**Control your computer with hand gestures**

Surely the future of the track pad technology lies in motion control, This technology enables us to use certain functions of the laptop with mere gestures.

The Concept was first Published at circuitdigest.com

Here we shall try to control a VLC media player file using gesture control

**Pre requisite knowledge:**

* Python
* The Arduino system

**Software required:**

* Anaconda (Spyder for Python 3.7)
* Arduino IDE

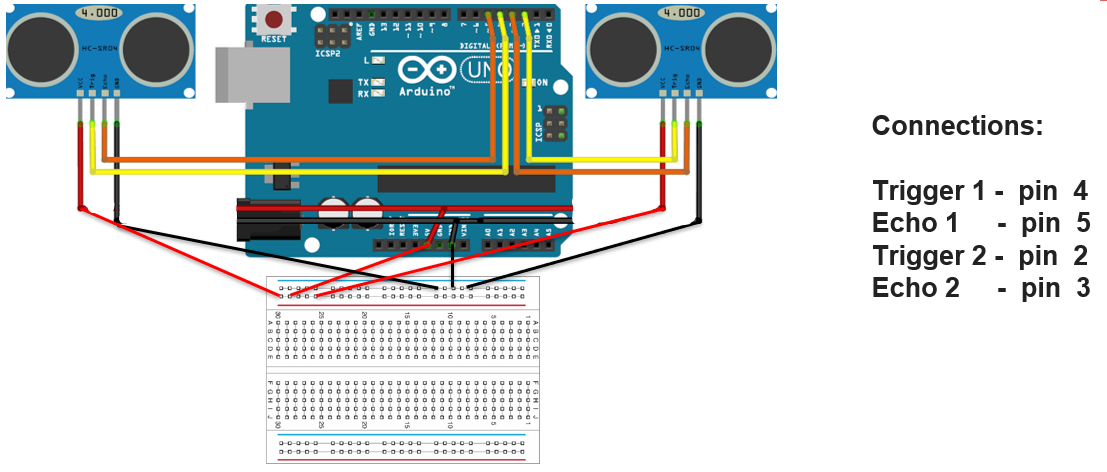
**Construction of the project**

wo ultrasonic sensors shall be placed in on one side of the laptop screen , these sensors will evaluate the distance between our hand and the screen with regard to which the we shall perform certain functions.

The pyautogui library of python shall perform certain functions when arduino sends commands to the computer.

We first connect two ultrasonic sensors to the arduino board . We shall connect the board to the laptop by means of the serial port , this shall serve purposes of communication and power.

Once all that is taken care of , we carefully attach the sensors on one side f the laptop screen by means of a tape. We shall now proceed with the programming.



**Programming the arduino:**

As we already know , any function derived from this procedure is a result of gestures, in order for the computer to understand these gestures it has to sense the distance between the hand and the screen , this purpose is fulfilled by the means of an ultrasonic sensor. Since we plan to implement this on a media player, we must therefore implement certain gestures that correspond to functions in the media player.

These gestures are described below

**Gesture** 1: When the hands are held sufficiently far from the sensor then the video pauses or plays

**Gesture** 2: when the right hand is placed up at a sufficiently far distance

**Gesture** 3: When left hand is placed up at a sufficiently far distance then the video should Rewind one step.

**Gesture** 4: When right hand is placed up before the sensor at a particular near distance and then if moved towards the sensor the video should fast forward and if moved away the video should Rewind.

**Gesture** 5: when the left hand , kep at a sufficiently far distance is moved towards the sensor it increases the volume and when moved away it decreases the volume.

I/O Pin Allocation:

The US sensors are connected to digital pins 2,3,4,5 and powedred by 5V Pin

The Serial communication between Arduino and python takes places at a baud rate of 9600

Arduino Code:

const int trigger1 = 2; //Trigger pin of 1st Sesnor

const int echo1 = 3; //Echo pin of 1st Sesnor

const int trigger2 = 4; //Trigger pin of 2nd Sesnor

const int echo2 = 5;//Echo pin of 2nd Sesnor

long time\_taken;

int dist,distL,distR;

void setup() {

Serial.begin(9600);

pinMode(trigger1, OUTPUT);

pinMode(echo1, INPUT);

pinMode(trigger2, OUTPUT);

pinMode(echo2, INPUT);

}

/\*###Function to calculate distance###\*/

void calculate\_distance(int trigger, int echo)

{

digitalWrite(trigger, LOW);

delayMicroseconds(2);

digitalWrite(trigger, HIGH);

delayMicroseconds(10);

digitalWrite(trigger, LOW);

time\_taken = pulseIn(echo, HIGH);

dist= time\_taken\*0.034/2;

if (dist>50)

dist = 50;

}

void loop() { //infinite loopy

calculate\_distance(trigger1,echo1);

distL =dist; //get distance of left sensor

calculate\_distance(trigger2,echo2);

distR =dist; //get distance of right sensor

//Uncomment for debudding

/\*Serial.print("L=");

Serial.println(distL);

Serial.print("R=");

Serial.println(distR);

\*/

//Pause Modes -Hold

if ((distL >40 && distR>40) && (distL <50 && distR<50)) //Detect both hands

{Serial.println("Play/Pause"); delay (500);}

calculate\_distance(trigger1,echo1);

distL =dist;

calculate\_distance(trigger2,echo2);

distR =dist;

