Arduino Based Motion Tracking Keyboard Using Machine Learning

Aditya Priyadarshi[1], Bharath V[1], Aptha KS[1], Sanjit S[1]
Sahana V[2]
[1]UG Students, [2]Assistant Professor
[1,2]Department of ISE,
JSS Academy of Technical Education, Bangalore, Karnataka, India
Email: *adityapriyadarshiex@gmail.com*

DOI:

Abstract

In spite of the emergence of various sensors and input devices, the current methods have not been able to provide an accurate recognition system for gestures. Recognition system such as the one in Kinect using machine learning algorithms shows some of the use cases for these gestures to newer system and domains. The one in our paper provides a far more accurate result along with faster computational speed as it uses the support vector machine (SVM) algorithm. This motion recognition system tracks motion made in mid-air by device in a 3D space, log its speed, angular velocity, distance covered and some other variables at real time, and in turn convert the device captured data of motion into characters of English alphabets. The device presented in this paper presents solution based on support vector algorithms and discusses about some concepts raised from the device.

Keywords: *motion recognition system, support vector machine*

INTRODUCTION

The most common techniques in the communication between the human and computer are performed via simple devices such as mouse and keyboard. Automated gesture tracking devices can serve as a very efficient tool in the case to serve social purpose like the sign language communication for the deaf and verbally impaired person. Although the current devices are intended for easy and user friendly interaction with the user, still they lack when there is a massive data input at some reasonable speed. For these reasons, we need some kind of a new input method working at a greater degree as people communicate that is through gestures.

The problem of understanding the sign language, besides being challenging, can be of great social interest as it would be helpful to the deaf and verbally impaired people to get discrete, personal and effective service in the day to day life. We envision a system to track gestures made in mid-air, where a computer will automatically translate the motion made by the device by the user to characters of the English alphabet system. The problem of gesture recognition deals with the detection, analysis and recognition of character from sequence of motion of device.

The hardware needed is not complex and inexpensive as it consists of an Arduinopro micro, an mpu-6050 accelerometer and an hc-06 module for Bluetooth communication.

Recognition of gestures is a compound problem that has widely attracted attention. Initial methods for this recognition considered of setups fit only for laboratories, i.e. finger movements being converted into voltage signal sent to the computer via a wired glove being worn by a person. Obviously, these kinds of setups are not at all appropriate for general usage as limitations such as the length of the wire. Later, the glove was replaced by a camera like the Microsoft Kinect 3D which was very costly.

We put forth a method for gesture recognition via motion tracking as our step to solve these kinds of problems. Our device is a gesture tracking device which uses machine learning's support vector machine model to convert accelerometer data to a sequence of alphabets. The main aim of the project is to build a device using an Arduino pro micro that translates gestures into words wirelessly.

PROBLEM ANALYSIS

The problems faced while developing a robust recognition system are numerous and very much complicated. Some of these problems are as follows:

Object Analysis: We can represent a gestureby some feature vectors which are a collection of derived primitives. The vector shall serve as a compact and correct representation of the gesture. Feature selection is thereforeapplication dependent and studying performance of every feature vector consumes a lot of time. The final selection of this vector is based on mechanical and statistical performance measures such as to maximize between class pattern variability, minimize within class pattern variability and having a robust classification for unknown patterns.

Classifier needs some detailed representation as gesture is a very complicated pattern in order to produce a correct output. Our system can do this feature selection on its own as it would be using SVM.

Image Segmentation: Gesture contourshould be retained on the foreground only. Accuracy of the representation should be there, which means that problems like 3D hand movements should be dealt. If in case many cameras are being used then the creation of hidden spots could be done which will lead to performance degradation as the gesture vocabulary is large and variances between some gestures are less.

Pattern Recognition: We present twoproblems:
Two successive gestures have a discrimination gap which is not easy to define. The separating gap between such gestures is not defined, as in case of image sequences within time domain.

Every pattern needs to be tested out for similarity, using many classes. In case of an alphabet recognition system, the gestures are tested to match against 52 classes (A-Z, a-z). Our device, which is intended to perform translation of gestures into characters, every gesture of the vocabulary uses a different data set.

This paper is aimed at studying the technology behind an Arduino based motion recognition device that tracks the gestures made in mid-air and translates these gestures into characters of the English alphabets as bits. Using this device, people can create gestures which can be a new system of communication for the blind. It can be even used by the teachers as a completely new style of teaching as teaching culture has been continuously advancing fromchalk and blackboard to modern day methods such as use of projectors. This device will be having a very high accuracy as it will be using support vector machine of machine learning and will be able to efficiently work upon the gestures.

Throughout the span of this project, system objectives have been created. They resemble the outputs used to create this system.

