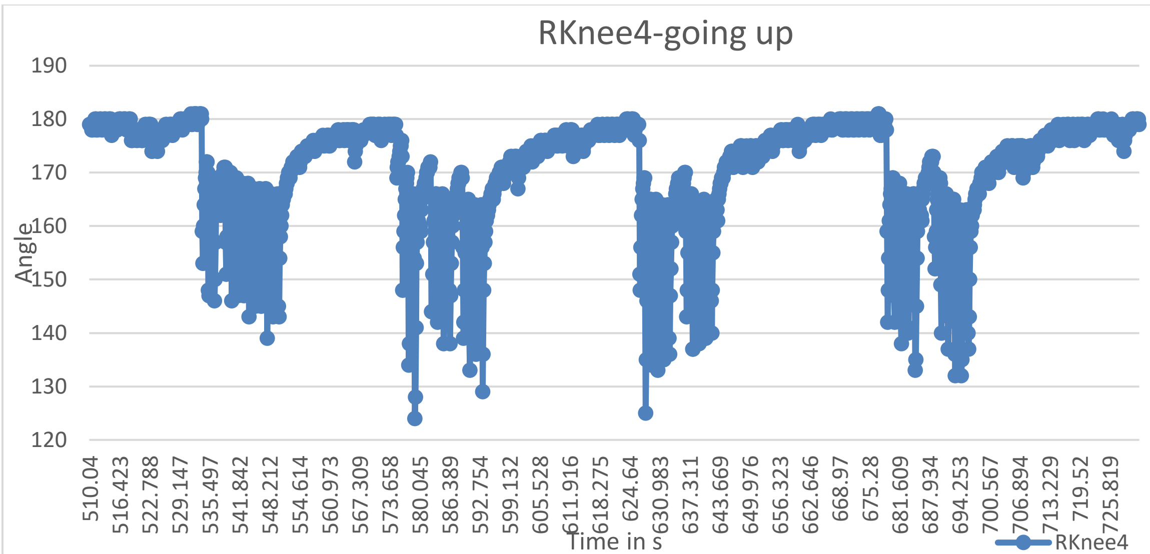

Figure 23: Part of walking upstairs.

4.3.3 Hurdles

Another experiment conducted was performing hurdles. The flex-sensors was used only on right leg. This experiment was conducted for couple of minutes where 15inch blocks were used as hurdles for the experiment. The result for hurdles are shown in the below Figure 24.

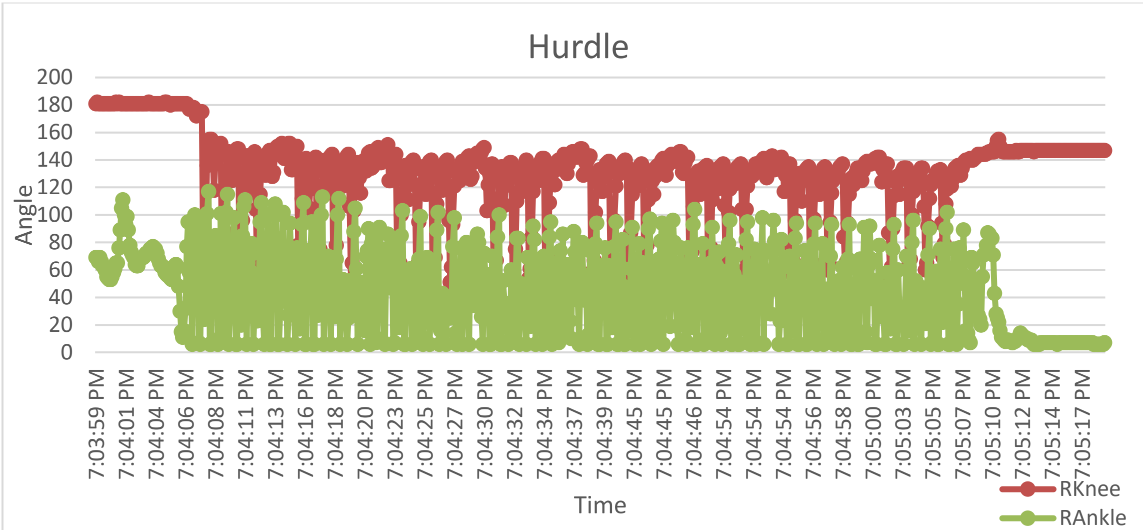

Figure 24: Hurdles graph

The angles of knee and ankle during hurdles are plotted in the above graph. The right knee angle starts at around 180 degrees as athlete would stand straight before starting the hurdles and the ankle angle would be around 80-100 degrees. Once the hurdles start, the knee angle drops down to around 50 degrees as knee would be bent a lot more when compared to spot jogging or walking or running. So, the knee angle moves from 180 to 50 and goes back to 150 degrees and repeats going from 150 to 50 and back to to 150 till the hurdles are over. The more details can be seen in the next Figure 25 as it has only knee angle.

The ankle angle starts moving from around 80-100 degrees mark and goes down till 20 degrees and moves up till 120 degrees and repeats going from 120 to 20 and back to 120 degrees while performing the hurdles as shown in the graph. As there was a kink on flex-sensor and due to more usage, the value of ankle angle goes beyond 20 degrees as well. The knee angle for hurdles is studied in more details below.

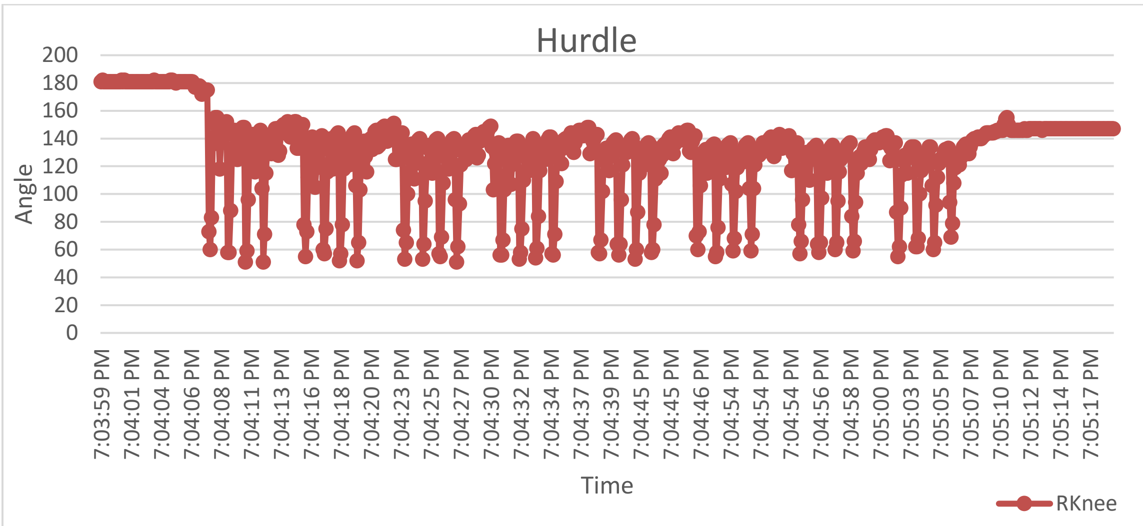

Figure 25: Hurdles graph for only knee angle.

From Figure 25 it is visible that there is only knee angle plotted for the hurdles. As already stated above it starts from 180 and goes down till 50 and comes back to 150 degrees, this can be seen in the above figure. There were total four hurdles of 15inch height, so total four jumps and running back again doing all the four hurdles to reach the start point. The graph also has 4 spikes down depicting the crossing of hurdle and some gap depicting running part and again 4 spikes going from 150 to 50 degrees and this repeat till the end thus, showing the result of knee angle for hurdles.

4.3.4 Jumping from toe and landing on heel (Accuracy Test).

In this experiment, the flex-sensors are placed under toe and heel to relate the pressure of toe and heel against knee and ankle angles but here knee and ankle angles are calculated by Tracker software tool which has protractor to track markers attached on hip, knee, ankle, and toe to calculate knee and ankle angles.

