A Project Report on

IoT Based

Smart Gesture Control

Submitted by

SumaLatha G(R151615)

Punya Bhanu Priya T(R151664)

Tejesh P(R151541)

Mukunda P.M(R151518)

Submitted to

Under the supervision of

Mr. T Chandrasekhar

Assistant Professor

Computer Science & Engineering

AP-IIIT RK Valley

as a part of

Partial fulfilment of the degree of Bachelor of Technology in Electronics and Communication Engineering

Date: 28-10-2019

CERTIFICATE

This is to certify that the report entitled "**IoT Based Smart Gesture Control**" submitted by **Tejesh Palagiri(R151541)** in partial fulfillment of the requirement for the award of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out by her under my supervision and guidance.

The report hasn't been submitted previously in part or in full to this or any other university or institution for the award of any degree.

Mr. T Chandrasekhar,
Project internal guide,
Computer Science & Engineering,
RGUKT, R.K Valley.

Mr. N Chandra Sekhar,
Head of the department,
Computer Science & Engineering
RGUKT, R.K Valley.

Declaration

I Tejesh P hereby declare that this report entitled "**IoT Based Smart Gesture Control**" submitted by me under the guidance and supervision of **T Chandrasekhar** is a bonafide work. I also declare that it has not been submitted previously in part or in full to this university or other university or institution for the award of any degree or diploma.

Date: 28-10-2019
Place: RK Valley

(Tejesh P)
Id.No:R151541

Acknowledgement

On the very outset of this report. I would like to extend my sincere & heartfelt obligation towards all the personages who have helped me in this endeavor.
Without their active guidance, help, cooperation & encouragement, I wouldn't made headway in the project.

I am ineffably indebted to **Mr. T Chandrasekhar**, my project internal guide for conscientious guidance and encouragement to accomplish this project.

I am extremely thankful and pay my gratitude to **Mr. N Chandra Sekhar, HOD CSE**, for his valuable guidance and support on completion of this project in its presently.

I extend my gratitude to **Rajiv Gandhi University of Knowledge and Technologies, R.K.Valley** for giving me this opportunity.

Any omission in this brief acknowledgement does not mean lack of gratitude.

Thanking you
Tejesh P
(R151541)

Abstract

Gesture recognition and control is a type of perceptual computing (Recognizing what's happening around it) user interface that allows computers to capture and interpret human gestures as commands. Gesture control is the ability of a computer to understand gestures and executes commands based on those gestures.

Gesture control can help to operate Laptop or PC by anyone simply by giving gestures or by the app. We can also operate Laptop or PC remotely with the help of app(Blynk).

Gesture Control can make Life Easier for the Disabled.

For example:

- PlayStation games such as "JUST DANCE" and "Kinect Sports".
- Google Pixel4 mobiles have gesture sensing feature.
- Controlling home appliances.

Introduction

IoT - Internet of Things

A concept where all the things around us are connected and communicate with each other. In IoT the things around us communicate with each other without the interference of the humans.

IoT Architecture

Key Elements

1) **Node**: Node is an electronic device that is attached to a network, and is capable of receiving, or transmitting information over a communication channel. A Node can be a combination of a Sensor or an Actuator or both with Processor and Communication Module.
    - **Sensor**: A Sensor is a device, module, machine, or subsystem whose purpose is to detect events or changes its environment and send the information to other electronics, frequently a computer processor.
    - **Actuator**: An Actuator is a component of a machine that is responsible for moving and controlling a mechanism or system.
    - **Processor**: A Processor, or "microprocessor" is a small chip that resides in computers and other electronic devices. Its basic job is to receive input and provide the appropriate output after processing it.
2) **Network**: A group of two or more devices (nodes) are connected and communicate with each other either wired or wireless.
3) **Cloud**: Cloud stores, access and processes the data and sends the information to the nodes.

Benefits

- Automation and Control
- Monitor
- Information
- Time saving
- Better Quality of Life
- Power & Security

IoT Verticals

- Smart Home describe
- Smart Agriculture
- Smart Retail(Ex:Amazon Go,Surprise Sweet Button)
- Smart Cities

- Industrial Automation

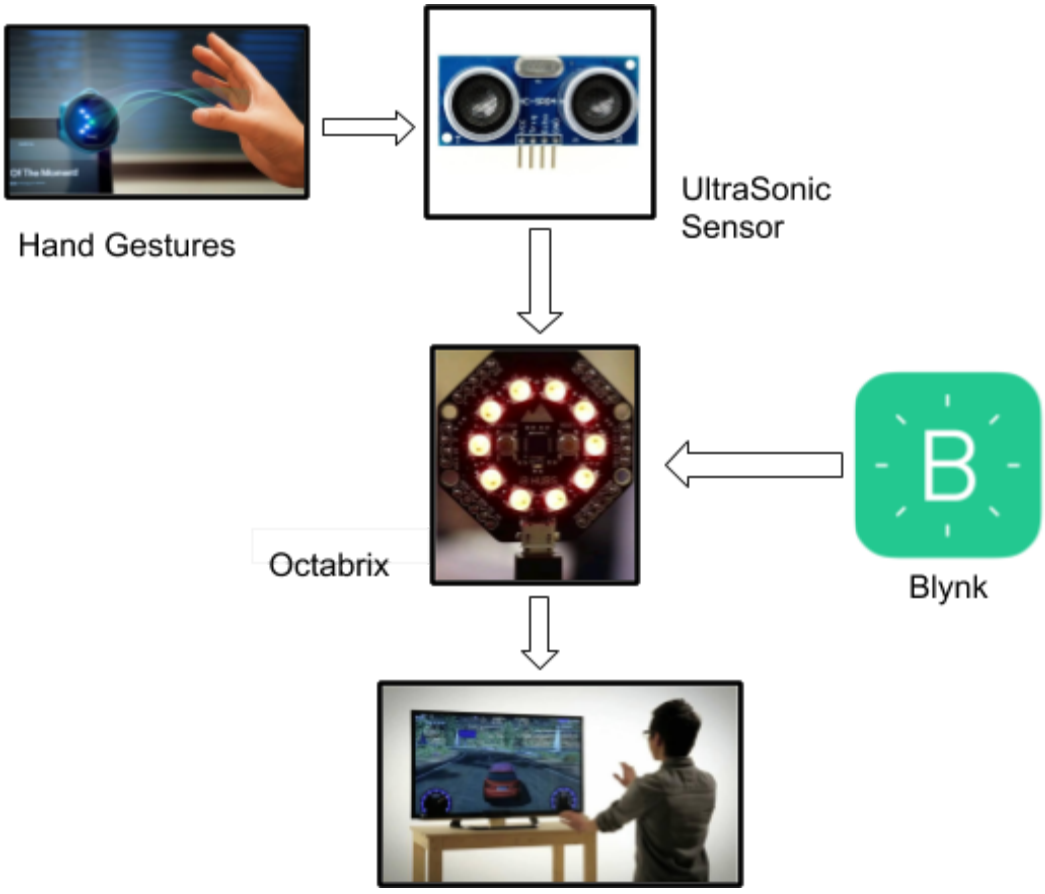
- **The total installed base of Internet of Things (IoT) connected devices is projected to amount to 75.44 billion worldwide by 2025.**

About Our Project

However as mentioned above the IoT has Prominent role in the future too, As a part of IoT we take up the project "**Smart Gesture Control**".

Smart Gesture Control is the Communication between Humans and Computers with Gestures. As Soon as a human give a Gesture to a Computer it takes as a command and executes based on them.

Approach



Hardware and Software components used

Hardware

- **Octabrix**
- **USB Cable**
- **UltraSonic Sensor**
- **Jumper Wires(Female - Female)**

Software

- **Arduino IDE**
- **Python**
- **PIP**
- **Pyserial**
- **PyautoGUI**
- **Blynk**

Features of the project

Using Blynk App:

1. **Arrow Keys Control:**

Joystick Widget(V1) enables all the operations which are done by the arrow keys.

2. **Space Key & Enter Key Control:**

Button Widget control the Space Key(V4) and the Enter Key(V7) on the Keyboard .

3. **Text Control:**

By Simple press on the Button(V5) Widget we can type a predefined Text on the PC.

4. **Close Key Control:**

Button(V8) widget Close can close the application and shut downs the system.

5. **Delete Key Control:**

Button Widget(V6) can delete a file or text.

Using UltraSonic Sensor:

1. **Play & Pause Gesture:**
   While a Video is going on, if we give a gesture with hand below 15 cm from the UltraSonic sensor then the video pauses/play automatically.
2. **Go to Next Gesture:**
   In a playlist we can jump to the next track or video if we give a gesture with hand above 15cm from UltraSonic Sensor.

Step by Step implementation

Step1: Install Arduino IDE , Python , Pyserial , Pyautogui , PIP Libraries.
Step2: Install Blynk App in mobile
Step3: Sign-up to the app.
Step4: Create a new project.
Step5: Now you'll get authentication token to your mail make note of that.
Step6: Add the required widgets.
   A. Joystick and add virtual pins
   B. Buttons and add virtual pins
Step7: Interface the UltraSonic(HC-SR04) and Octabrix with as shown in the Circuit Diagram.
Step8: Write the code in Arduino IDE.
Step9: And make script of python with given code.
Step10: Connect the Octabrix using USB to the Laptop or PC.
Step11: Upload the arduino code to the Octabrix.
Step12: Run the python Script.

1. Install Arduino IDE , Python , Pyserial , Pyautogui ,PIP Libraries.

Arduino IDE:

The Arduino integrated development environment is a cross-platform application that is written in the programming language Java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards

Python:

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.

PIP:

PIP is the package installer for Python. You can use pip to install packages from the Python Package Index and other indexes

Pyserial:

Pyserial helps us in getting serial data that is transmitted by the Octabrix.

Pyautogui:

The purpose of PyAutoGUI is to provide a cross-platform Python module for GUI automation for human beings. The API is designed to be as simple as possible with sensible defaults.

2. Install Blynk App in mobile

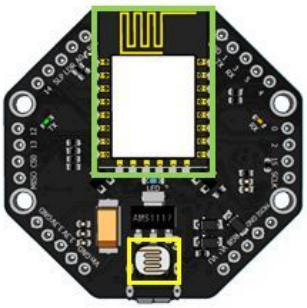
Blynk:

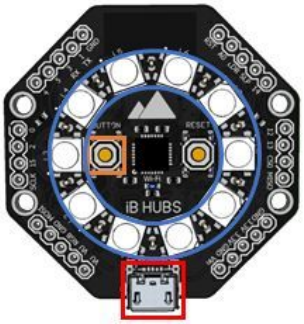
Blynk is a Cloud Platform to control things over the Internet. It's a digital dashboard where you can build a graphic interface for your project by simply dragging and dropping widgets ,analyze telemetry data, and manage deployed products at scale.

- **Sign-up to the app.**
- **Create a new project.**
- **Now you'll get authentication token to your mail make note of that.**
- **Add the required widgets.**
  - Joystick and add virtual pins
  - Buttons and add virtual pins

3.Interface the UltraSonics(HC-SR04) with the Octabrix as shown in the Circuit Diagram.

Octabrix:





It is a development board with the famous and low cost ESP8266 WiFi SoC and minimal space requirement.

