

8-2017

# Statistical Movement Classification based on Hilbert-Huang Transform

Shanqi Sun

*University of Arkansas, Fayetteville*

Follow this and additional works at: <http://scholarworks.uark.edu/etd>



Part of the [Controls and Control Theory Commons](#)

---

## Recommended Citation

Sun, Shanqi, "Statistical Movement Classification based on Hilbert-Huang Transform" (2017). *Theses and Dissertations*. 2476.  
<http://scholarworks.uark.edu/etd/2476>

This Thesis is brought to you for free and open access by ScholarWorks@UARK. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of ScholarWorks@UARK. For more information, please contact [scholar@uark.edu](mailto:scholar@uark.edu), [ccmiddle@uark.edu](mailto:ccmiddle@uark.edu).

Statistical Movement Classification based on Hilbert-Huang Transform

A thesis submitted in partial fulfillment  
of the requirements for the degree of  
Master of Science in Electrical Engineering

by

Shanqi Sun  
Shenyang University of Technology  
Bachelor of Science in Mechanical Engineering, 2012

August 2017  
University of Arkansas

This thesis is approved for recommendation to the Graduate Council.

---

Dr. Jingxian Wu  
Thesis Director

---

Dr. Zhong Chen  
Committee Member

---

Dr. Shengfan Zhang  
Committee Member

## **Abstract**

The goal of this project is to introduce an automatic movement classification technique of finger movement signals using Hilbert-Huang Transform (HHT). Due to the nonlinear and nonstationary processing behavior, movement signals are analyzed with the Hilbert-Huang Transform (HHT). The slope of auto-correlation function and mean of frequency from first three Intrinsic Mode Functions (IMFs) was used as feature parameters for each category. Finally, performing support vector machine (SVM) for pattern classification completes classifying types of finger movement. According to the records of 669 trial samples of two types of finger movement signals (thumb and pinky), average accuracy is 93.28%. In another case of movement (thumb and pinky), average accuracy is 100%. All in all, the feature extraction method based on Hilbert-Huang transform (HHT) can be used to achieve effective movement classification.

## Acknowledgements

I would like to sincerely express my gratitude to all the people who help me to complete this thesis.

Firstly, I want to offer my gratitude to my advisor, Dr. Jingxian Wu, who help me a lot during my academic life in University of Arkansas. When I met some challenging problems in my research, he always gave me some good advices and help overcome them. Without his guidance and help, this thesis would not have been finished.

Moreover, I would like to appreciate my thesis committee members, Dr. Chen and Dr. Zhang who had been my defense committees at the last moment. Without their help, I could not pass final defense. the Under their patient reviews, I could improve the quality of my thesis.

## Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>                             | <b>1</b>  |
| 1.1      | Background and Motivation                       | 1         |
| 1.2      | Literature Survey                               | 2         |
| 1.3      | Thesis Outline                                  | 4         |
| <b>2</b> | <b>Movement Detection Data</b>                  | <b>5</b>  |
| 2.1      | Overview of Switch Devices                      | 5         |
| 2.2      | Movement Detection Data                         | 6         |
| 2.3      | Data of Activation and Non-activation           | 9         |
| <b>3</b> | <b>Feature Extraction</b>                       | <b>12</b> |
| 3.1      | Basic Principle of Hilbert-Huang Transform      | 12        |
| 3.2      | Feature Extraction Based on HHT                 | 15        |
| <b>4</b> | <b>Classification Method</b>                    | <b>21</b> |
| 4.1      | Introduction to Statistical Learning            | 21        |
| 4.2      | Supervised Learning                             | 22        |
| 4.3      | Unsupervised Learning                           | 23        |
| 4.4      | Introduction of SVM                             | 24        |
| 4.4.1    | Basic Concepts                                  | 24        |
| 4.4.2    | The Optimal Margin Classifier                   | 25        |
| 4.4.3    | Kernels   | 27        |
| 4.4.4    | SVM Parameters                                  | 29        |
| 4.5      | Multi-class Classification                      | 30        |
| 4.6      | Application of SVM on Movement Data             | 30        |
| <b>5</b> | <b>Experiment Result</b>                        | <b>31</b> |
| 5.1      | Classification of Pinky and Thumb               | 31        |
| 5.2      | Classification of Activation and Non-activation | 34        |
| 5.2.1    | Feature Extraction                              | 34        |
| 5.2.2    | Experiment Result                               | 37        |
| <b>6</b> | <b>Conclusions and Future Work</b>              | <b>40</b> |
| 6.1      | Conclusions                                     | 40        |
| 6.2      | Future Work                                     | 41        |

|   |           |    |
|---|-----------|----|
| 7 | Reference | 42 |
| 8 | Appendix  | 44 |

## List of Figures

|     |   |    |
|-----|---|----|
| 2.1 | The switch device with IMU [1] . . . . .                              | 6  |
| 2.2 | Complete Setup Assembled on: (a) Hand and (b) Wrist [1] . . . . .     | 6  |
| 2.3 | The movement signal of thumb . . . . .                                | 8  |
| 2.4 | The movement signal of pinky . . . . .                                | 8  |
| 2.5 | The movement signal of activation . . . . .                           | 10 |
| 2.6 | The movement signal of non-activation . . . . .                       | 11 |
|     |   |    |
| 3.1 | Envelope . . . . .  | 13 |
| 3.2 | The empirical mode decomposition components . . . . .                 | 16 |
| 3.3 | Hilbert Spectrum . . . . .  | 17 |
| 3.4 | Hilbert Spectrum of Movement from Thumb and Pinky . . . . .           | 18 |
| 3.5 | Auto-correlation of Energy of Movement from Thumb and Pinky . . . . . | 20 |
|     |   |    |
| 4.1 | Binary SVM classifier . . . . .                                       | 25 |
| 4.2 | Optimal Margin . . . . .  | 26 |
| 4.3 | Linear and nonlinear classifier. . . . .                              | 28 |
|     |   |    |
| 5.1 | Cross Validation . . . . .  | 32 |
| 5.2 | Hilbert Spectrum of Movement from Thumb and Pinky . . . . .           | 35 |
| 5.3 | Auto-correlation of Energy of Movement from Thumb and Pinky . . . . . | 36 |

## List of Tables

|     |  |    |
|-----|--|----|
| 2.1 | The Movement Data . . . . .                    | 7  |
| 2.2 | The Movement Data of non-activation . . . . .  | 10 |
| 5.1 | Confusion matrix using all features . . . . .  | 33 |
| 5.2 | Confusion matrix using original data . . . . . | 33 |
| 5.3 | Confusion matrix without HHT . . . . .         | 34 |
| 5.4 | Confusion matrix using all features . . . . .  | 38 |
| 5.5 | Confusion matrix using original data . . . . . | 38 |
| 5.6 | Confusion matrix without HHT . . . . .         | 39 |

# Chapter 1

## Introduction

### 1.1 Background and Motivation

For people with severe disabilities, their daily activities and actions are always limited. Thus, the assistive technology switches are used by disabled people in order to help them in daily life.

Assistive technology switches provide people who experience a severe physical disability or limitation to interact with devices or technologies that would otherwise require a lot of body movements. Such technology utilizes binary switch whose output is toggled on and off when activated. One common usage of assistive technology switch is in controlling power wheelchair that requires upper limb movement. For people who cannot control their limb movements (e.g., Muscular Dystrophy), assistive technology switches allow them to control the wheelchair by head movements or even by breathing. Another common usage of assistive technology switches is to control electronic devices such as laptops and smart phones which usually require mouse, keyboard or touchscreen [1].

In our case study, a disabled woman was left with only her thumb and pinky as reliable communication methods and switch access movements due to her brain stem stroke. A switch developed using the IMU is intended to increase her ability to reliably access switches. The machine learning aspect is intended to give us the ability to train the switch to recognize the user's switch access movement to generate the switch closure.

