IoT Based 3D Motion Model

Project Team

Mohammad Shaoor Fahim	P16-6102
Muneeb Muzammal	P16-6093
Hamza Amir	P16-6142

Session 2016-2021

Supervised by

Dr. Omar Usman Khan

Department of Computer Science

National University of Computer and Emerging Sciences
Peshawar, Pakistan

January, 2021

Student's Declaration

We declare that this project titled "*IoT Based 3D Motion Model*", submitted as requirement for the award of degree of Bachelors in Computer Science, does not contain any material previously submitted for a degree in any university; and that to the best of our knowledge, it does not contain any materials previously published or written by another person except where due reference is made in the text.

We understand that the management of Department of Computer Science, National University of Computer and Emerging Sciences, has a zero tolerance policy towards plagiarism. Therefore, We, as authors of the above-mentioned thesis, solemnly declare that no portion of our thesis has been plagiarized and any material used in the thesis from other sources is properly referenced.

We further understand that if we are found guilty of any form of plagiarism in the thesis work even after graduation, the University reserves the right to revoke our BS degree.

Mohammad Shaoor Fahim	Signature: ________________
Muneeb Muzammal	Signature: ________________
Hamza Amir	Signature: ________________

____

Verified by Plagiarism Cell Officer
Dated:

Certificate of Approval

The Department of Computer Science, National University of Computer and Emerging Sciences, accepts this thesis titled *IoT Based 3D Motion Model*, submitted by Mohammad Shaoor Fahim (P16-6102), Muneeb Muzammal (P16-6093), and Hamza Amir (P16-6142), in its current form, and it is satisfying the dissertation requirements for the award of Bachelors Degree in Computer Science.

Supervisor

Dr. Omar Usman Khan Signature: ____________________

Mashal Khan

FYP Coordinator
National University of Computer and Emerging Sciences, Peshawar

Dr. Hafeez ur Rehman

HoD of Department of Computer Science
National University of Computer and Emerging Sciences

Acknowledgements

We are very thankful to Almighty Allah, who is most Merciful and most Beneficent for giving us strength and courage to choose such challenging project and for completing Final Year Project-I Report with great dignity and grace. However, it would not have been possible without the kind support and help of many individuals. We would like to extend our sincere thanks to all of them. We would like to express deep gratitude to our project supervisor Dr. Omar Usman khan for their patient guidance, enthusiastic encouragement and useful critiques of this research work. We are grateful to the final year project (FYP) committee for extending their cooperation to us and helping us out in completion of our project. We would also like to thank our family and friends for their support and help throughout our academic career. We are highly indebted to FAST NUCES university Peshawar for providing us a platform to showcase our work.

Mohammad Shaoor Fahim

Muneeb Muzammal

Hamza Amir

Abstract

The aim of this thesis is to develop and deploy a 3D Model of a human which captures the movements of a human host in real time, as the movements of a human body carry a tremendous amount of meaningful data the applications of our project are quite diverse. The approach used was a wireless inertial sensing technology consisting of 15 sensors placed on the different limbs of the human body which would provide us with the essential readings. However, Movement status solution is still the problem that prevents further advancements of 3D motion modelling systems. As per the features of a 3D human body model is proposed. Our project aims to create a framework for further research on motion capture applications, which would be useful in the fields of Medical, Improvement in Robotics and Gait Analysis. The techniques we have used are Rotations of the X, Y and Z axis, namely the Roll, Pitch and Yaw values in the form of Euler angles along our path to create the system we discovered that these values include an ever-increasing error that is caused by the earth's gravitational pull and the gyroscope bias drift, so we worked on applying a different methodology to remove these errors and came to use the Madgwick's filter which would eliminate these errors and produce an accurate real time representation. lastly, all of the performance of the 3D motion model is to be verified by a systematic inspection of the system by scanning continuous dynamic movements.

Contents

List of Figures

Chapter 1

Preliminaries and Introduction

1.1 Introduction

In the field of computer science, 3D Motion rendering and monitoring technology has become quite popular; motion capture technology has a wide range of applications in fields such as sports, television and film production, sports training and medicinal science. In the department of game development, Other games may include sports, for example: "Cricket" as well as "Football (FIFA)," take use of the motion of athletes and pair it with virtual effects, so the outcome is a clear, accurate and vivid display. In the field of movies, many movies, for example: “The Avengers” and “kingkong,” mainly use motion pictures and systematic effects using a green screen. The department of medical sciences, these methodologies are primarily used for orthopedics, surgery, therapy of damaged limbs and ligaments, various fields to help physicians understand the disabilities of patients by comparing the mobility of the patient before and after the recommended treatments. The department of athletic training, live tracking of athletic training is possible by displaying real-time tracking interface for athletes who wear the sensors on their body in the form of a suit or just the wireless sensors.

In our study, we found that many of the challenges faced by researchers that were fixated on algorithms and the effectiveness of live time systems. The only exception is the popular companies, for example: “3dSuit” and “Xsens”, have achieved the most accurate 3D motion modelling renders. Dependent upon self-regulation and various organizational problems, the main crux of the algorithm is still unknown and remains a major problem not only affecting the accuracy of the motion detection device and real-time structures. Based on the already present algorithms, these processes depict the in-depth sensory information and establish an algorithmic model that drives data from sensors to human models.To establish the accurate and effective communication between model and person. The real-time structures of the program are also significantly better than the predecessors to many of the basic system.

1.2 Objective

The prime objective of the project is to develop an integrated IoT based system that helps us in the real time motion and construction of a 3D model of the human figure. The system is first calibrated and the raw values of the accelerometer,gyroscope and magnetometer. Moreover, converting that raw data into meaningful Euler angles in the form of roll, pitch and yaw rotations is going to be key. Some of the major objectives to proceed the beginning are given below:

- The Fabrication of IoT devices.
- Massive amount of data collection from sensors with the help of IoT devices.
- Real time visualization of human body motion.
- Wireless data transfer.
- Accuracy of data and coordinance with the 3D object.

The objective of this project is to develop a general framework of a 3D model which can be used by future applications that may include work in the games, films, sports industry and medical science.

1.3 Scope

This project is unique as it can branch into any field such as, medical sciences, film industry, sports industry and the gaming world. In the Medical sciences department, it can be used to monitor the contraction of muscles and then use that data to correct the errors in the posture and the way a particular exercise is done which focuses on a particular muscle group at a time. In the films industry it can be used to alter the shape of the human model into a creature of sorts and be used in an animated film or for CGI. In the sports industry the movement of a cricketer can be recorded and patterns can be followed as guidance from a professional. In the gaming industry it can be used to map the movement of a character and with the integration of virtual reality it is a completely separate genre.

1.4 Idea of the Project

The main idea of the project is to use the trending technology that is IoT (Internet of Things) and combine it with computer graphics to create a visual effect of the body so that it nay be easier to analyse. So, what we have done is to develop a basic framework on which future work can be done to help all of our possible applications into fruition.

