

MINI PROJECT REPORT
ON
“Gesture Controlled VLC Media Player”

Submitted in
Partial Fulfillment of requirements for the Award of Degree
of
Bachelor of Technology
In
Computer Science and Engineering
By

(Group Number: 19)

Abhay Pratap Singh Parihar (Roll No: 1816410007)
Abhinav Shukla (Roll No: 1816410011)
Akashdeep Soni (Roll No. 1816410036)

Under the supervision of
Sir Kumar Saurabh
(Assistant Professor)

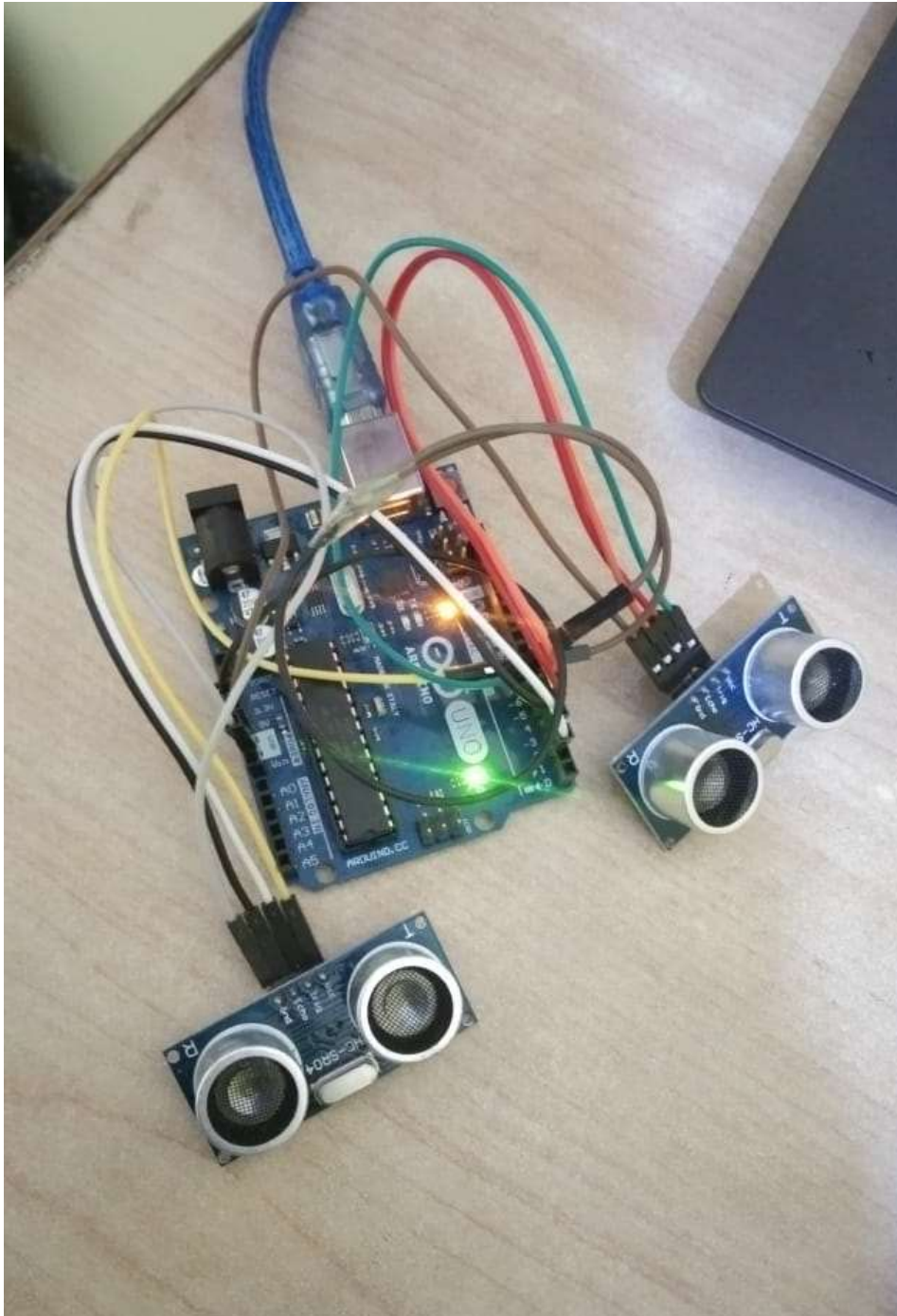


Pranveer Singh Institute of Technology.
Kanpur - Agra - Delhi National Highway - 2
Bhauti -Kanpur - 209305.
(Dr. A.P.J. Abdul Kalam Technical University)

Objective

- The objective of this project is to develop an Arduino and Python based hand gesture control system that can be used to control various multimedia players.
- Here, we will use hand gestures to control various functionality of the VLC media player using the Arduino UNO R3 and 2 Ultrasonic Sensors.

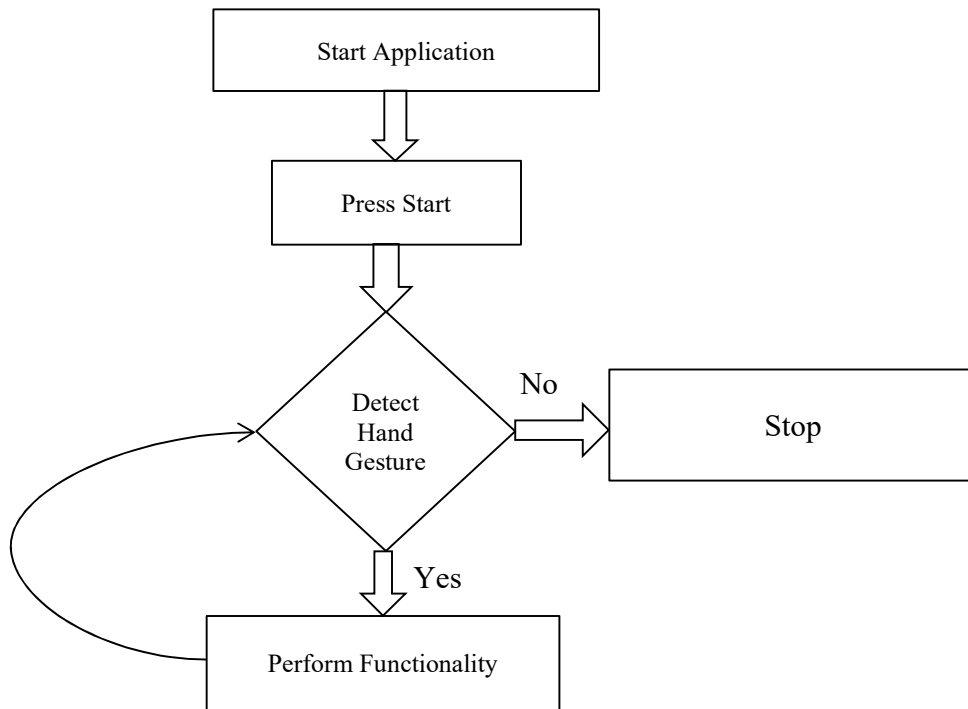




Introduction

- Using Arduino and Python we can control VLC media player with hand gestures.
- We are going to do the same, standing in front of ultrasonic sensors you can control the volume as well as other media playback controls only by using your hands
- To increase the volume we have to raise our left hand in the upward direction holding our hands in front of the ultrasonic sensors, similarly push our left hand to decrease the volume.
- This will revolutionized the era of media controls as we will not need to touch again and again the input devices to control the VLC media player.
- We will be sitting on the place enjoying the video and controlling it by only hand gestures.
- Only we need to have an Arduino and Ultrasonic sensors.