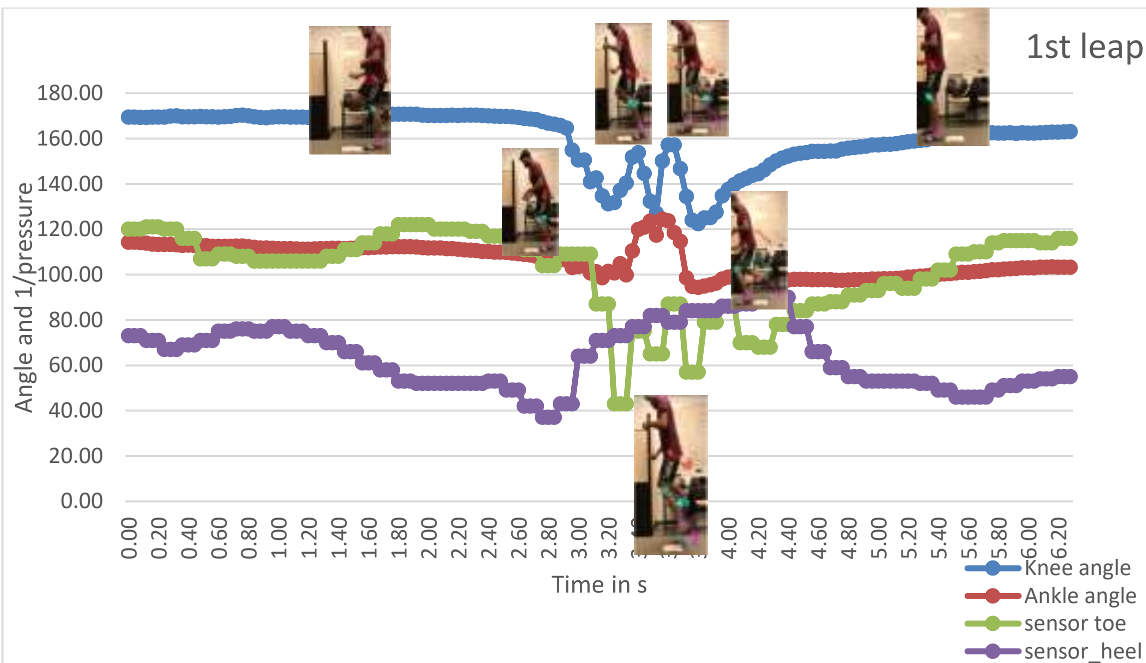

Figure 26: First leap from toe and landing on heel.

The Figure 26 depicts a leap over A4 size paper where the takeoff of leap starts on toe and lands on heel. The images on graph are kept correspondingly with knee angles. Each image is the position of corresponding crests and trough of knee angle graph. The x-axis is time in seconds and y-axis is angle and inverse of pressure (i.e., higher the angle value, lower the pressure). Pictures are also attached to show the leap process to relate angles and pressures accordingly. When standing straight on one leg to start the leap, knee angle is 170 degrees, ankle angle is 90 degrees, pressure on toe is less and more on heel. Just before takeoff the leg slightly bends forward putting pressure on toe to takeoff. Thus, knee angle starts decreasing from 180 degrees to lower, ankle angle also decreases a little just before takeoff and accordingly the pressure on toe increases and on heel decreases.

Just after takeoff, the knee straightens up a little in the air, then bends a bit during the motion then straighten up again before landing and bends again as heel lands and straightens up to stand after the entire motion. Thus, in this process, the knee angle works accordingly moving up and down from 160 to 120 degrees. When it straightens up a bit it comes to 160 degrees as it does not straighten up completely to 180 and when it bends it will be 120 degrees as shown in the Figure 26.

In this process the ankle angle remains low only at two positions and they are one at takeoff and another after landing. Other than that, the ankle does not bend in the entire motion of leap and remains around 135 degrees. Similarly, the pressure on toe initially remain normal and then goes down deep i.e., pressure increases at takeoff and after that pressure slowly decreases i.e., goes up in the graph. Pressure on heel is bit high than toe before starting the leap i.e., in standing position (line in the graph is lower) and pressure decreases during the motion (line in the graph increases) and when landing on heel, pressure increases (line in the graph decreases). This entire process of leaping is repeated two more times whose graphs are in the Figures 27 and 28 below.

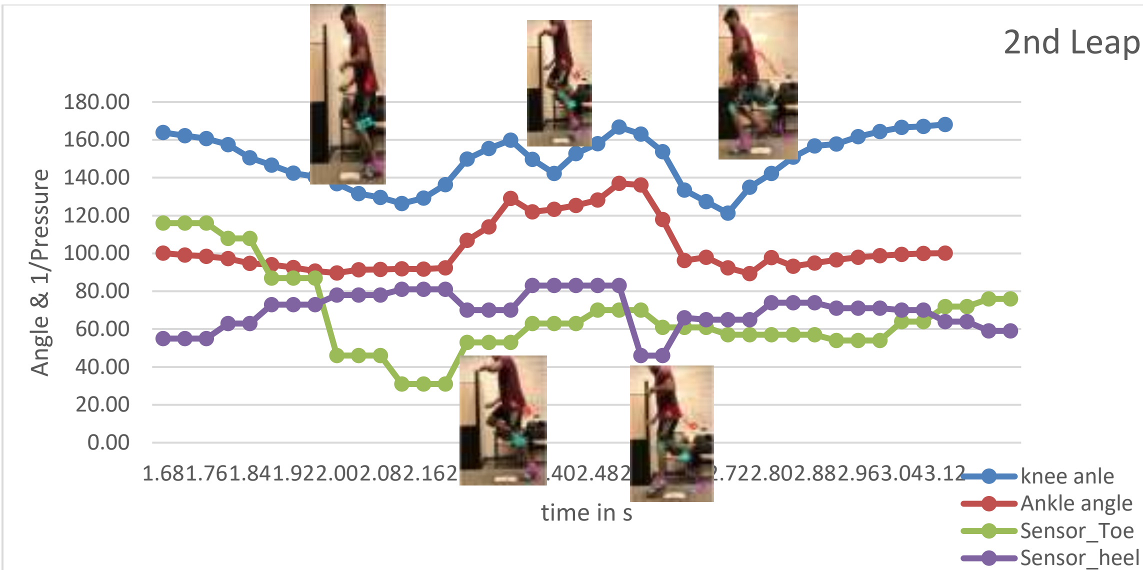

Figure 27: Second leap from toe and landing on heel.

The images on graph are kept correspondingly with knee angles. Each image is the position of corresponding crests and trough of knee angle graph.

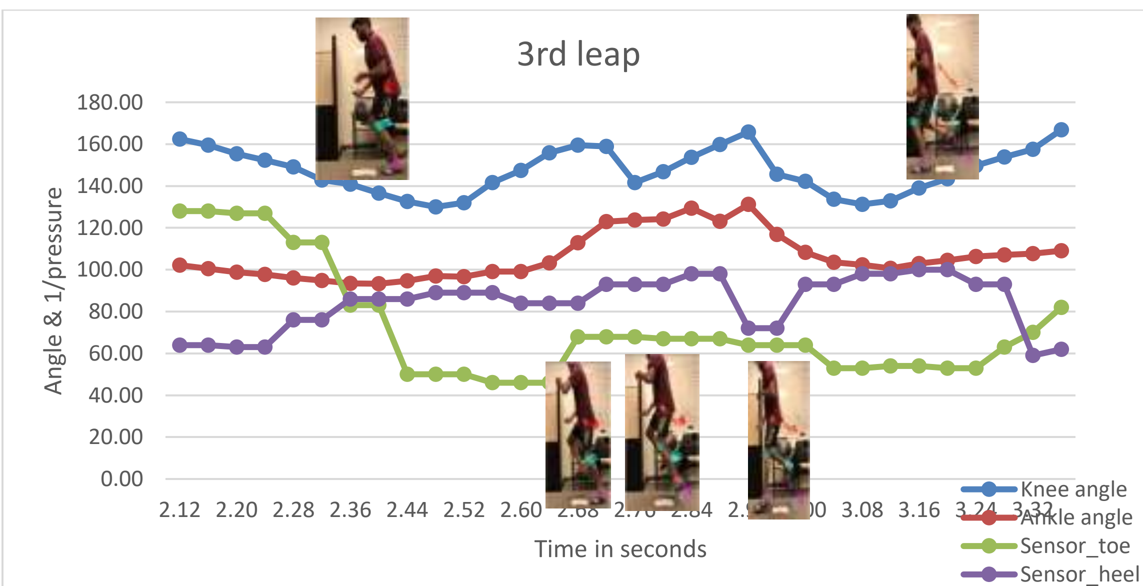

Figure 28: Third leap from toe and landing on heel.

4.4 Comparison of Hardware with Software (Tracker)

Single leg rise with ankle movement was performed with flex-sensors placed in knee sleeve on back of the knee and markers are placed on hip, knee, ankle, and toe for recording the video. Knee and ankle angles are taken from both hardware and software and compared for this experiment. Initially leg is raised and kept in 90 degrees position and then ankle is moved up and down for few seconds and then leg is brought back to normal position.