1.5 SWOT Analysis

1.5.1 Strengths

- Our product is versatile.
- Enhance performance of humans.
- Increased market demand.
- Highly accurate and self correcting model.
- 3D Motion trackers to integrate real time applications.

1.5.2 Weaknesses

- Less Researched on field.
- Expensive Hardware.
- The Market contains Laymen.
- Hardware Failure

1.5.3 Opportunities

- Motion capture for Sports, Research and Ergonomics.
- No national level competitor.
- Paradigm shift towards IT may benefit our project.
- We can take advantage of government projects.

1.5.4 Threats

- Magnetic field effect.
- Day by day improvements.
- Budget Restrictions.
- International Level Competitors.

Chapter 2

Review of Literature.

This chapter provides a comprehensive survey of the similar work to IoT based 3D motion model. In this chapter, we study the work done on this project and to what extent previously work has been done. For instance, we have studied different research papers and below listed the work done by them one by one.

2.1 Orientation Detection using an Accelerometer

2.1.1 Features

- The difference between any linear acceleration in the and the earth's gravitational field vector is measured by the accelerometer. the absence of a corresponding acceleration, the accelerometer effect is measured by the vector magnetic field and can be used to determine the height of the accelerometer and the direction of the model roll.

- The output of the device is dependent upon the way in which the rotations of the device are applied.

- Accelerometers are not sensitive to the rotation along the earth's gravitational field. An alternate solution is presented to prevent this instability occurring.

- We use algebraic expressions to compute the tilt of the accelerometer from vertical or the rotation angle between any two readings of the accelerometer.

  Accelerometers are sensitive to the difference between the acceleration of the sensor line and the local magnetic field. The details of any accelerometer will point to the x, y, and z axes in the sensor package and, accordingly, this is explained so that the acceleration of the straight lines of these axes will give the best accelerometer effect.

2.1.2 Roll and Pitch angle estimation

The position of a 3D object can be described by its roll, pitch and yaw rotations from an initial frame of reference. The roll, pitch and yaw rotation matrices are the ones, which transform a vector (such as the earth's gravitational field vector g) under a rotation of the coordinate system by angles in the roll, pitch and yaw angles about the x, y and z axes respectively, are:

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi & sin\phi \\ 0 & -sin\phi & cos\phi \end{bmatrix}$$

$$R_y(\theta) = \begin{bmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{bmatrix}$$

$$R_z(\psi) = \begin{bmatrix} cos\psi & sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

These are the equations used to calculate the Roll, Pitch and Yaw angles respectively.

2.2 Madgwick's filter (MARG) working

The Magnetic Angular and Rate Gravity (MARG) implementation used the distortion in the magnetic field and the gyroscope bias drift compensation to eliminate errors. The filter uses a different than the usual representation in quaternion form, which allows the magnetometer and accelerometer data to be used in a mathematically derived and optimized form of a gradient-descent algorithm which enables us to compute the direction of the gyroscope measurement error as a derivative of a quaternion.

2.3 Wireless Motion Capture on a Human in Real-Time

In this paper, from the movement of the skeletal muscles of the human body, they develop a new way of calculating an attitude based on inertial sensing theory. They also confirmed the method of calculating the movement of the limbs. After that, they developed a 3D human motion model which uses an artificial modeling model based on computer simulations. Finally, they investigated the operation of the system by taking photographs of the effect of motion capture with a high-speed camera. Test results show that the scheme can achieve real-time tracking of the 3D model of the human body.

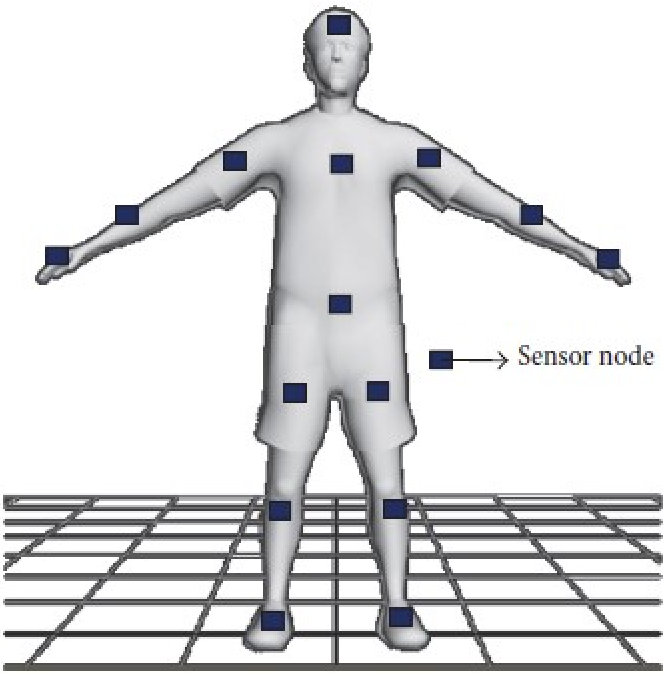

Figure 2.1: Main body structure

Chapter 3

IoT and Architecture

In order to proceed, we have done some literature review in which we studied different previously research works related to IoT (Internet of Things) to have an exact view of the industry requirements and upcoming advancements using IoT. We focused on the 3D graphic model based upon IoT and its importance in the modern world. After a thorough study and keeping in mind the previous work done on IoT based Motion capture devices, we came up with an idea that a proper real time monitoring system that can be developed with the help of IoT devices, which includes a 9-Axis Accelerometer, NodeMCU and a Lithium ion battery. For Instance, by the integration of IoT devices, we can get some data in the form of X,Y and Z coordinates. Experimentally, getting the big data from the devices that are required for a much more refined and well performing device. Therefore, we started with integration of devices by configuring the micro controllers to get some data.

3.1 What is IoT (Internet of Things)?

Internet of Things is defined as the association of items or gadgets to the web. An organization of web associated objects or physical devices to gather and share information via wireless technology is called as Internet of Things.
The Internet of Things is the idea of interfacing any gadget (For instance, it has an on/off change) to the Internet and to other associated devices. The IoT is a monster

organization of associated things and individuals – all of which gather and offer information about the manner in which they are utilized and about nature around them.
The Internet of Things is really a straightforward idea, it implies taking all the things on the planet and interfacing them to the web.

3.1.1 How IoT works?

When something is associated with the web that implies that it can send data or get data, or both. This capacity to send and additionally get data makes things smart, and that is good.

In the Internet of Things (IoT) devices can be categorised into three classifications:

- Devices that gather information and afterward send it.
- Devices that get information and afterward follow up on it.
- And both, that send as well as receive data.

3.2 IoT Devices

In order to develop a smart water system, it is required to have a thorough look at the 9-Axis Accelerometer (MPU-9250) and Node MCU esp-8266 and their working mechanism.

