CHAPTER 1

INTRODUCTION

1.1 GESTURE RECOGNITION AND AUGMENTED REALITY

Human gestures are an efficient and powerful way of interaction. These are sometimes used to express ourselves. For example nodding may be used to show our consent, or raising a hand can be used to indicate that we are willing to speak something. A number of techniques are used to convert these gestures into required output, typically either image based or device based, although hybrids are beginning to come about. This technology is still in its emerging state, a number of applications have been implemented in real time. But according to our ideology virtual images in real life scenario will make the conversation effective.

With the rapid advancements in technology, the use of computers in our daily life has increased manifolds. Our aim is to design a system that can understand the sign language accurately so that the signing people may communicate with the non signing people without the need of an interpreter. It can be used to generate speech, text or image. Unfortunately, there has not been any system with these capabilities so far. A huge population in India alone is of the deaf and dumb. It is our social responsibility to make this a part of the growing world. In this project we have used few sample signs for testing purposes. This is not according to any standard sign language systems.

Augmented Reality is research area which is in progressive stage and focuses on wearable technology like goggles, contact lenses which will be commonly used in future. The main characteristics of Augmented Reality system are combining real and virtual environment, real time interaction and registered in 3D.

The basic goal of Augmented Reality system is to enhance the user perception and interaction with real world through supplementing the real world with 3D virtual objects that appear to co-exist in some space as real world. This project was initially focused on developing a system for disabled people using this gesture recognition technique. But after learning the prolific technology of augmented reality, it was extended as a presentation tool and a learning aid.

1.2 CHRONOLOGY

Initially the data glove was made of accelerometers and microcontrollers. This glove converted the hand gestures into text messages based on specific American or British sign language system alone. It was made wireless by using GSM technology and the message was simply displayed in a LCD screen.

Then the hand gesture recognized was converted into speech signals. In this the gestures were predefined according to ASL and stored in an EEPROM.

As these gesture based products gained popularity, image processing methods were also added to these techniques. Based on the requirement, Leap Motion sensor or Kinect sensor was used. These two were mainly used for gaming and computer controls. The Leap Motion Sensor consists of 2 IR cameras and 3 IR LEDs. LED generates IR light signal and camera generates 300 frames per second of reflected data. These signals are sending to the computer through USB cable for further processing.

The Kinect Sensor consists of RGB camera, depth sensor and multi-array microphone. It recognizes facial movement and speech. A multi colour glove was used along with it. It detects different color region based on Gaussian colour model.

1.3 EXISTING SYSTEM

Nowadays, a data glove is treated as an interactive device, resembling a glove worn on the hand, which facilitates tactile sensing and fine-motion control in robotics and virtual reality. In case of robotic applications, a wireless data glove was developed to control a Talon robot. Sensors mounted on the glove send signals to a processing unit, worn on the user's forearm that translates hand postures into data. An RF transceiver, also mounted on the user, transmits the encoded signals representing the hand postures and dynamic gestures to the robot via RF link.

Commands to control the robot's position, camera, claw, and arm include "activate mobility," "hold on," "point camera," and "grab object." Data gloves are one of the several types of electromechanical devices used in haptics applications. Haptics is the science of applying control to interact with computer applications. Haptics offers an additional dimension to a virtual reality or 3-D environment. Within a virtual reality room or other VR environment, a data glove can allow you to interact normally with objects – for example, vibration while being shot by someone in the game. This enables user engagement in virtual environments, particularly for applications like VR gaming.

Figure 1.1 VR glove products

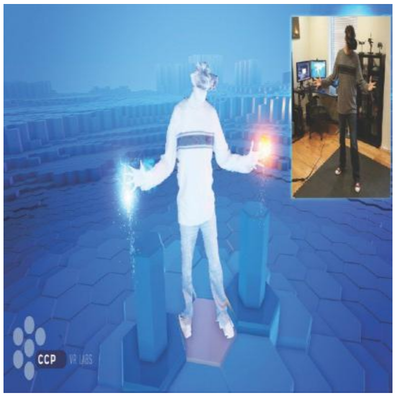

Figure 1.2 Virtual application

1.4 DRAWBACKS

The unanswered problems in all these systems are

- What if a mute and illiterate deaf person wants to communicate?
- What if there is no common language for the mute person and the listener?

In the above two cases the gesture to text conversion system is of no use.

- What if a mute and a deaf person want to communicate?
- What if the mute person decides to give an entire presentation?
- When several gestures are to be given how a microcontroller will be used?

In the above cases the gesture to speech conversion system is of no use.

- What is the point of the virtual world if the person they are interacting with is not visible?
- What if the robotic hand is not satisfactory as a human hand?

In this, the gesture to virtual world and gesture to robotic arm doesn’t provide a valid explanation.

1.5 PROPOSED SYSTEM

The basic concept involves the use of data gloves that can be use for real time interaction. These gloves are designed using flex sensors which is stitched to the glove. Flex sensors function as analog voltage dividers. All the gestures made are analyzed by the Arduino which interfaces the input and sends the information to the Bluetooth module. A Bluetooth module is used for wireless communication in this project. This signal is used to create augmented images in the laptop. Here augmented reality is achieved using a software called as UNITY 3D. There are many methods to do this but unity has a wide source for the implementation. So it will easier during the product development stage.

1.6 ORGANISATION OF THE REPORT

First, the abstract and literature Survey which are referred from Journals and Conference papers are provided. Then the system design and hardware description have been explained. Later, the software tools required for the data glove implementation is explained. The result and implementation follows the software tools. Finally, conclusion and the future enhancement of this project are highlighted.

CHAPTER 2

LITERATURE SURVEY

Aaisha Parveen S et.al., [1] have presented the design of a system that converts gestures into text messages for communication. All the gestures are analyzed and sent to a GSM modem which operates using a SIM card for wireless communication. This information is sent to mobile phones as SMS. But this system has two major drawbacks. One is the difficulty in coding the microcontroller for several gestures and the other is its incapacity to converse with strangers.

David L. Quam [2] has contributed his research work on identifying the classes of gesture recognition with human hand manipulating the glove. A total of 22 gestures in three classes were investigated. The first class contained gestures which only involved finger flexure. The second class contained gestures which required both finger flexure and hand orientation. The third class of gestures required finger motion, in addition to flexure and orientation. The major drawback was that more gestures couldn’t be recognized.

Manisha U et.al., [3] have provided a review on different methods to implement the sign language recognition system. Here the processes through image processing were mainly spotlighted. Since image processing was given due considering the drawbacks on edge detection was spoken as a disadvantage. Also there were few details on others systems also.

V.Padmanabhan et.al., [4] have focused on interpreting a meaning for every motion and converting it into speech signals. In this the messages are kept in a database and later on the gestures are matched with data base and produced as voice. The system includes wired method of text to speech conversion (TTS).

Ronald Azuma HRL Laboratories et.al., [5] in their research work highlighted the new developments in this field of Augmented Reality. It clearly defined the trending approaches in various fields and the possible advancements that can be made. It also stated the future works that can be evolved. But unfortunately there was not much reference to interaction with embedded systems.

Vajjarapu Lavanya et.al., [6] have found that the conversion of hand gestures into speech signals can be made wireless by using RF transmitter and RF receiver. In this the ASL is used where each alphabet of English vocabulary, A-Z is assigned a unique gesture. The main pitfall in this is the language barrier.

D. Vishnu Vardhan et.al., [7] have outlined the possibility of using an entire word for a particular gesture rather than assigning a particular alphabet for the same. This system uses MEMS technology to identify the gestures and transforms it into voice signals. The potential of mobility is the major benefit that is underscored.

Analyzing all the above systems and their limitations, a unique link for the embedded data glove with the augmented world has been identified. Thus a more advanced and integrated system design has been discussed in the chapter 3.

CHAPTER 3

SYSTEM DESIGN AND REQUIREMENTS

3.1 SYSTEM DESIGN

Arduino Nano is used to control the overall operations of the sensing system. Flex Sensors are used to recognize the bend on the index and the middle fingers. The three dimensional hand movements are sensed by the MEMS sensor. Bluetooth Modules are used for wireless transmission. Webcam is used for the capturing the user, the image target and thereby provide an augmented output.

3.1.1 DATA GLOVE MODULE

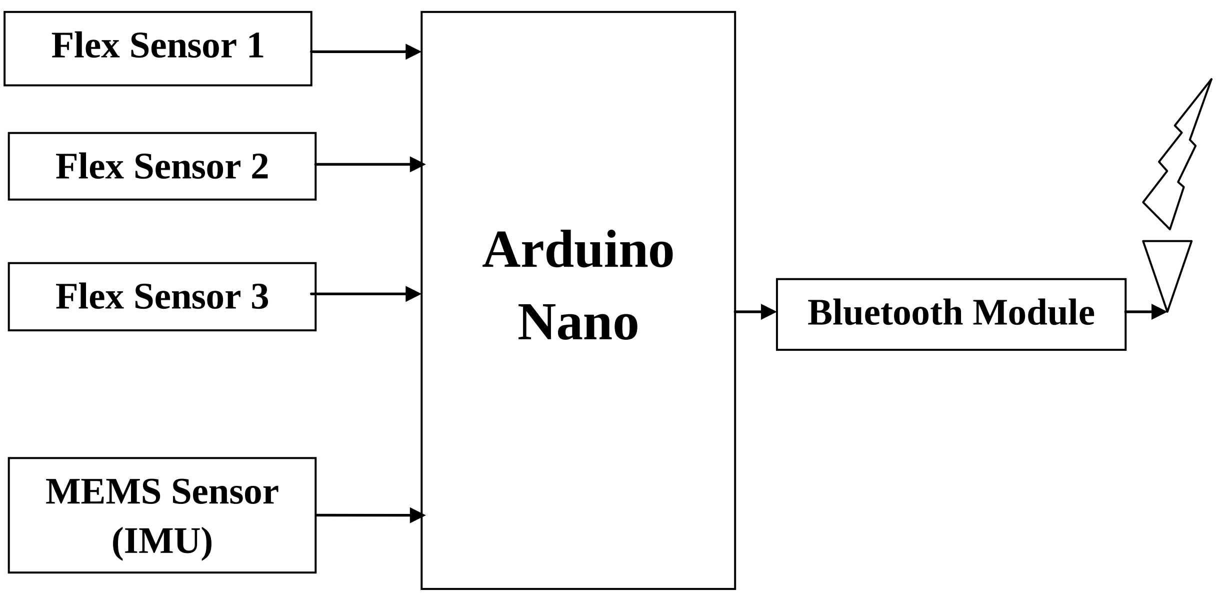

Figure 3.1 Block Diagram of the Transmitter

The Hand Data Glove comprises of two Flex sensors, a MEMS sensor, Arduino UNO and a Bluetooth transmitter. A bend on the user's finger is recognized by Flex sensors and corresponding voltage is produced by the sensors. The analog signals are then transmitted to the Arduino. The three dimensional hand movements are sensed by the MEMS sensor interfaced with the Arduino. The generated digital signal from the Arduino is transmitted by the Bluetooth transmitter to the Bluetooth Dongle.