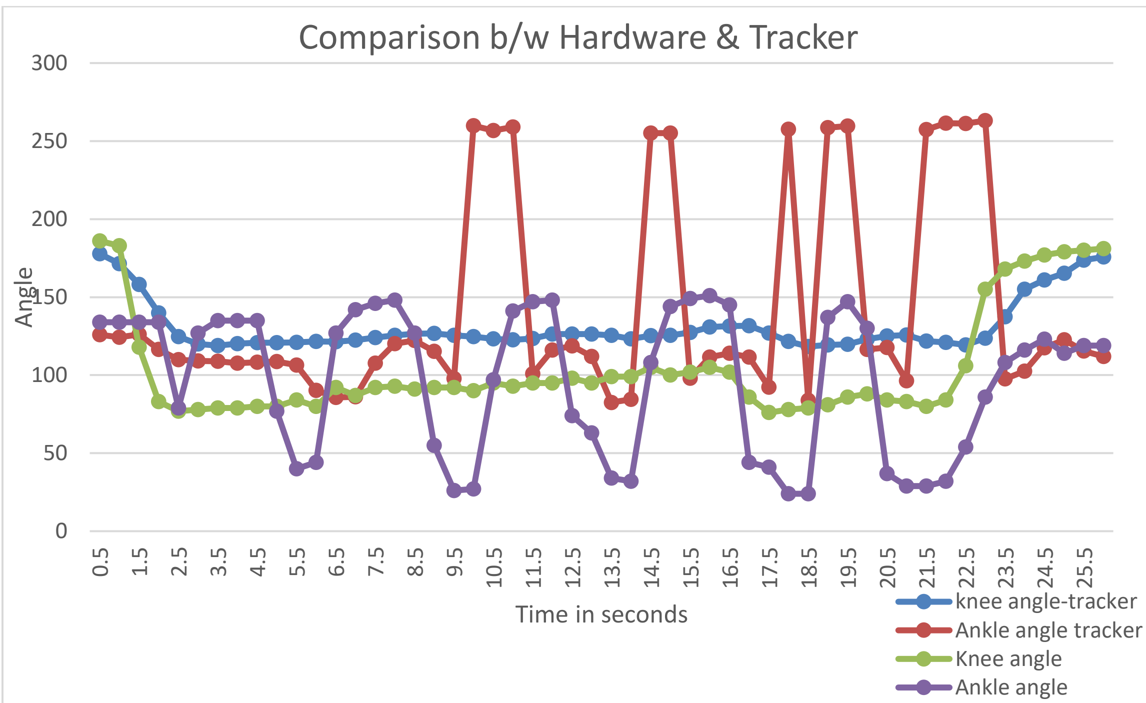

Figure 29: Comparison of hardware and software (Tracker) result for one experiment.

The knee angle (green) from flex-sensor starts around 180 degrees goes to around 90 degrees and stays there for few seconds and comes back to 180 degrees which is standing position. The knee angle (blue) from Tracker software also behaves in similar fashion but there is a change in value. It goes from 180 to 120 and back to 180 degrees and this is because of the position of markers on body and slight movement of leg during the experiment. When the leg is raised in the air for this experiment the knee angle should be ideally 90 degrees but the average error difference between hardware and Tracker software value of knee angle when raised in the air is 35 degrees with average value of knee angle from flex-sensor when leg was in air was 89.7 degrees and from Tracker software was 124 degrees.

Similarly, the ankle angle (purple) from flex-sensor starts at 120 degrees in the standing position and when leg is raised to move the ankle up and down, ankle angle goes from 150 to 40 degrees and finally when leg is brought back to normal position, ankle angle goes back to around 120 degrees. It should be around 90 degrees but due to a kink on flex-sensor it deviates some amount of angle. Ankle angle (red) from Tracker software starts from 120 and when leg raised, and ankle moves up and down then, value of ankle angle moves from 120 to 250 degrees and back to 120 degrees when leg is down to standing position. It should move from around 150 to 80 degrees and back to 150 degrees ideally when ankle is moving up and down. Though Tracker software is accurate enough, but it was an error in the ankle angle for this experiment.

CHAPTER 5: VISUALIZATION TOOL

A software tool was built in MATLAB to visualize the data for the Wearable Knee Angle Sensor. The data was collected from the tool called Tracker which tracked angles and (x,y) coordinates of hip, knee, ankle and toe. By having markers at these positions on an athlete and recording the event and feeding it to tracker gave the required data which in-turn was used with MATLAB to create the simulation. Simulated various leg motions which can be found in Figure 34.

5.1 Simulation Nomenclature

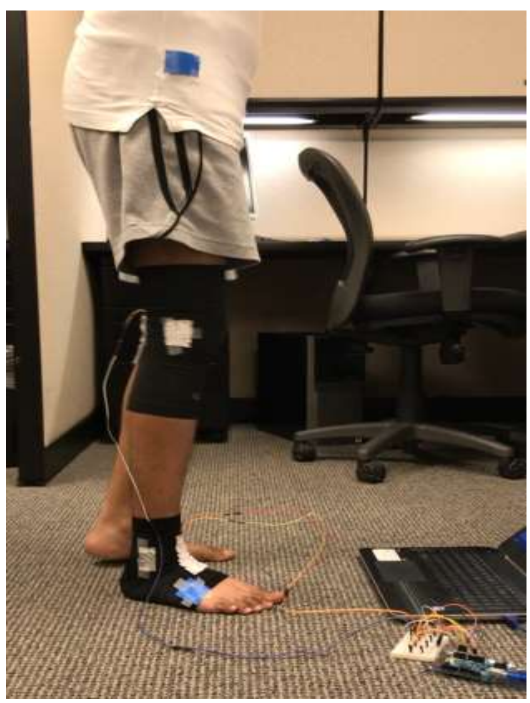

Figure 30: Markers on the body

The figure above shows the markers on hip, knee, ankle, and toe. This experiment was single leg bend conducted as a lab test which was recorded and feed to Tracker software tool which can give us (x,y) coordinates of all the four points, angles of knee and ankle. But initially only (x,y) coordinates are taken and given to MATLAB for simulation which calculates Relative Hip Angle (RHA), Relative Knee Angle (RKA), Relative Ankle Angle (RAA), Absolute Knee Angle (AKA) and Absolute Ankle Angle (AAA) to display the motion. Even vice-versa can also be performed i.e., taking angles from Tracker and calculating (x,y) coordinates of hip, knee, ankle and toe to display the motion.

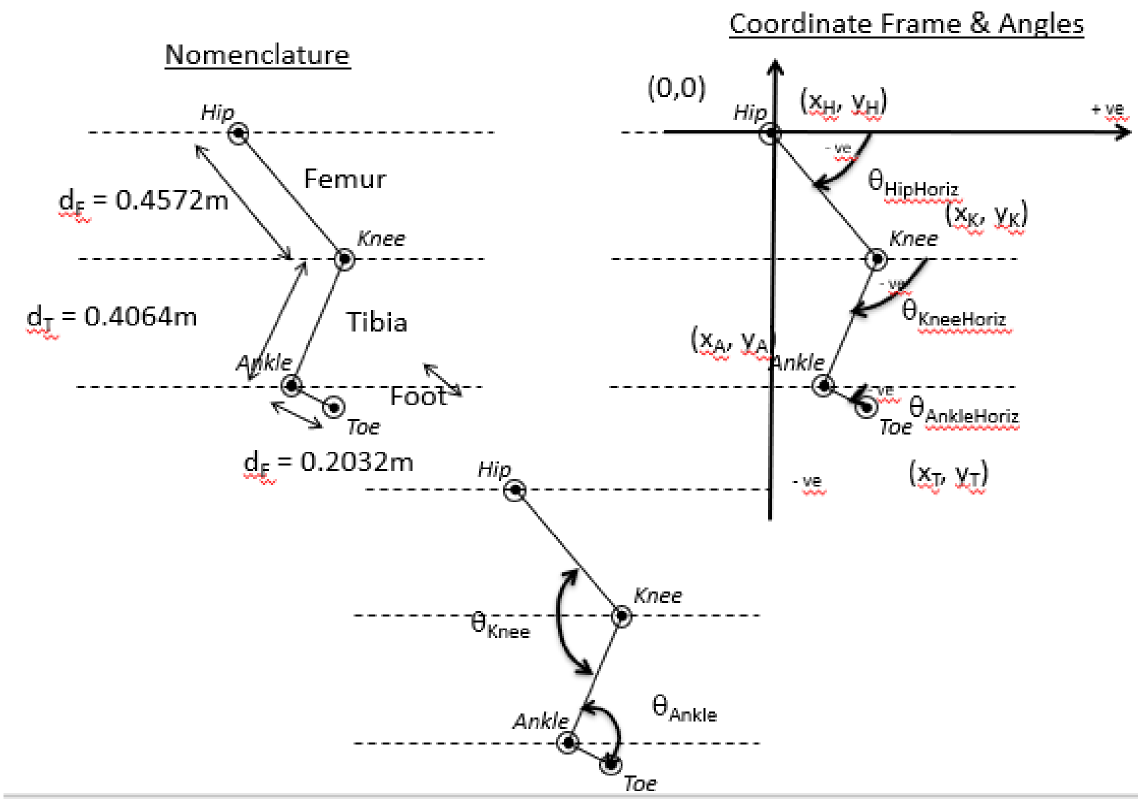

Figure 31: Simulation model nomenclature

Figure 31 shows the simulation model nomenclature where, θ_{Knee} is Absolute Knee Angle (AKA), θ_{Ankle} is Absolute Ankle Angle (AAA), $\theta_{HipHoriz}$ is Relative Hip Angle (RHA), $\theta_{KneeHoriz}$ is Relative Knee Angle (RKA) and $\theta_{AnkleHoriz}$ is Relative Ankle Angle (RAA). These angles are calculated using suitable equations and simulation is displayed accordingly as shown in Figure 32. The entire plot is divided into two subplots where plot on left side displays the motion with (x,y) coordinates with x-axis as x coordinates and y-axis as y coordinates along with all the angles in real time and plot on right side shows the waveforms of these angles with x-axis as time and y-axis as angles. Figure 32 is a screenshot of leg bend motion where the leg is in standing position and Figure 33 shows bending position.