- The creation of an interface to show results
- Successfully implementation of methods to detect gestures
- Successfully logging data by Arduino pro micro and mpuaccelerometer
- Sending of this data to computer via the hc module
- Successfully implement support vector machine to extract useful features
- Conversion of these gestures into characters of the English alphabet

When someone presses the switch, the module starts to read the gestures and sends accelerometer data to the computer. When the button is released, the transmission of data stops and conversion takes place. It first reads the movements of the device via mpuand Arduino and then sends the data collected via the hc module to a computer. With the machine learning algorithm, translation of the motion readings into characters is performed. The code is written in python and uses the scikit-learn library for the SVM algorithm. The module gets powered by a power bank which is connected to the Arduino pro micro. The cost of production for this device will be very less compared to the current recognition system such as the Microsoft Kinect 3D. It is designed in such a way that user can use the device in a user friendly manner and have a very sensitivity as it would be using the six-axis gyroscope and accelerometer.

METHEDOLOGY

Flask the sketch to Arduino, open serial monitor in the Arduino IDE and press the button on Arduino, the reading of gesture would start. Now everything will be working, so we are ready to use the library. We will start by deleting the data folder and its contents and replace it with a new created dataset. This library has been originally designed to make a keyboard such that each gesture is associated with a character also being case sensitive. This means we can teach the algorithm about 53 different gestures.

Start a new recording batch, by recording a sample for a specific gesture. Open the terminal and type: python start.py target=a: 0 port=COM6

- The "target" argument will tell module that we are recording a new sample for a gesture.
- The "a" character characterizes a gesture, and should be unique as it has length of 1 for every gesture.
- The "0" character is the batch number which has to be different each time when we register a new batch so that overriding of sample is avoided.
- The "port" represents the serial port to which Arduino is connected to.

When start.py is running, recording of different gestures by pressing and releasing the button on the Arduino is done. When the button is pressed, recording of data from the accelerometer is done. Every sample gets saved in the data folder as a different file. After this the training of model is done. When dataset is ready, we can use it to train our algorithm. For this open the terminal and write: python learn.py

To check out if the model is working, open the terminal and type: python start.py port=<YOUR_SERIAL_PORT> predictNo we can make gestures and see it getting predicted correctly. In Python, scikit is a popular library which is used to implement machine learning algorithm, Support Vector Machine is also available in scikit library and follows the structure same as object creation, import library, fitting model and prediction. Sample code below shows implementation:

```
#Importing Library

fromsklearn import svm

#Assuming, X (predictor) and Y (target) for tr
aining data set and x_test(predictor) of test
dataset

# CREATING SVM object

Model = svm.svc(kernel='linear', c=1,
gamma =1)
model.fit(X, y)

model.score(X, y)

#Predicting Output

Predicted= model.predict(x_test)
```

SOFTWARE AND HARDWARE SPECIFICATIONS

The library which is being used is written in Python and uses Scikit Support Vector Machine algorithm to translate the gestures into letters. SVM is a supervised machine learning classification algorithm.

There are limitless ways in which analysis of datasets with a variety of Python libraries can be done. We will only check the dimensions of the data and see few records. Scikit contains the SVM library, which has built in class for different SVM algorithms which perform making of predictions, evaluating the algorithm and giving results.

In order to get the accelerometer data, we build a module using an Arduino, a MPU-6050 as accelerometer which also acts as a gyroscope for better accuracy and a HC-06 module to enable Bluetooth communication. The entire module is being powered by using a power bank which is connected to the Arduino. Our proposed system can be also used via wired connection as the Arduino will be directly connected to the system using USB cable.

When someone presses the switch, the module starts to read the gestures and sends accelerometer data to the pc. When the button is released, the transmission of data stops and conversion takes place. It first reads the movements of the device via MPU and Arduino and then sends the data collected via the hc module to a computer. With the machine learning algorithm, translation of the signal into characters is performed. The following are the hardware components being used:

Arduino pro micro

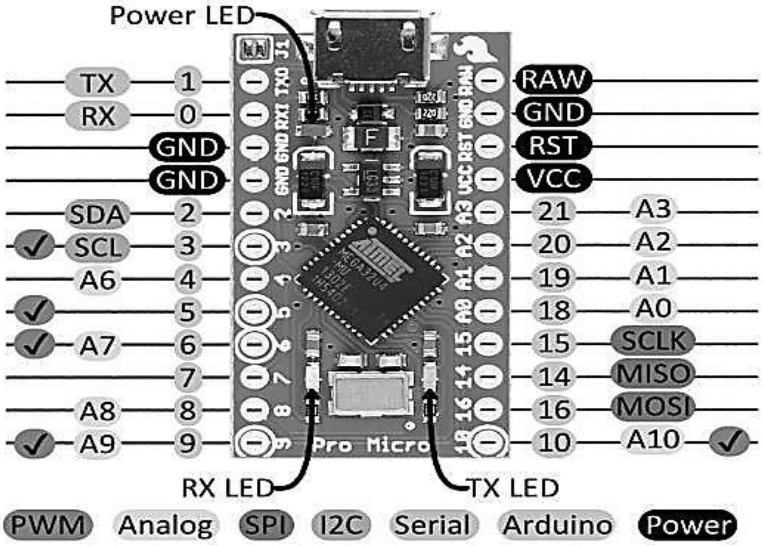

Figure 1: *Ardunio pro micro module.*

Itisa5V16MHzCompatibleBoard.Itis basedon theATmega32U4. The ATmega32U4 has an internal USB transceiver so that no bulky external USB chip is required.

