Abstract

The project ‘Machine Learning on Wearable Devices’ is aiming at exploring interaction between users and programs on wearable displays through hand gesture recognition.

Testing device

The device we used in our research is based on a set of motion capture device of Axis Neuron. Because the device was bought 5 years ago, some IMUs did not function properly. The IMU set for right hand was found broken and unable to pass the device completeness check on Axis Neuron software, so we had to switch to left hand. However, the IMU for middle finger on left hand was also found malfunctioning and we decided to substitute it with an Arduino IMU.

Testing Environment Design

Thanks to Axis Neuron who officially provides an interface library to export IMU data, we are able to connect Axis Neuron software and Unity to access to the data of IMUs. The testing environment is consisted with several scripts: a python program for XGBoost recognition algorithm, three C# scripts for Unity to record, transmit data and handle with data from Arduino IMU.

Findings and reasons

We found that static gestures (the first five gestures in our list) could be successfully recognized despite sometimes the program may be confused among them. A phenomenon I learnt before in signal processing course, cluster, inspire me of one possible reason. In signal modulation, different signals could be transmitted in different

phases simultaneously. However, when more signals are transmitted, signals become more similar with each other on their phase and “cluster”, which means it becomes harder to distinguish signals and an increasing error rate for decoding the signal. This idea can also apply to our gestures. Because all five gestures are tipping figures together, their features share a high similarity and error occurs more frequently when the program recognizes them. This also indicates a tradeoff between the amount of gestures command and recognition accuracy: if more gesture is defined, surely the system would be able to perform more sophisticated functions, but it also leads an increase similarity between gestures and lower accuracy.

On the other hand, other eight gestures which are non-static are never recognized. This is probably an issue with algorithm design. In our program, data samples are recorded and recognized frame by frame without considering the frames before or after it. It functions well with static gestures because it only requires determining the shape of hand, where one frame is enough. When it comes to non-static gestures, the program needs to determine the whole process of movement, then each sample has to be a data frame containing a series of data and the data format will be completely different with static gesture recognition. Additionally, determining when the movement start, stop and how long it continues are difficult. This issue is still unsettled.