3.2.1 NodeMCU ESP 8266

Node Microcontroller Unit is an IoT platform that is open source. It is a type of microcontroller which has fixed a module of Wi-Fi in them. It is a bite-sized Wi-Fi enabled microcontroller. It is a lua based firmware. It has a flash memory to store code. It has 4MB of flash memory whereas Arduino UNO has only of 32kb. It is

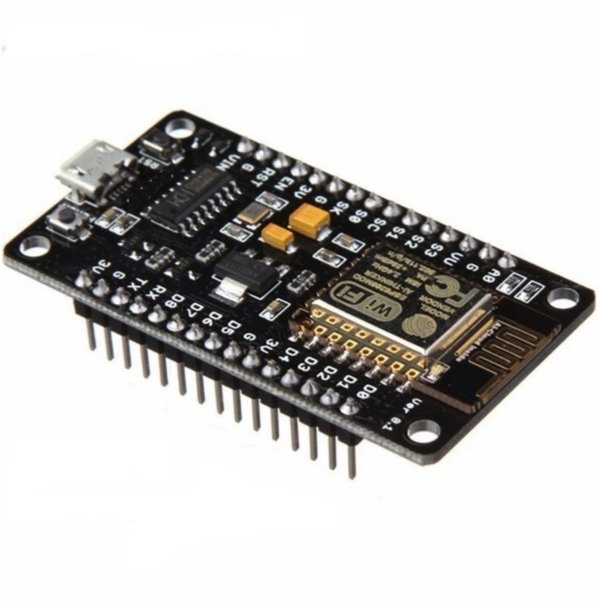

Figure 3.1: Node Microcontroller Unit

faster than UNO because of USB port and it is also smaller in size than Arduino UNO.

3.2.2 Components

- Esp-12e

  It is a WI-FI module present in a Node MCU used for establishing a wireless network connection for microcontroller. It also have 128kb Random Access Memory and 4MB of memory for the code to be stored and sent to server or cloud. It not only connect to Wi-Fi itself but also allow other devices to connect to it and act as a Wi-Fi hotspot which make more versatile.

- Serial Communication

  It has CP2102 USB-to UART Bridge controller to transform serial signals from USB signals and allow PC to code using Arduino IDE and transfer data to ESP8266 Wi-Fi module.

- Voltage

  As the working voltage scope of ESP8266 is 3V to 3.6V, the board accompanies a LDO voltage controller to keep the voltage consistent at 3.3V. 3V3 is the pin used for giving the voltage to the micro controller and its internal parts.

3.2.3 ESP8266 NodeMCU Pin out

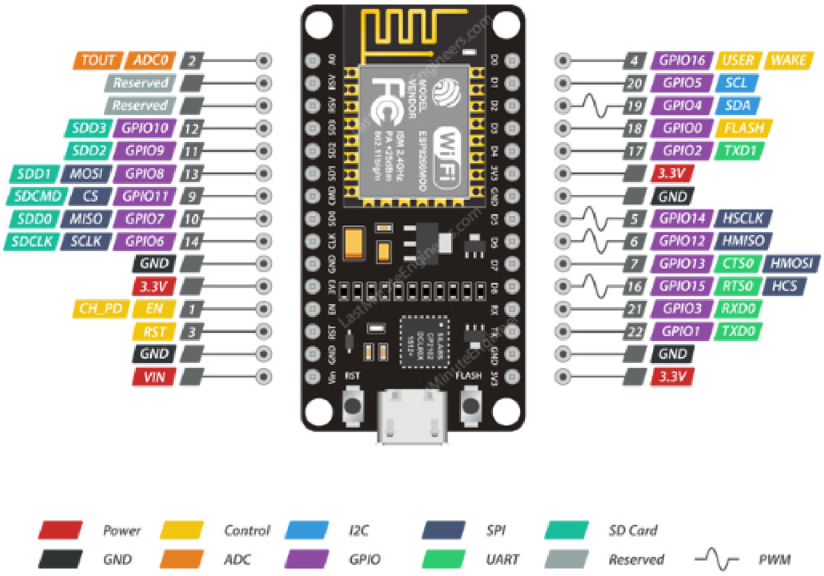

Figure 3.2: Node MicrocontrollerPinout

3.2.4 ESP8266 Development Platforms

Moving on to the interesting stuff that is the development platform. An operating system for IoT devices makes it easy to write code and transfer it to the board. Works on Windows, Mac OS X, and Linux. The nature of the program is Java-based and depends on Processing and other open source programs. This product can be used with any Arduino board.

3.2.5 ESP8266 Operating Modes

– Station Mode (STA)

– Soft Access Point Mode (AP)

3.3 9-Axis Accelerometer MPU-9250

Sensors play an important role in the automation of systems nowadays. Sensors are little, ease and solid gadget, and are anything but difficult to install with bigger hardware.The MPU-9250 is a multi-chip (MCM) module with two themes integrated into a single QFN package. The mortal man sits on a 3-Axis gyroscope and a 3-Axis accelerometer. Some of the dead have an AK8963 3-Axis magnetometer from Asahi Kasei Microdevices Corporation. Therefore, the MPU-9250 is a 9-axis Motion Tracking tracking device that includes a 3-axis gyroscope, 3-axis accelerometer, 3-axis magnetometer and Digital Motion Processor TM (DMP) all in a small package -3x3x1mm available as pin-compatible upgrade from MPU6515.

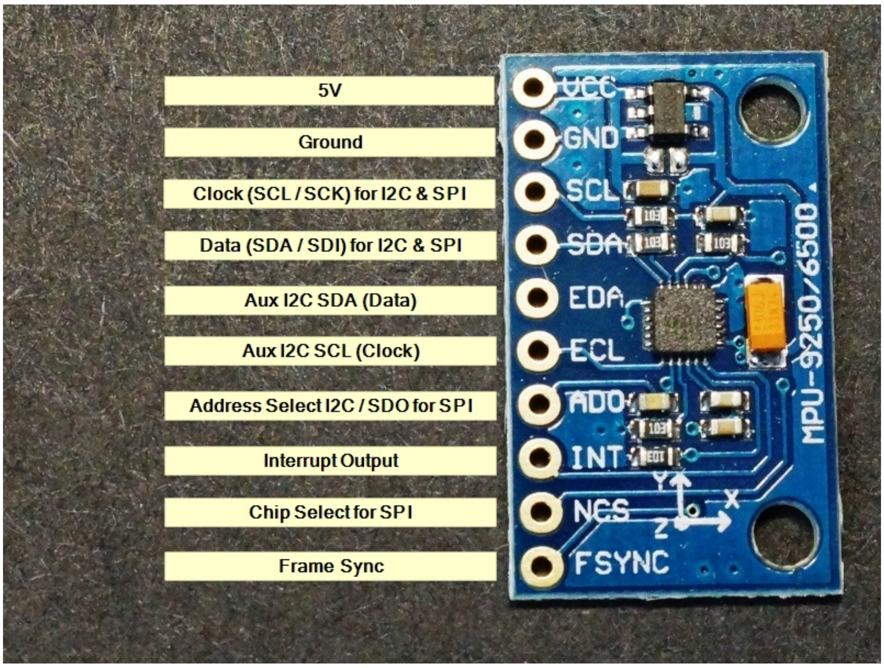

Figure 3.3: 9-Axis Accelerometer MPU-9250

3.3.1 Accelerometer Features

The MPU-9250 includes a triple-axis accelerometer which has a range of features:

- A digital emitting accelerometer with an axis with a digital range with a full set distance of 2g, 4g, 8g and 16g with 16-bit ADCs combined.
- Current Accelerometer for normal operation: 450A.
- Current low power accelerometer mode: 8.4A at 0.98Hz, 19.8A at 31.25Hz.
- Current sleep mode: 8A.
- Easy-to-use instructions.
- Wake-on-motion disrupts the low power of the application processor.
- Self-Examination.