It has 10-bit ADC, 12 DIOs, 5 PWM pins and hardware serial connections TX and Rx. A voltage regulator on board is present which accepts voltage up to 12VDC.

Features of the Arduino Pro Micro are:

- ATmega32U4 which runs at 5V/16MHz
- On Board micro USB connector
- 12x Digital I/Os
- 4x10 bit ADC pins
- Over current and reverse polarity protected
- Rx and tx Connections which are serial
- DC input of 5V which can go up to 12V
- On board status and power LEDs

Mpu 6050 accelerometer + gyroscope

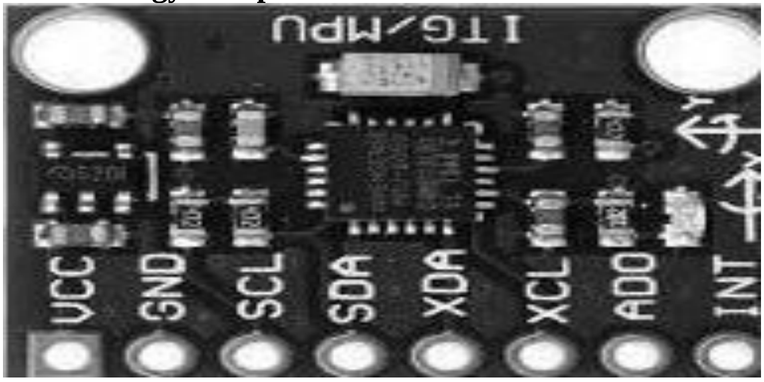

Figure 2: *MPU 6050 accelerometer.*

The MPU-6050 is a motion processing device. It combines a 3-axis accelerometer and a MEMS 3-axis gyroscope on the same silicon chip together with a Digital Motion Processor capable of processing 9-axis Motion Fusion algorithm.

Applications

- Movements and gestures detection
- Angular rate detection
- User interface which is motion activated
- Platform Stability
- Navigational Boards

CONCLUSION

The issue related with this recognition technique is what technology to use for collecting raw data from the device. A large number of recognition techniques are available like the feature extraction, active shape models, template matching. There are also a few segmentation algorithms such as Hand segmentation using HSV Colour, Anticipated static gesture set and sampled storage approach algorithm provide for recognition without much noise. But in the situation of the "Arduino based movement monitoring keyboard using machine learning," the results will be more precise as it will use the scikit learning (support vector machine algorithm) and thus better predictions with less noise to classify signals into letters. The cost of production of this device will be much less from the current methods of recognition and it will also be working mostly in all the environments such as areas with not proper light conditions. It can be of great social interest too as it would be helpful to the deaf and mute people to get discrete, direct and very effective service in the day to day life.

REFERENCES

1. Shakunthaladevi, M., R. B. Revathi (2014), "Real time hand gestureRecognitionusingAVR microcontroller", *International Conference on Information Communication and Embedded Systems (ICICES2014)*.
2. Andres G. Jaramillo, Marco E. Benalcazar (2017),"Real-time hand gesture recognition with EMG using machine learning", *2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM),*
3. Viblis, M. K., K. J. Kyriakopoulos (1999),"Gesture Recognition: The Gesture Segmentation Problem", *Advances in Intelligent Systems,*
4. Tsung-Ming Tai, Yun-JieJhang, Zhen-WeiLiao, Kai-Chung Teng, Wen-JyiHwang (2018), "Sensor-Based Continuous Hand Gesture Recognition by Long Short Term Memory", *IEEE Sensors Letters,*
5. Shukla, Jaya, AshutoshDwivedi (2014),"A Method for Hand Gesture Recognition", *2014Fourth InternationalConference on Communication Systems and Network Technologies*
6. S. Ghotkar, RuchaKhatal, SanjanaKhupase, SurbhiAsati, MithilaHadap (2012),"Hand gesture recognition for Indian Sign Language", *2012 InternationalConference on Computer Communication and Informatics*
7. www.slideshare.net
8. https://ieeexplore.ieee.org/search/searchresult.jsp?newsearch=true&queryText=big%20data
9. https://www.springer.com/gp
10. https://www.tutorialspoint.com/arduino/arduino_functions.htm

Cite this article as: