Core Project Report

on

Application Control using Hand based gestures

Submitted by

Sai Sankeerth Reddy	Sai Sumanth Reddy	Vivek Nimmagadda
15B00253	15B00254	15B00288
CSE-1	CSE-1	CSE-1

2015 Batch

Under the guidance

Of

Ms. Tamanna Sehgal

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF ENGINEERING AND TECHNOLOGY

BML Munjal University
Gurugram-122413, India
December, 2018

Acknowledgements

We are indebted to BML Munjal University for their constant supervision as well as for providing necessary information regarding the project. We are extremely thankful to Ms. Tamanna Sehgal for giving us all the support and guidance which made us traverse this far in completing our project. We would also like to express our gratitude to the other panel members for their valuable feedback which made us complete our project.

BML MUNJAL UNIVERSITY, GURGAON, HARYANA

CANDIDATE'S DECLARATION

We hereby declare that the work done in this core project entitled **"APPLICATION CONTROL USING HAND BASED GESTURES"** submitted in the fulfillment of the completion of the 7th semester of Bachelor of Technology (B.Tech) program in Computer Science & Engineering at BML Munjal University, Gurugram is an authentic record of our original work carried out under the guidance of Ms. Tamanna Sehgal due acknowledgements have been made in the text of the project to all other material used. This core project work was done in full compliance with the requirements and constraints of the prescribed curriculum.

Place: Gurugram, Haryana
Date: 10th December, 2018

C O Sai Sumanth Reddy	Chirra Reddy Sai Sankeerth Reddy	Vivek Nimmagadda
15B00254	15B00253	15B00288

Abstract

The purpose of this project was to develop a music application which utilizes Hand Gesture Recognition in aspect to Human Computer Interaction to eliminate the problems faced by current systems which involve conventional peripheral devices. The project makes use of two HC-SR04(ultrasonic sensors) and an MPU 6050(gyroscope and accelerometer sensor) which characterize the gestures through the Doppler shifts which are caused by the ultrasonic tones emitted by the ultrasonic sensors and the direction of the hand relative to the MPU 6050. The gestures which have been characterized can potentially be used to control our application. By adding this hand gesture mechanism, this project is not only easy to use, but also has a sophisticated feel to it.

Table of Contents

1. Introduction

Gesture technology has revolutionized the electronics Industry since its arrival. This technology allows systems to capture and construe gestures made by human actions as commands has paved the way to numerous applications and devices around us which utilize these gestures to provide us with services such as the basic necessities of comfort. Instead of the traditional and cumbrous task of typing with keys, a motion sensor can be used to perceive and interpret our movement as the source of data input. These devices which control computers/laptops through gestures are coming into high demand recently.

Some of these examples include Smartphone applications, Automated homes, and Smart Television setups etc. The Multinational Technological Moghul, Microsoft's gesture recognition platform 'Kinect' allows users to operate computers purely through gestures and speaking. A technology called the leap-motion enables us to control certain functions of a particular application according to the necessities of the application.

Example of a few gestures include:

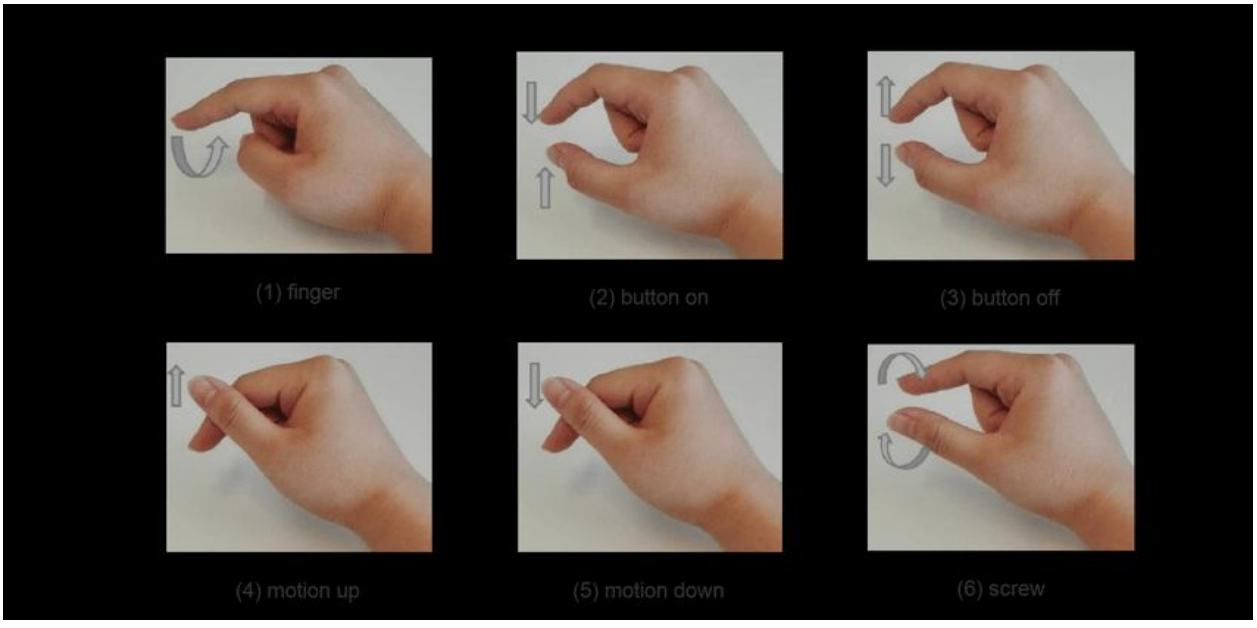

Figure 1.1

Sometimes, the gesture detection is somehow implicitly encoded in the measurement input signals and there may not be an unambiguous link between the gesture motion and measured quantities. These type of ambiguities could be solved using recognition and Classification phase of Machine Learning.

This technology, after looking at the feasibility can also be implemented to perform simple tasks on a device by using simple gestures like waving our hand, placing your hand at a particular distance from the sensor etc. Using this technology a simple music player application can be built that uses such simple gestures to perform certain functions and machine learning can be used to recommend playlists to the users.

2. Problem Definition

Windows, icons, menus, pointers (WIMP) together with the keyboard and the mouse, have been largely successful in providing the flexibility for use of computers. But this is rather a restrictive way of interaction with the end user systems. Mainly in the case of Multimedia applications which involve various hotkeys for different functionalities, such as “Alt+Enter” for enlarging the screen and “Shift + Upward Arrow” for increasing the volume etc.

Thus, the present technology, i.e, the Gesture-based Computing came into existence. This technology has shown promise and is also in high demand presently. These gestures can be used to eliminate the intricacies involved in remembering hotkeys by bringing about a Perceptual User Interface (PUI) for the Application.

Problem Statement:

The problem in gesture technology is the inability of cameras to be able to detect gestures or motion in real time in constrained environments deteriorating the real world applications that can be implemented otherwise.

A number of challenges which might occur while implementing these gestures include:

2.1 Detecting distinct fingers

The system may fail to recognize distinct fingers in the left-hand side of the image in Figure 2.1, although both the left-hand and right-hand side of the image represents the same gesture.

Figure 2.1

2.2 Detecting the full gesture

An ambiguity might occur in detecting and approximating the gesture shown on the left-hand side of the image in Figure 2.2, even though the left-hand and right-hand side of the image represents the same gesture.

Figure 2.2

2.3 Detecting blurry images

The image of the gesture might be blurred often times due to poor camera quality or other factors involved like in Figure 2.3, which in turn results in the system failing to recognize that specific gesture.

Figure 2.3

2.4 Bad lighting conditions

Bad lighting conditions such as excessive or poor illumination onto the hand may be challenging to the user's system and may bring about ambiguities in detecting the gestures.