3.3.2 Gyroscope Features

The triple gyroscope on the MPU-9250 features a variety of features:

- Digital output X-, Y-, and Z-Axis angular rate sensors (gyroscopes) with range 250, 500, 1000, and 2000 / sec and ADCs 16-bit.
- Low-pass filter which is programmable digitally.
- Gyroscope currently operates: 3.2mA.
- Current sleep mode: 8A.
- Sensitivity scale factor calibrated by the factory.
- Self-Examination.

3.3.3 Magnetometer Features

The triple-axis Magnetometer in the MPU-9250 features a wide range of features:

- 3-axis silicon monolithic Hall-effect magnetic sensor with magnetic concentrator.
- Powerful scalability and high precision with low current consumption.
- 14-bit data output correction (0.6T / LSB).
- The full range distance is 4800T.

- Magnetometer current for normal operation: 280A with a frequency of 8Hz.
- The function of self-examination of the internal magnetic field is to ensure the effectiveness of the magnetic sensor in the final products.

Chapter 4

Implementation

4.1 Background

Now, before coming towards the implementation we have enough knowledge of our basic project, the architecture required (9-Axis Accelerometer and the NodeMCU) and how to integrate the devices. Means we have studied a number of research papers and obviously surf the internet for some material to start the implementation. Firstly, we have to install the 9-Axis Accelerometer with NodeMCU coded via Arduino IDE and get some data by the basic configuration techniques. The data will be monitored on Serial Monitor and then on Web based application. We managed to send the data directly to the database and later on, perform some analytics on the generated data.

4.2 Interfacing a NodeMCU with an Accelerometer

Before we start, it’s very important to know about the proper working mechanism of the sensor.

4.3 Working Mechanism of an Accelerometer

An accelerometer is a device that measures the vibration, or acceleration of a building's motion. The force created by vibrations or shifts in motion (acceleration) causes the mass to "compress" the accelerometer's equipment which charges an amount of electricity equal to the energy produced. Since the charge is equal to the force, and the magnitude remains constant, then the charge is also equal to the speed.

4.4 Connecting the Accelerometer to NodeMCU

The association required for this Accelerometer with the NodeMCU is exceptionally simple. There are just four wires originating from the Accelerometer. The 3.3V VCC (Orange wire), the Ground (Black wire), the SCL(Serial Clock) (Cyan wire) and the SDA(Serial Data) (Brown wire). Associate the VCC and Ground of the Accelerometer to the NodeMCU's VCC and Ground. SCL (Cyan wire) of the Accelerometer is associated with the NodeMCU's advanced pin D6 and the SDA (Brown wire) is associated with the NodeMCU's advanced D7 pin. These pins are responsible for serial data communication.This is how we have managed our Accelerometer with the Node MCU ESP8266.

4.5 Uploading Code to the Node MCU for measurement of Accelerometer data

For programming, the Accelerometer is coded via Arduino IDE. The programming language is very similar to C language, it is known as even driven programming. The D6 that is a GPIO(General Purpose Input Output) pin used for the serial clock and the D7 pin is used to transfer the serial data. It reads the X,Y and Z axis of the

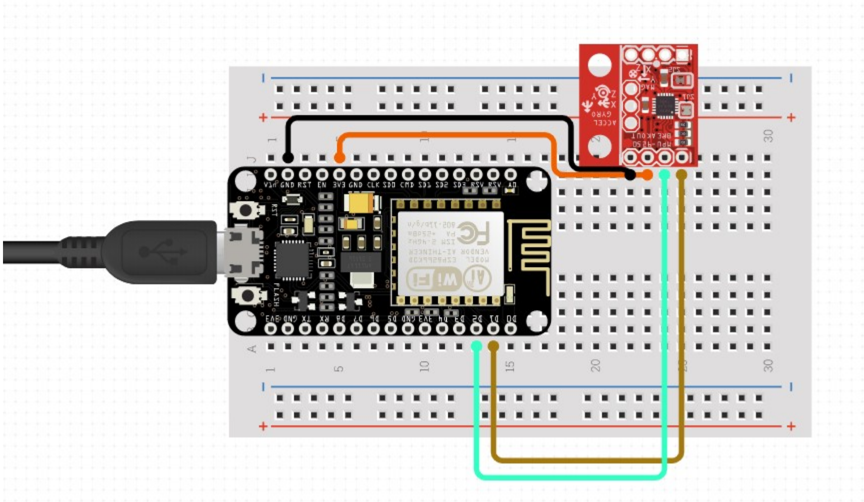

Figure 4.1: Accelerometer with NodeMCU

accelerometer, the gyroscope and magnetometer data emitting out from the sensor MPU-9250.After the data is detected by the device, the data is normalized using a scaling and sensitivity factor according to our needs. In this way, we will get an optimal result.

4.6 Node MCU ESP8266 as HTTP Server (Access Point Mode)

After uploading code, 9600 should be the value set for baud rate. By pressing the RESET button on the wifi module will end up by showing the beginning message of HTTP server. All of the values are monitored on the Serial Monitor. Then, find the device that you can join to the Wi-Fi network, that device may be your laptop, Mobile phone or tabet. Then search for a network named as Node MicroController Unit. After joining to your NodeMCU access point network, set up a browser and guide it to IP address. Web page will be served by Arduino showing current status of the network. The web page shows the number of readings, flow rate, date, time, location and the number of devices.

4.7 Experimentation

Experiments have been performed on the hardware model by associating the data to a 3D planar object. Firstly, we install only a single device to see the overall behaviour and then later on, we have increased the number of sensors from 1 to 3 and 3 to more. Finally, a model has been completed with 15 devices and complete experimentation has been performed with keeping a close eye on every single device from the body to the render.

4.7.1 Case I: Using a single device

While using a single device, we have noticed that the 3D object is accumulating a gyroscope bias drift and magnetic field distortion along the Z axis as time passes by the error grows.To remove this error we tended towards a different approach, an algorithm that removes these errors as they are generated, this algorithm is known as the Madgwick's filter.

4.7.2 Case II: Using multiple devices

In this case, we have managed to install 3 devices and perform some experimentation in order to collect values for analysis. Here, we have found out that the limbs have some free range of motion and the complete range of motion may not be possible and equal for the body parts, hence introducing an angle limit between the limbs.