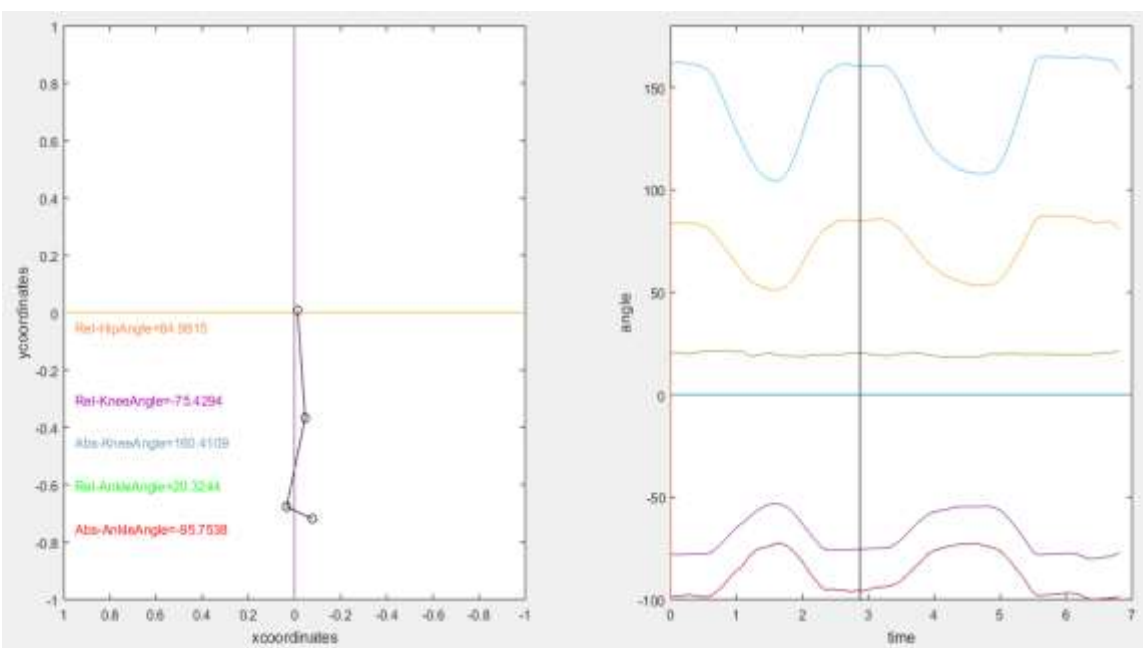

Figure 32: Simulation image of single leg bend with-standing position.

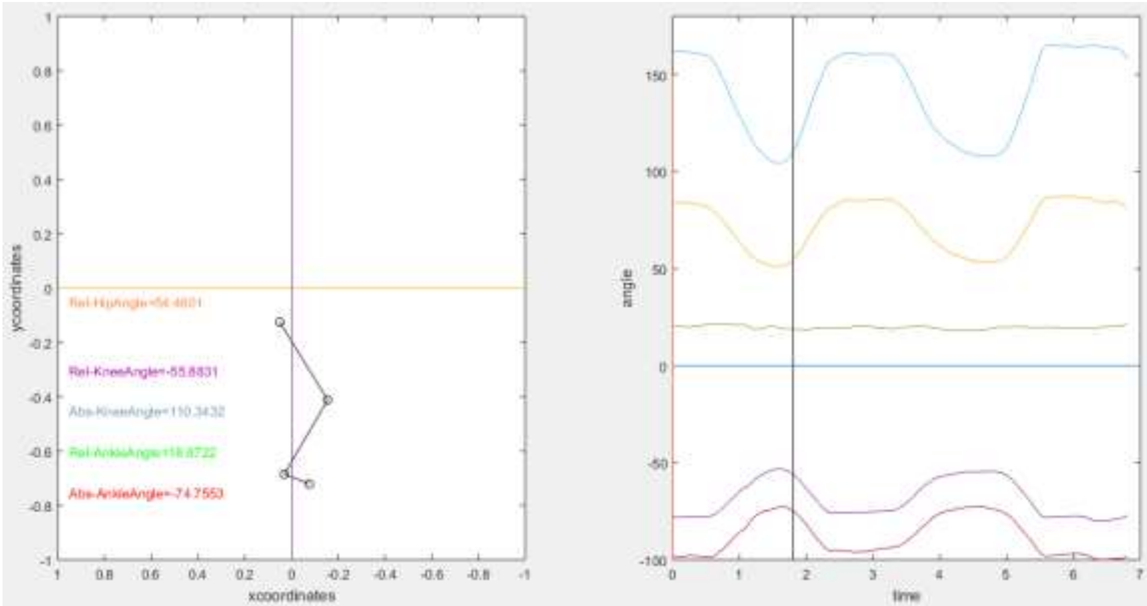

Figure 33: Simulation image of single leg bend with-bending position.

5.2 Simulation of Walking

There are some other simulations which are performed using same techniques but with little modifications to it. Walking is another example shown in Figure 34. Blue is right leg and red is left leg. In this figure there are two circles on left foot (red in color) which indicates the foot is on ground and if there are no circles on foot that means the foot is in air. Similar to the previous experiment, the entire plot is divided into two subplots left and right. The left subplot displays the motion of walking with (x,y) coordinates along with angles distinguishing between right and left leg by indicating it in two different colors. The right subplot is again divided into 5 plots displaying all the angles for both left and right legs. The upper part of the body is not considered since didn’t calculated Absolute Hip Angle (AHA) for walking thus, only lower limbs are displayed.

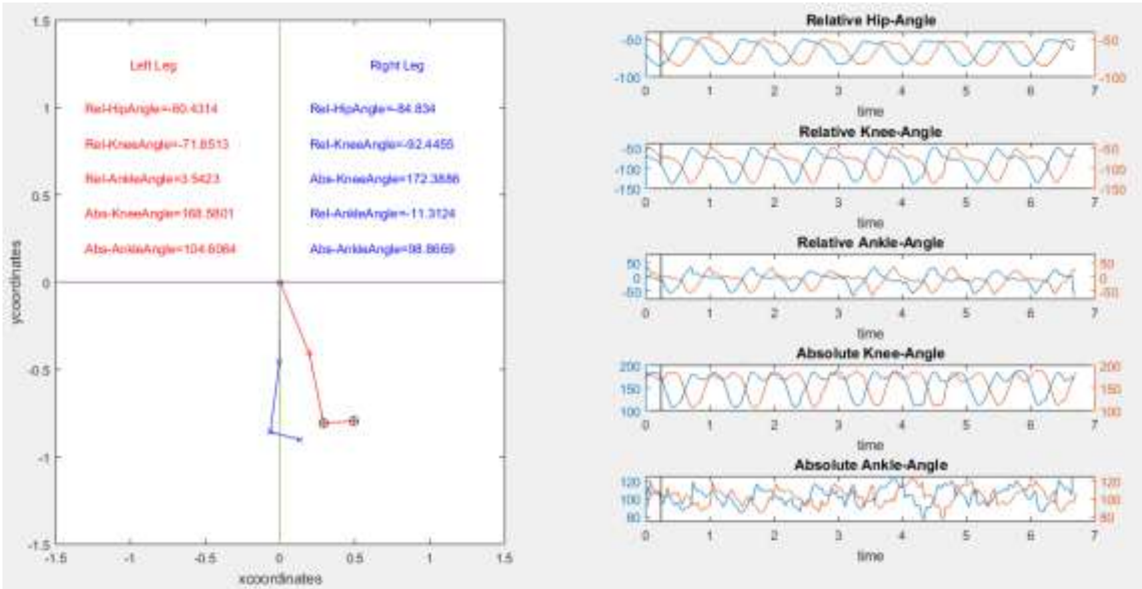

Figure 34: Simulation image of walking.