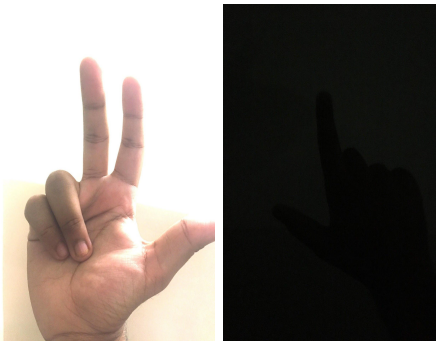

Figure 2.4

Apart from the challenges mentioned above, other significant problems which can arise are:

- **Size:** Size of the hand may differ from person to person and system may not recognize.
- **Location:** the system may not detect the gesture sometimes if not in the center.
- **Occlusion:** System's camera might be blocked which ultimately leads to the failure of gesture recognition.

3. Objective

Goal: To develop a fully functional multimedia application suited for practical use which performs basic actions based on motion-based gestures even in constrained environments by interfacing Arduino and python modules and recommending playlists using machine learning techniques.

The ultimate goal of our project is to develop a music player application which performs basic actions like play/pause music, fast forward, rewind, change volume etc. by detecting motion-based gestures, i.e, detecting gestures using sensors rather than a real-time camera. To make it viable for practical use it should be able to perform and detect the gestures even in constrained environments. Using motion-based gestures will help eliminate intricacies in gesture detection in constrained environments. Along with being able to detect gestures and perform actions, it should be able to recommend playlists to the user which can be implemented using machine learning techniques. As a result of using these gestures, it simplifies the interaction process of humans with modern-day computers. This is also very

useful where multiple tasks need to be performed at the same time and also very useful in cases when partially blind people use the application. They do not have to go through the hassle of pressing buttons or touching the screen, they just have to perform simple gestures like waving a hand or facing a palm towards the system.

4. Literature Survey

Ever since the gesture technology was introduced, a substantial amount of research has been done in the field and this section highlights different researches that have already been done that are useful for our project.

4.1 Gesture Recognition

The methods and systems that are used to interact with computers have taken an innovative turn with the emergence of Human-Computer Interactions (HCI). One such system includes a project by Deepak K. Ray et al, 'Hand Gesture Recognition using Python[1]' is a simple but effective implementation of a setup which recognizes Hand Gestures utilizing Python modules namely Pyautogui.

4.2 Gestures using Motion and Image comparison

Another intriguing approach that can be carried out with the help of Human-Computer Interaction (HCI) is being able to control the mouse and carry out various functions with it. This has been explained in 'Gesture Based Interface Using Motion and Image Comparison[2]' by Shany Jophin et al. The paper explains a new approach to implement mouse functions using a real-time camera that can recognize gestures through motion and image comparison. They used a camera, colored substance, image comparison technology, and motion detection technology to control mouse movement and its functions.

4.3 Gesture Recognition using Ultrasonic Sensors

The motivation behind our decision to choose Ultrasonic sensors instead of a Camera was mainly due to the work of Kaustubh Kalgaonkar and Bhiksha Raj's 'One-handed Gesture Recognition using Ultrasonic Doppler SONAR[3]'. A new device is presented by this paper which is based on ultrasonic sensors to recognize one-handed gestures which are distinguished and characterized through a sonic tone emitted by the transmitter. The device consists of an Acoustic Doppler Sonar(ADS) which is a setup that includes three ultrasonic receivers and a transmitter. The ultrasonic tone emitted by the transmitter is reflected by the user's moving hand and causes a Doppler shift that corresponds to the current velocity of the

user's hand. The signals which are obtained by the receivers are used for recognizing a particular gesture.

