Gesture Recognizer Strap using Motion Sensor

Bengi Günay
Advisor:Dr. Faris Serdar TAŞEL

Çankaya University, Department of Computer Engineering

Abstract

The purpose of project ‘Gesture Recognizer Strap using Motion Sensor’ is to develop an gesture recognizer software. The aim is to develop affordable multipurpose device. To reach the project aims there will be used an accelerometer, gyroscope, compass sensor. These sensors are the one of the most affordable solution for gesture recognition problem. As the wearable device a wristband will be used. They will be on the wristband to track the basic movements of arm of the user like linear movements of arm (such as up, down, left, right) or angular movements of the arm.

Keywords :Gesture Recognition, Gesture Recognizers Software, Wearable Technology, Accelerometer, Gyroscope, Compass Sensor

Introduction

Gesture recognition is the process of getting data from a sensor and using an algorithm understand which gesture corresponds that data in the database. Gesture recognition is a brand topic in computer science. Today this technology is used in healthcare sector, automotive sector, transit sector, gaming sector, defense sector and electronics sector.

There are different technologies for gesture recognition in literature. These technologies categorized in two sub-categories as Image based, and Non-image based. Image based technologies includes a single camera, a stereo camera and a depth sensor. Non-image based technologies includes wearable materials such as a glove or a band or non-wearable technologies.

Our solution for gesture recognition problem is to develop a software with using a wearable technology. Our wearable device will be an wristband consist of accelerometer, gyroscope and compass sensor. Our solution is cheap and designed for multipurpose as usage.

Solution

Our software basically contains 2 different mode: a training mode and a testing mode.

In the training mode system takes raw data from the sensor, then it detects the gestural information. This gestural information is saved in the JSON file.The activity diagram of training mode can be seen at figure 1.

In the recognition mode the systems again take raw data from the sensor and detects the gestural information. Compares this information with the predefined gesture information (from training) to see if there is a match. If there is a match system recognizes the gesture. The activity diagram of recognition mode can be seen at figure 2.

The gestural information is hold in an array called feature vector.Our feature vector holds euler angles and accelerometer values for each axis

.

The method for comparison is distance measuement.Firstly the distance for each angle and each accelerometer value in the feature vector is calculated separately. If input gesture is closer to the train gesture in all 6 distance types, it is considered as an candidate. If there are more than one candidate, a total distance calculation will be made for the feature vector.The train gesture with the closest distance will be accepted as the recognized gesture.If a test gesture is rejected by partial distances, for all gesture.It is assumed as not recognized.

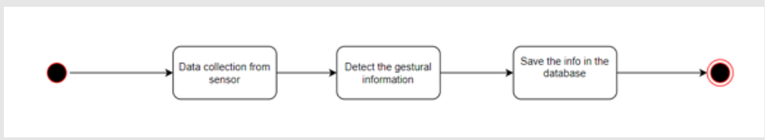

Figure 1 – Activity Diagram of Training Mode

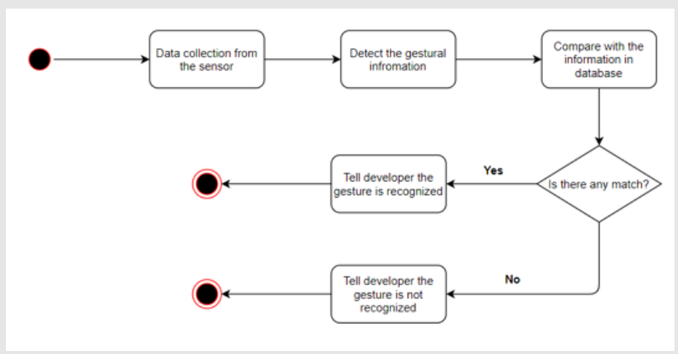

Figure 2 – Activity Diagram of Testing

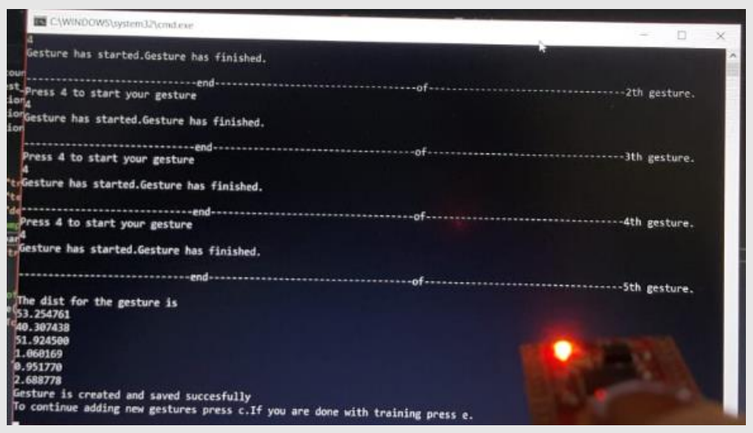

Figure 3 – Finished Product

Results & Conclusion

During the development process, there are lots of different creations of feature vector and different comparison techniques are applied.

At first it is planned to hold only angles and time in the feature vector, but then the accelerometer values are included for each axis. The last version of the of the holds only angles and accelerometer values. After holding accelerometer values, there is no need to hold time seperately.

At comparison process, it is planned to calculate only one distance for each vector. When the distance is calculating, a normalization is processed so that every different angle and accelerometer value is accepted as same effect on the distance.

Due to accelerometer values, the program started to make some wrong recognition.As en example if there is irrelevant gesture is recorded fast in the training, and the gesture that we made is recorded more slowly at the train, but faster in test, it could be recognized as some other fast gesture.

To correct this mistake, the accelerometer values are saved in a logaritmic scale but the results didnt bring satisfactory. Therefore lastly each distance of each angle and calculated separately and used in elimination separately which brings the best results until now

Acknowledgement

I would like to thank Faris Serdar Taşel, advisor of the project, for providing opinion and expertise to help in this project.