3.1.2 RECEIVER MODULE

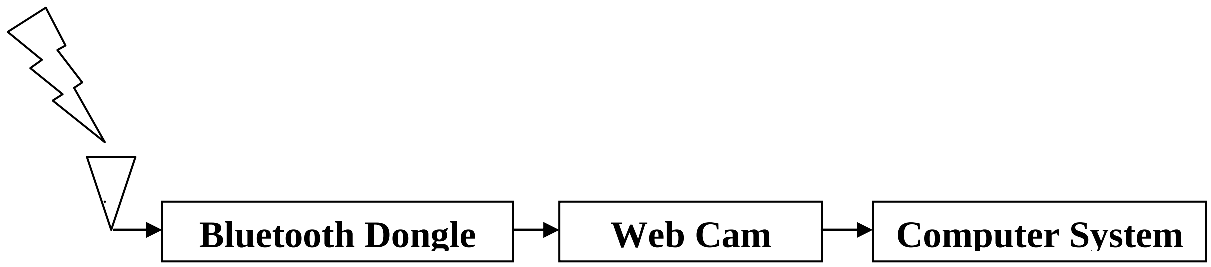

Figure 3.2 Block Diagram of the Receiver

The receiver module comprises of Bluetooth dongle which is connected to the USB port of the Laptop or the Computer. It also includes a Serial Communication port for connecting the module to the computer system. The transmitted signal is received by this and the corresponding function is performed on the computer screen.

3.2 CIRCUIT DESIGN

- At transmitter, Flex sensors are interfaced to the Arduino at A0, A1and A2 of the analog input pins.
- The MEMS sensor is interfaced at A5 & A6 of the analog pins and pin 2 of the digital pins.
- The signals from the sensor are converted from analog to digital form by the in-built ADC of the controller.
- The output from the Arduino at pin 3 & pin 4 is connected to the TX & RX of the Bluetooth module and is transmitted to the receiver.

- At receiver, the transmitted signals are received by the Bluetooth Dongle and are serially communicated to the computer system.
- UNITY 3D software is used that enables to perform the required operations.

The proposal was initially developed using an Arduino Uno. But at the later developed stages, it was replaced by Arduino Nano due to size constraints.

3.2.1 CIRCUIT USING ARDUINO UNO

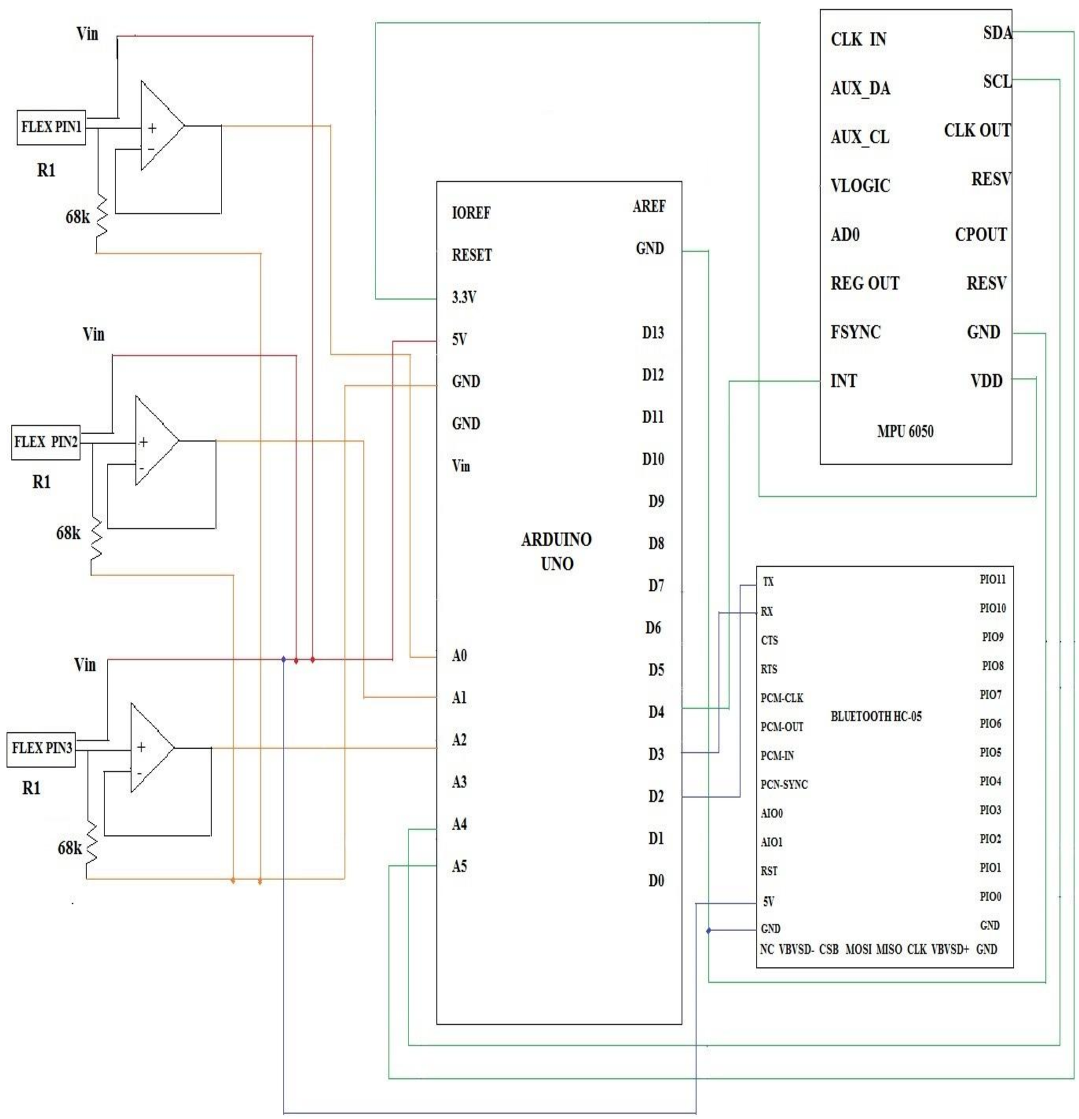

Figure 3.3 Circuit Diagram using Arduino Uno

3.2.2 CIRCUIT USING ARDUINO NANO

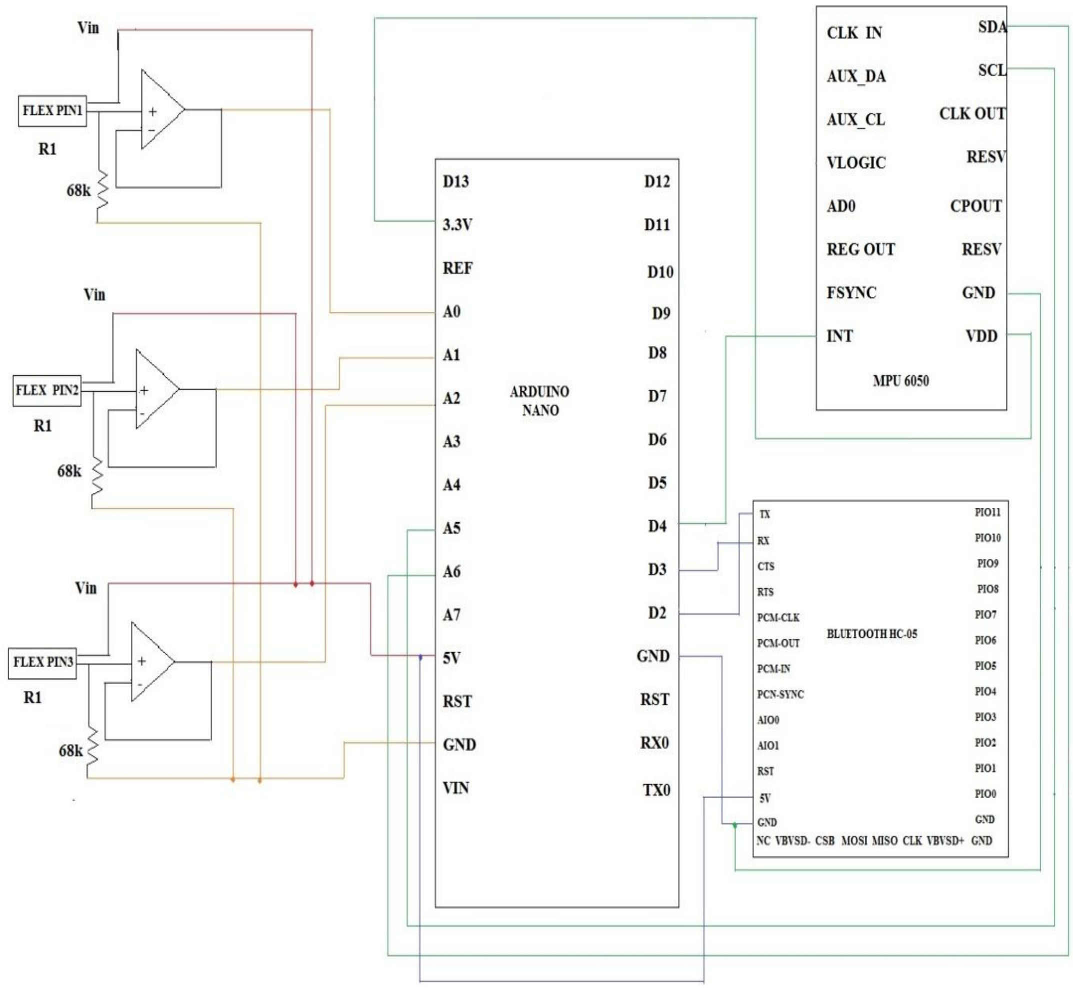

Figure 3.4 Circuit Diagram using Arduino Nano

3.3 PROJECT DESCRIPTION

The circuit interconnections are done as per the design. The sensors are mounted onto a glove. Receiver is connected to the computer system and the system is configured using the UNITY 3D Software. The Flex sensors 1, 2 & 3 are interfaced to channel 0 &1 of the controller. The X, Y, Z terminals of the MEMS sensor are connected to the channel 2, 3, 4 respectively. The user may either use the Flex sensor or the MEMS to send a signal to the controller. Depending upon the value of a predefined variable the controller will recognize the signal at one of its 5 channels.