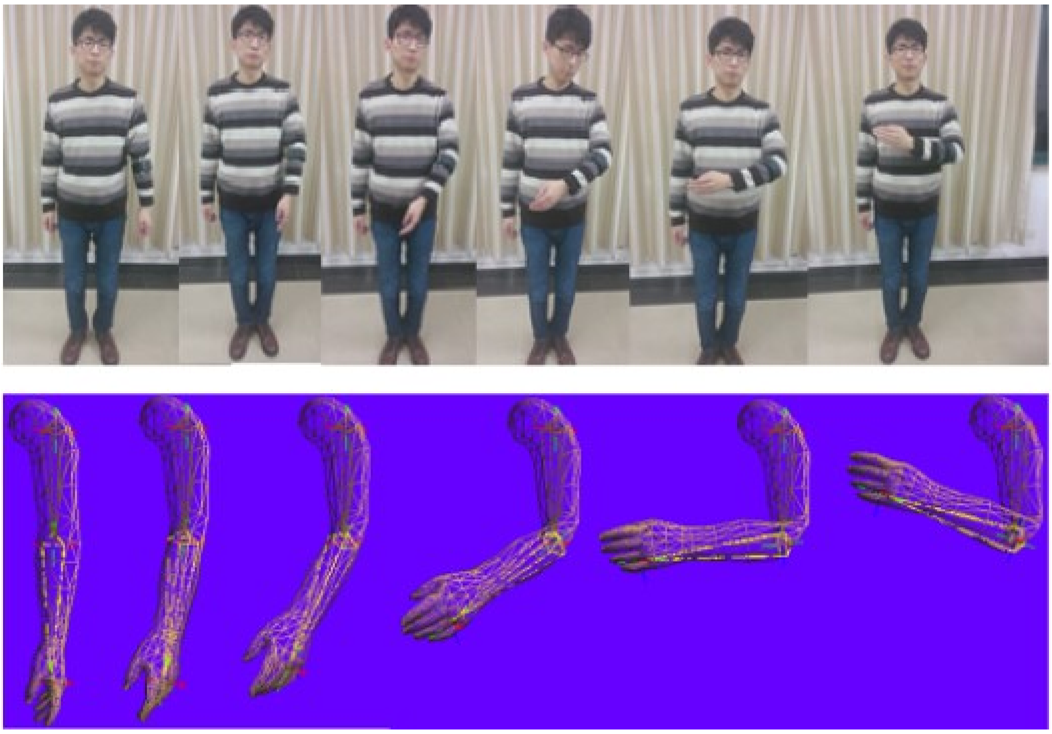

Figure 4.2: 3 devices replicating the movement of an arm

4.7.3 Hardware Model for Experimentation

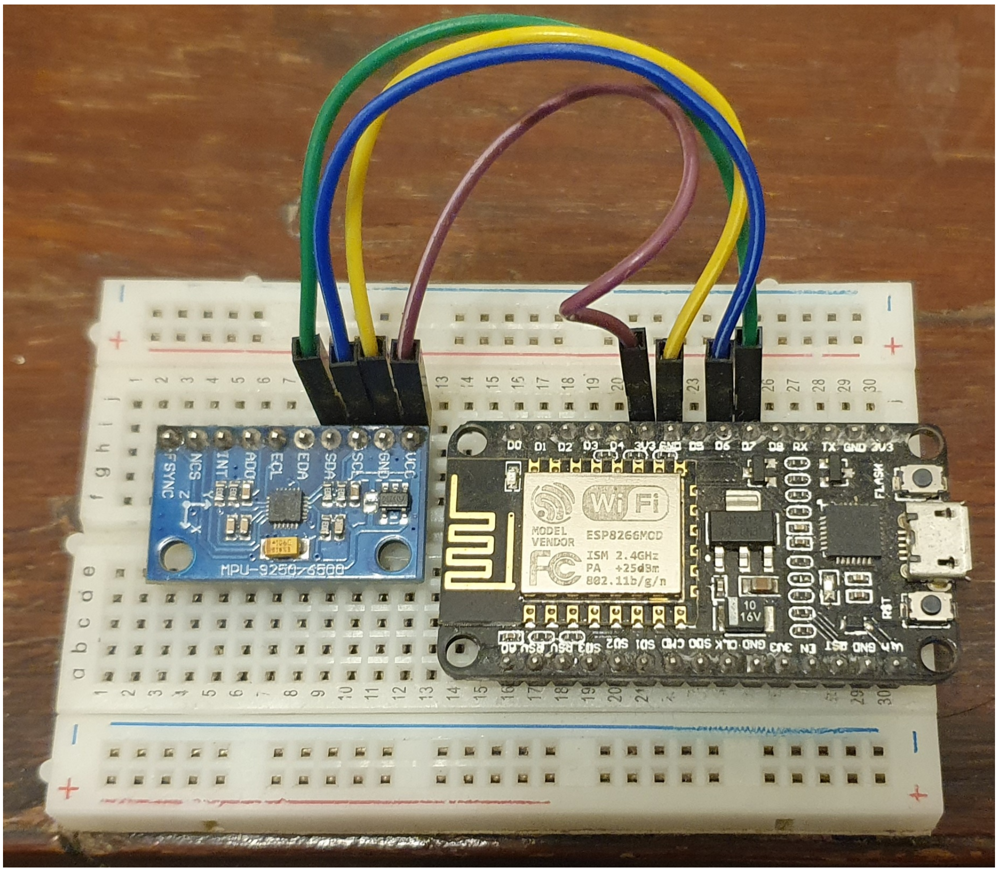

Figure 4.3: Hardware Model

Chapter 5

System Analysis and Design

To start the implementation, first we have to design the use case diagram to show the usefulness of our framework showing use cases of the system and the actors who perform the actions. The Block diagram is the second step for the design, in which the prime functions are portrayed by the squares name as blocks associated by lines that show the connections between them.

