**Project Report**

Gesture Keyboard

Contents

[Introduction 3](#_Toc23999019)

[Problem Statement 3](#_Toc23999020)

[Motivation for the project 4](#_Toc23999021)

[Project Diagrams 4](#_Toc23999022)

[Implementation 6](#_Toc23999023)

[Project Snapshots 6](#_Toc23999024)

[Project Development Time Schedule 7](#_Toc23999025)

[Learning and Reflections 8](#_Toc23999026)

[Conclusion 8](#_Toc23999027)

[Limitations 8](#_Toc23999028)

[Future Enhancements 8](#_Toc23999029)

[References 9](#_Toc23999030)

# Introduction

We are making a gesture keyboard which will be inputting some characters to a device. The general idea behind this is that the user would press/click a button and hold it down while dragging the pointer across the letters in the desired word before releasing the button to complete the word. Gesture Keyboard is a device that translates the gestures into characters . It sends the data to a computer that using accelerometer , with a Machine Learning algorithm, translates the motion readings into characters . The library itself is written in Python and uses the Scikit-learn library for the SVM(machine - learning) algorithm to classify the signals into characters .

# Problem Statement

The Specially - abled community people have miserable lives. They have a lot of difficulty in performing day to day and necessary tasks. Also, one of the most faced difficulty by them is not being able to communicate, which stops them from leading a life.

The Gesture Keyboard created by us, helps facilitate communication between the specially abled community, as one can convey whatever he/she wants to convey. This makes their lives somewhat easier.

Also, in corporate jobs and in educational fields, there is often a large amount of typing work. It can sometimes become a hectic task, and might use up time, which could be devoted to perform better yielding tasks.

Hence, the project, Gesture Keyboard could also be used as a typing aid, and can prove to be a handy tool for those who work in corporate fields, and educational fields, as it would reduce effort, and save time.

# Motivation for the project

The main aim of building this project is to create a natural interaction between the humans and computers where the recognised characters translated from the gestures given by a human can be understood from Arduino pro micro with accelerometer and Bluetooth module, by implementing SVM-support vector machine (machine learning) in python.

This gesture keyboard does a gesture recognition which translates it into characters/letters has an extraordinary incentive in numerous applications, for example - communication via gestures recognition, enlarged reality (computer-generated reality), gesture-based communication translators for the defence sectors, gaming sector and robot control and especially beneficial for people who are deaf , or can't talk or see as it allows interactive communication with people around them.

# Project Diagrams

Block diagram:

A close up of text on a whiteboard

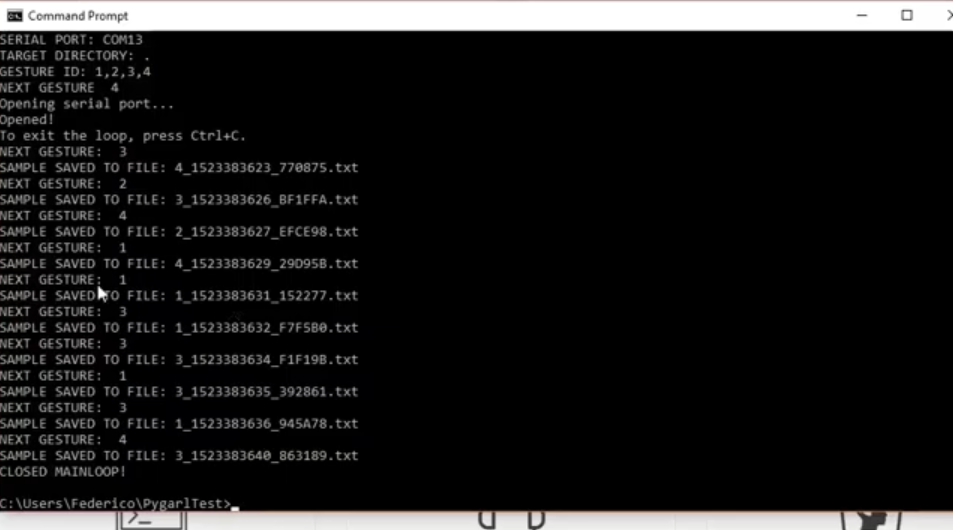
Description automatically generated

# Implementation

* The components used for the project are: - an Arduino, an MPU 6050 Accelerometer, a breadboard, connecting wires.
* We are using the library Pygarl, used for gesture analysis and recognition and the library sklearn. These libraries are used to build gesture recognition systems.
* Algorithms of Machine Learning will be used to train the model so that it can identify the gesture accordingly.

**This worked earlier in python 2 but our soldered device was damaged so we have to go for solderless and without Bluetooth module , so as you can see in 2nd picture instead of output it started showing some encrypted values so we couldn’t train our model.**

**This is actually work in progress and its 80% done!**

 A close up of text on a black background

Description automatically generated

# Project Snapshots

A circuit board

Description automatically generated

Fig:- Arduino Set-up

A person sitting at a table using a computer

Description automatically generated

Fig:- Device Set-up connected

# 

# Project Development Time Schedule

Project Planning:

24th August 2019 - Deciding what project we should do, then we came up with the idea of Pedestrian Detection using tensor-flow.

29th August 2019 - We started researching on the idea we came up with and started installing necessary software applications and python libraries required for the project.

20th September 2019 to 1st October 2019 - For Pedestrian detection we started doing the things required for this project i.e. updating drivers and installing tensor-flow again since it was failed.

28th September 2019 - We were facing problems while downloading tensor-flow (CPU version) , and for training the model with tensor-flow CPU version which is very slow process so we have to drop this idea of building a 'pedestrian detection' and also we didn't have the access to compatible systems(super computer) for training , even if we try to train the small models in our laptop it still won't work as we would have lost our laptop's motherboard.

2nd October 2019 - Now we have dropped the previous idea and had to come up with the idea of making gesture keyboard using Arduino which does a gesture recognition then it is translated into characters.

4th October 2019 - Now we are clear with our idea about the project, so we divided the work and started researching and working on its code.

# Learning and Reflections

Initially, we just had a little knowledge of Arduino and python, in the process of building the Gesture Keyboard system, we learnt new concepts, techniques, and improved our Computer Science knowledge. We learnt basics of SVM (support vector machine), and came to know how to develop, test and train the model.

We got acquainted to the installation and worked on the library Pygarl. Enhanced our C and Python skills with the help of the project. Also, we made a step to enter the fascinating technology of Gesture Recognition.

# 

# Conclusion

We achieved some of the goals we set during the project.

Our model is having some limitations, and the coding part having some errors, that need to be fixed, still looks to perform well and satisfy most of the needs, that are expected from the Gesture Keyboard.

# Limitations

Some **Limitations** of the project are: -

1. Gesture Keyboard doesn’t involve sound, so the text printed on the screen couldn’t be read out loud.

2. The system requires to be connected to a Power bank, so it’s battery charging is an issue in case if we use Bluetooth module

3. since we made it without soldering and Bluetooth module so its design is little bit complex(connections and wiring).

# 

# Future Enhancements

Creating the hands-free version of the Gesture Keyboard: - Presently, we need to hold the Arduino system in our hand and move it using our whole arm, later, we see to eliminate the need to hold the system.

Maybe we’ll try to make it more handy where one has to use only his fingers.

# References

1. <https://github.com/gesture-keyboard>
2. <https://youtu.be/wUYZWPJRUbk>
3. <https://www.geeky-gadgets.com/arduino-gesture-keyboard-13-03-2017/>
4. University Seniors and faculties: - The seniors helped us with the errors in our codes & lab faculty helped us with the part of soldering.