The goal of this thesis is to extract useful information from signals of different finger movements and to propose an automatic classification technique of finger movements.

Due to the fact that the movement signal is nonlinear and nonstationary, the application of traditional data analysis, such as spectrum analysis based on Fourier transform, has problems. However, Hilbert-Huang transform (HHT) represents an effective signal processing for nonlinear and nonstationary data.

HHT was developed by Norden E. Huang in the 1990s [2]. Hilbert-Huang transform can show the physical mechanisms hidden in data by empirical basic functions. Through this way, the signals are decomposed into a finite number of components which are referred to intrinsic mode functions (IMFs) by an empirical mode decomposition (EMD) process [3]. The EMD is based on the local behavior of signal, so it is suitable for nonlinear and nonstationary processes. The relationship among instantaneous amplitudes and frequencies, and time can be found by Hilbert transform. The combination of EMD and Hilbert transform is called Hilbert-Huang transform. The application of the HHT is growing in many fields. In this thesis we presented a new application of Hilbert-Huang transform method on movement classification. With the Hilbert-Huang transform, the relationship of instantaneous frequencies and time of movement signals and the slope of auto-correlation function of energy-time distribution which was used as features parameters for each class was computed and the support vector machine(SVM) method was applied for classification.

## 1.2 Literature Survey

Some literature are reviewed before this thesis work. [1] introduces the disability assistive technology, sensor technology, and the hardware device used to determine switches.

[2] is the original paper introducing the methods of HHT which was written by Norden E. Huang. It offers very detailed information about HHT including some concepts and applications. HHT contains two parts which are termed as empirical mode decomposition (EMD) and Hilbert spectrum and is suitable for nonstationary and nonlinear time series analysis. It transfers the signal from time domain to amplitude-frequency-time domain. This paper also talks about how to achieve EMD by using the sifting process and some mathematical and physical relationships among amplitude (or energy), frequency, and time. The last section gives some examples about the application of HHT on classic nonlinear data analysis.

[3] gives an application of HHT on electroencephalogram's signal (EEG) to classify sleep stages. It describes the basic principle of HHT including EMD process and Hilbert spectrum. The physical meaning of Hilbert spectrum is also discussed. Furthermore, the feature extraction of sleep EEG signal based on HHT is performed. Both [4] and [5] are good papers to learning how to use HHT on signal analysis.

Support vector machine is a supervised learning algorithm for classification machine learning problems. It separates data into two groups by building a hyperplane with maximum margin between them.

[6] and [7] introduce the method of SVM including how the algorithm is established and the knowledge of Gaussian kernels.

[8] offers a method predicting if the putt is successful or not based on EEG data by SVM. It describes some concepts of SVM, Gaussian radial basis kernels, and important parameters in detail. Moreover, [9] applied SVM to speaker identification and speech recognition. It provides another good example of SVM application on time series signal analysis.

The next section will give a outline of this thesis work including the organization and content of every chapter.

### **1.3 Thesis Outline**

This thesis includes the movement data collecting system, feature extraction, statistical learning methods, especially support vector machine (SVM), experiment results, and future work.

Chapter 1 gives an overview of the whole thesis. It introduces the background and motivation of this thesis work, literature survey and the organization of this thesis. Chapter 2 covers the application of the switch device and the data collecting system. How the switch device was used and how movement signals were achieved will be reviewed in detail.

Chapter 3 introduces the HHT method and how to process this method. Furthermore, how the features are extracted from movement signal based on HHT will be discussed in detail. Chapter 4 starts from a review of statistical learning including supervised learning and unsupervised learning. Especially, the method of SVM and Gaussian radial basis kernels will be discussed. Moreover, how to apply this algorithm on our movement classification problem will be introduced.

Chapter 5 discusses the experimental simulation results. The simulation accuracies with different features will be listed. The last chapter will summarize the whole thesis, give some conclusions and discuss the future work based on the previous results.

## Chapter 2

### Movement Detection Data

#### 2.1 Overview of Switch Devices

We obtained data from a disabled person when she made movement with her pinky or thumb. Our goal is to train a model to classify the types of her movement: thumb or pinky. Furthermore, we collected another type of movement data in order to classify the types of movement: activation and non-activation.

Fig. 2.1 shows the switch device we used to collect movement data. An Inertial Measurement Unit (IMU) sensor embedded in a 3D printed ring is included in the hardware [1]. The ring should be worn on the middle finger. Other two additional 3D printed rings are also shown in Fig. 2.1. These two rings should be worn on the thumb and pinky respectively. How switch devices were assembled on the body is displayed on Fig. 2.2.

The device collected the data using a 3-axis magnetometer, a 3-axis accelerometer, and a 3-axis gyrometer. Each of them consist of three sensing axes with the common orthogonal axis system. Thus, the finger movement can be defined in nine degree-of-freedom. The magnetometer can measure the distance and location of a magnet relative to the sensor. The Accelerometer is used to measure acceleration along a defined axis. The gyrometer is designed to relative the rotation functioned by the around a fixed axis [1].

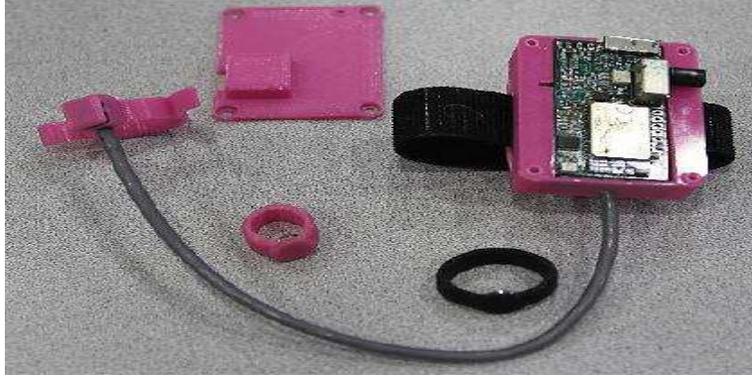
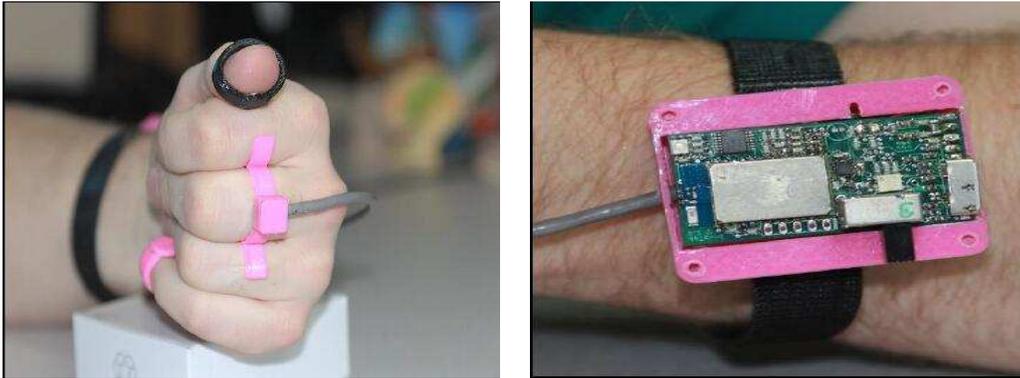


Figure 2.1: The switch device with IMU [1]



(a) Assembled on Hand

(b) Assembled on Wrist

Figure 2.2: Complete Setup Assembled on: (a) Hand and (b) Wrist [1]

## 2.2 Movement Detection Data

For thumb and pinky case, there are  $N$  trials in the database where  $N = 667$ . There are  $N_{thumb} = 306$  trials representing thumb and other  $N_{pinky} = 361$  trails representing pinky among the database. Each trial is a signal whose duration is  $T = 4$  seconds with a sampling frequency  $f_s = 100\text{Hz}$  of 9 axis of sensor measurements including accelerated movements on axis X, Y, Z, gyroscopic movements on axis X, Y, Z, and magnetic movements on axis X, Y, Z. An example of the movement signals is shown on Table 2.1. Furthermore, Fig. 2.3 and Fig. 2.4 are curves of an example of the movement signals from thumb and pinky.

| Time/s | Accel |       |       | GYRO |      |      | MAG |     |     |
|--------|-------|-------|-------|------|------|------|-----|-----|-----|
|        | X     | Y     | Z     | X    | Y    | Z    | X   | Y   | Z   |
| 0.01   | 132   | -9234 | 11413 | 122  | -126 | -324 | 591 | 285 | 781 |
| 0.02   | 162   | -9209 | 11447 | 177  | -200 | -407 | 593 | 279 | 793 |
| 0.03   | 389   | -9105 | 11482 | 199  | -227 | -463 | 591 | 285 | 789 |
| ...    | ...   | ...   | ...   | ...  | ...  | ...  | ... | ... | ... |
| 3.99   | 419   | -9251 | 11433 | 142  | 42   | -262 | 600 | 278 | 802 |
| 4.00   | 199   | -9280 | 11336 | 95   | 186  | 25   | 599 | 279 | 799 |

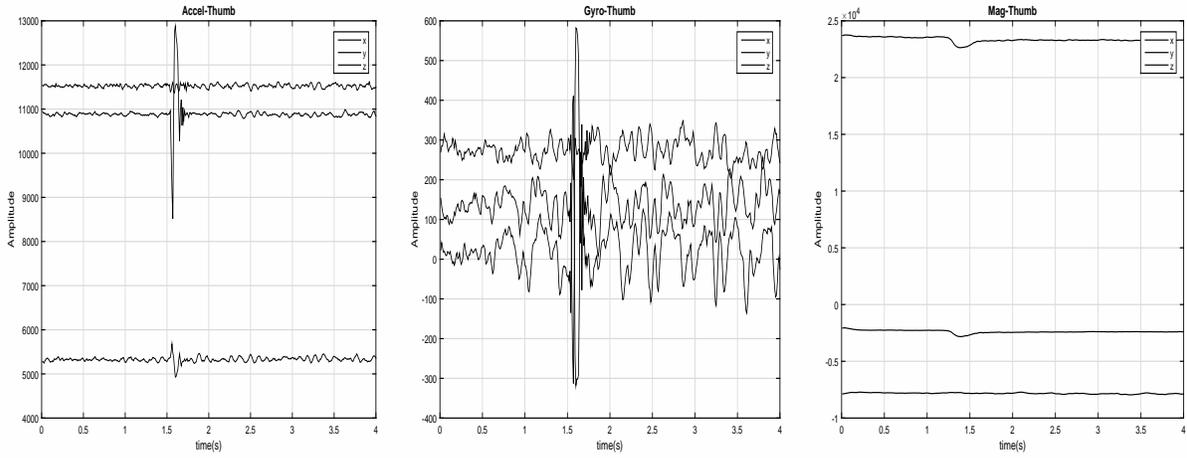
Table 2.1: The Movement Data

The accelerated movement signals on axis X, Y, Z are denoted as  $x_{a_n}(t)$ ,  $y_{a_n}(t)$ ,  $z_{a_n}(t)$  respectively. The gyroscopic movement signal on axis X, Y, Z are denoted as  $x_{g_n}(t)$ ,  $y_{g_n}(t)$ ,  $z_{g_n}(t)$  respectively. The magnetic movement signal on axis X, Y, Z are denoted as  $x_{m_n}(t)$ ,  $y_{m_n}(t)$ ,  $z_{m_n}(t)$  respectively, where  $n = 1, 2, 3, \dots, N$  and  $t = 0.01, 0.02, 0.03, \dots, 4s$ . Each of them is a column vector including 400 elements.

Thus, every trial is a  $400 \times 9$  dimension matrix denoted as

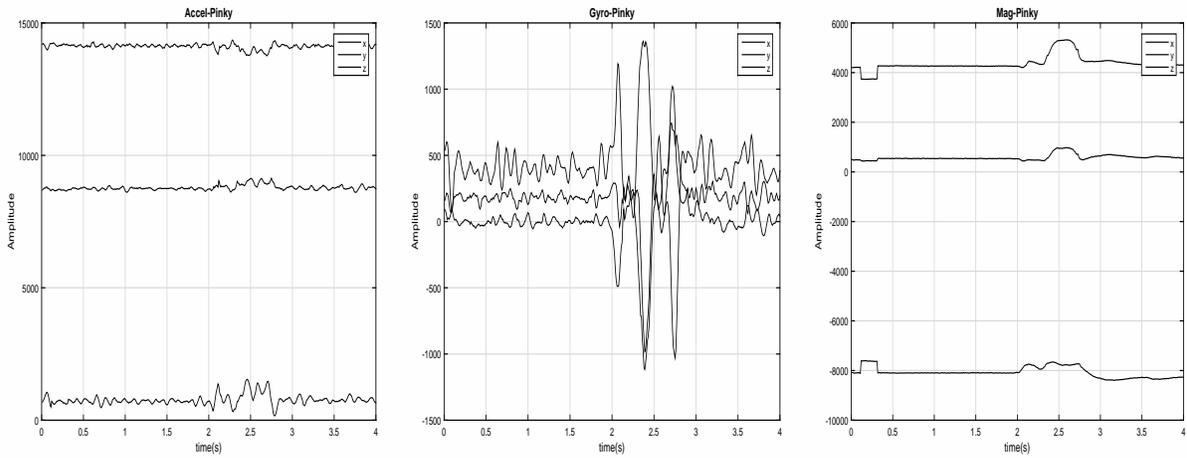
$$M_n = [x_{a_n}^T(t), y_{a_n}^T(t), z_{a_n}^T(t), x_{g_n}^T(t), y_{g_n}^T(t), z_{g_n}^T(t), x_{m_n}^T(t), y_{m_n}^T(t), z_{m_n}^T(t)]$$

where  $n = 1, 2, 3, \dots, N$  and  $t = 0.01, 0.02, 0.03, \dots, 4s$ .



(a) The acceleration of thumb in three dimensions (b) The gyroscope movement of thumb in three dimensions (c) The magnetism movement of thumb in three dimensions

Figure 2.3: The movement signal of thumb



(a) The acceleration of pinky in three dimensions (b) The gyroscope movement of pinky in three dimensions (c) The magnetism movement of pinky in three dimensions

Figure 2.4: The movement signal of pinky

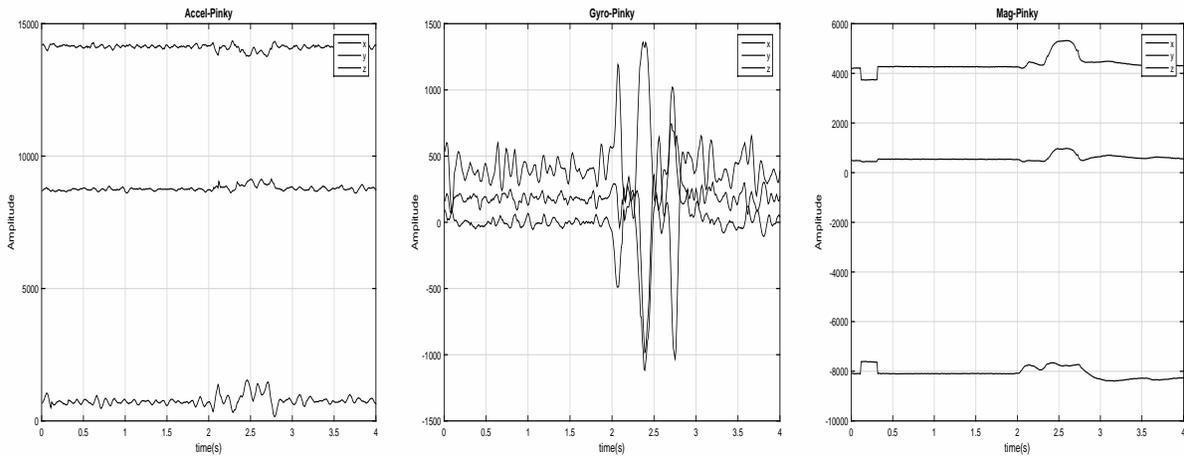
### 2.3 Data of Activation and Non-activation

The activation case contains the movement of thumb, pinky and together. The type non-activation does not mean the hand keep static. The hand can still move, but fingers do not move relatively, such as shaking hands. These kind of data is not from the disabled person. They were from two candidates who used 3 hours to collect the data at the Intelligent Information Processing lab at the University of Arkansas.

For activation and non-activation case, there are totally 654 trials collected. For the first candidate, there are  $N^{(1)} = 374$  trials including  $N_{non}^{(1)} = 185$  trials representing non-activation and  $N_{act}^{(1)} = 188$  trials representing non-activation. For the second candidate, there are  $N^{(2)} = 180$  trials including  $N_{non}^{(2)} = 92$  trials representing non-activation and  $N_{act}^{(2)} = 88$  trials representing activation. Each trial is also a signal whose duration is  $T = 4$  seconds with a sampling frequency  $f_s = 100\text{Hz}$  of 9 axis of sensor measurements. An example of the non-activation signals is shown on Table 2.2. The Fig. 2.5 and Fig. 2.6 are curves of an example of the movement signals from activation and non-activation.

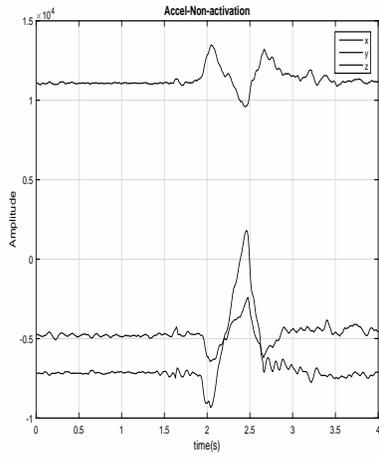
| Time/s | Accel |       |       | GYRO |      |      | MAG |     |     |
|--------|-------|-------|-------|------|------|------|-----|-----|-----|
|        | X     | Y     | Z     | X    | Y    | Z    | X   | Y   | Z   |
| 0.01   | -22   | -3630 | 14164 | -65  | -226 | 83   | 258 | 736 | 370 |
| 0.02   | -82   | -3446 | 13960 | -3   | -312 | 63   | 262 | 730 | 368 |
| 0.03   | -104  | -3344 | 13888 | -43  | -445 | -19  | 254 | 732 | 364 |
| ...    | ...   | ...   | ...   | ...  | ...  | ...  | ... | ... | ... |
| 3.99   | 711   | -3358 | 14250 | 21   | -105 | -46  | 320 | 724 | 512 |
| 4.00   | 696   | -3408 | 14351 | -38  | 41   | -110 | 323 | 731 | 517 |

Table 2.2: The Movement Data of non-activation

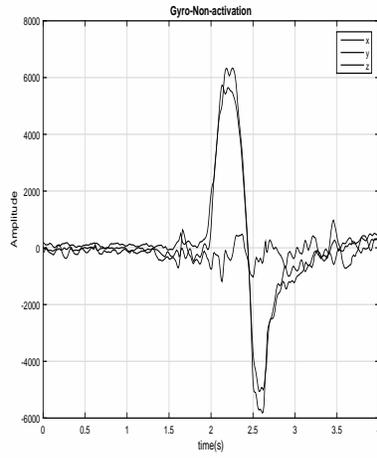


(a) The acceleration of activation  
 (b) The gyroscope movement of activation  
 (c) The magnetism movement of activation

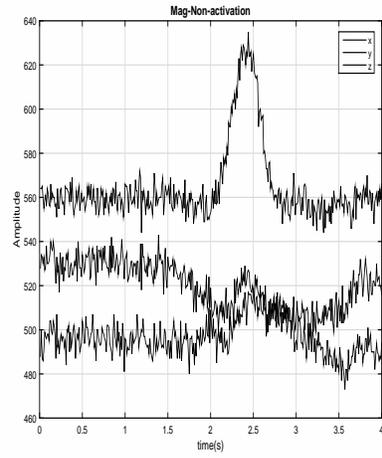
Figure 2.5: The movement signal of activation



(a) The acceleration of non-activation



(b) The gyroscope movement of non-activation



(c) The magnetism movement of non-activation

Figure 2.6: The movement signal of non-activation

## Chapter 3

### Feature Extraction

#### 3.1 Basic Principle of Hilbert-Huang Transform

Feature extraction is the most important analysis in this thesis, which directly affects the classification result. From the simulation result, we can see the accuracy is just 76.53% if we only use original data. Thus, our goal is to extract a meaningful difference as the feature between two classes from the data.

Hilbert-Huang Transform is a process that is widely used to decompose signals into a number of Intrinsic Mode Function (IMF), and further to obtain the instantaneous frequency and Hilbert spectrum from the signals. As the definition, it can be split into two sequential processes, which are known as the empirical mode decomposition (EMD) and Hilbert spectral analysis [3].

In the EMD process, signals will be decomposed into a series of IMFs until it meets a pre-defined criteria. Intrinsic Mode Function (IMF) is a special function satisfying the following two properties: 1) The difference between the number of extrema and the number of zero crossings is less than 1 and 2) the mean value between the upper and lower envelope (derived from local maxima and local minima respectively) is always 0. The following steps describe a special sifting process to extract all IMFs.

To prepare for the sifting process, for a given signals  $x(t)$ , calculate its upper-envelope  $u(t)$  and its lower-envelope  $v(t)$  and the mean value between the two envelopes  $m_1(t)$ , which

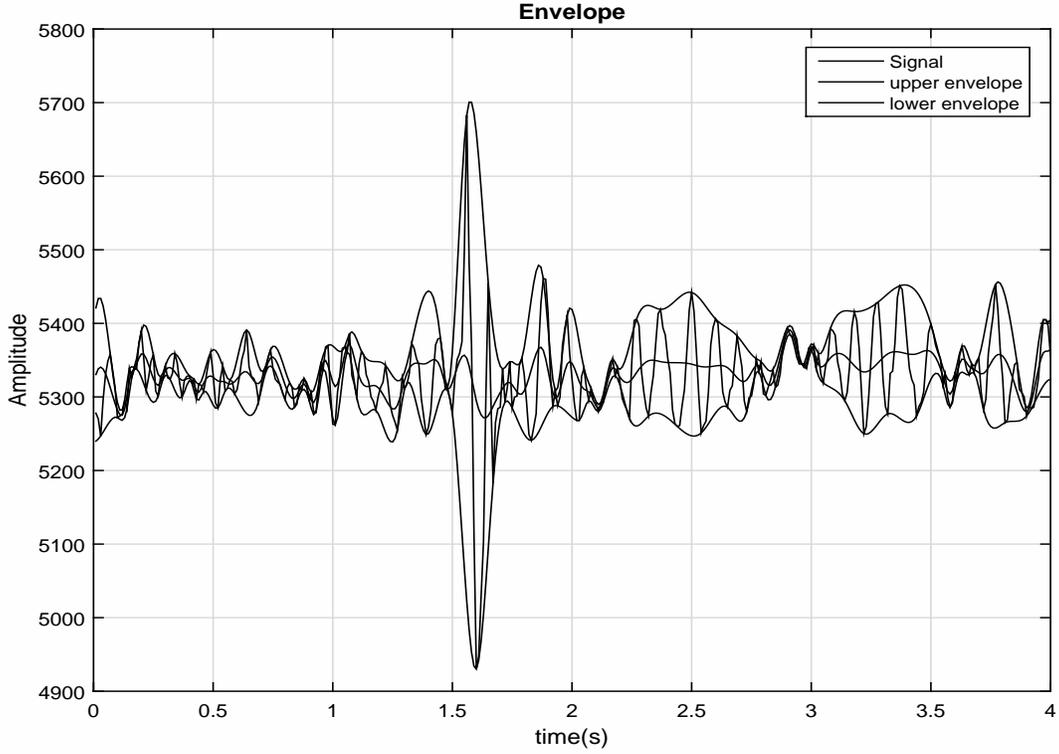


Figure 3.1: Envelope

are shown on Fig. 3.1.

$$m_1(t) = [u(t) + v(t)]/2 \quad (3.1)$$

First, get the difference between the original signals and its calculated mean vector.

$$h_1(t) = x(t) - m_1(t) \quad (3.2)$$

To get a IMF that meets the two criteria, this sifting process usually need to be applied multiple times, in which case the obtained difference  $h_k(t)$  becomes the signal in the next sifting step. After the  $(k + 1)$ th sifting, take the difference  $h_{1k}(t) = h_{1(k-1)}(t) - m_{1k}(t)$  as the first IMF if it satisfies the two properties of IMF. This first IMF will be denoted as  $c_1(t)$ :

$$c_1(t) = h_{1k}(t) \quad (3.3)$$

In practice, a standard deviation (SD) criterion will be checked to help decide whether

the  $h_{1k}(t)$  is good enough to satisfy the IMF properties or not using the following inequality:

$$\text{SD}(k) = \sum_{t=0}^T \frac{|h_{1(k-1)}(t) - h_{1k}(t)|^2}{h_{1(k-1)}^2(t)} \leq 0.2 \quad (3.4)$$

Where  $T$  is the length of the data.

Next, the second IMF  $c_2(t)$  can be obtained using the same sifting process where the first residual  $r_1(t) = x(t) - c_1(t)$  will be treated as the new signals. Repeat this process  $n$  times until the last residual  $r_n(t)$  becomes monotonic. In this way, the original signals can be represented by the series of IMF and the final residual as:

$$x(t) = \sum_{i=1}^n c_i(t) + r_n(t) \quad (3.5)$$

Now since all the IMFs have been extracted from the signals, we can obtain the Hilbert spectrum and the instantaneous frequency based on them.

Get the Hilbert transform of all IMFs  $c_i(t)$  using:

$$H[c_i(t)] = \frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{c_i(\tau)}{t - \tau} d\tau \quad (3.6)$$

Form a complex function  $z_i(t)$  with the  $c_i(t)$  as the real part and the Hilbert transform of  $c_i(t)$  as the imaginary part:

$$z_i(t) = c_i(t) + jH[c_i(t)] = a_i(t)e^{j\theta_i(t)} \quad (3.7)$$

Where  $j = \sqrt{-1}$ ,  $a_i(t)$  is called the instantaneous amplitude and  $\theta_i(t)$  called the instantaneous phase of IMF  $c_i(t)$ . Then calculate the instantaneous frequency  $\omega_i(t)$  of  $c_i(t)$  using:

$$\omega_i(t) = \frac{d\theta_i(t)}{dt} \quad (3.8)$$

In this way, we can represent the original signals  $x(t)$  as the following equation:

$$x(t) = \text{Re}\left[\sum_{i=1}^n a_i(t)e^{j \int \omega_i(t) dt}\right] + r_n(t) \quad (3.9)$$

where the residue  $r_n(t)$  is ignored. On the time-frequency plane, plot the instantaneous amplitude which provides a special distribution of each IMF signals' energy. Equation (3.9) enables us to represent the amplitude and the instantaneous frequency as functions of time in a three-dimensional plot, in which the amplitude can be contoured on the time-frequency plane. This time-frequency distribution of the amplitude is designated as the Hilbert amplitude spectrum  $H(\omega, t)$ .

With the Hilbert spectrum defined, we can also define the marginal spectrum,  $h(\omega)$ , as

$$h(\omega) = \int_{-\infty}^{+\infty} H(\omega, t) dt \quad (3.10)$$

The marginal spectrum is a function of amplitude respect to frequency. It represents such the energy appeared with a high probability at the frequency  $w$ .

### 3.2 Feature Extraction Based on HHT

Feature extraction of movement signals for each axis,  $x(t)$ , based on HHT transform is performed:

1) EMD: The movement signal for each axis  $x(t)$  is decomposed into a series set of IMFs.

a) Mean value calculation: Calculate upper-envelope  $u(t)$  and lower-envelope  $v(t)$  of  $x(t)$ .

For the upper-envelope, the local maxima should be obtained firstly. Then it can be achieved by using cubic spline interpolant to those local maxima points. In the same way, the lower-envelope can be obtained just by replacing local maxima with local minima. Secondly, the mean value  $m_1(t)$  of them can derived got by equation (3.1).

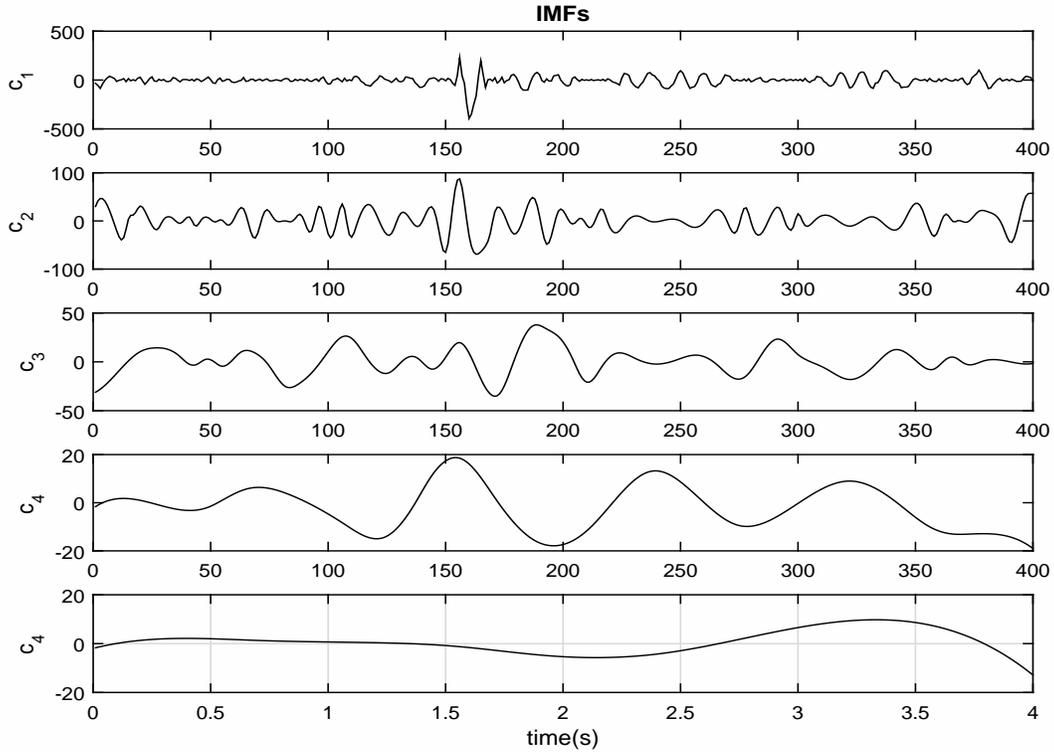


Figure 3.2: The empirical mode decomposition components

b) Sifting process and IMFs computation: Take the difference  $h_1(t)$  between  $x(t)$  and  $m_1(t)$  using equation (3.2). In most cases,  $h_1(t)$  can not satisfy the two IMF properties. Treat  $h_1(t)$  as new signal and repeat previous steps by using equation (3.1) and (3.2) until it satisfies (SD) criterion (3.4). If  $h_{1k}(t)$  satisfies (SD) criterion (3.4) after the  $(k + 1)$ th sifting, it will be denoted as  $c_1(t)$  which is the first IMF component. Next, the second IMF  $c_2(t)$  can be obtained using the same sifting process where the first residual  $r_1(t) = x(t) - c_1(t)$  will be treated as the new signals. Repeat this process n times until the last residual  $r_n(t)$  is monotonic. The movement signal can be decomposed as 5 IMFs in Fig. 3.2 When the EMD is completed. The residue is negligible.

2) Hilbert Spectrum: Once the IMFs are obtained, the Hilbert transform will be applied on them in order to get the relationship among amplitude, frequency, and time.

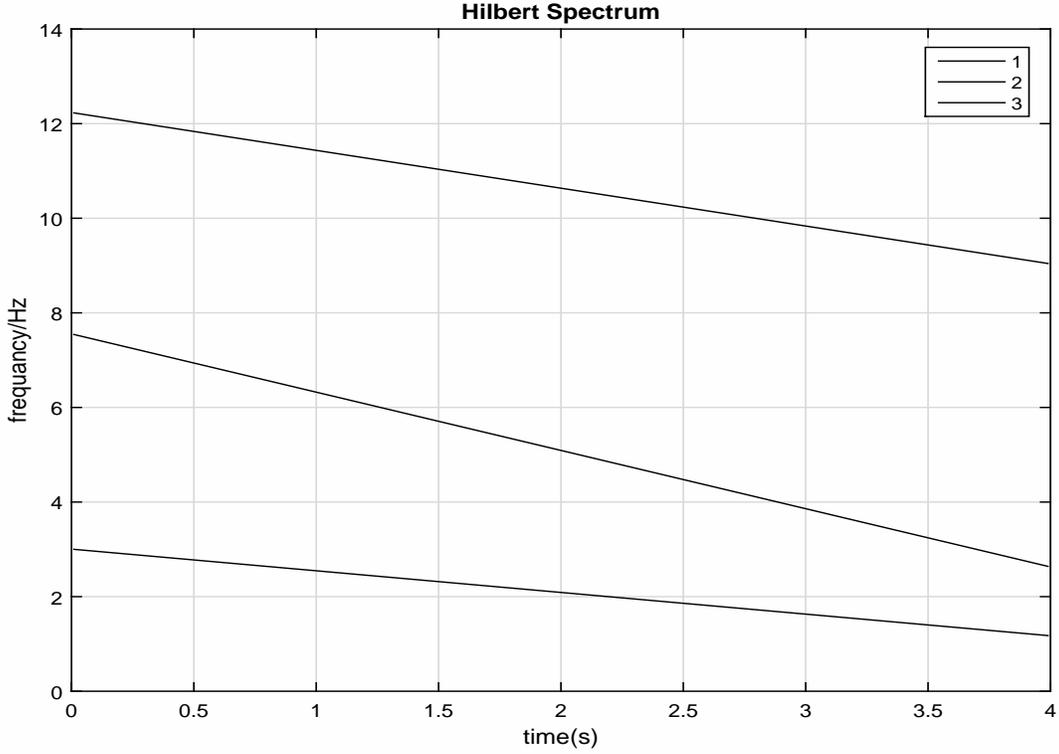


Figure 3.3: Hilbert Spectrum

Implementing the Hilbert transform to every IMF component by equation (3.6). Then we can achieve the instantaneous frequencies of each IMF component according to equation (3.8) and  $f_i(t) = \omega_i(t)/2\pi$ . Corresponding instantaneous frequencies of first 3 IMFs is given in Fig. 3.3

In our case, we try to find the significant difference from Hilbert Spectrum between different classes. Fig. 3.4 is the Hilbert Spectrum from thumb and pinky.

From those figures, we can get the frequencies (first 3 IMFs) from thumb are higher than the frequencies (first 3 IMFs) from pinky. So we decide to use the mean of frequency from the first 3 IMFs,

$$Y_1 = [m_f(1), m_f(2), m_f(3)]$$

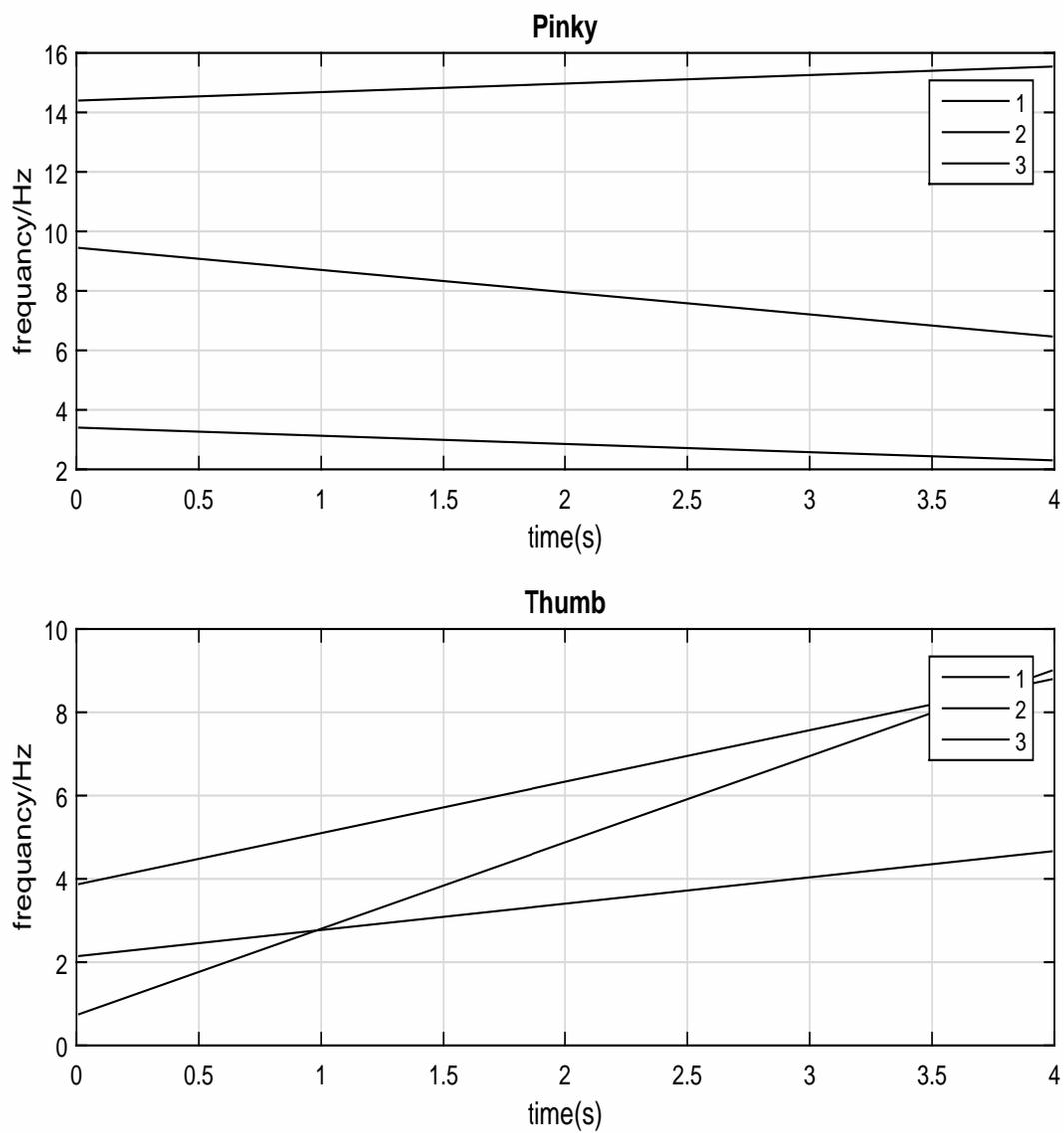


Figure 3.4: Hilbert Spectrum of Movement from Thumb and Pinky

as a feature, where

$$m_f(i) = \frac{1}{K} \sum_{k=1}^K f_i(k), \quad K = 400 \quad (3.11)$$

Another way we get the feature by computing the auto-correlation function of the energy from first 3 IMFs in different classes.

$$E_i(t) = a_i^2(t) \quad (3.12)$$

$$R_i(\tau) = \int E_i(t)E_i(t - \tau)dt \quad (3.13)$$

Fig. 3.5 is the plot of the auto-correlation function from thumb and pinky.

From those figures, we can get the slopes (first 3 IMFs) from thumb cases are higher than the slopes (first 3 IMFs) from pinky cases. So we decide use the points from  $R(10)$  to  $R(30)$  from the first 3 IMFs,

$$Y_2 = [R_1(10), R_1(11), \dots, R_1(30), R_2(10), R_2(11), \dots, R_2(30), R_3(10), R_3(11), \dots, R_3(30)]$$

as another feature which is a  $1 \times 63$  vector.

Now we get  $X_l$  which is a  $1 \times 66$  vector,

$$X_l = [m_f(1), R_1(10), \dots, R_1(30), m_f(2), R_2(10), \dots, R_2(30), m_f(3), R_3(10), \dots, R_3(30)]$$

as the feature for each axis.  $l \in \{xa, ya, za, xg, yg, zg, xm, ym, zm\}$  represents the axes.

Thus, for total signal, the feature  $X$  is a  $1 \times 634$  vector,

$$X = [X_{xa}, X_{ya}, X_{za}, X_{xg}, X_{yg}, X_{zg}, X_{xm}, X_{ym}, X_{zm}]$$

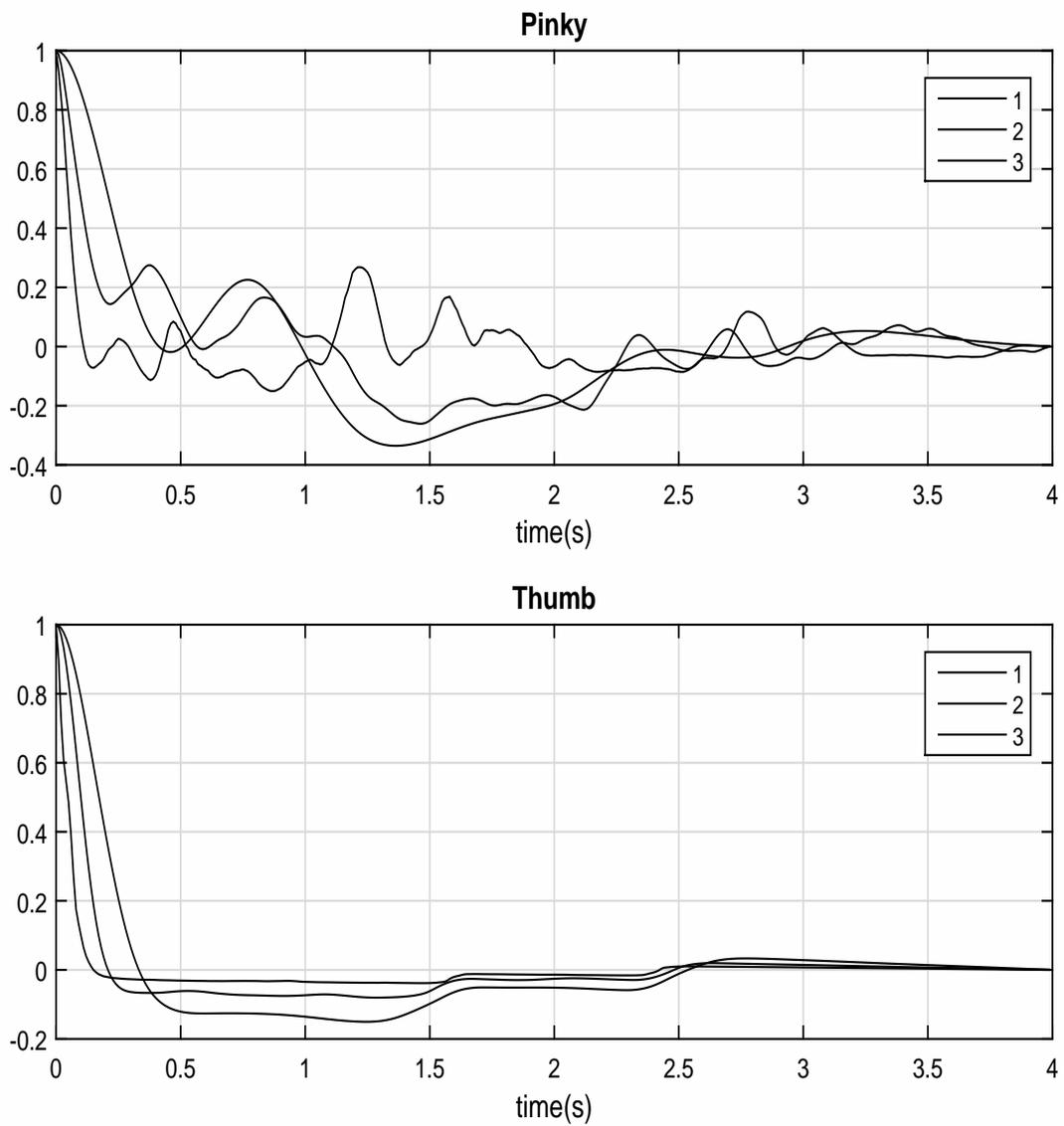


Figure 3.5: Auto-correlation of Energy of Movement from Thumb and Pinky

## Chapter 4

### Classification Method

#### 4.1 Introduction to Statistical Learning

Statistical learning is a method which refers to using statistical models to analyze and predict data. The object of statistical learning is data including number, text, image, video, audio, and their combinations. Statistical learning use the knowledge of statistics to mine the structure, property, law, and other information hidden in the data. There is an assumption that the same or similar types of data share the same statistical property in applications. With statistical learning people can use past data and current data to predict the outcome of future data. Statistical learning is also called machine learning which can give computers the ability to learn and improve by themselves like human beings. It is a popular technology for Artificial Intelligence and other industry fields and its application is widely used nowadays.

For example, a person need to evaluate the price of a new house before he buy it. He need to consider several factors such as location, size, age, facilities nearby, and other information of the house in order to make the best decision. He can collect such data and use them to analyze the relationship between the price of the house and those information by using statistical learning method. Then he can predict the price of a new house and make a better decision. Another example of its application is spam filters in email services. As we known, many email services are able to decide spam from emails received with high accuracy

automatically. Building a spam classifier is based on choosing features from spam and non-spam emails. For spam, the content of that email always contains the word like ‘buy’, ‘deal’ or something like that. If the word ‘discount’ occurs in the text, this email is more likely to be spam. However, if my name appears in a piece of email, it is probably non-spam, because the sender has already known who I am. According to the number of those sensitive words occur in email, the computer can classify the emails to non-spam and spam.

From previous two examples, we can find a common point in both of them: the output has been told either a real number or a category labeled. However, in the real world, a lot of unlabeled data need to be classified into different groups for some special cases. Speech recognition and social network analysis in Facebook are examples for this type of statistical learning problem.

According to previous discussion, there are two major kinds of problems in statistical learning and machine learning: supervised learning and unsupervised learning.

## **4.2 Supervised Learning**

Supervised learning refers to the type of machine learning problem whose goal is to infer a model from labeled data. For instance, the examples from last section, we know what the output is. In the housing price case, the output is a real number. In the spam filters case, the output is labeled category: non-spam and spam. Therefore, there are also two types of problems in supervised learning depending on value of output. When predicting a continuous value output, we call this case a regression problem. Another type of problems has been termed a classification problem if the output is discrete.