Flowchart



Feasibility Study

The product is easily affordable in terms of price and will be easy to use.

- **Technical feasibility:**

The method of production includes some codes and a development environment including some devices.

Project requirements are a working ultrasonic sensor, few jumper wires, a breadboard along with an Arduino UNO, and a computer.

- **Legal feasibility:**

Our product does not violate any legal requirements. We are developing our software by keeping all the product guidelines and rules regarding copyright and all.

- **Operational feasibility study:**

The setup can be used with any computer having a VLC media player and having USB support and in any situation. It is free of cost so easily accessible by everyone. It only needs an extra setup for the functioning of the gesture controls. The setup includes only Arduino and ultrasonic sensors.

- **Project duration feasibility study:**

The project will complete in approximately 3 weeks.

Technology Used

- The project will be using the most trending language used nowadays i.e. python. We will be using a few libraries of python.
- The libraries that are used are –
 - pySerial (For establishing the communication between Arduino and Python)
 - pyAutoGUI (For controlling keyboard and mouse)
 - Tkinter (For GUI)
- We will also be using Arduino C for programming the Arduino.

Hardware/Software Required (In case of Open Source Software/datasets/Libraries mention URL/Credits/Source)

Hardware used:

- Arduino UNO R3
- Ultrasonic sensors x2
- Jumper wires
- A computer with USB support

Software used:

- Arduino 1.8.13
- Python 3.9
- pySerial
- pyAutoGUI
- VS code

Coding

vlc_control.ino:

- The core Arduino C programming is present in this file.
- We will use Arduino and Ultrasonic sensor to detect distance between hand and sensor.
- Then we will do conditional programming to print various commands which can be later used by python to perform various functionality

```
int trg1 = 2;
int echo1 = 3;
int trg2 = 4;
int echo2 = 5;

void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600);
    pinMode(trg1,OUTPUT);
    pinMode(echo1,INPUT);
    pinMode(trg2,OUTPUT);
    pinMode(echo2,INPUT);
}

void loop() {
    // put your main code here, to run repeatedly:
    int d,dl,dr;
    dl = get_dist(trg1,echo1);
```

```
dr = get_dist(trg2,echo2);

// Play/Pause
if((dl>=40 && dl<=50) && (dr>=40 && dr<=50))
{
    Serial.println("Play/Pause");
    delay(1000);
}
else if(dl>=15 && dl<=25)
{
    delay(100);
    while(dl<=40)
    {
        dl = get_dist(trg1,echo1);
        if(dl<=15)
        {
            Serial.println("VolumeDown");
            delay(100);
        }
        else if(dl>=25)
        {
            Serial.println("VolumeUP");
            delay(100);
        }
    }
}
```

```

else if(dr>=15 && dr<=25)
{
    delay(100);
    while(dr<=40)
    {
        dr = get_dist(trg2,echo2);
        if(dr<=15)
        {
            Serial.println("Forward");
            delay(100);
        }
        else if(dr>=25)
        {
            Serial.println("Backward");
            delay(100);
        }
    }
}

```

```

float get_dist(int trg, int echo)
{
    digitalWrite(trg, LOW);
    delayMicroseconds(2);
    digitalWrite(trg, HIGH);

```

```
    delayMicroseconds(10);  
    digitalWrite(trg, LOW);  
  
    int t,d;  
    t = pulseIn(echo, HIGH);  
    d = (t * 0.034)/2;  
    return d;  
  
}
```

vlc_control.py

This python code is only for testing purpose which does not include GUI.

```
import serial as sl
import pyautogui as pag
from time import sleep

s_obj = sl.Serial('com6', 9600)
print("Starting")
sleep(2)
print("Successful")

while True:
    msg = s_obj.readline().decode('utf-8').strip()
    print(msg)

    if msg == "Play/Pause":
        pag.press('space', interval=0.2)
    elif msg == "VolumeUP":
        pag.press('up')
    elif msg == "VolumeDown":
        pag.press('down')
    elif msg == "Forward":
        pag.press('right')
    elif msg == "Backward":
        pag.press('left')
    else:
```

```
print("Nothing")
```