To have a clear picture subplot 1 and subplot two are shown individually below.

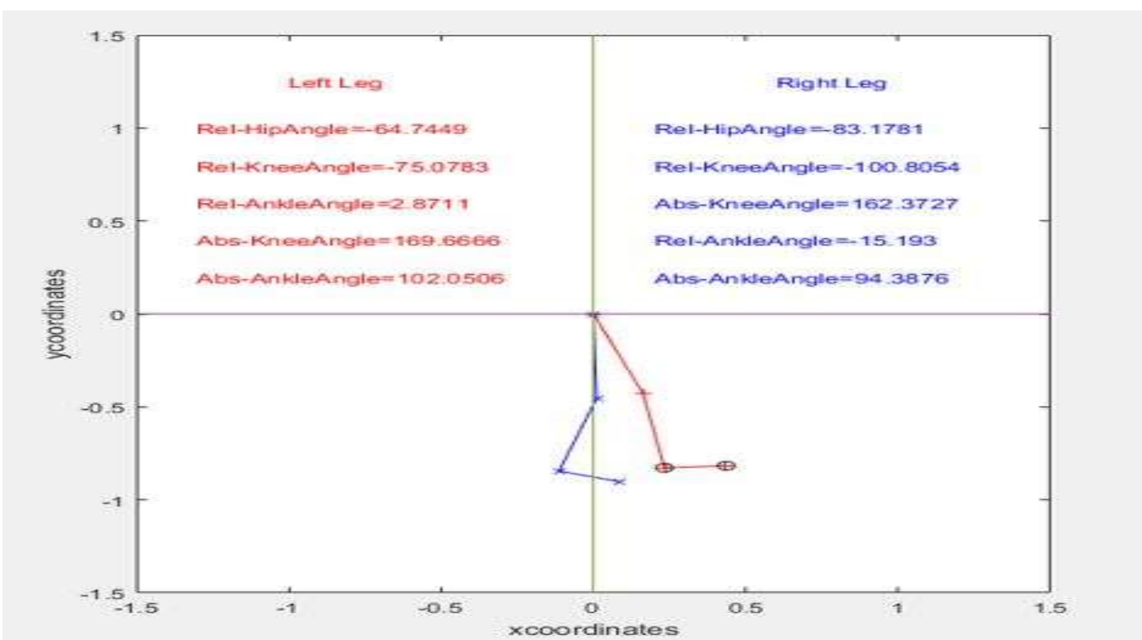

Figure 35: Simulation images of limbs walking.

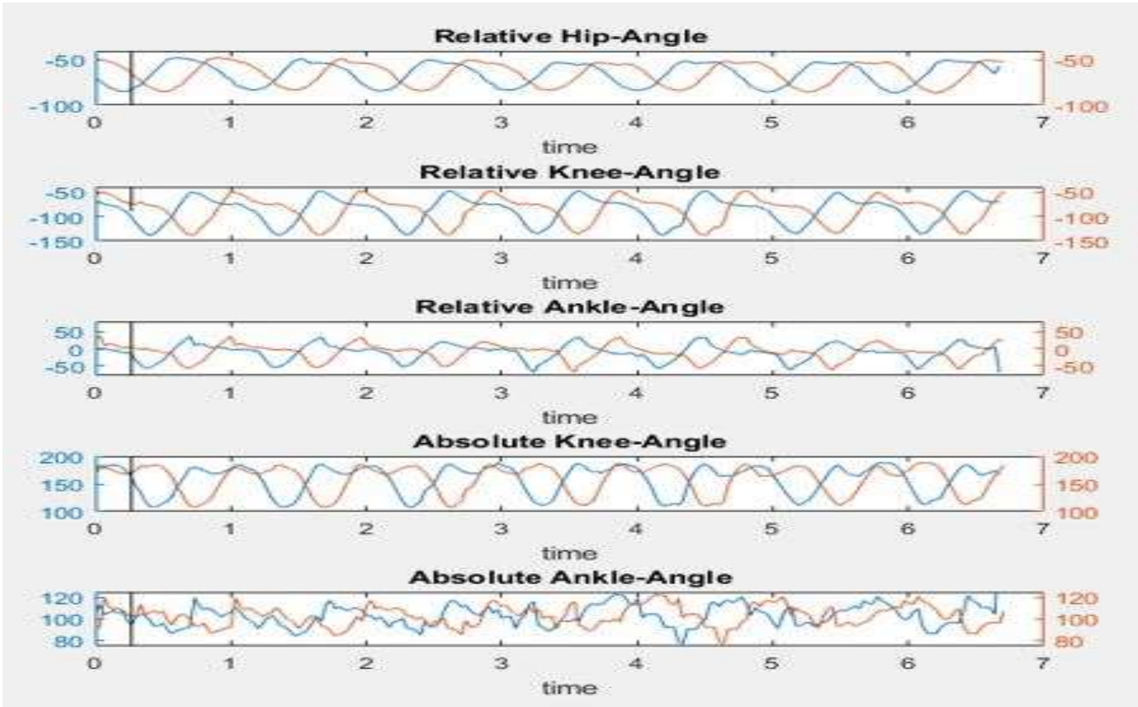

Figure 36: Simulation image of graph for walking.

5.3 Simulation of Anterior Cruciate Ligament (ACL) Injury

For catching a ball, the upper part of the body is also considered in the display and this was possible after taking Absolute Hip Angle (AHA). The Figure 37 shows Simulation image of Anterior Cruciate Ligament (ACL) injury due to one-legged, heel first landing at 180 degree, the screenshot of this plot with right subplot divided into 6 subplots after including Absolute Hip Angle. The upper part of the body is given a fixed length and it is in blue color. The left leg is on ground thus, left foot has two circles. All the equations can be found in the MATLAB sample code given in appendix.

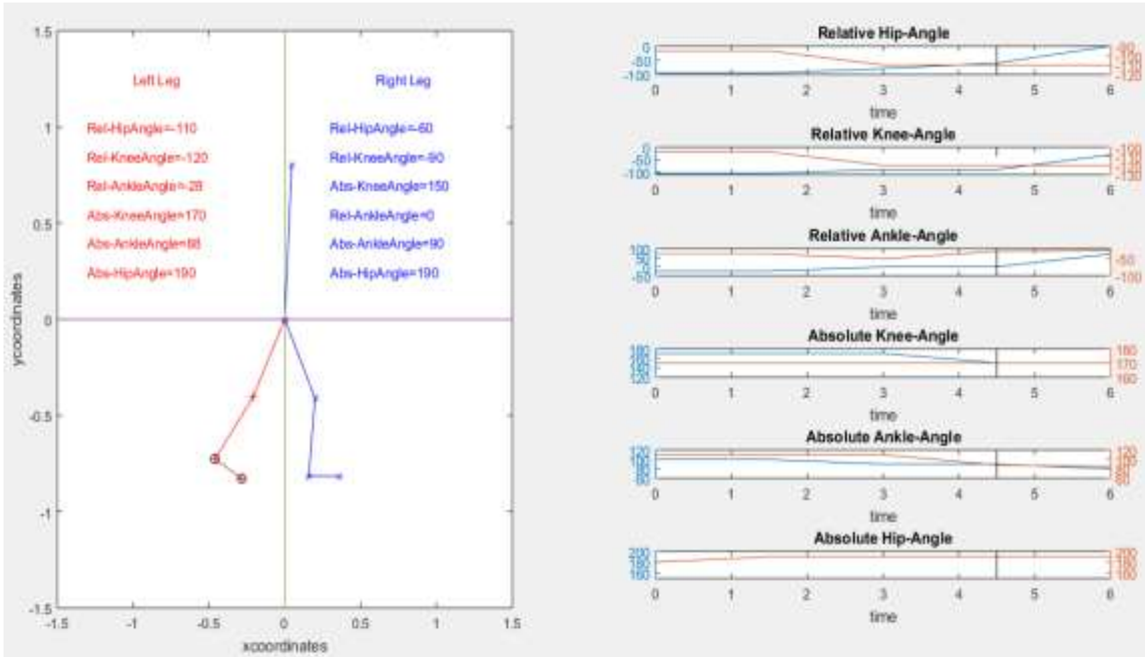

Figure 37: Simulation image of ACL injury due to one-legged, heel first landing.

CHAPTER 6: CONCLUSION AND FUTURE WORK

6.1 Conclusion

This thesis describes the design and testing of a Wearable Knee Angle Sensor (WKAS) designed for operation in contact sport game situations. The WKAS was configured with a Flex-Sensor mounted in a senor-sleeve behind the knee attached to a knee sleeve. An Arduino was used to collect and process the data and store it on an SD card. The data from the SD card can be downloaded and analyzed in visualization software developed in MATLAB.

6.1.1 Limitations of WKAS Design

Flex-Sensor gets kinks and must be recalibrated. Sensor-sleeve was redesigned to prevent kinking. Sometimes knee sleeve shifts a little which must be adjusted. Sensors rides up and down in knee sleeve and sensor pins break-off during extreme bending.

6.2 Future Work

Though there are few limitations for flex-sensor, it is the most suitable one with all the other advantages of being light weight, flexible, comfortable, accurate (Until before it breaks). The WKAS configured with flex-sensor can be modified in the design in order to overcome the limitations above. The future proposed design is shown below having two blocks (yellow in color) holding the flex-sensor in the middle, these two blocks are 3D-printed, and it is not the final design as there can be changes in the size and shape of these blocks, but the concept would he holding the flex-sensor between two blocks connected to

Arduino and lot of testing need to be done on this design. This design also can hold a miniaturized Arduino board inside it and that would be the future Wearable Knee Angle Sensor to measure knee angles to help in performance assessment (e.g., leg motion while sprinting) and for injury prevention (e.g., leg motion leading to specific injuries).

Figure 38: Future design of WKAS.

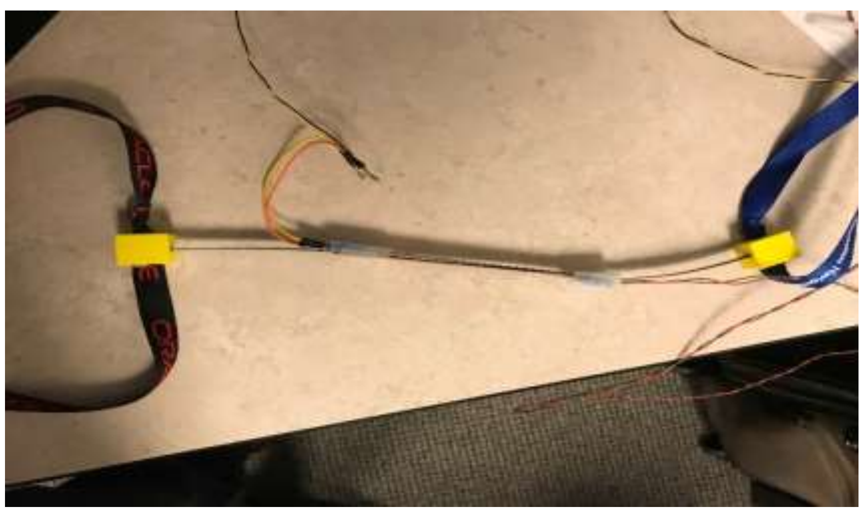

Figure 39: Future design testing.

APPENDICES

APPENDIX A: MICROCONTROLLER CODE

```
#include <SD.h> //Load SD card library

#include<SPI.h> //Load SPI Library

const int chipSelect = 4;

File myFile;

unsigned long time;

void setup() {

// Open serial communications and wait for port to open:

Serial.begin(9600);

//  while (!Serial) {

; // wait for serial port to connect. Needed for native USB port only

// }

Serial.print("Initializing SD card...");

// see if the card is present and can be initialized:

if (!SD.begin(chipSelect)) {

myFile.println("Card failed, or not present");

// don't do anything more:

return;

}

myFile.println("card initialized.");
```

```
}
void  loop() {
// put your main code here, to run repeatedly:
//read the input on analog pin 0,1,2,3:
int sensorvalue1 = analogRead(A0);
int sensorvalue2 = analogRead(A1);
int sensorvalue3 = analogRead(A2);
int sensorvalue4 = analogRead(A3);
int degree1, degree2, degree3, degree4;
myFile = SD.open("Data.txt", FILE_WRITE);
if (myFile) {
Serial.print("Time: ");
time = millis();
//prints time since program started
Serial.print(time);
//print out the value you read:
Serial.print(" device1= ");
Serial.print(sensorvalue1);
degree1 = ((0.1759*sensorvalue3) + 6.7192);
Serial.print(" angle1= ");
Serial.print(degree1);
delay(1);  //dealy in between reads for stability
```

```
//print out the value you read:
Serial.print("  device2= ");
Serial.print(sensorvalue2);
degree2 = ((0.1759*sensorvalue3) + 6.7192);
Serial.print(" angle2= ");
Serial.println(degree2);
delay(100);  //dealy in between reads for stability
//print out the value you read:
Serial.print("  device3= ");
Serial.print(sensorvalue3);
degree3 = ((0.1759*sensorvalue3) + 6.7192);
Serial.print(" angle3= ");
Serial.print(degree3);
delay(100);  //dealy in between reads for stability
//print out the value you read:
Serial.print("  device4= ");
Serial.print(sensorvalue4);
degree4 = ((0.1759*sensorvalue3) + 6.7192);
Serial.print(" angle4= ");
Serial.println(degree4);
delay(100);  //dealy in between reads for stability
myFile.print("Time: ");
```

```
time = millis();

//prints time since program started

myFile.print(time);

myFile.print(", ");

myFile.print(" device1= ");

myFile.print(sensorvalue1); //write temperature data to card

myFile.print(", ");

myFile.print(" angle1= ");

myFile.print(degree1);

myFile.print(", "); //write a commma

myFile.print("device2= ");

myFile.print(sensorvalue2); //write temperature data to card

myFile.print(", ");

myFile.print(" angle2= ");

myFile.println(degree2);  //write pressure and end the line (println)

myFile.print(", ");

myFile.print(" device3 ");

myFile.print(sensorvalue3);  //write temperature data to card

myFile.print(", ");

myFile.print(" angle3= ");

myFile.print(degree3);

myFile.print(", ");
```

```
myFile.print(" device4= ");

myFile.print(sensorvalue4); //write temperature data to card

myFile.print(", ");

myFile.print(" angle4= ");

myFile.println(degree4);

myFile.close(); //close the file

}

}
```

APPENDIX B: MATLAB SAMPLE CODE FOR SIMULATION

0) Inputs (Df, Dt, ...)

The inputs given to this program are:-

```
rxh = 0; ryh = 0; % x,y positions of right hip [rxh- right x-position
of hip]
Lxh = 0; Lyh = 0; % x,y positions of left hip [Lyh- left y-position of
hip]

df = 0.4572;  % distance of Femur
dt = 0.4064;  % distance of Tibia
dfo = 0.2032; % distance of Foot
sheet = 1;    % sheet number of Excel spreadsheet

i = 2; % i represents row number of spreadsheet
t = 5; % t resembles 5th column of spreadsheet in which time is present

rrh = 1; Lrh = 8; % right, left relative hip angles present in 1st & 8th
column
rak = 2; Lak = 9; % right, left absolute knee angles present in 2nd, 9th
column
raa = 3; Laa = 10; % right, left absolute ankle angles present in 3rd ,
10th column
```

1) Reading data from spreadsheet (Abs Knee, Abs Ankle, Rel Hip)

```
dataset = xlsread('Angles', 'sheet1', 'A1:q204');
       % reads the spreadsheet called Angles of sheet1 in the given
range a1 to q204

dataset(:,rrh)= yh % reads all the rrh(right relative hip) angles at a
time                                which are in 1st column

dataset(:,rak)= yak % reads all the rak(right absolute knee) angles at
a time                         which are in 2nd column

dataset(:,raa)= yaa % reads all the raa(right absolute ankle) angles at
a                              time which are in 3rd column

dataset(:,Lrh)= Lyrh % reads all the Lrh(Left relative hip) angles at a
time                            which are in 8th column

dataset(:,Lak)= Lyak % reads all the Lak(Left absolute knee) angles at
a time                          which are in 8th column

dataset(:,Laa)= Lyaa % reads all the Laa(Left absolute ankle) angles at
a                               time which are in 8th column

dataset(i,rrh) = rrha % reads rrh(right relative hip) angles from 1st
column                                  by each row-i where i increases
with time

dataset(i,rak) = raka % reads rak(right absolute knee) angles from 2nd
column                                  by each row-i where i increases
with time

dataset(i,raa) = raaa % reads raa(right absolute ankle) angles from 3rd
column                            by each row-i where i increases with time

dataset(i,Lrh) = Lrha% reads Lrh(left relative hip) angles from 8th
column by                              each row-i where i increases with
time

dataset(i,Lak) = Laka% reads Lak(left absolute knee) angles from 9th
column by                       each row-i where i increases with time

dataset(i,Laa) = Laaa% reads Laa(left absolute ankle) angles from 10th
column                                 by each row-i where i increases with
time
```