5.1 Use Case Diagram

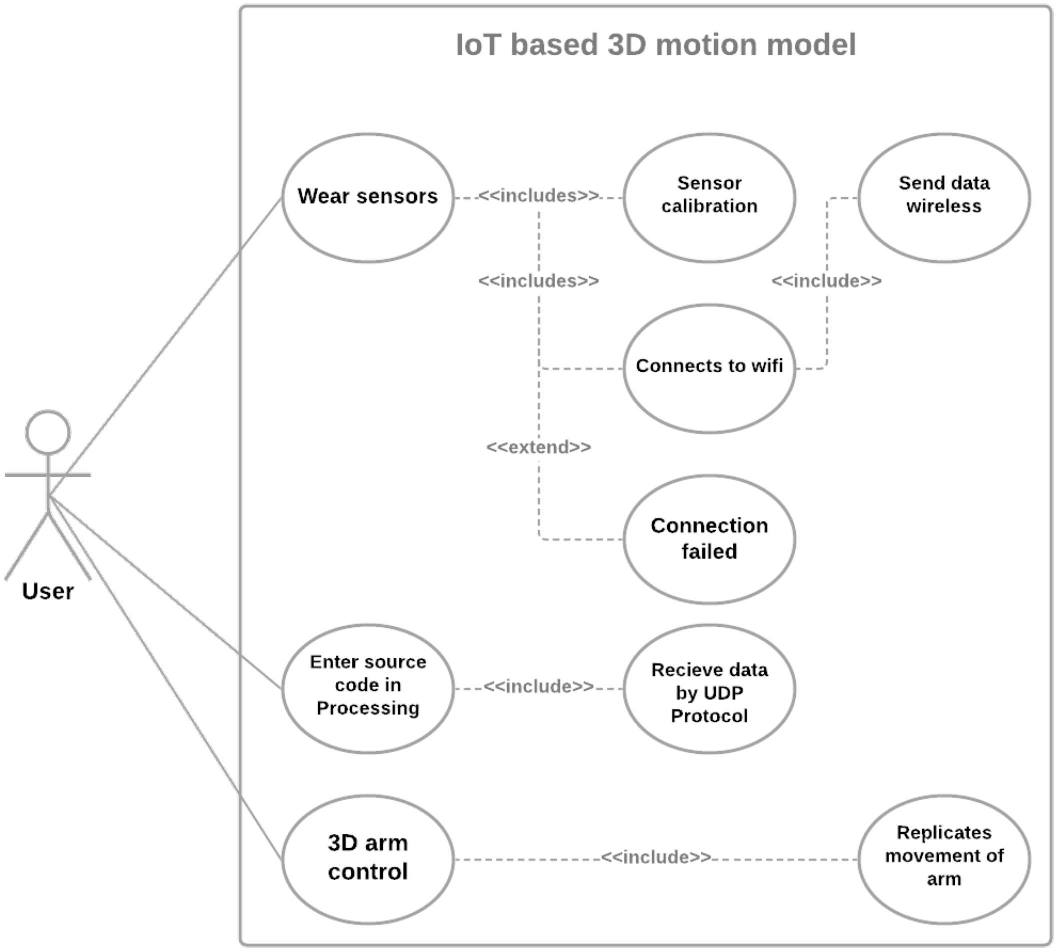

Figure 5.1: Use Case Diagram of IoT Based 3D Motion Model

5.2 Block Diagram

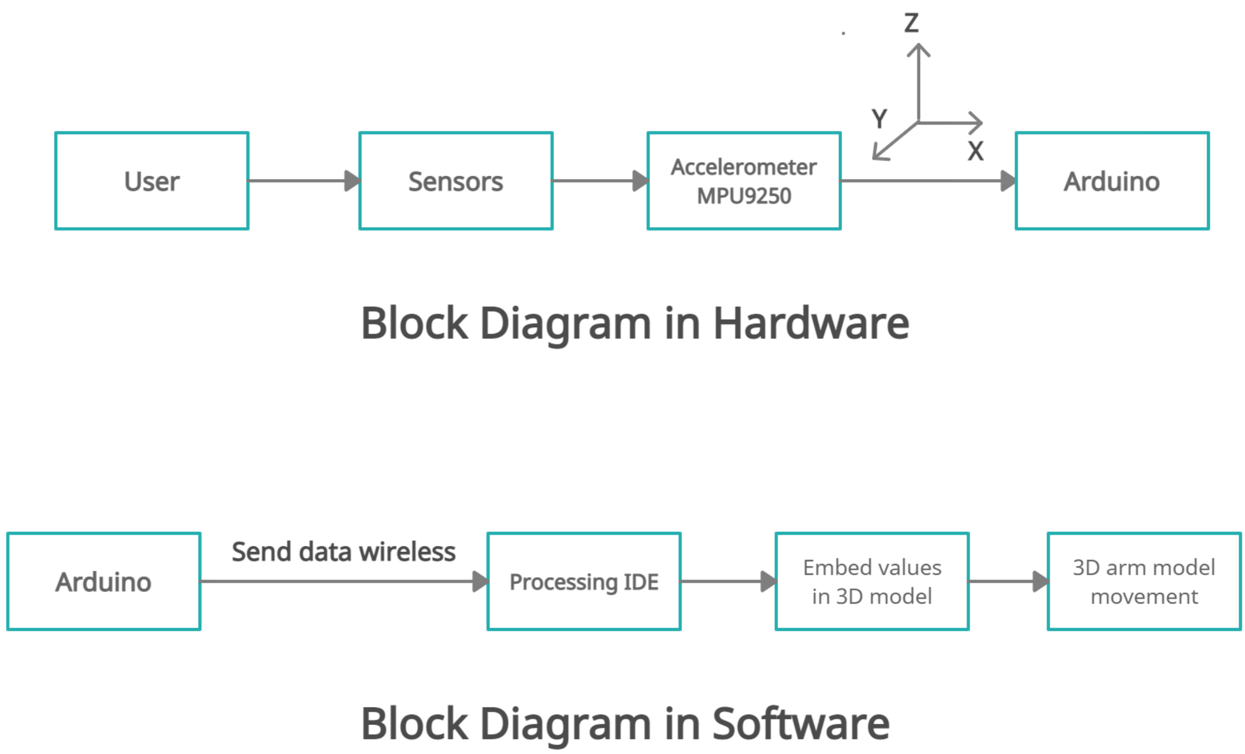

Figure 5.2: Block Diagram

Chapter 6

Implementation of Algorithm

6.1 Introduction

We have made a 3D Render of a planar object to depict the movement of a limb. Using IoT, we collected data from sensors through the Wifi module of NodeMCU ESP8266. Collected Real time values at the serial monitor and then visualize all the data on a Processing development environment which uses OpenGL libraries for better monitoring . It's like a limb (arm) which shows the user, change in the position of the limb as we move the limb.

6.2 Orientation Determination Algorithm

To determine the orientation of the device we need some quantifiable values and those values come in the form of angles, namely the angles would be along the X axis which would be Roll, the Y axis which would be pitch and the Z axis which would be Yaw. These angles help us in determining the change in the axis and hence, the change in the orientation of our model. Angles roll, pitch and yaw about the x, y and z axes respectively, are:

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi & sin\phi \\ 0 & -sin\phi & cos\phi \end{bmatrix}$$

$$R_y(\theta) = \begin{bmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{bmatrix}$$

$$R_z(\psi) = \begin{bmatrix} cos\psi & sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

6.2.1 Roll and Pitch Angles:

The simplified version of Roll becomes:

$$\tan\theta_{xyz} = \left(\frac{-G_{px}}{G_{py}\sin\phi + G_{pz}\cos\phi}\right) = \frac{-G_{px}}{\sqrt{G_{py}^2 + G_{pz}^2}} \tag{6.1}$$

The simplified version of Pitch becomes:

$$\tan\phi_{yxz} = \frac{G_{py}}{\sqrt{G_{px}^2 + G_{pz}^2}} \tag{6.2}$$

6.2.2 Regions of Instability

The complications that arise with the calculation of these Roll and Pitch angles are that they become unstable.

This happens due to the 3D planar object being aligned with the x-axis pointing vertically upwards or downwards. Even if the 3D object is not exactly vertical, the inverse tangent calculation is mainly controlled by the accelerometer noise in the numerator and the denominator of Equation 6.1 which produces and essentially random and unstable estimate of the Roll angle.