For the case of Flex sensor, the sensor generates two types of analog values depending upon whether it is bent or not. The MEMS has 4 values, one for each of its 3 axes and one for the idle case. Once the channel is identified the type of value is recognized by the controller. This analog value is converted to digital data by the ADC module. We use the interrupts of the controller to enable quick response by the controller to the changing hand gestures.

The digital data is wirelessly transmitted using Bluetooth which accepts 8 bit data and transmit 32 bit signal. At the receiver side, the signal is received by the Bluetooth Dongle and serially transmitted to the computer.

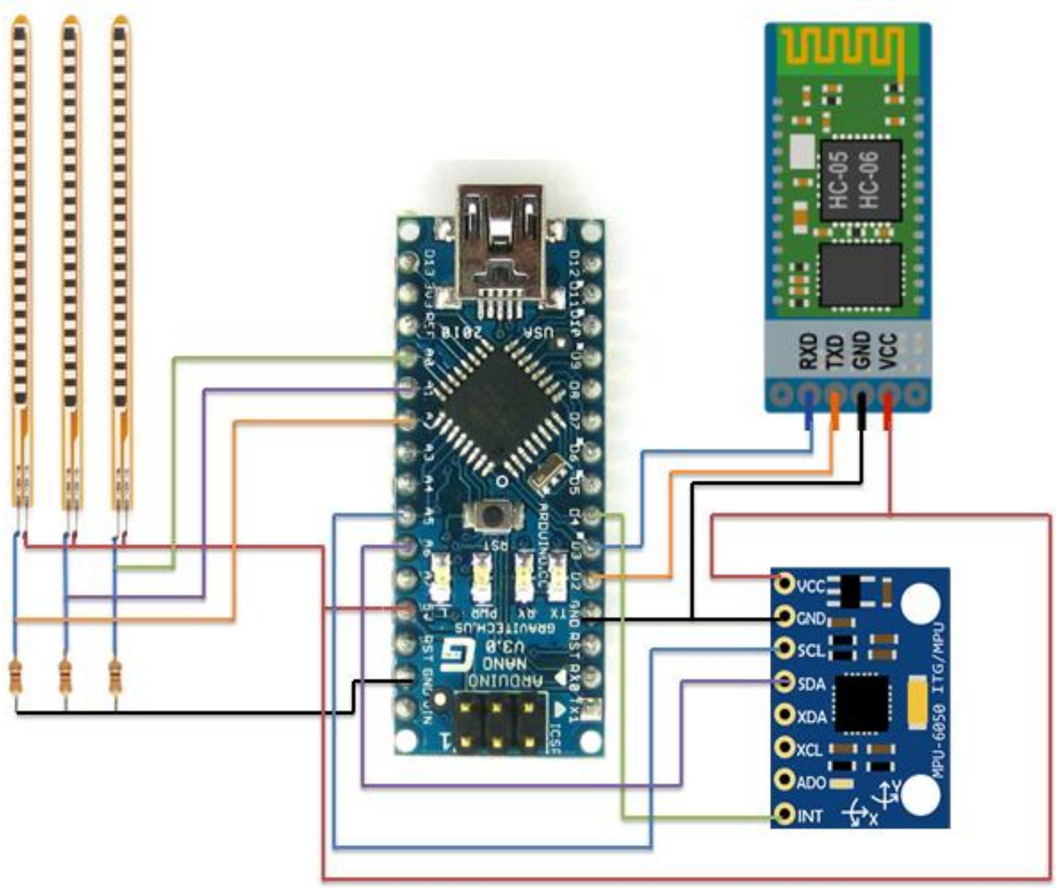

Figure 3.5 Hardware Connections

3.4 HARDWARE MODULES

The hardware modules used are –

- Arduino Nano
- Flex Sensor
- MPU 6050
- Bluetooth Module
- Bluetooth Dongle

3.4.1 ARDUINO NANO

Arduino Nano is a surface mount breadboard embedded version with integrated USB. It is a smallest, complete, and breadboard friendly. It has more analog input pins and onboard +5V AREF jumper. This new version 3.0 comes with ATMEGA328 which offer more programming and data memory space. It is two layers. That make it easier to hack and more affordable.

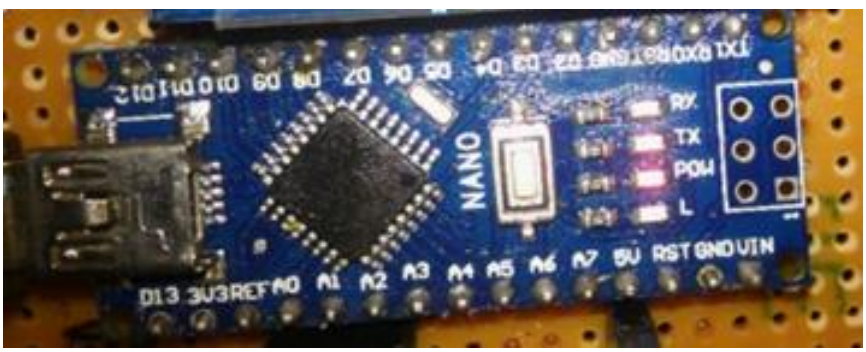

Figure 3.6 Arduino Nano

There are 14 digital pins on the Nano. It has 8 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts. Pins for I2C: 4 (SDA) and 5 (SCL). Support I2C (TWI) communication using the Wire library.

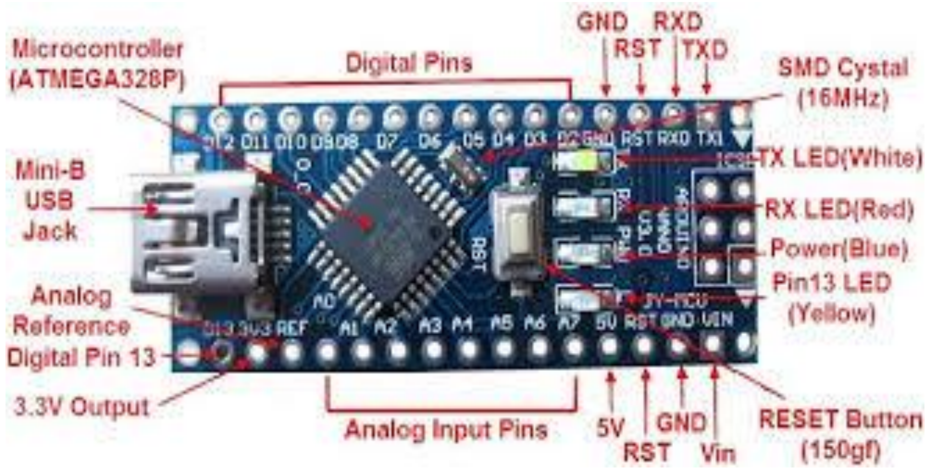

Figure 3.7 Pin description of Arduino Nano

The features that aided in the project are -

- The chip on the board plugs straight into your USB port and supports on your computer as a virtual serial port. The benefit of this setup is that serial communication is an extremely easy protocol which is time-tested and USB makes connection with modern computers and makes it comfortable.
- It is an open source design and there is an advantage of being open source is that it has a large community of people using and troubleshooting it. This makes it easy to help in debugging projects.
- It is a 16 MHz clock which is fast enough for most applications and does not speeds up the microcontroller.
- It is very convenient to manage power inside it and it had a feature of built-in voltage regulation. This can also be powered directly off a USB port without any external power. You can connect an external power source of upto 12v and this regulates it to both 5v and 3.3v.
- An on-board LED is attached to digital pin 13 to make fast the debugging of code and to make the debug process easy.

- Finally, it has a button to reset the program on the chip.

3.4.2 FLEX SENSORS

The flex sensor is a variable resistor. The resistance of the flex sensor increases as the body of the component bends. Flex sensors are available in two sizes: one 2.2" (5.588cm) long and another coming in at 4.5" (11.43cm) long. Left flat, these sensors will look like a 30kΩ resistor. As it bends, the resistance between the two terminals will increase to as much as 70kΩ at a 90° angle. By combining the flex sensor with a static resistor to create a voltage divider, one can produce a variable voltage that can be read by a microcontroller's analog-to-digital converter.

Figure 3.8 Flex Sensor

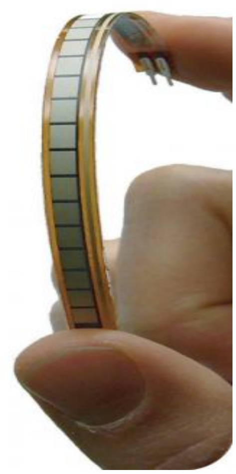

Figure 3.9 Bend Direction

3.4.3 MPU 6050

IMU sensors usually consist of two or more parts. Listing them by priority, they are the accelerometer, gyroscope, magnetometer, and altimeter. The MPU 6050 is a 6 DOF (Degrees of Freedom) or a six-axis IMU sensor, which means that it gives six values as output. Three values from the accelerometer and three from the gyroscope are obtained. The MPU 6050 is a sensor based on MEMS (Micro Electro Mechanical Systems) technology. Both the accelerometer and the gyroscope are embedded inside a single chip. This chip uses I2C (Inter Integrated Circuit) protocol for communication.