//Control Modes

//Lock Left - Control Mode

if (distL>=13 && distL<=17)

{

delay(100); //Hand Hold Time

calculate\_distance(trigger1,echo1);

distL =dist;

if (distL>=13 && distL<=17)

{

Serial.println("Left Locked");

while(distL<=40)

{

calculate\_distance(trigger1,echo1);

distL =dist;

if (distL<10) //Hand pushed in

{Serial.println ("Vup"); delay (300);}

if (distL>20) //Hand pulled out

{Serial.println ("Vdown"); delay (300);}

}

}

}

//Lock Right - Control Mode

if (distR>=13 && distR<=17)

{

delay(100); //Hand Hold Time

calculate\_distance(trigger2,echo2);

distR =dist;

if (distR>=13 && distR<=17)

{

Serial.println("Right Locked");

while(distR<=40)

{

calculate\_distance(trigger2,echo2);

distR =dist;

if (distR<10) //Right hand pushed in

{Serial.println ("Rewind"); delay (300);}

if (distR>20) //Right hand pulled out

{Serial.println ("Forward"); delay (300);}

}

}

}

delay(200);

}

**Code Snippet:**

const int trigger1 = 2; //Trigger pin of 1st Sesnor

const int echo1 = 3; //Echo pin of 1st Sesnor

const int trigger2 = 4; //Trigger pin of 2nd Sesnor

const int echo2 = 5;//Echo pin of 2nd Sesnor

void setup() {

Serial.begin(9600);

pinMode(trigger1, OUTPUT);

pinMode(echo1, INPUT);

pinMode(trigger2, OUTPUT);

pinMode(echo2, INPUT);

}

We have to write a function:

calculate\_distance()

to calculate the distance between the hand and the sensor

**Code Snippet:**

/\*###Function to calculate distance###\*/

void calculate\_distance(int trigger, int echo)

{

digitalWrite(trigger, LOW);

delayMicroseconds(2);

digitalWrite(trigger, HIGH);

delayMicroseconds(10);

digitalWrite(trigger, LOW);

time\_taken = pulseIn(echo, HIGH);

dist= time\_taken\*0.034/2;

if (dist>50)

dist = 50;

}

We must use two variables distL and distR which update the current distance value

**Code Snippet:**

calculate\_distance(trigger1,echo1);

distL =dist; //get distance of left sensor

calculate\_distance(trigger2,echo2);

distR =dist; //get distance of right sensor

if both the hands are placed at a distance of 40 cm, the word **“Play/Pause”** will be sent out through serial port

**Code Snippet**

if ((distL >40 && distR>40) && (distL <50 && distR<50)) //Detect both hands

{Serial.println("Play/Pause"); delay (500);}

If the Right hand alone is placed before the module the word **“Forward”** will be sent out through serial port. if it is left hand the word **“Rewind”** will be sent out through serial port.

**Code Snippet:**

if ((distL >40 && distL<50) && (distR ==50)) //Detect Left Hand

{Serial.println("Rewind"); delay (500);}

if ((distR >40 && distR<50) && (distL ==50)) //Detect Right Hand

{Serial.println("Forward"); delay (500);}

The code for volume is shown below, Based on the action, here the word “Vup” or “Vdown” will be sent out through serial port

**Code Snippet:**

//Lock Left - Control Mode

if (distL>=13 && distL<=17)

{

delay(100); //Hand Hold Time

calculate\_distance(trigger1,echo1);

distL =dist;

if (distL>=13 && distL<=17)

{

Serial.println("Left Locked");

while(distL<=40)

{

calculate\_distance(trigger1,echo1);

distL =dist;

if (distL<10) //Hand pushed in

{Serial.println ("Vup"); delay (300);}

if (distL>20) //Hand pulled out

{Serial.println ("Vdown"); delay (300);}

}

}

}

**PYTHON PROGRAMMING :**

**In order to code we must use the pyautogui module**

**Installing pyautogui module for windows using Anaconda command prompt:**

* Open Anaconda Command prompt and use the command to install the pyautogui module

“*python –m pip install pyautogui*”

**For Mac:**

Execute the following code on the terminal:

* 1. pip install --upgrade pip
  2. conda install -c conda-forge pyautoguipip
  3. install pyserial

To find the port addresses on a mac use the terminal command:

python -m serial.tools.list\_ports

**Python Code**

import serial #Serial imported for Serial communication

import time #Required to use delay functions

import pyautogui

ArduinoSerial = serial.Serial('com18',9600) #Create Serial port object called arduinoSerialData

time.sleep(2) #wait for 2 seconds for the communication to get established

while 1:

incoming = str (ArduinoSerial.readline()) #read the serial data and print it as line

print incoming

if 'Play/Pause' in incoming:

pyautogui.typewrite(['space'], 0.2)

if 'Rewind' in incoming:

pyautogui.hotkey('ctrl', 'left')

if 'Forward' in incoming:

pyautogui.hotkey('ctrl', 'right')

if 'Vup' in incoming:

pyautogui.hotkey('ctrl', 'down')

if 'Vdown' in incoming:

pyautogui.hotkey('ctrl', 'up')

incoming = "";