6.2.3 Solution

The solution to this problem is simple in theory, but we need to eliminate the gyroscope drift and the error caused by the earth's magnetic field, hence we applied a different formula to eliminate these errors which is the Madwick's filter.

6.3 Madgwick's Filter

The Magnetic Angular and Rate Gravity (MARG) implementation used the distortion in the magnetic field and the gyroscope bias drift compensation to eliminate errors. The filter uses a different than the usual representation in quaternion form, which allows the magnetometer and accelerometer data to be used in a mathematically derived and optimized form of a gradient-descent algorithm which enables us to compute the direction of the gyroscope measurement error as a derivative of a quaternion.

6.3.1 Magnetic distortion compensation

As mentioned before the earth's magnetic field causes distortion in our results and those distortions are due ti the ferromagnetic elements in the surrounding area of the magnetometer. The main culprits of these errors include the following sources:

- Electrical appliances.
- Metal furniture.
- Metal structures within a buildings' construction.

The measured direction of the earth's magnetic-field in the earth frame at time t, can be computed as the normalized magnetometer measurement, and rotated by the estimated orientation of the sensor provided by the filter. The effect of an erroneous inclination of the measured direction earth's magnetic field, can be corrected if the filter's reference direction of the earth's magnetic field, is of the same inclination. This is achieved by computing has to be normalized to have only components in the earth frame x and z axes.

$$^{E}\hat{h}_t = \begin{bmatrix} 0 & h_x & h_y & h_z \end{bmatrix} = {}^{S}_{E}\hat{q}_{est,t-1} \otimes {}^{s}\hat{m}_t \otimes^{S}_{E} \hat{q}^{*}_{\text{eat },t-1} \tag{6.3}$$

$$^{E}\hat{b}_t = \begin{bmatrix} 0 & \sqrt{h_x^2 + h_y^2} & 0 & h_z \end{bmatrix} \tag{6.4}$$

6.3.2 Gyroscope bias drift compensation

As mentioned earlier, the gyroscope zero bias will drift as time passes by, with the temperature and the motion of the object.
The normalised direction of the estimated error in the rate of change of orientation, It may be expressed as the angular error in each gyroscope axis. The gyroscope bias is represented by the DC component and may removed as the integral of weighted by an appropriate gain. This would yield the compensated gyroscope measurements as shown in equations. The first element of is always assumed to be 0.

$$^{S}\omega_{\varepsilon,t} = 2^{S}_{E}\hat{q}^{*}_{est,t-1} \otimes^{S}_{E} \dot{\hat{q}}_{\varepsilon,t} \tag{6.5}$$

$$^{S}\omega_{b,t} = \zeta \sum_{t} {}^{S}\omega_{\varepsilon,t} \Delta t \tag{6.6}$$

$$^{S}\omega_{c,t} = {}^{S}\omega_t - {}^{S}\omega_{b,t} \tag{6.7}$$

Chapter 7

Results

In this chapter, we discuss how our device converts raw data into meaningful data and then uses that data to deduce Roll, Pitch and Yaw values which help us in figuring out the orientation of the device and finally applying the Madgwick's Filter to remove any errors that may accumulate over time.

7.1 Real Time Monitoring System

This is the visual representation of the values of the Roll Pitch and Yaw angles of the senso. The raw values from the accelerometer are taken and go through a sensitivity scale factor, which calibrates the sensor to the sensitivity of our choosing then the orientation determination algorithm is applied so that the errors may be removed.

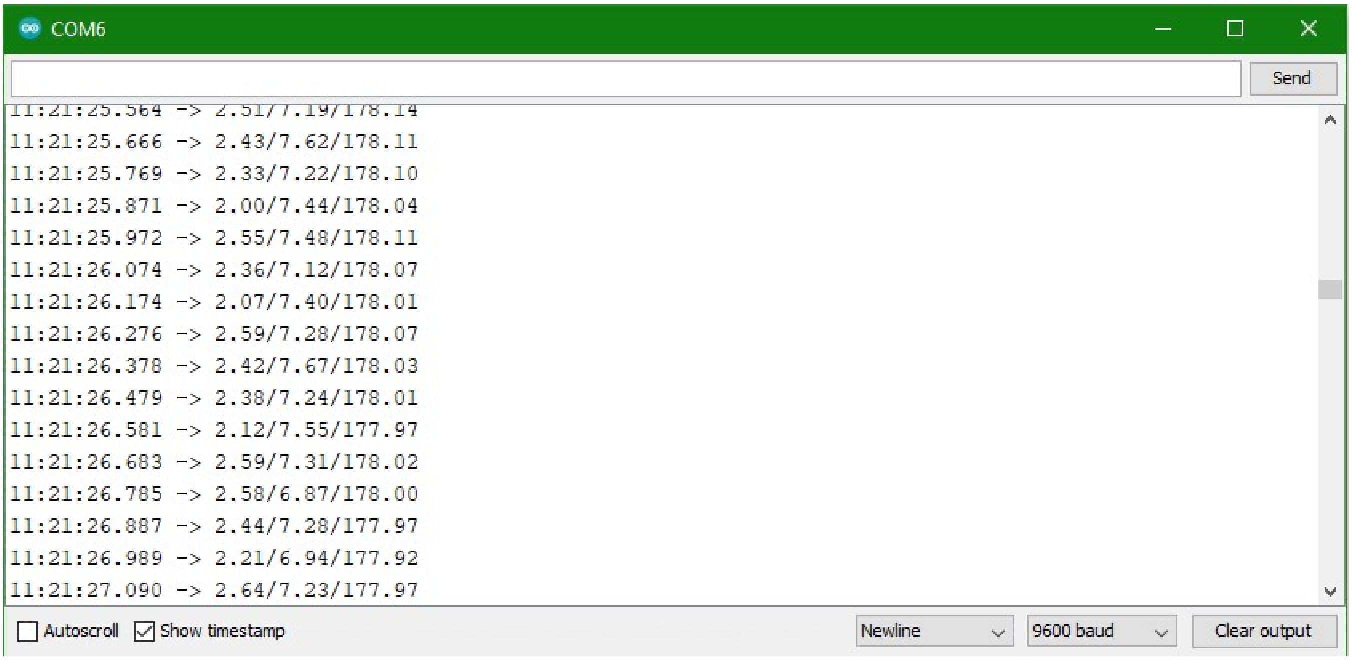

Figure 7.1: Roll, Pitch and Yaw angles on the Serial monitor

When we visualize this data we notice a drift in the gyroscope and interference of the earth's magnetic field which causes distortion in the orientation of our model as time passes, this error accumulates.

7.1.1 NodeMCU ESP8266 as an Access Point

This is the part where we make our device truly wireless by the integrating a lithium ion battery to power the NodeMCU and the Accelerometer. The NodeMCU acts as an access point which connects to the wifi router and begins to transmit its data, all that is needed to check and receive that data is an IP address which is unique for all the devices hence, all the limbs have a separate IP address and it becomes an easy task to deal with each of the devices.