vlcTk.py

- All the python code is here.
- We will create a GUI using Tkinter.
- We will get Arduino signal using pySerial module.
- After getting signals from Arduino we will control keyboard using pyautogui module.

```
from tkinter import *
from PIL import ImageTk, Image
import serial as sl
import pyautogui as pag
from time import sleep

root = Tk()
root.title("VLC Process Window")
root.geometry("640x360")

s_obj = sl.Serial('com6', 9600)
sleep(2)

# BG Image
bg = ImageTk.PhotoImage(file="bg_img.jpg")
bg_image = Label(root, image=bg).place(relwidth=1, relheight=1)

# Frame
fm = Frame(root, bg="white")
fm.place(x=40, y=60, height=200, width=250)
```

```

title = Label(fm, text="VLC Process Window",
font=("Impact",20,), fg="#d77337", bg='white')
title.place(x=10,y=10)

status = Label(fm, font=("Goudy Old Style",20,),
fg="#d77337", bg='white')

def fun():

    msg = s_obj.readline().decode('utf-8').strip()

    status.config(text=msg)

    status.place(x=10,y=60)

    print(msg)

    if msg == "Play/Pause":

        pag.press('space',interval=0.2)

    elif msg == "VolumeUP":

        pag.press('up')

    elif msg == "VolumeDown":

        pag.press('down')

    elif msg == "Forward":

        pag.press('right')

    elif msg == "Backward":

        pag.press('left')

    root.after(10,fun)

def start():

```



```
root.after(1,fun)

start_bt = Button(fm, text="Start", command = start,
fg="white", bg="#d77337", font=("Times New Roman",20))
start_bt.place(x=80,y=120,width=100,height=40)

root.mainloop()
```

Conclusion

After all the study and project development we are hereby to conclude that:

- We will be using Arduino UNO and ultrasonic sensors and few jumper wires and a few sets of code.
- Our code will be written in Arduino C and Python language and using a few of its libraries
- VLC media player controls will be automated using ultrasonic sensors and other tools
- The user will be able to raise the volume by raising his hand and similar for volume down.
- The user will be able to forward/backward video playback using hand gestures.
- The user will be able to play/pause the video.

Future Scope

In the current world, many facilities are available for providing input to any application some need physical touch and some without using physical touch (speech, hand gesture, etc.). But not many applications are available which are controlled using current and smart facility of providing input which is by hand gesture. By this method, the user can handle the application from distance without using a keyboard and mouse. This application provides a novel human-computer interface by which a user can control the media player (windows) using hand gestures. The application defines some gestures for controlling the functions of the windows player. The user will provide a gesture as input according to the interesting function. The application provides the flexibility of defining user interest gestures for a specific command which makes the application more useful for physically challenged people, as they can define the gesture according to their feasibility. As a prospect of this research we are also going to investigate with a large number of gestures with different persons and motion type hand gestures are developed. We are also going to generalize our system so that it can be useful for other different media players available in the market.

References

- [0] The major reference of the project is Youtube and websites available on internet and Arduino learning through the Udemy and official documentation.

- [1] RuizeXu, Shengli Zhou, Wen J. Li. MEMS Accelerometer Based Nonspecific-User Hand Gesture Recognition. IEEE, 2012. Vol:12, 1166-1173.

- [2] Sheng-Yu Peng, Wattanachote K., Hwei-Jen Lin and Kuan-Ching Li. A Real-Time Hand Gesture Recognition System for Daily Information Retrieval from Internet. In 4th International Conference on Ubi-Media Computing (UMedia), pages 146{151. IEEE, July 2011

- [3] Ginu Thomas. Review of Various Hand Gesture Recognition Techniques. VSRD-IJEECE, Vol. 1 (7), 2011, 374-383