Specifications:

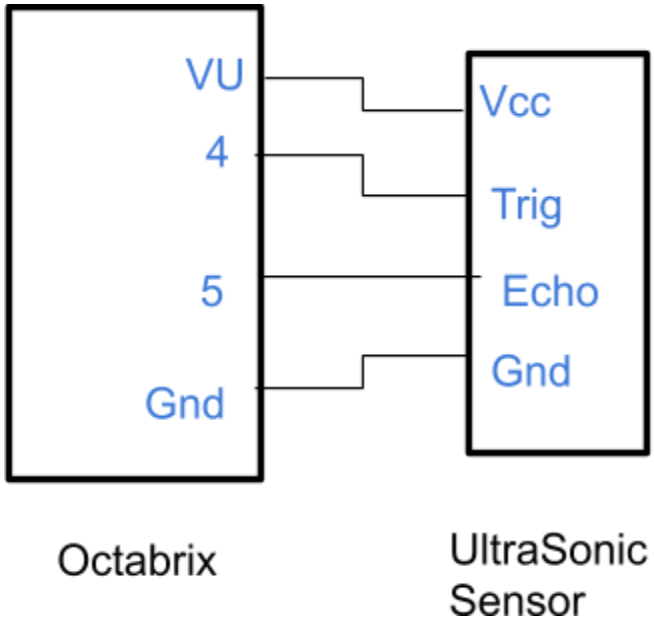
A. ESP8266
B. Operating Voltage(3.3V)
C. Voltage Regulator(5V - 12V)
D. Micro USB input port(Type A)
E. On Board LED
F. RAM(32KB)
G. In-Built Wi-Fi(802.11 b/g/n)
H. Flash Memory(4MB)
I. Octabrix has 10 digital I/O pins with current 12mA(source), 20mA(sink) also capable of PWM Signals with one Analog pin(A0).
J. 10 – Pixel RGB-LED Ring
K. On-board Light Sensor
L. Programmable push Button
M. Vin(To connect to the external voltage)
N. VU(Gives out the voltage supplied through USB
O. 3.3V(Gives regulated voltage output)
P. GND(Ground pin)
Q. The ESP8266 is a low-cost Wi-Fi microchip with full TCP/IP stack and microcontroller capability. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style Commands.

UltraSonic Sensor(HC-SR04):

- As the name indicates, ultrasonic sensors measure distance by using ultrasonic waves.The sensor emits an ultrasonic wave and receives the echo reflected back from the target. Ultrasonic Sensors measure the distance to the target by measuring the time between the emission and reception.

4. Write the code in Arduino IDE.
5. And make script of python with given code.
6. Connect the Octabrix using USB to the Laptop or PC.
7. Upload the arduino code to the Octabrix.
8. Run the python Script.