2) Calculating Rel Knee, Rel Ankle

The following equations calculate relative Knee & Relative Ankle angles:

```
rrk = dataset(:,rrh) + dataset(:,rak) - 180; % right relative knee
rra = -dataset(:,raa) + rrk + 180; % right relative ankle

Lrk = dataset(:,Lrh) + dataset(:,Lak) - 180; % left relative knee
Lra = -dataset(:,Laa) + Lrk + 180; % left relative ankle
```

3) Calculate X, Y for Hip, Knee, Ankle

```
rxk = rxh + (cos(dataset(i,rrh)/53.7)*df);
% x-position of right knee equals x-position of right hip + product of
distance of Femur and cosine of right relative hip angle

ryk = ryh + (sin(dataset(i,rrh)/53.7)*df);
% y-position of right knee equals y-position of right hip + product of
distance of Femur and sine of right relative hip angle

rxa = rxk + (cos(rrk(i)/53.7)*dt);
% x-position of right ankle equals x-position of right knee + product
of distance of Tibia and cosine of right relative knee angle

rya = ryk + (sin(rrk(i)/53.7)*dt);
% y-position of right ankle equals y-position of right knee + product
of distance of Tibia and sine of right relative knee angle

rxt = rxa + (cos(rra(i)/53.7)*dfo);
% x-position of right toe equals x-position of right ankle + product of
distance of Foot and cosine of right relative ankle angle

ryt = rya + (sin(rra(i)/53.7)*dfo);
% y-position of right toe equals y-position of right ankle + product of
distance of Foot and sine of right relative ankle angle


Lxk = Lxh + (cos(dataset(i,Lrh)/53.7)*df);
% x-position of left knee equals x-position of left hip + product of
distance of Femur and cosine of left relative hip angle

Lyk = Lyh + (sin(dataset(i,Lrh)/53.7)*df);
% y-position of left knee equals y-position of left hip + product of
distance of Femur and sine of left relative hip angle

Lxa = Lxk + (cos(Lrk(i)/53.7)*dt);
% x-position of left ankle equals x-position of left knee + product of
distance of Tibia and cosine of left relative knee angle

Lya = Lyk + (sin(Lrk(i)/53.7)*dt);
% y-position of left ankle equals y-position of left knee + product of
distance of Tibia and sine of left relative knee angle
```

```
Lxt = Lxa + (cos(Lra(i)/53.7)*dfo);
% x-position of left toe equals x-position of left ankle + product of
distance of Foot and cosine of left relative ankle angle

Lyt = Lya + (sin(Lra(i)/53.7)*dfo);
% y-position of left toe equals y-position of left ankle + product of
distance of Foot and sine of left relative ankle angle
```

4) Data Structure (Matrix ?)

```
x = [rxh rxk rxa rxt]; % x-position of right hip, knee, ankle, toe
y = [ryh ryk rya ryt]; % y-position of right hip, knee, ankle, toe

Lx = [Lxh Lxk Lxa Lxt]; % x-position of left hip, knee, ankle, toe
Ly = [Lyh Lyk Lya Lyt]; % y-position of left hip, knee, ankle, toe

xx = [rxa rxt]; % x-position of right ankle, toe for different color
foot
yy = [rya ryt]; % y-position of right ankle, toe for different color
foot

xxx = [Lxa Lxt]; % x-position of left ankle, toe for different color
foot
yyy = [Lya Lyt]; % y-position of left ankle, toe for different color
foot
```

5) Stick Figure GUI

```
subplot(5,2,[1,3,5,7,9]); % subplot of human legs with 5 rows, 2
columns with                            column 1, 3, 5, 7 & 9 combined


plot(max(x),max(y));
hold on;
plot(-max(x),-max(y));
% above three lines are for shifting the origin to the center

plot(x,y, '-xb'); %blue colour right % plot of right leg
plot(Lx,Ly, '-+r'); %red colour left % plot of left leg
% condition for making right foot to change its color when landed on
ground
    if (rra(i)<5) && (rra(i)>-5)
       sz = 40;
       scatter(xx,yy,sz,'MarkerEdgeColor',[0 0 0]);
    end
```

```
% condition for making left foot to change its color when landed on
ground
    if (Lra(i)<5) && (Lra(i)>-5)
       sz = 40;
       scatter(xxx,yyy,sz,'MarkerEdgeColor',[0 0 0]);
    end


xlim ([-1.5 1.5]); % x-limit of subplot
ylim([-1.5 1.5]); % y-limit of subplot


xlabel('xcoordinates') % x-label of subplot
ylabel('ycoordinates') % y-label of subplot


    % right
% x and y positions of strings where m is x position & n is y position
m0 = 0.6; m1 = 0.2; m2 = 0.2; m3 = 0.2; m4 = 0.2; m5 = 0.2;
n0 = 1.25; n1 = 1; n2 = 0.8; n3 = 0.6; n4 = 0.4; n5 = 0.2;


% strings with angle values
    str0 = (['Right Leg']);
    str1 = (['Rel-HipAngle=',num2str(rrha)]);
    str2 = (['Rel-KneeAngle=',num2str(rrk(i))]);
    str3 = (['Rel-AnkleAngle=',num2str(rra(i))]);
    str4 = (['Abs-KneeAngle=',num2str(raka)]);
    str5 = (['Abs-AnkleAngle=',num2str(raaa)]);


% text which consist of string & angle values at x,y position with blue
color
text(m0,n0,str0,'color', [0 0 1])
text(m1,n1,str1,'color', [0 0 1])
text(m2,n2,str2,'color', [0 0 1])
text(m3,n3,str4, 'color', [0 0 1])
text(m4,n4,str3, 'color', [0 0 1])
text(m5,n5,str5, 'color', [0 0 1])


%left

% x and y positions of strings where m is x position & n is y position
m6 = -1.3; m7 = -1.3; m8 = -1.3; m9 = -1.3; m10 = -1.3; m11 = -1;
n6 = 1; n7 = 0.8; n8 = 0.6; n9 = 0.4; n10 = 0.2; n11 = 1.25;


% strings with angle values
    str6 = (['Rel-HipAngle=',num2str(Lrha)]);
    str7 = (['Rel-KneeAngle=',num2str(Lrk(i))]);
    str8 = (['Rel-AnkleAngle=',num2str(Lra(i))]);
```

```
    str9 = (['Abs-KneeAngle=',num2str(Laka)]);
    str10 = (['Abs-AnkleAngle=',num2str(Laaa)]);
    str11 = (['Left Leg']);


% text which consist of string & angle values at x,y position with blue
color
text(m6,n6,str6,'color', [1 0 0])
text(m7,n7,str7,'color', [1 0 0])
text(m8,n8,str8, 'color', [1 0 0])
text(m9,n9,str9, 'color', [1 0 0])
text(m10,n10,str10, 'color', [1 0 0])
text(m11,n11,str11, 'color', [1 0 0])


% below three lines are for drawing x and y axis in the center of the
subplot
     x=get(gca,'xtick');
y=get(gca,'ytick');
plot([-max(abs(x)) max(abs(x))],[0 0]);
plot([0 0],[-max(abs(y)) max(abs(y))]);
```