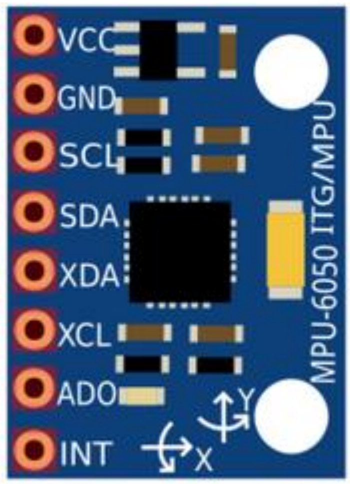

Figure 3.10 IMU

The features that aided in the project are -

- Programmable interrupt supports gesture recognition, panning, zooming, scrolling, and shake detection
- VDD Supply voltage range of 2.375V–3.46V
- Full Chip Idle Mode Supply Current: 5µA
- 400kHz Fast Mode I²C or up to 20MHz SPI (MPU-6000 only) serial host interfaces
- User self-test
- 10,000g shock tolerant
- Smallest and thinnest package for portable devices (4x4x0.9mm QFN)

3.4.4 HC-05 BLUETOOTH MODULE

HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It has the footprint as small as 12.7mmx27mm.

Figure 3.11 HC-05 Bluetooth Module

The features that aided in the project are -

- Typical -80dBm sensitivity
- UART interface with programmable baud rate. Supported baud rate: 9600,19200,38400,57600,115200,230400,460800.
- Auto-connect to the last device on power as default.
- Permit pairing device to connect as default.
- Auto-pairing PINCODE:”0000” as default
- Auto-reconnect in 30 min when disconnected because of exceeded range of connection.

3.4.5 BLUETOOTH DONGLE

A USB Bluetooth dongle is a quick and inexpensive means to allow Bluetooth Wireless communications onto personal computer. Simply plug in the

USB Bluetooth Dongle adapter into the computer's USB (Universal Serial Bus) port and any Bluetooth peripheral device will be able to communicate with the PC! This includes computer-to-computer communications, cell phones, printers, and PDAs.

It transfers data at rates up to 3MB and has a LED status indicator. Its slim, pocket-sized design is ideal to fit into even the most crowded USB port. It has an operational wireless range of 2 to 100 meters

Figure 3.12 Parts of Bluetooth Dongle

Figure 3.13 Bluetooth Dongle

3.5 SOFTWARE TOOLS

The Softwares used are Arduino IDE and Unity 3D. C++ is used for first one and C#/ JAVA is used for the later

3.5.1 ARDUINO IDE

Arduino is a prototype platform (open-source) with ready-made software called Arduino IDE (Integrated Development Environment), which is used to write and upload the computer code to the physical board.

(i) ARDUINO INSTALLATION AND BASICS

Step 1: Download Arduino IDE Software. Power up the board by connecting it to the computer using the USB cable.

Step 2: To create a new project, select File --> New.

Step 3: Select the Arduino board. Go to Tools -> Board and select your board.

Step 4: Select the serial port. Select the serial device of the Arduino board. Go to Tools -> Serial Port menu.

Step 8: Upload the program to your board. Before that, the function of each symbol appearing in the Arduino IDE toolbar is studied.

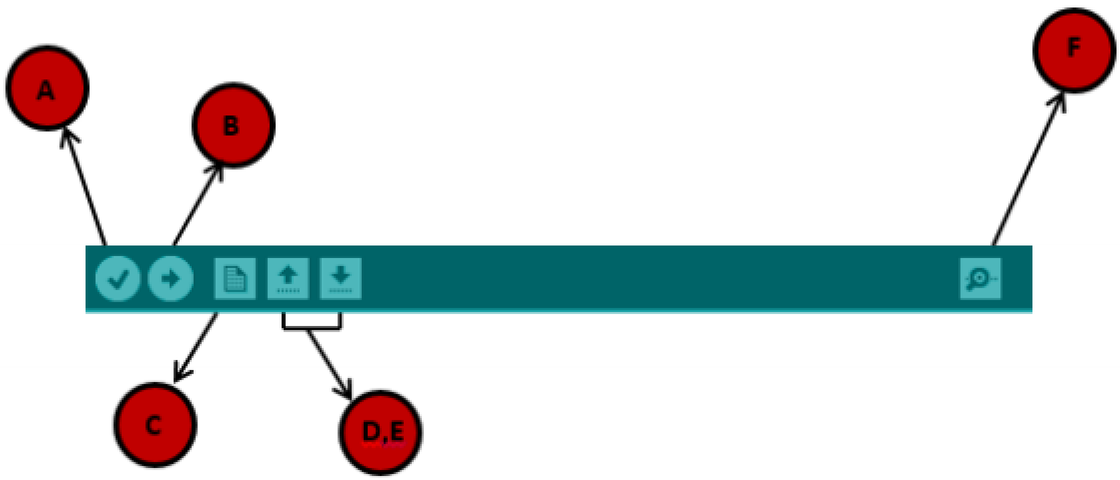

Figure 3.14 Arduino IDE Toolbar

A- Used to check if there is any compilation error.

B- Used to upload a program to the Arduino board.

C- Shortcut used to create a new sketch.

D- Used to directly open one of the example sketches.

E- Used to save your sketch.

F- Serial monitor used to receive serial data from the board and send the serial data to the board.

Now, simply click the "Upload" button in the environment. If the upload is successful, the message "Done uploading" will appear in the status bar.

(ii) PROGRAMMING STRUCTURE

Arduino programs can be divided in three main parts: Structure, Values (variables and constants), and Functions. Software structure consists of two main functions:

- Setup( ) function
- Loop( ) function

```
Void setup ( )
{
}
```

PURPOSE: The setup() function is called when a sketch starts. Use it to initialize the variables, pin modes, start using libraries, etc. The setup function will only run once, after each power up or reset of the Arduino board.

```
Void Loop ( )
{
}
```

PURPOSE: After creating a setup() function, which initializes and sets the initial values, the loop() function does precisely what its name suggests, and loops consecutively, allowing your program to change and respond. Use it to actively control the Arduino board.

(iii) SERIAL COMMUNICATION MODULE

Serial communication is a Protocol with a set of rules, which must be applied such that the devices can correctly interpret data they mutually exchange. Arduino automatically takes care of this, so that the work of the programmer/user is reduced to simple write (data to be sent) and read (received data).

Serial communication can be further classified as –

- Synchronous - Devices that are synchronized use the same clock and their timing is in synchronization with each other.

- Asynchronous - Devices that are asynchronous have their own clocks and are triggered by the output of the previous state.

(iv) INTER INTEGRATED CIRCUIT

Inter-integrated circuit (I2C) is a system for serial data exchange between the microcontrollers and specialized integrated circuits of a new generation. It is used when the distance between them is short. Connection is established via two conductors. One is used for data transfer and the other is used for synchronization (clock signal).

The I2C bus consists of two signals: SCL and SDA. SCL is the clock signal, and SDA is the data signal. Following are the pins for Arduino UNO boards – A4 (SDA), A5 (SCL)

The following functions are used to initialize the Wire library and join the I2C bus as a master or slave. This is normally called only once.

- Wire.begin(address) – Address is the 7-bit slave address

• Wire.beginTransmission(address) – Begin a transmission to the I2C slave device with the given address.

• Wire.write(value) – Queues bytes for transmission from a master to slave device.

• Wire.endTransmission() – Ends a transmission to a slave device.

3.5.2 UNITY 3D

Unity3D is a powerful cross-platform 3D engine and a user friendly development environment. The user interface is well organized and the panels can be fully customized by dragging and dropping.

(i) SCRIPTING

Scripts are known in Unity as behaviours. It allows the user to take assets in the scene and make them interactive. Multiple scripts can be attached to a single object, allowing for easy code reuse. To create a C# script the following procedure has to be performed-

1. Click Assets > Create > New C# Script
2. Rename the new script in the Project panel
3. Double click the script to open it in MonoDevelop

C# class names must be the same as their file name and are case sensitive. All scripts have a **start()** method and an **update()** method. The start() method is run once when the object is first created, while the update() method run once per frame.

Now that the script is done, so assign it to the asset: Drag the script onto the example asset in the scene.

```
using UnityEngine;
using System.Collections;

public class PlayerScript : MonoBehaviour {
        // Use this for initialization
        void Start () {
        }

        // Update is called once per frame
        void Update () {
        }
}
```

Figure 3.15 Sample Script

(ii) UNITY3D WITH VUFORIA

Vuforia is an Augmented Reality Software Development Kit (SDK) for mobile devices that enables the creation of Augmented Reality applications. It uses Computer Vision technology to recognize and track planar images (Image Targets) and simple 3D objects, such as boxes, in real-time.

❖ CREATE A PROJECT

1. Create a new project in Unity.
2. Go to: https://developer.vuforia.com/downloads/sdk and Download the SDK for Unity
3. Accept the import of the package into your Unity project.
4. Then proceed to the “Develop” page in Vuforia.

❖ OBTAIN A LICENSE KEY

A license key is a unique ID which is required to create an app in Unity which uses Vuforia. To create a license key, head over to “Develop >> License

Manager" and select "Add License Key". We will be prompted with a page like this one:

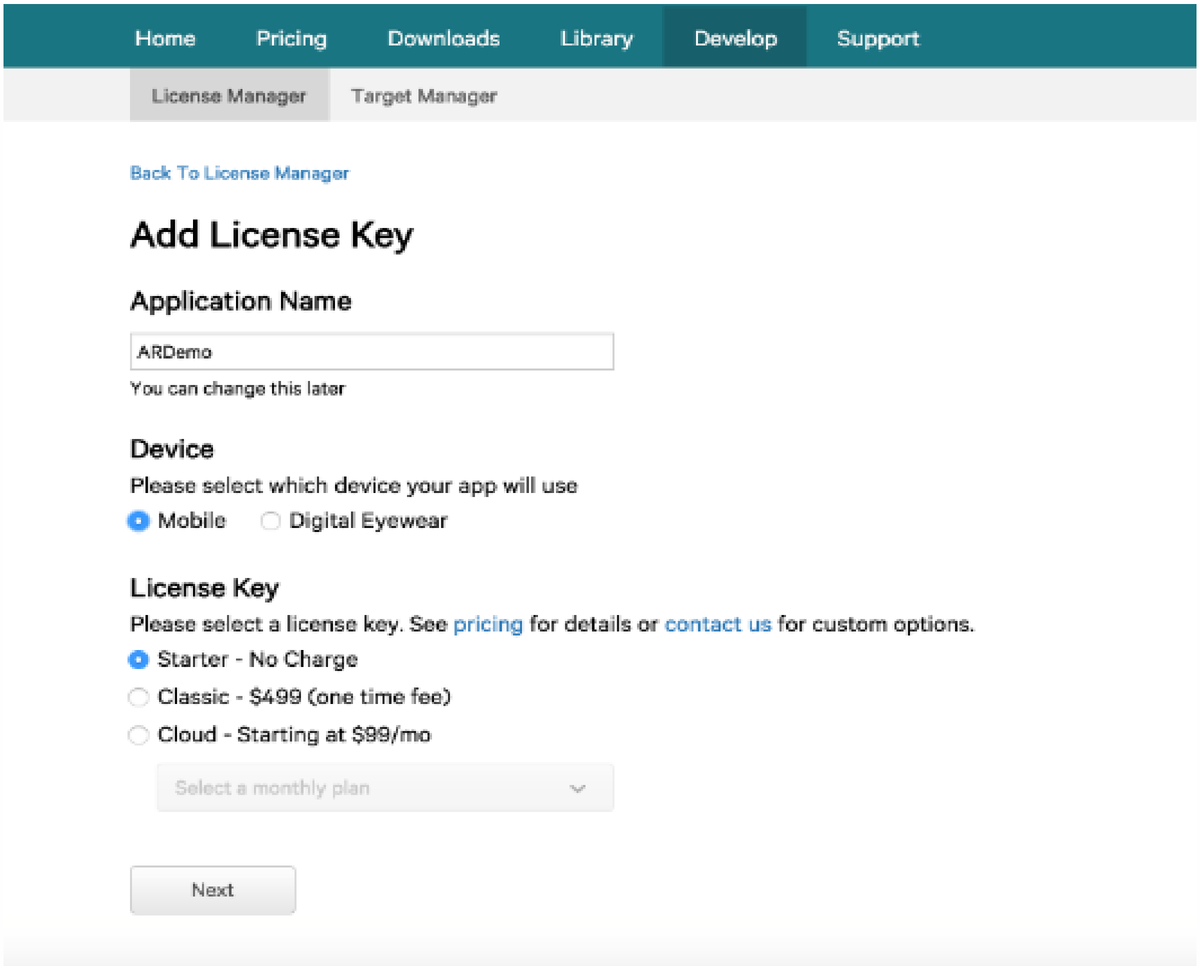

Figure 3.16 License Key Creation in Vuforia

Then select "Next" and then "Confirm". This will be directed back to the "License Manager" page where the application's name will appear. Select the intended application name to get the license key. This key will be used later in Unity.

❖ CREATE A DATABASE AND AN IMAGE TARGET

The next step is to set up an image as a target. Go to "Develop >> Target Manager" and select "Add Database" and fill in the details:

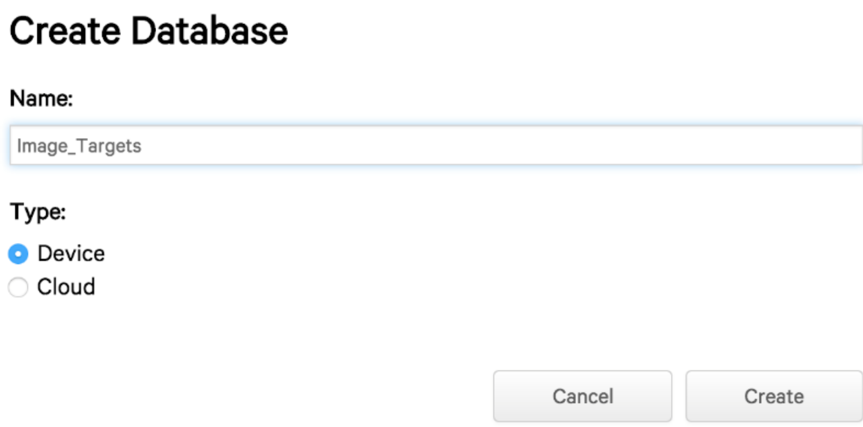

Figure 3.17 Database Creation in Vuforia

Next, add images to the newly made database by selecting the database and clicking on "Add Target.

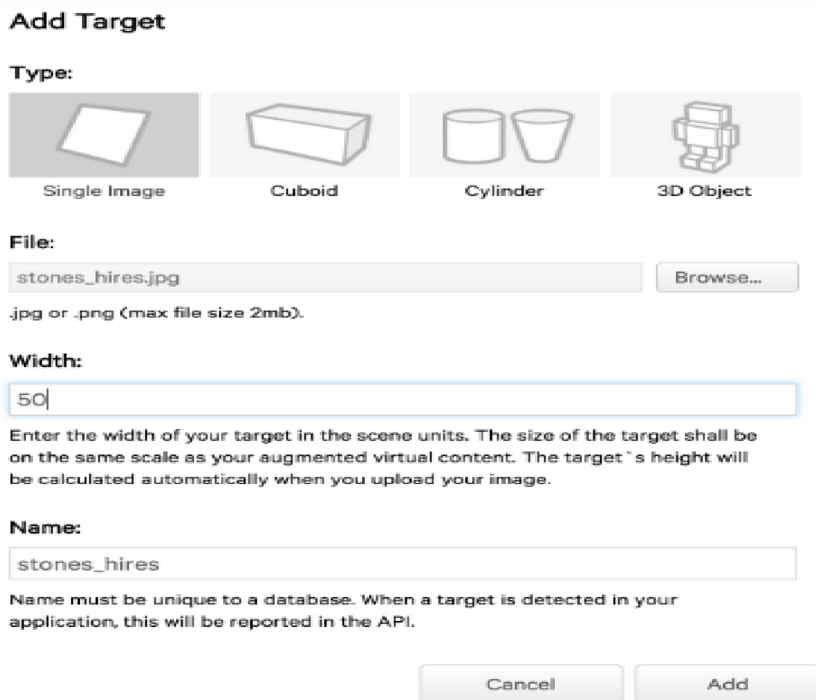

Figure 3.18 Adding Targets in Vuforia

After a short upload time, the target will be added and we will have a "Download Dataset" option.

An important thing to note here is the “Rating”. A good rating means that it can be used for tracking, whereas a bad rating means that the image does not have enough feature points.

Figure 3.19 Rating in Vuforia

Download the dataset by selecting “Download Dataset >> Unity editor”.

❖ STEPS IN UNITY EDITOR

In “Assets >> Vuforia >> Prefabs” and drag ARCamera into the scene. “ARCamera” is an Augmented Reality camera prefab from Vuforia and is very similar to a regular camera. It is. Also delete the Main Camera from the scene (Hierarchy Panel, “Right Click >> Delete”).

For ARCamera to run, add the App License Key. In the Inspector panel, paste the unique key previously received from Vuforia’s License Manager.

Next, add the ImageTarget to the scene. It can be found under “Assets >> Vuforia >> Prefabs”. Configure both to load and activate in the Datasets field of the AR Camera’s Vuforia Configuration asset, which is accessible from the AR Camera's Inspector panel via the Open Vuforia Configuration button. This allows the user to use targets from both Device Databases at the same time in the Unity scene.

Next, let’s drag the 3D image onto the Scene. It can be found under “Assets >> prefab >> name”. For the 3D image to appear over Image Target, it needs to be made a child of Image Target. Then, whenever the Image Target is detected by a mobile device's camera, all the children of the target will also appear together.

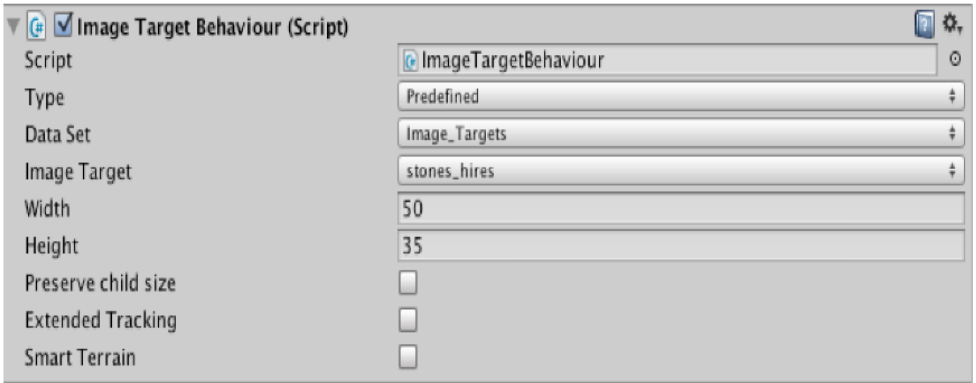

Figure 3.20 Image Target Behaviour

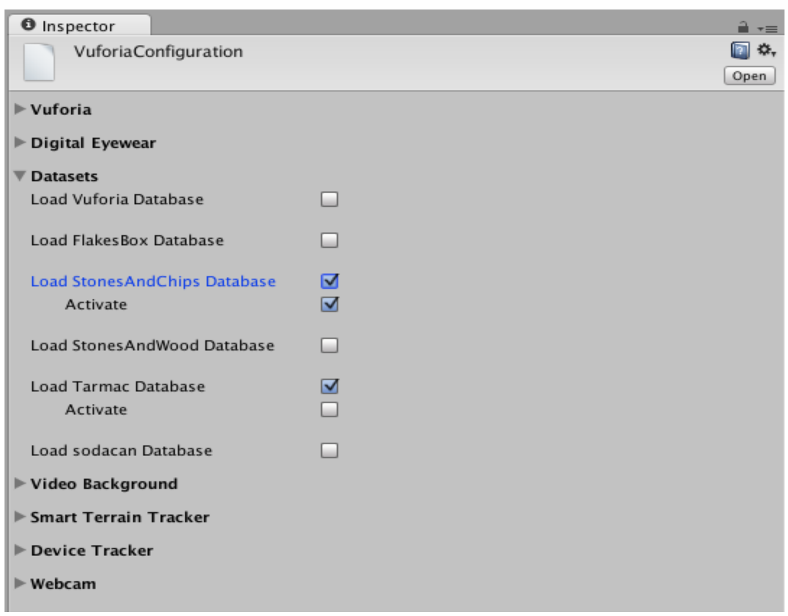

Figure 3.21 Vuforia Configuration in Unity Editor

At this point, select "Run" and point a print out of the image towards the webcam, a 3D image is seen on it. Finally the scene is saved.

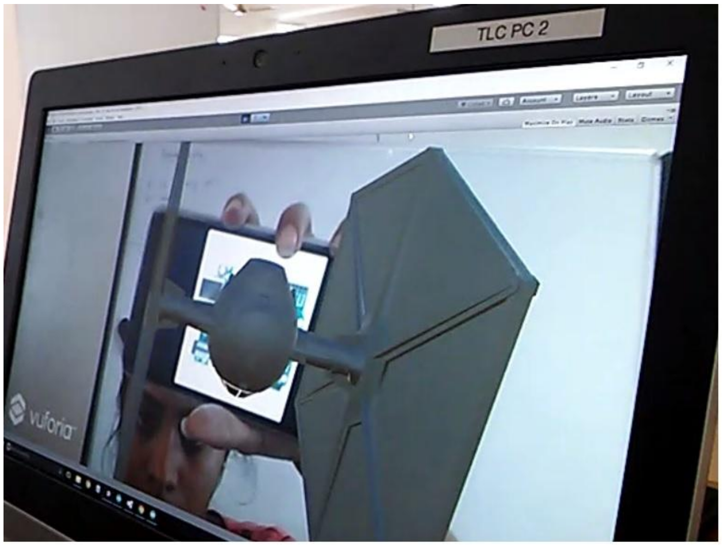

Figure 3.22 3D image on the Image Target

Learning about the modules gives a basic overview of the project implementation process which is discussed in the next chapter. Also the obtained result of the project has been shown in that chapter.

CHATPER 4

IMPLEMENTATION AND RESULT ANALYSIS

4.1 IMPLEMENTATION

The steps implemented in the system are-

STEP 1: Get the gestures from the user and store it.

STEP 2: Scan the digital pattern for the data gloves.

STEP 3: Acquire data from flex sensors and send it to the Arduino and compare it with the stored data.

STEP 4: Transmit the data as RF signal through Bluetooth

STEP 5: Receive the data via UART Bluetooth

STEP 6: Convert the obtained data into 3D augmented images using "UNITY 3D" software.

STEP 7: Display in the laptop

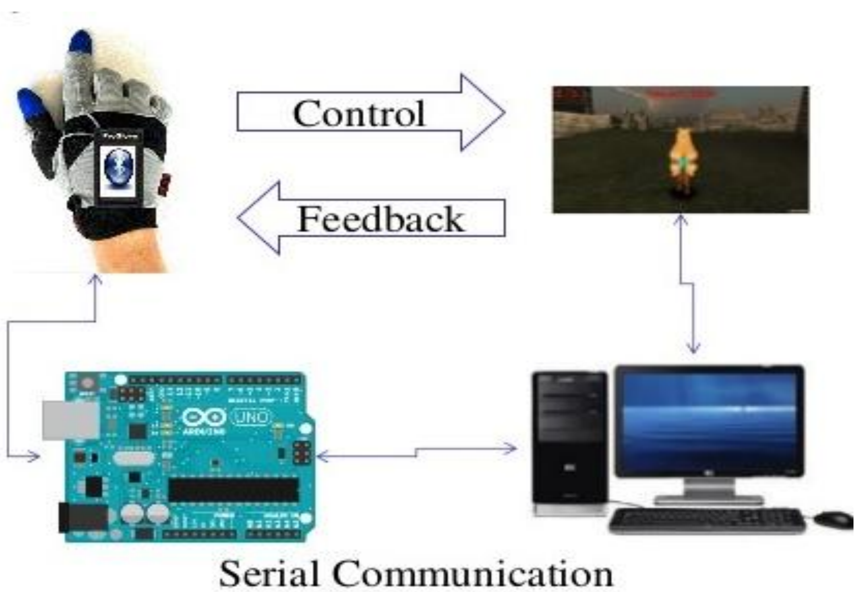

Figure 4.1 Procedural View

The flow of the system can be explained as,

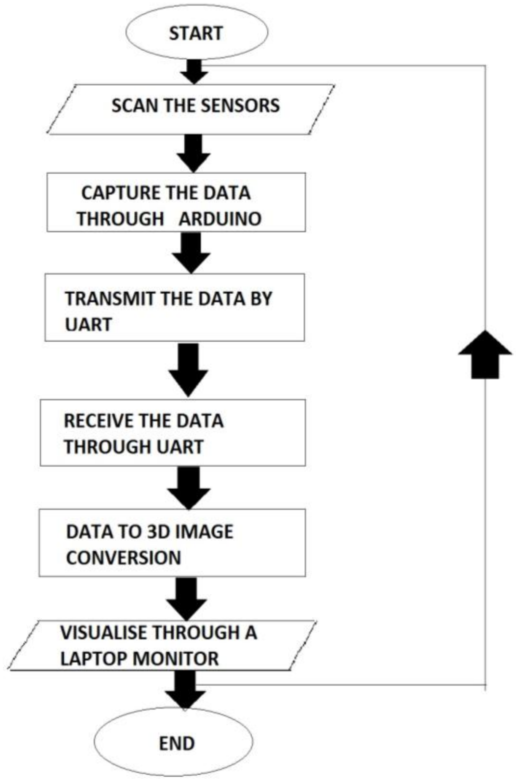

Figure 4.2 Flow Chart for Proposed System

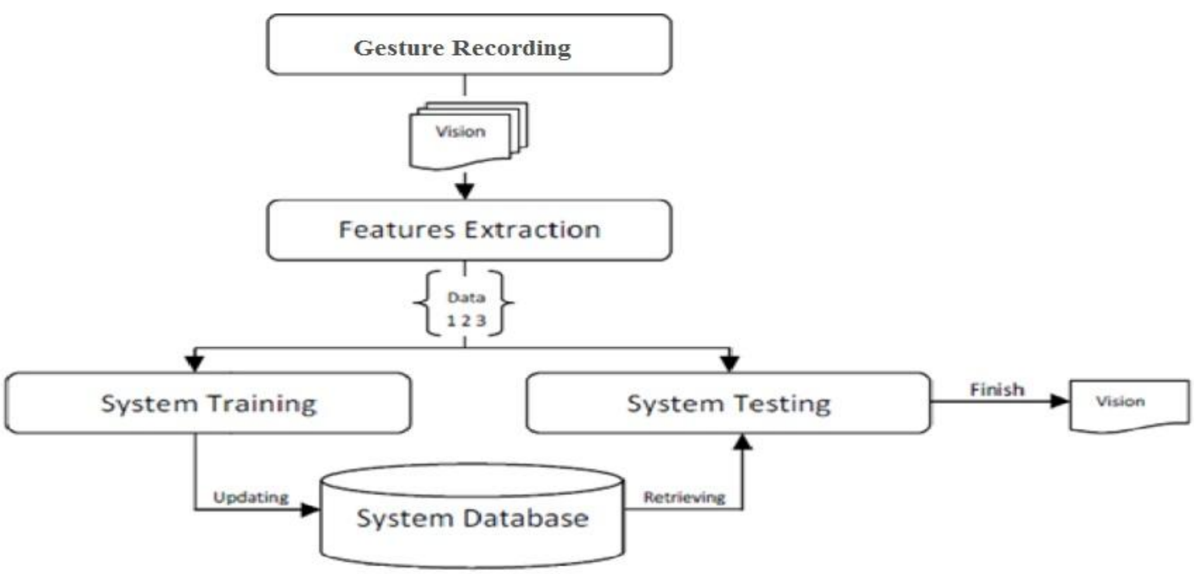

Figure 4.3 Flow of the code

In the initial stages of the project, Arduino Uno was used. The entire set up was made in a bread board and an old woolen glove for testing purposes.

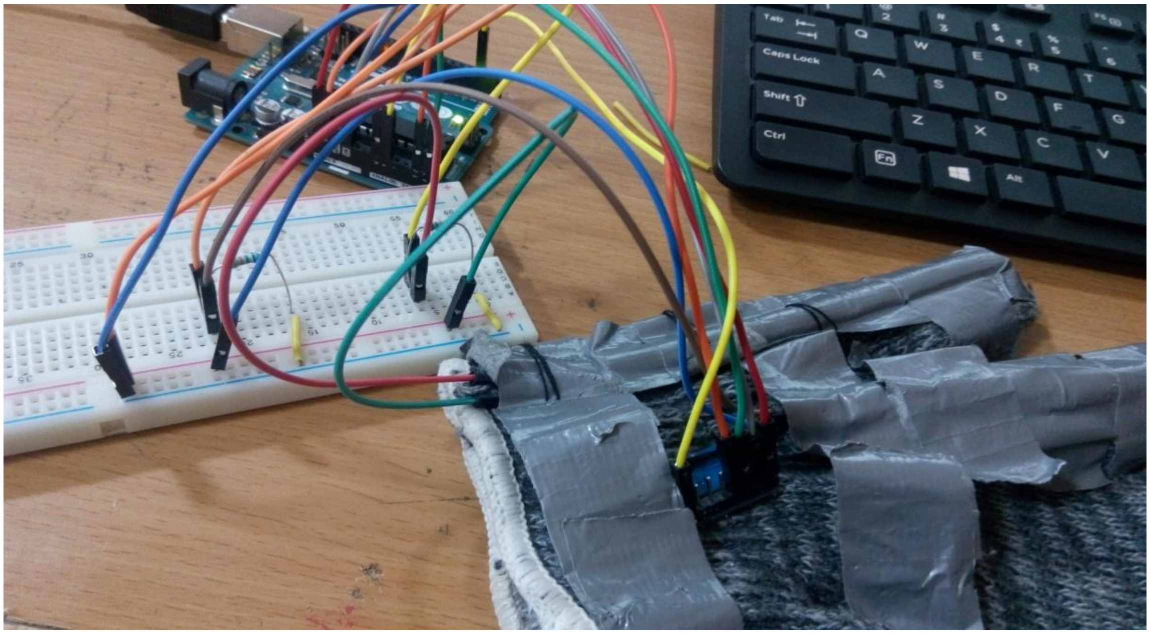

Figure 4.4 Test model

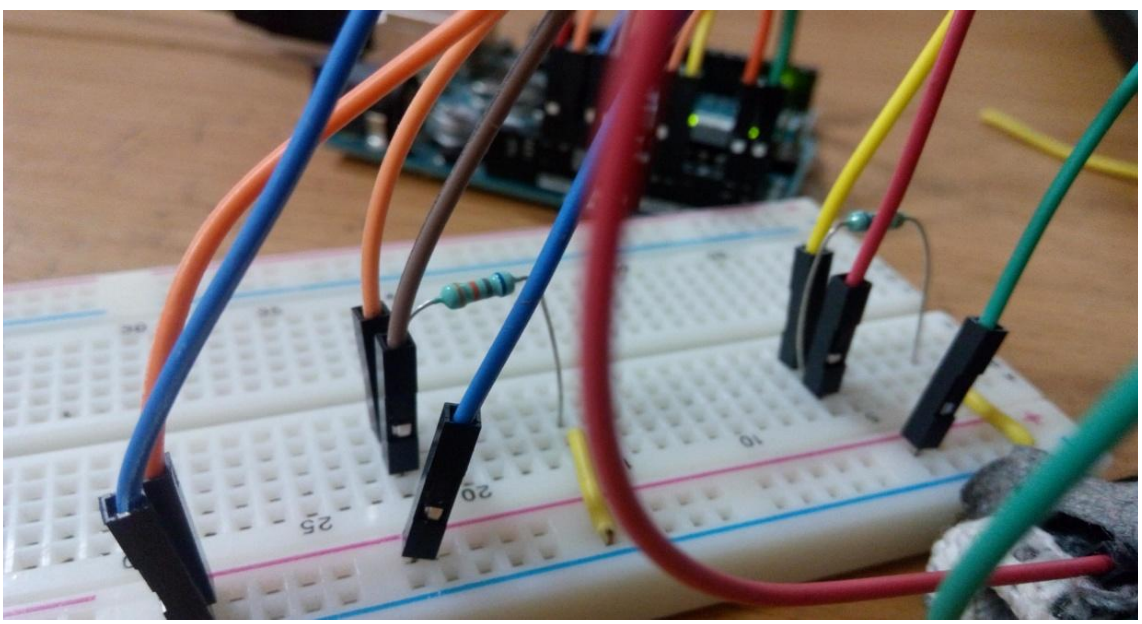

Figure 4.5 Voltage divider circuit in the Breadboard

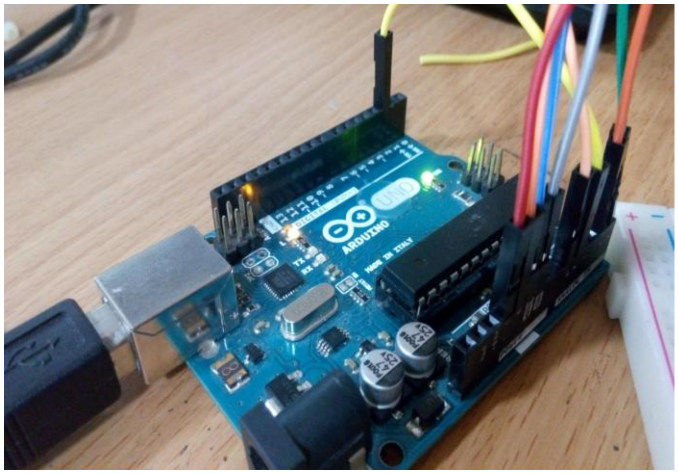

Figure 4.6 Connected Arduino Uno

At the product development stage, the Arduino Uno was replaced by Arduino Nano for size constraints. Then the bread board components were soldered on a strip board. Flex sensors and the IMU were stitched onto a new cotton glove and the strip board was stuck on a wrist band.

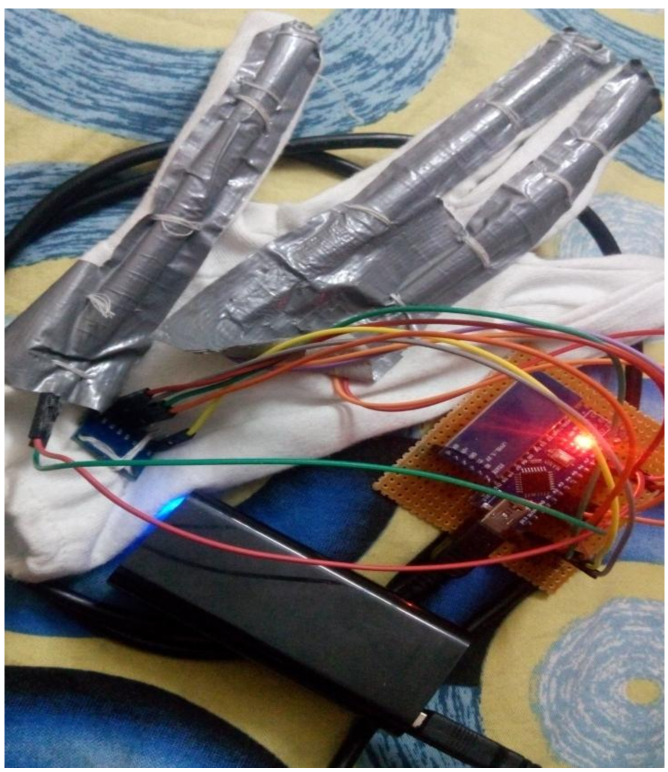

Figure 4.7 Product view

Figure 4.8 Flex Sensors with IMU

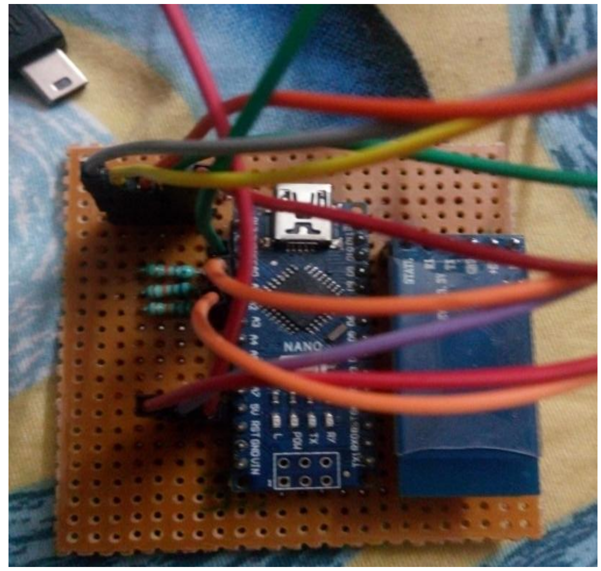

Figure 4.9 Soldered Strip Board

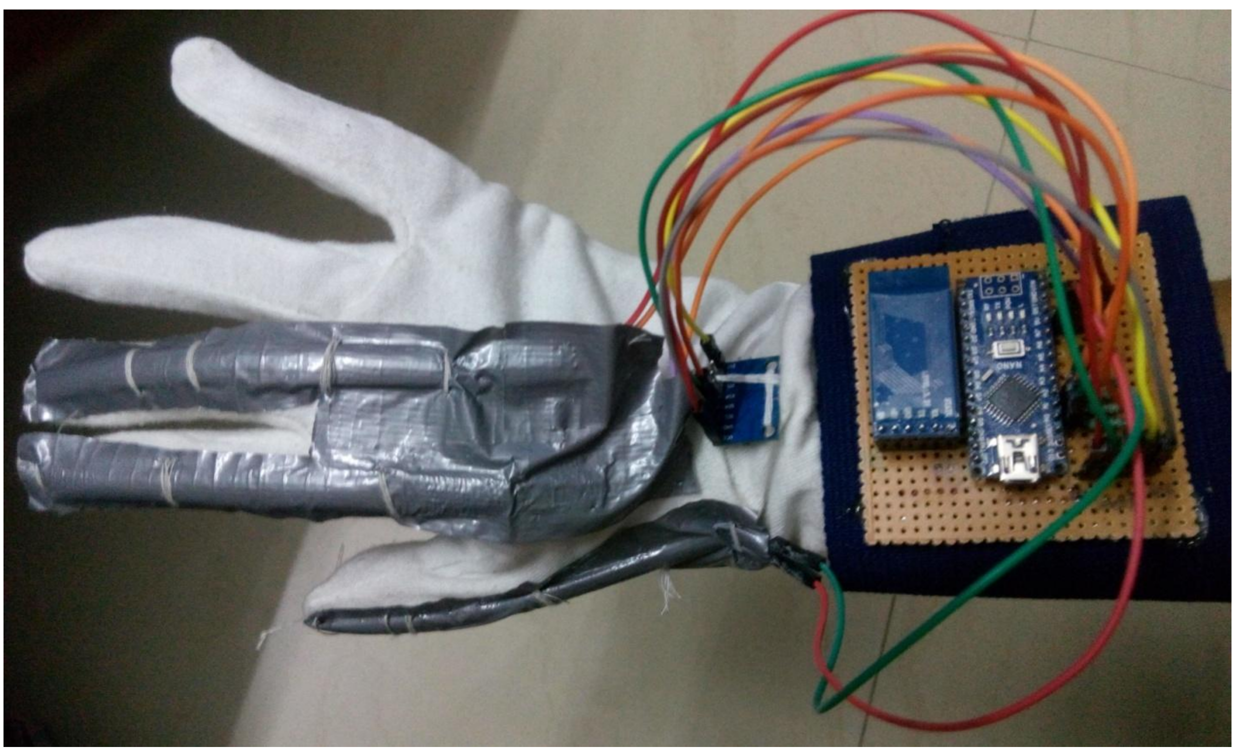

Figure 4.10 Glove worn by the user

Figure 4.11 Glove with the Power Source

4.2 RESULT ANALYSIS

The project was demonstrated in 3 different ways. One was by displaying the power point of the project. The next was the presentation of educational 3D images that can be rotated 180 degrees. The last and the main one was for the communication of the mute people which was shown by images with their interpretation.

The glove was worn by the user. When the first gesture is shown, the first image was observed. Then the image changes for every other gesture. The IMU function was given to the last gesture. So every time the last gesture assigned is made, the object begins to rotate based on the movement of the hand.

Basically, when the device is turned on, it goes into the calibration mode. This mode is where the gestures that are going to be used are initially registered. This is stored in the Arduino and when the actual Play Mode begins, every gesture made is compared with this database for detecting the symbol. This communication is made wireless by using Bluetooth on both the ends. So the user can stand anywhere within the range of the Bluetooth and the movement of hand is more flexible.