Hardware circuit diagram



Code

```
/*arduino(Octabrix) code */

int echo = 5;
int trig = 4;
float cm = 0;
long duration = 0;
#define BLYNK_PRINT Serial  //Defines the object that is used for printing(To get the
information about the HDK whether it is connected or not)
#include<BlynkSimpleEsp8266.h>  //To use BLYNK app functionalities
#include<ESP8266WiFi.h> //To connect to WiFi
char auth[] = "Qt1dzc1BzY4HGhIXUJQpRmnzRXI7uN1g"; //The authentication code we got from mail to
connect with the app
char ssid[] = "JioFi_10319F1";  //Wi-Fi Network name
char pass[] = "359@tejesh";   //EncryptionCode(Password)
BLYNK_WRITE(V1)
{
  uint16_t x = param[0].asInt();    //Getting the x coordinate from the joystick
  uint16_t y = param[1].asInt();    //Getting the y coordinate from the joystick
  if(x == 1023)
  {
      Serial.println("Right");
```

```
  }
  if(x == 0)
  {
      Serial.println("Left");
  }
  if(y == 0){
      Serial.println("Down");
  }
  if(y == 1023)
  {
      Serial.println("Up");
  }

}
BLYNK_WRITE(V2)
{
  if(param.asInt() == 1)
  {
      Serial.println("mouseleft");
  }
}
BLYNK_WRITE(V3)
{
  if(param.asInt() == 1)
  {
      Serial.println("mouseright");
  }
}
BLYNK_WRITE(V4)
{
  if(param.asInt() == 1)
  {
      Serial.println("pause");
  }
}
BLYNK_WRITE(V5)
{
  if(param.asInt() == 1)
  {
      Serial.println("a");
  }
```

```
}
BLYNK_WRITE(V6)
{
  if(param.asInt() == 1)
  {
      Serial.println("delete");
  }
}
BLYNK_WRITE(V7)
{
  if(param.asInt() == 1)
  {
      Serial.println("enter");
  }
}
BLYNK_WRITE(V8)
{
  if(param.asInt() == 1)
  {
      Serial.println("lock");
  }
}
void setup()
{
Serial.begin(115200);               //Starting the serial monitor to get the data that is being
sent by the Octabrix
  Blynk.begin(auth,ssid,pass);      //Connecting the Octabrix to the wifi
  pinMode(echo,INPUT);
  pinMode(trig,OUTPUT);
}

void loop()
{
Blynk.run();      //It makes to get the input from the Blynk App
  BLYNK_WRITE(V1);      //Data we got from the Joystick will be sent to the function
  BLYNK_WRITE(V2);      //Left Click of mouse
  BLYNK_WRITE(V3);      //Right Click of mouse
  BLYNK_WRITE(V4);      //Space
  BLYNK_WRITE(V5);      //Getting some Text
  BLYNK_WRITE(V6);      //Delete Key
  BLYNK_WRITE(V7);      //Enter Key
```

```
  BLYNK_WRITE(V8);
  digitalWrite(trig,LOW);
  delay(20);
  digitalWrite(trig,HIGH);
  delay(20);
  digitalWrite(trig,LOW);
  pinMode(echo,INPUT);
  duration=pulseIn(echo,HIGH);
  cm=(duration/2)*0.0343;
  //Serial.println(cm);
  if(cm < 10)
  {
      Serial.println("Right");delay(1000);

  }
  if(cm > 10 && cm < 20)
  {
      Serial.println("Left");delay(1000);
  }
}
/*Python script */
```

```
import serial                                          #Including the library to
manipulate the serial data from the Octabrix
import pyautogui                                       #Including the library to control
the GUI(PC) with the coming serial inputs

Arduino_Serial = serial.Serial("COM4",115200)          #Creating a variable to get
connected with the octabrix and to get the serial data
print ("Connected")
while 1:                                               #Goes on taking serial inputs till
the connection ended
    incoming_data = str (Arduino_Serial.readline())    #Typecasting the collected data
into a string
    print (incoming_data)

    if 'Up' in incoming_data:                          #If the serial data we get is Up
        pyautogui.hotkey('up');                        #Presses the UP arrow key

    if 'Down' in incoming_data:                        #If the data is Down
```

```
        pyautogui.hotkey('down');                          #Presses the Down arrow key

    if 'Left' in incoming_data:                            #If the data is Left
        pyautogui.hotkey('left');                          #Presses the Left arrow key

    if 'Right' in incoming_data:                           #If the data is Right
        pyautogui.hotkey('right');                         #Presses the Right arrow key

    if 'mouseleft' in incoming_data:                       #If the data is mouseleft
        pyautogui.click();                                 #Presses the LeftClick mouse key

    if 'mouseright' in incoming_data:                      #If the data is mouseright
        pyautogui.click(button='right');                   #Presses the RightClick mouse key

    if 'stop' in incoming_data:                           #If the data is pauseHello World
        #pyautogui.hotkey('space')                          #Presses the Space Bar
        #pyautogui.keyUp('space')
        pyautogui.click();

    if 'next' in incoming_data:
        pyautogui.hotkey('shift','n');

    if 'a' in incoming_data:                               #If the data is a
        pyautogui.typewrite("Hello World")                 #We can also give some text input

    if 'delete' in incoming_data:                          #If the data is delete
        pyautogui.hotkey('del');                           #Presses Del key

    if 'enter' in incoming_data:                           #if the data is enter
        pyautogui.hotkey('enter');                         #

    if 'lock' in  incoming_data:
        pyautogui.hotkey('alt', 'f4')


    incoming_data = "";                                    #Here we will make the data empty
every time because it holds previous data
```

Challenges/Bottleneck of the project

- ➢ We can also implement the Gesture Control with more features by using various Sensors(Soli,Kincet,SpectraSymbol Flex Sensor,Bend Sensor).
- ➢ The Octabrix can't connect if there is much distance from the network source.
- ➢ Data transferring from the cloud to node(Octabrix) takes some time.

LoRaWAN® Technology - Building scalable IoT solutions

What is LoRaWAN?

LoRaWAN stands for Long Range Wide Area Network. It's a standard for wireless communication that allows IoT devices to communicate over large distances with minimal battery usage.

It has a wide range of 20KMs in rural areas and 2-3 KMs in urban areas.

LoRaWAN Will be inevitable Technology in Future Smart City Applications together with Internet of Things.

It is based on spread spectrum modulation techniques derived from Chirp Spread Spectrum (CSS) technology.

It is Low Power Protocol that enables the Connection of a Sensor to the internet ensuring the lowest energy consumption possible.

LoRaWAN operates in unlicensed radio spectrum. This means that anyone can use the radio frequencies without having to pay million dollar fees for transmission rights. It is similar to WiFi, which uses the 2.4GHz and 5GHz ISM bands worldwide. Anyone is allowed to set up WiFi routers and transmit WiFi signals without the need for a license or permit.

Features:

1. Low Battery Consumption (Nearly 4-5 years).
2. Wide range.
3. Can be used in creating network with IoT nodes.
4. Low cost.
5. Easy setup.

6. It can serve at a data rate of 27 to 50 KB/s.
7. More feasible than Bluetooth and Wi-Fi and other communication technologies.

Applications:

1. Smart city
    - Smart Lighting
    - Smart Parking & Vehicle Management
    - Fire Detection and Management
    - Waste Management

2. Smart Agriculture
    - Temperature and Moisture Monitoring
    - Water Level Sensors and Irrigation Control
3. Smart Home
    - Security
    - Home Automation for IoT enables smart appliances.
4. Health Care
    - Heat Stroke Detection in field areas
    - Wearables
    - Health Monitoring devices and Management
5. Industrial
    - Radiation & Leak Detection
    - Item Location & Tracking
    - Shipping & Transportation

Implementation of LoRaWAN in Smart Gesture Control:

Future scope of the project

- ➢ We can achieve a lot more applications by using camera as well, by predefining the gestures as commands.
- ➢ Can use the Gesture Control for Home Automation.
- ➢ Can Control Car interiors or Bike Head Lights.
- ➢ Mini Drones.
- ➢ Gesture - Controlled Smart Clock.
- ➢ Designing(Architecture, Textile, Machines)

References

1. https://www.electronicshub.org/arduino-based-hand-gesture-control-computer/
2. https://pyautogui.readthedocs.io/en/latest/introduction.html
3. https://stackoverflow.com/questions/8290913/pyserial-error-5-access-is-denied
4. https://pythonhosted.org/pyserial/pyserial.html#installation
5. https://pyautogui.readthedocs.io/en/latest/install.html
6. https://www.python.org/downloads/windows/
7. https://stackoverflow.com/questions/27596776/installing-and-using-pyserial-in-python-3-4-windows-8-1
8. https://arduino.stackexchange.com/questions/10175/cant-communicate-with-arduino-using-python-windows
9. https://www.youtube.com/watch?v=HQ83Xz2K-ps

Project specific reference -

10. https://www.marxentlabs.com/what-is-gesture-recognition-defined/
11. https://interestingengineering.com/how-gesture-recognition-will-change-our-relationship-with-tech-devices