4.4 Hand Gestures Interfacing

In Anupam Agrawal and Siddharth Swarup Rautaray's, 'A Vision-based Hand Gestures Interface for Operating VLC Media Player Application[4]', the K nearest neighbor algorithm has been exploited to recognize various gestures and the distinguished gestures have been used to control VLC media player features like Play, Pause, Full Screen, Stop, Increase Volume and Decrease Volume. Also, Lucas Kanade Pyramidal Optical Flow algorithm, which recognizes moving points in the input image has been used to recognize hand from the input video. Later, K-means is used to find the center of the hand. This system uses a database that stores various hand gestures. On comparing the input with these stored images of the gestures, VLC Media Player is controlled. Albeit it has a simple implementation, the present application of this system is less robust.

4.5 Gestures using Accelerometer and Gyroscope

In 'Studying rotational dynamics with a smartphone—accelerometer versus gyroscope[5]', Brasken Mats and Ray Pörn discuss the different functions that can be carried out by an accelerometer and a gyroscope comparing both of them in terms of functionalities in rotational dynamics. They describe the uses of accelerometers and gyroscopes in smartphones and their possible uses for carrying out different functions for gestures.

4.6 Clap Gesture Detection

Another important gesture that can be used to perform functions is a clapping gesture. It is a very famous gesture that can be seen in movies and a lot of research has been done in this field including a heavy contribution from Cyril Arnould et al. who have written 'Clap detection with micro-controller[6]'. In this paper, they discuss how they took various samples of sounds other than claps to train the model and then using the microcontroller to detect claps based on a certain threshold. The main problem was of the complexity associated due to the decision variable used for the training the model.

4.7 Machine Learning for playlist recommendations

An important functionality in a Music Application would be the ability to recommend playlist based on the user's preferences or likings. Tushar Singh Bohra et al. did exactly that in their paper 'Segmenting Music Library for generation of Playlist using Machine Learning[7]'. People like to listen to different songs based on the work they are doing and using this phenomenon, the authors were able to segment the songs and group the songs that had a similar emotional effect. Though they didn't include important parameters like Genre, Artist,

Album, Tempo etc., they were successful in generating playlists with similar emotional effect with reduced redundancy compared to playlists with no segmentation.

5. Methodology

The methodology followed is quite different for the different modes that will be implemented. For simplicity, this section is further divided into subsections each explaining about the different modules present in the project.

5.1 Mode 1

In mode 1, an Arduino board will be needed with two Ultrasonic sensors and a python IDLE which is run on a laptop. Two Ultrasonic sensors on top of the laptop are connected to the Arduino board which is powered by an I/O serial port of the laptop. Using the Arduino software different conditions for the gestures will be defined and uploaded onto the Arduino board. The sensor detects the gestures based on conditions defined and a keyword specific to a gesture performed is printed on the serial monitor. Now a code is written in the python IDLE which enables the system to read the text printed on the serial monitor and accordingly perform an action using the Pyautogui module of python according to the text printed. The action is implemented based on the gestures performed.

5.2 Mode 2

In mode 2, an Arduino Nano board and an MPU6050, which is a gyro and accelerometer sensor are needed along with a glove and Python IDLE that is run on a laptop. The sensor is used to calculate the coordinates in 3D space. This sensor is put on a glove and similar to mode 1, connected to an Arduino which is powered by an I/O serial port in the laptop. Similar to mode 1, code is to be written to define the gestures and actions to be done in Arduino and python IDLE respectively. Whenever the sensor detects a change in coordinates within the gesture constraints, it prints onto the serial monitor which is read by the system through the python IDLE. Again, a specific action is implemented based on gestures detected by the MPU6050 sensor.

5.3 Music Player Application

The gestures that are detected and performed in modes 1 and 2 are implemented in a music player application that is to be developed. To develop the music player Python libraries like ‘Tkinter’ for GUI and ‘pygame’ are to be used for the music player. Using these python modules, the application will be developed which will allow the users to choose different songs and perform different functions like, pause/play song, increase volume, decrease

volume, next song, previous song etc. These functions will have gestures linked to them through interfacing the modes to this application.