6) Strip Chart GUI

```
subplot(5,2,2) % subplot of 2nd box with 5 row, 2 columns
xlabel('time') % x-label
ylabel('angle') % y-label
title('Relative Hip-Angle') % title of the subplot


yyaxis left % plotting y axis towards left
plot(x1,yh); % plot x1(time) vs yh(relative hip angle of right leg)
ylim([-100 -40]) % y-limit of this plot

yyaxis right % plotting y axis towards right
plot(Lx1,Lyrh); % plot Lx1(time) vs Lyrh(relative hip angle of left
leg)
ylim([-100 -40]) % y-limit of this plot

line([xx1 xx1], [yy1(1) yy1(end)], 'color', 'black'); % line which flow
from                                                          left to
right



subplot(5,2,4) % subplot of 4th box with 5 row, 2 columns
xlabel('time') % x-label
ylabel('angle') % y-label
title('Relative Knee-Angle') % title of the subplot
```

```
yyaxis left % plotting y axis towards left
plot(x1,rrk); % plot x1(time) vs rrk(relative knee angle of right leg)
ylim([-150 -40]) % y-limit of this plot

yyaxis right % plotting y axis towards right
plot(x1,Lrk); % plot Lx1(time) vs Lrk(relative knee angle of left leg)
ylim([-150 -40]) % y-limit of this plot

line([xx1 xx1], [yy1(2) yy2(end)], 'color', 'black'); % line which flow
from                                                                left to
right



subplot(5,2,6) % subplot of 6th box with 5 row, 2 columns
xlabel('time') % x-label
ylabel('angle') % y-label
title('Relative Ankle-Angle') % title of the subplot

yyaxis left % plotting y axis towards left
plot(x1,rra); % plot x1(time) vs rra(relative ankle angle of right leg)
ylim([-80 80]) % y-limit of this plot

yyaxis right % plotting y axis towards right
plot(x1,Lra); % plot Lx1(time) vs Lra(relative ankle angle of left leg)
ylim([-80 80]) % y-limit of this plot

line([xx1 xx1], [yy1(3) yy3(end)], 'color', 'black'); % line which flow
from                                                                left to
right

subplot(5,2,8) % subplot of 8th box with 5 row, 2 columns
xlabel('time') % x-label
ylabel('angle') % y-label
title('Absolute Knee-Angle') % title of the subplot

yyaxis left % plotting y axis towards left
plot(x1,yak); % plot x1(time) vs yak(absolute knee angle of right leg)
ylim([100 200]) % y-limit of this plot

yyaxis right % plotting y axis towards right
plot(Lx1,Lyak); % plot Lx1(time) vs Lyak(absolute knee angle of left
leg)
ylim([100 200]) % y-limit of this plot

line([xx1 xx1], [yy1(4) yy4(end)], 'color', 'black'); % line which flow
from                                                                left to
right
```

```
subplot(5,2,10) % subplot of 10th box with 5 row, 2 columns
xlabel('time')  % x-label
ylabel('angle') % y-label
title('Absolute Ankle-Angle') % title of the subplot

yyaxis left % plotting y axis towards left
plot(x1,yaa); % plot x1(time) vs yaa(absolute ankle angle of right leg)
ylim([75 125]) % y-limit of this plot

yyaxis right % plotting y axis towards right
plot(Lx1,Lyaa); % plot Lx1(time) vs Lyaa(absolute ankle angle of left
leg)
ylim([75 125]) % y-limit of this plot

line([xx1 xx1], [yy1(5) yy5(end)], 'color', 'black'); % line which flow
from                                                              left to
right
```

REFERENCES

[1] Thomas Seel, Jorg Raisch, Thomas Schauer, "IMU-Based Joint Angle Measurement for Gait Analysis," Sensors Journal, volume: 14, number: 4, pages: 6891--6909, year: 2014. [Online]. Available: https://doi.org/10.3390/s140406891.

[2] Bakhshi Khayani, Saba. "Development of Wearable Sensors For Body Joint Angle Measurement.", MS Thesis, University of Denver (2011).

[3] Sunghoon Ivan Lee, Jean-Francois Daneault, Luc Weydert, Paolo Bonato, "A novel flexible wearable sensor for estimating joint-angles," 2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks (BSN), Year: 2016, Pages: 377 – 382.

[4] L. Kun, Y. Inoue, K. Shibata, and C. Enguo, "Ambulatory estimation of knee-joint kinematics in anatomical coordinate system using accelerometers and magnetometers," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 2, pp. 435–442, Feb 2011.

[5] G. Cooper, I. Sheret, L. McMillian, K. Siliverdis, N. Sha, D. Hodgins, L. Kenney, and D. Howard, "Inertial sensor-based knee flexion/extension angle estimation," *Journal of Biomechanics*, vol. 42, no. 16, pp. 2678 – 2685, 2009.

[6] Y. Qi, C. B. Soh, E. Gunawan, K. S. Low, and A. Maskooki, "A novel approach to joint flexion/extension angles measurement based on wearable uwb radios," *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 1, pp. 300–308, Jan 2014.

[7] Y. Qi, C. B. Soh, E. Gunawan, K.-S. Low, and R. Thomas, "Lower extremity joint angle tracking with wireless ultrasonic sensors during a squat exercise," *Sensors*, vol. 15, no. 5, p. 9610, 2015.

[8] L. D. Toffola, S. Patel, M. Y. Ozsecen, R. Ramachandran, and P. Bonato, "A wearable system for long-term monitoring of knee kinematics," in *Proceedings of 2012 IEEE-EMBS International Conference on Biomedical and Health Informatics*, Jan 2012, pp. 188–191.

[9] J. L. Riskowski, "Gait and neuromuscular adaptations after using a feedback-based gait monitoring knee brace," *Gait & Posture*, vol. 32, no. 2, pp. 242 – 247, 2010.

[10] P. T. Wang, C. E. King, A. H. Do, and Z. Nenadic, "A durable, low cost electrogoniometer for dynamic measurement of joint trajectories," *Medical Engineering & Physics*, vol. 33, no. 5, pp. 546 – 552, 2011.

[11] D. Z. Stupar, J. S. Bajic, L. M. Manojlovic, M. P. Slankamenac, A. V. Joza, and M. B. Zivanov, "Wearable low-cost system for human joint movements monitoring based on fiber-optic curvature sensor," *IEEE Sensors Journal*, vol. 12, no. 12, pp. 3424–3431, Dec 2012. 381

[12] A. S. Silva, A. Catarino, M. V. Correia, and O. Fraz0Ł0o, "Design and characterization of a wearable macro bending fiber optic sensor for human joint angle determination," *Optical Engineering*, vol. 52, no. 12, p. 126106, 2013.

[13] Y. Enokibori and K. Mase, "Human joint angle estimation with an e-textile sensor," in *Proceedings of the 2014 ACM International Symposium on Wearable Computers*, ser. ISWC ' 14. New York, NY, USA: ACM, 2014, pp. 129–130.

[14] P. T. Gibbs and H. Asada, "Wearable conductive fiber sensors for multi-axis human joint angle measurements," *Journal of NeuroEngineering and Rehabilitation*, vol. 2, no. 1, pp. 1–18, 2005.

[15] Y. Menguc, Y.-L. Park, H. Pei, D. Vogt, P. M. Aubin, E. Winchell, L. Fluke, L. Stirling, R. J. Wood, and C. J. Walsh, "Wearable soft sensing suit for human gait measurement," *The International Journal of Robotics Research*, 2014.

[16] H. O. Michaud, J. Teixidor, and S. P. Lacour, "Soft flexion sensors integrating stretchable metal conductors on a silicone substrate for smart glove applications," in *2015 28th IEEE International Conference on Micro Electro Mechanical Systems (MEMS)*, Jan 2015, pp. 760–763.

[17] H. Zheng, N. D. Black, and N. D. Harris, "Position-sensing technologies for movement analysis in stroke rehabilitation," *Medical and Biological Engineering and Computing*, vol. 43, no. 4, pp. 413– 420, 2005. [Online]. Available: http://dx.doi.org/10.1007/BF02344720.

[18] https://www.idgconnect.com/blog-abstract/13524/perspective-how-injury-helped-develop-tech-solution.

[19] Jeroen H. M. Bergmann, Salzitsa Anastasova-Ivanova, Irina Spulber; Vivek Gulati, Pantelis Georgiou, and Alison McGregor, "An Attachable Clothing Sensor System for Measuring Knee Joint Angles," IEEE Sensors Journal, Year: 2013, Volume: 13, Issue: 10 Pages: 4090 – 4097.

[20] Castellano, Julen, and David Casamichana. "Heart Rate and Motion Analysis by GPS in Beach Soccer." Journal of Sports Science & Medicine 9.1 (2010): 98–103. Print.

[21] Roland Leser, Arnold Baca, and Georg Ogris, “Local Positioning Systems in (Game) Sports,” Sensors Journal, volume: 11, number: 10, pages: 9778--9797, year: 2011. [Online]. Available: https://doi.org/10.3390/s111009778.

[22] Brown, T. D., Vescovi, J. D., and VanHeest, J. L. “Assessment of Linear Sprinting Performance: A Theoretical Paradigm,” *Journal of Sports Science & Medicine*, *3*(4), 203–210, 2004.

[23] Majumdar, A.S., R A. Robergs “The Science of Speed: Determinants of Performance in the 100 m Sprint,” International Journal of Sports Science & Coaching, Volume: 6, Number: 3, 479, 2011.

[24] Boden BP, J.S.Torg, S.B. Knowles , Hewett T. E. “Video Analysis of Anterior Cruciate Ligament Injury: Abnormalities in Hip and Ankle Kinematics,” Am J Sports Med. 2009 Feb;37(2):252-9.

.

BIOGRAPHY

Venkata Naga Sai Chaitanya Neelamraju is a candidate for Master of Science in Computer Engineering, with a specialization in VLSI, Embedded Systems and Hardware Security from George Mason University. He has been a member of Dr. Lance Sherry's research lab, Center for Air Transportation and Systems Research (CATSR) Laboratory since June 2017 and also a member of Dr. Kris Gaj's Cryptographic Engineering Research Group (CERG). He received his Bachelor of Technology in Electronics and Communication Engineering from the Jawaharlal Nehru Technological University, Hyderabad, India, in 2016.