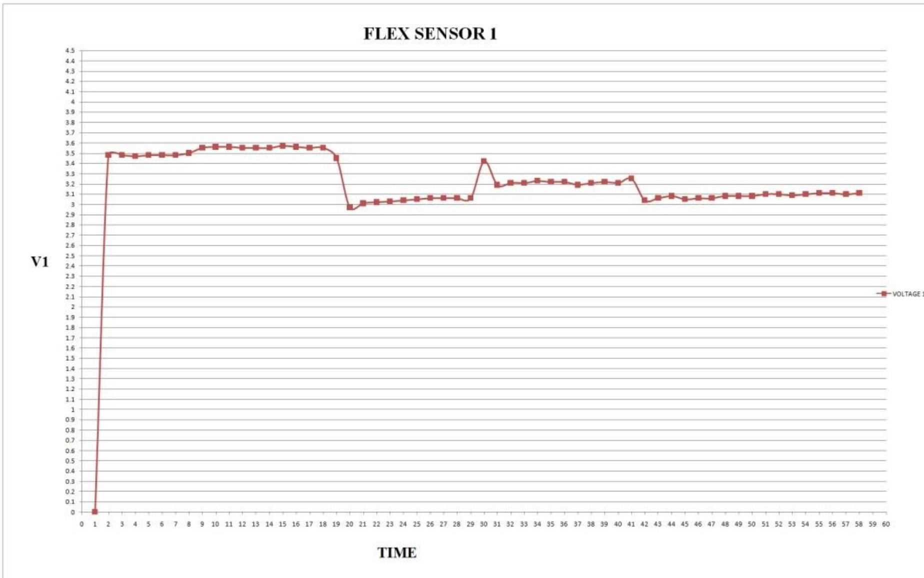

Figure 4.12 Gesture Analysis of Flex Sensor 1

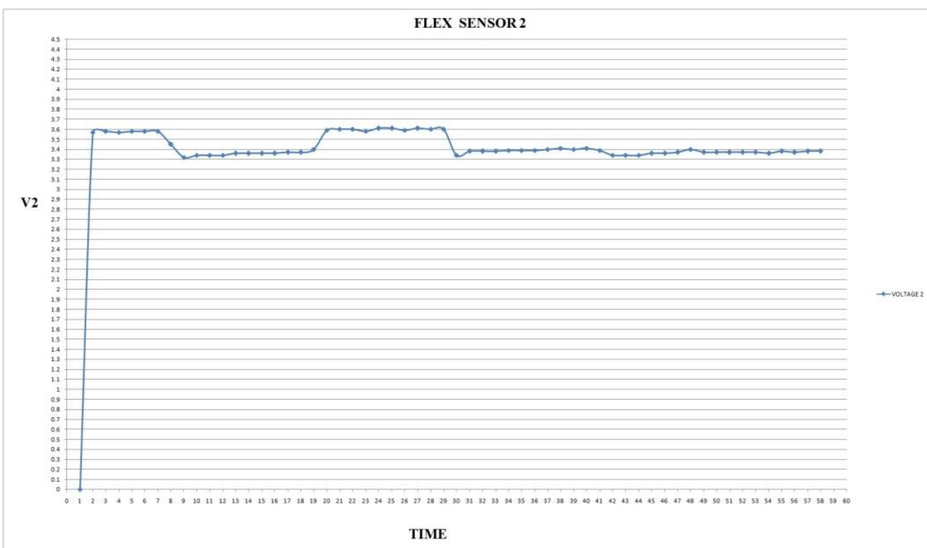

Figure 4.13 Gesture Analysis of Flex Sensor 2

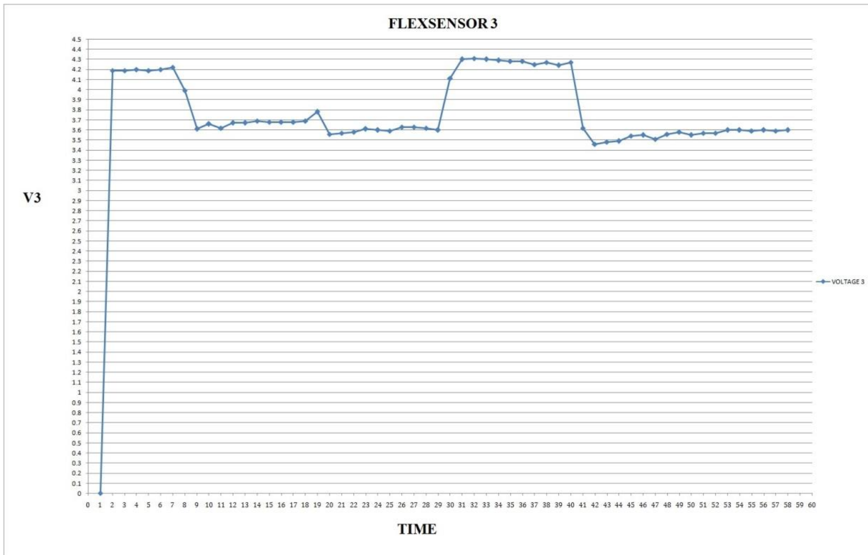

Figure 4.14 Gesture Analysis of Flex Sensor 3

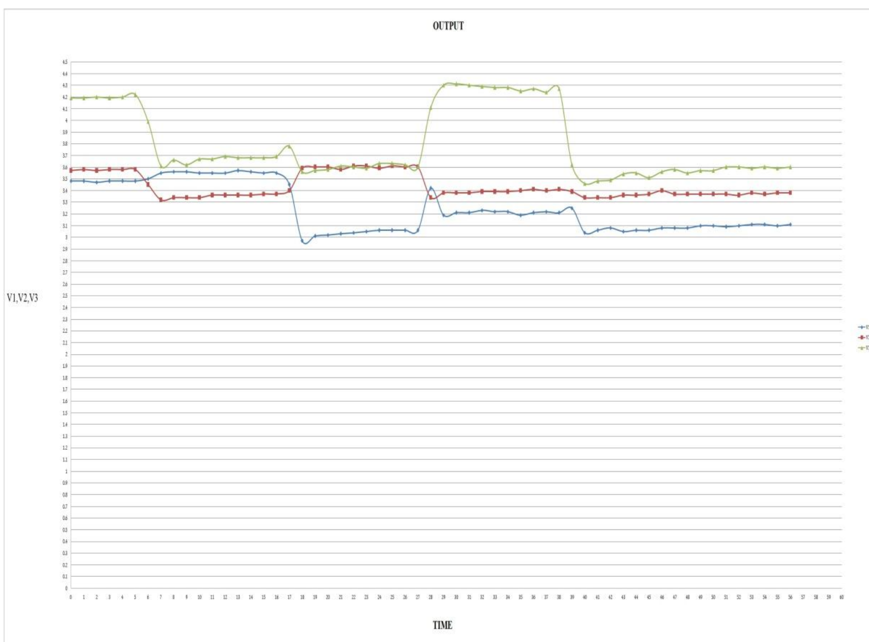

Figure 4.15 Comparison of the 3 Flex Sensors

Scrutinizing the results more thoroughly gives various ideas for the future enhancement of the ideology proposed and is hereby suggested in the chapter 5. Also the inference made from the current proposal is discussed in the same.

CHATPER 5

CONCLUSION AND FUTURE ENHANCEMENT

5.1 CONCLUSION

This project is designed to research on how a Hand Data Glove works and how it is used to interface between human and machine. It is found that the voice processing and text conversion are having many limitations with them, while augmented reality can be used for the same purpose without any limitations.

The degree of freedom (DoF) of data glove is more with augmented reality than the other technologies resulting better inputs in the world of virtualization. Hence, this is useful as a learning aid where the teaching process is interactive with visual 3D support. Its application for presenting power points makes it more handy.

Also, the project achieved user-friendliness due to the use of very simple hand gestures and these gestures need not be from ASL or BSL. It also achieved 93% of accuracy in gesture recognition. Moreover, the use of Bluetooth has lowered the power consumption and increased a long battery life.

The result is found to be very good and efficient with real-time. This experiment proves that such devices are a good technology for interacting and controlling the devices, software or hardware.

5.2 FUTURE ENHANCEMENT

In future work, Cloud Recognition and GUI operations can be included into the product for a more user friendly, wide database for 3D images and storage of database in the cloud. IOT is the trending technology these days. So using Cloud recognition broadens its application in IOT based researches. Also there can be a combination of two or more gestures to form a new complex gesture for a complex task to perform. Adding to this, HMD like Google glasses can be used for the vibrant experience of the product.

The OnGUI() function can be used to set up a particular image for particular gestures without the knowledge of coding. Once this process is enabled, the user can change different images for the same gesture. No changes in the script are required.

The Cloud Recognition Service is an enterprise class Image Recognition solution that enables developers to host and manage Image Targets online. This service is ideally suited for applications that use many targets, or targets that need to be updated frequently. Customers using the service can leverage the following benefits:

- Scale: More than one million targets can be used in an app.
- Flexibility: Integrate with existing content management systems.
- Time to Market: Deliver real-time, dynamically changing content and accelerate time to market

REFERENCES

1. Aaisha Parveen S and Rohitha.U.M, "Microcontroller based Hand Gesture Recognition System using Flex Sensor for Disabled People", in the proceedings of the National conference on Electronics and Communication(NCEC), 2015.
2. David L. Quam, "Gesture Recognition with a DataGlove", in the proceedings of the IEEE National Conference on Aerospace and Electronics, pp. 755-760, 1990.
3. Manisha U. Kakde, Mahender G. Nakrani, Amit M. Rawate, "A Review Paper on Sign Language Recognition System For Deaf And Dumb People using Image Processing ",International Journal of Engineering Research & Technology (IJERT), Vol. 5, No. 03, pp. 590-592, 2016.
4. Padmanabhan.V and M.Sornalatha, "Hand gesture recognition and voice conversion system for dumb people", International Journal of Scientific & Engineering Research, Vol. 5, No. 5, pp 427-431, 2014.
5. Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier and Blair MacIntyre, "Recent Advances in Augmented Reality", in the proceedings of the IEEE Conference on Computer Graphics and Applications, Vol. 21, No. 6, pp. 34-47, 2001.
6. Vajjarapu Lavanya, Akulapravin and Madhan Mohan, "Hand Gesture Recognition And Voice Conversion System Using Sign Language Transcription System", International Journal of Electronics & Communication Technology, Vol. 5, No. 4, pp.145-150, 2014.
7. Vishnu Vardhan.D and P. Penchala Prasad, "Hand Gesture Recognition Application for Physically Disabled People", International Journal of Science and Research (IJSR), Vol. 3, No. 8, pp.765-769, 2014.