5.4 Machine Learning for Playlist recommendation

In the music player application, the user will have an option to get a recommended playlist for every type of work or mood. The songs will be classified and grouped into lists which will have the same emotional effect. Using data of liked songs or songs that have already been heard as training data, and using parameters like genre and ratings for classification, a playlist can be recommended to the user.

5.5 Additional Features

An additional feature has been added, which is an in-app Tic-Tac-Toe game, in order to keep the user occupied while listening to his/her favourite songs. With this feature, the users now have the choice to play a game with friends while still listening to the song of their choice. All the functionalities and gestures supported by the Music player are still active while the game is in play.

6. Project Requirements

6.1 Functional Requirements

- The system should take in hand gestures as input and display the specified functionality as the output.
- The system should be able to recognize the specified gestures according to the distance calculated by the Ultrasonic sensors.
- The system should be able to recognize the specified gestures according to the relative position of the hand with respect to the accelerometer and gyro sensor.
- The system should be able to suggest playlist recommendations to the user according to the user's preferences by adopting Machine Learning algorithms.
- The system should perform the basic functionalities such as Play, Pause and Skip etc. assigned to the GUI buttons on the application.

6.2 Non-Functional Requirements

- Usability
- Effectiveness
- Efficiency
- Robustness
- Performance

7. Results and Discussions

A fully functional Music Player Application has been successfully implemented which displays the music library in a list box and includes fundamental GUI functionalities like pause/play, next/previous songs, stop music and quit application. Each of these functions along with volume control have been integrated to a gesture which performs its specified function. Crucial features like Menubar, Toolbar and Status bar that are usually available in Mainstream Media Players have been included in this Application as well to provide additional functionality. The application was able to accurately recognize the gestures through the Ultrasonic Sensors and through the MPU6050 sensor. Along with this, the application was able to recommend playlist according to the users likings. Apart from controlling music using the application, the user has access to additional features that have been embedded in the toolbar like playing a game of Tic-Tac-Toe or viewing some fun facts on music.

Figure 7.1
Music Player GUI

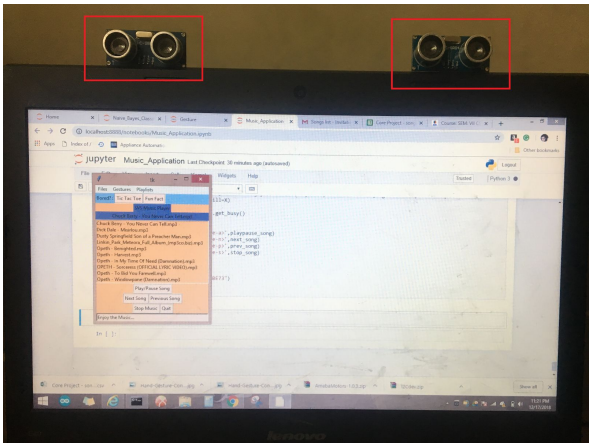

Figure 7.2
Mode 1: Using ultrasonic sensors

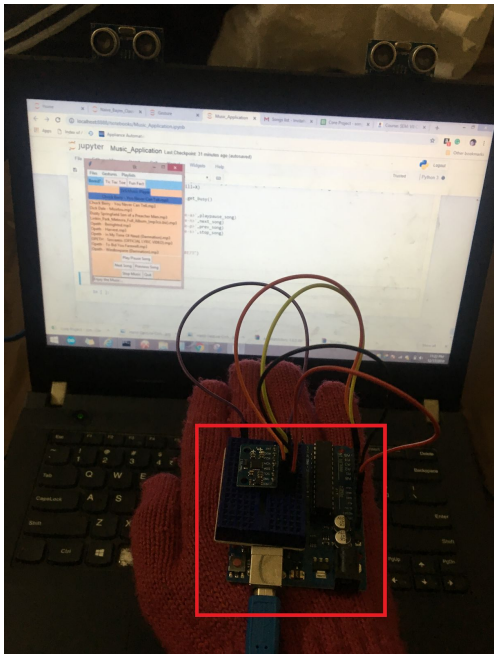

Figure 7.3
Mode 2: Using MPU 6050 (Accelerometer and Gyro sensor)