7.1.2 Madgwick's Filter

Lastly, after calculating the orientation of the device and making the device truly wireless we needed to eliminate any errors that may occur in the orentation of our device as time went by. We had already concluded that the errors were due to the gyroscope bias drift which accumulates over time and the earth's gravitational field which interferes with the magnetometer in out accelerometer hardware. To eliminate these errors we found Madgwick's filter to be the optimal way as it can eliminate both of the above-mentioned errors with ease by using already available data from the accelerometer device.

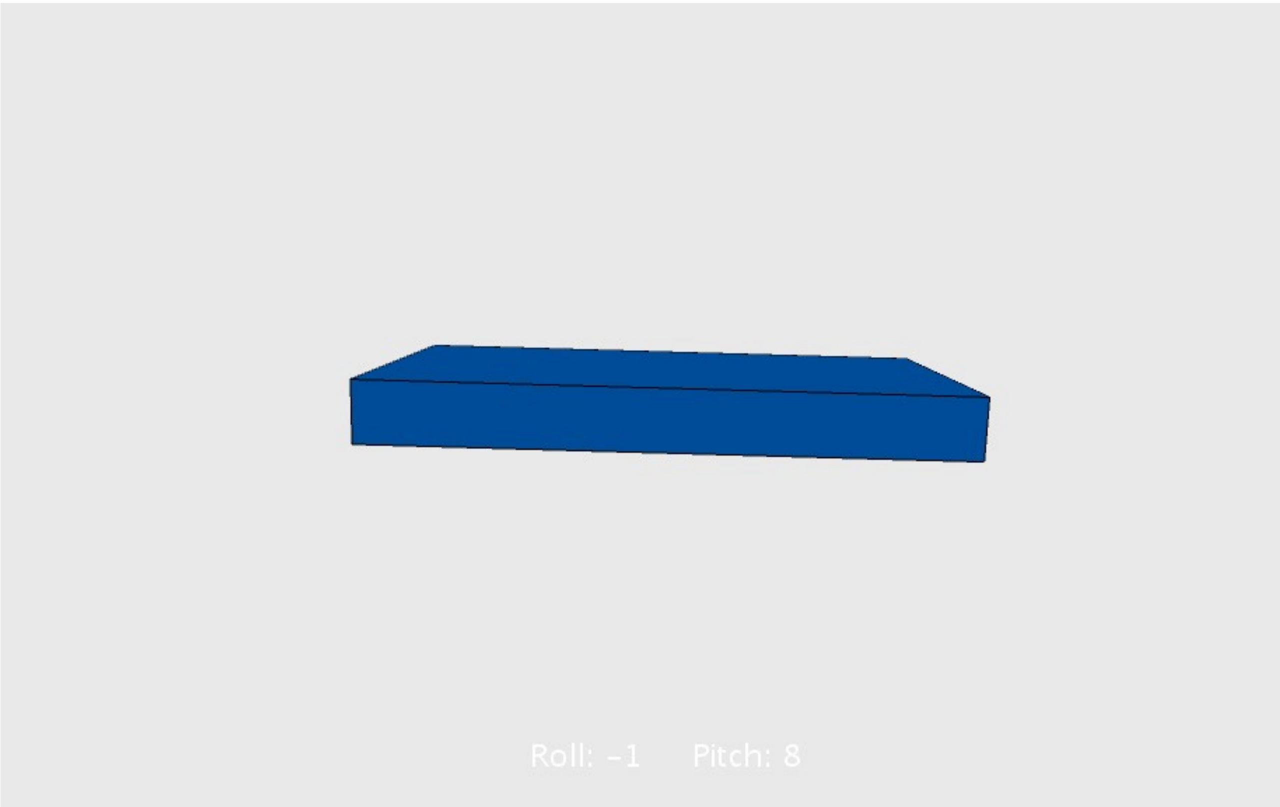

Figure 7.2: 3D model depicting a single limb.

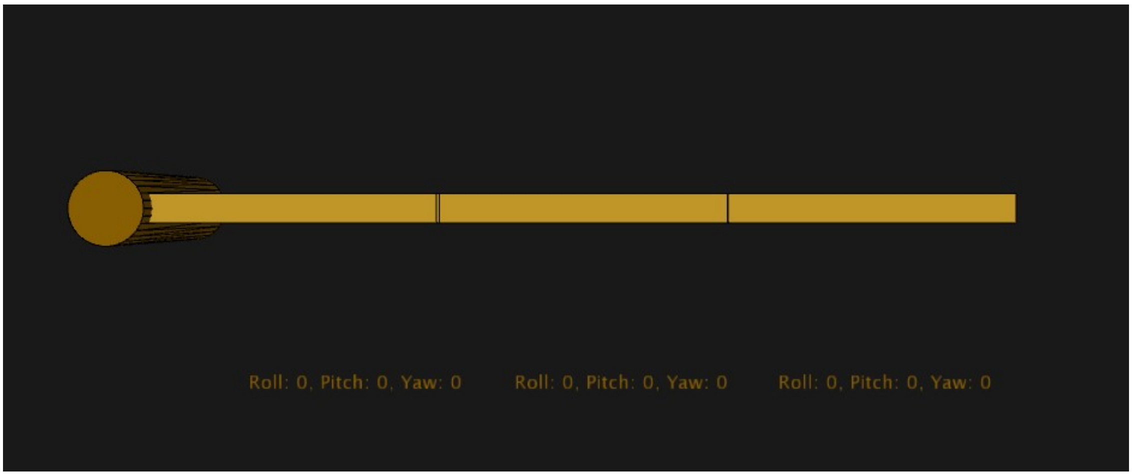

Figure 7.3: 3D model depicting an arm (3 Limbs).

7.2 Results of Algorithm on Application

By the application of this algorithm we can conclude that the device is now able to eliminate any errors that may occur and correct them as they form and integrate the corrected data to the model in real time, creating a real time illustration of the movement of the body.

Chapter 8

Conclusions

8.1 Conclusions

We have a motion sensing and rendering of a device, from our applications we have concluded that these kinds of sensors are fairly complex and require further work to be done on them, this field is still in need of a lot of research and there is quite a lot of growth opportunity in this department.

The system can capture the movement of a human host efficiently, but it still is lacking in some areas. Time loss is a factor that eventually occurs as the host computer takes about 19ms and in the real world the refresh rate is higher than 40 ms, hence, the human eye can easily detect the delay when it occurs. The quality and sensitivity of the sensor also plays an important role, we used China made devices which were not up to standards and were harder to configure than normal, original devices. The industry leaders such as the "Xsens MVN" group have made huge advancements in the department of motion capture, being a private company, the real crux of the algorithm still remains a mystery, the use of high precision, fast reaction times and high sensitivity of the sensors also play an important role but, they are expensive and not for mere testing devices. Taking the costs into consideration our device the MPU-9250 has a certain gap in sensitivity compared to the special inertial sensors. Overall, the project is complete as a framework, but there is a lot of room for improvement and a lot of applications are still possible.