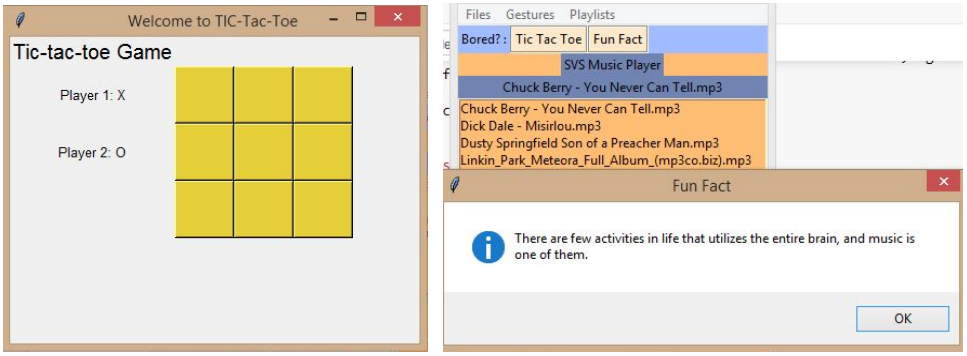

Figure 7.4
Tic-Tac-Toe game(left) and Fun fact(Right)

By implementing this project it has been observed that 2 or more components of the above mentioned literary survey were combined which in turn helped in solving the specified problems upto an extent. This project solves the problem of recognizing gestures even in bad lighting and recommends playlists to the user using Machine Learning algorithms. It has also been observed that in some cases it becomes difficult to recognize gestures through the ultrasonic sensors due to various environmental factors like temperature and physical factors like hand position in front of sensor etc. This is an area of research which can be worked on to improve the efficiency of the gesture recognition.

8. Future Scope

The use of hand-based gestures to control applications has shown great promise and potential for future scope. Although the use of these gestures has been limited to control various operations of our Music application, this technology could be exploited in many areas in the near future. There are many improvements that can be made to our system like improving the sensitivity of the gesture recognition, adding other features such as changing the size of the active windows a person's palm and multiple fingers. The User Interface of our application could also be improved by using Django, which supports many rich illustrations, unlike Tkinter which is bound to simple interfaces. We plan to add a new functionality which allows the user to add songs to the application, to increase the efficiency in recommending playlists.

9. References

[1] Deepak K. Ray, et al. “Hand Gesture Recognition Using Python”. *International Journal on Future Revolution in Computer Science & Communication Engineering (Ijfrcsce)*, 2018, Volume 4 Issue 6.

[2] Shany Jophin, et al. “Gesture Based Interface Using Motion and Image Comparison.” *International Journal of Advanced Information Technology (IJAIT)*, June 2012.

[3] Kaustubh Kalgaonkar, and Bhiksha Raj. “One-Handed Gesture Recognition Using Ultrasonic Doppler Sonar.” *IEEE Xplore Digital Library*, 26 May 2009, Document 4959977.

[4] Siddharth Swarup Rautaray, and Anupam Agrawal. “A Vision Based Hand Gesture Interface for Controlling VLC Media Player.” *International Journal of Computer Applications*, Nov. 2010, Volume 10 No.7.

[5] Mats Braskén and Ray Pörn, “Studying rotational dynamics with a smartphone—accelerometer versus gyroscope”, *IOP Science,* 6 June 2017, Physics Education Volume 52 No. 4.

[6] Cyril Arnould, et al. “Clap Detection with Microcontroller.” *Technische Informatik Und Kommunikationsnetze*, 12 June 2013.

[7] Tushar Singh Bohra, et al. “Segmenting Music Library for Generation of Playlist Using Machine Learning.” *IEEE Xplore Digital Library*, 8 Oct. 2015, Document 7293429.