All in all, if the problem is to study a labeled dataset in which a dependent or target

variable (Y) is always provided given a set of independent variables (Xs), it can be called a supervised learning problem. A supervised learning algorithm studies the training data and provides prediction for any new sets of independent variables. It can be further divided into a classification or a regression problem based on the type of the target variable. If the target variable is a numeric value, it is called a regression problem. Linear regression is one of the most common regression algorithms, which tries to minimize the square error between the predicted value and the true target variable. If the target variable is a categorical value, in which case you are trying to classify each input into one of the known classes, it is a classification problem. There are a wide range of classification algorithms including naive Bayes, decision tree, support vector machine and so on. Supervised learning requires data to be divided into training and testing dataset or use other cross-validation methodologies to prevent over-fitting, to decide which algorithm to use and to tune the model parameters. Such evaluation of accuracy or error is unique to supervised learning.

### **4.3 Unsupervised Learning**

On the other hand, if the model is only trained on independent variable (such data is not labeled) and the purpose of the model is to learn the structure or relationship between different observations (rows) or different variables (columns), it is called an unsupervised learning problem. One of the most common unsupervised learning is clustering in which you are trying to group observations into different clusters in order to maximize the difference between clusters and to minimize the difference within each cluster, and in this way to learn the structure of the data. Another common usage of unsupervised learning is to extract hidden information from the dataset to explain the variance in the data, such as Principle

Component Analysis, component analysis and singular value decomposition. Since the data is unlabeled, there is no concept of evaluating the model based on the accuracy or the error. Sometimes it requires domain knowledge to decide which is the most suitable unsupervised learning model.

## 4.4 Introduction of SVM

Support Vector Machine (SVM), which is the algorithm of separating data with a wide gap, was first introduced by Bernhard E. Boser [6]. It is a popular supervised learning algorithm for classification problem.

### 4.4.1 Basic Concepts

For binary case, the input of the data can be labeled as two classes which are negative and positive examples. SVM can determine a hyperplane separating those data into two groups. This hyperplane is an optimal decision boundary with a large gap to classify two classes examples. We denote the data  $x_i \in R^n$ ,  $i = 1, 2, 3, \dots, l$ , labeled by  $y_i \in \{-1, +1\}$  and try to create a decision function  $f(x)$  to divide those data in the high dimension space.

Fig. 4.1 is an example of binary SVM classifier.  $f(x) = w^T x + b$  is the decision function.

The hyperplane

$$\{x : f(x) = w^T x + b = 0\} \tag{4.1}$$

separates the data into two groups: the points above the hyperplane are negative examples labeled by -1 and the points under the hyperplane are positive examples labeled by +1.

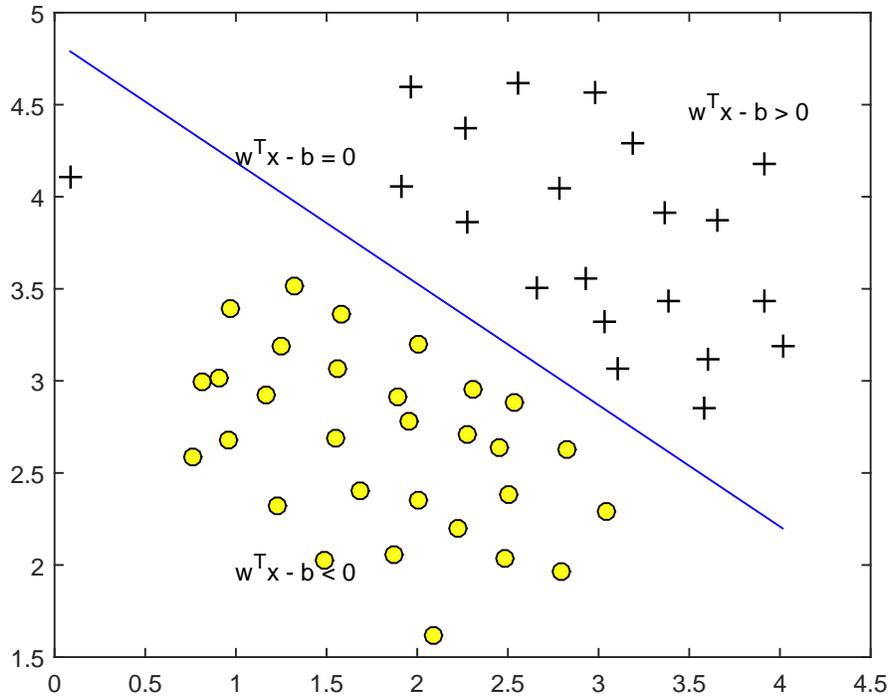


Figure 4.1: Binary SVM classifier

#### 4.4.2 The Optimal Margin Classifier

In general, there are a lot of separating hyperplanes which can be found between two different classes in the space. The goal is to find the separating hyperplane that maximizes the margin, since it can predict new data with a very high accuracy.

Mathematically, the distance of  $i$ -th data point from  $w^T x + b = 0$  is

$$d_i = \frac{|w^T x_i + b|}{\|w\|} = \frac{y_i(w^T x_i + b)}{\|w\|} \quad (4.2)$$

Comparing Fig. 4.1 and Fig. 4.2, there is a hyperplane with a large margin in second figure. Obviously, the smallest distance between the positive and negative examples is  $\frac{2}{\|w\|}$ . We try to maximize  $\frac{2}{\|w\|}$  which is equal to minimize  $\frac{\|w\|^2}{2}$ . Thus, the objective function should

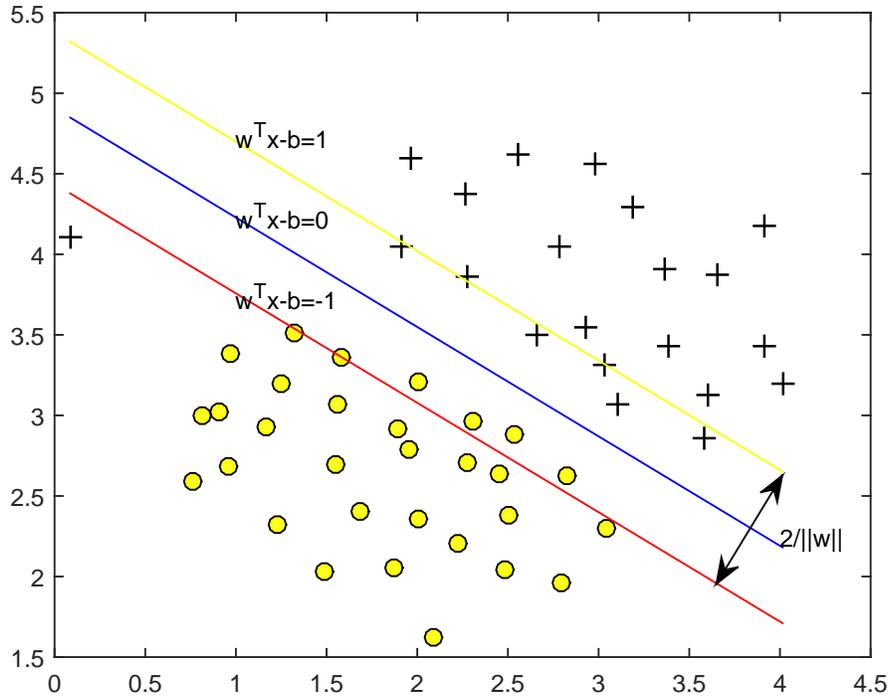


Figure 4.2: Optimal Margin

be

$$\min_{w,b} \frac{1}{2} \|w\|^2 \quad (4.3)$$

$$s.t. \quad y_i(w x_i + b) \geq 1 \quad (4.4)$$

From the knowledge of logistic regression, the probability that  $y = +1$  given  $x$  is

$$Pr(y = 1|x) = \frac{1}{1 + \exp(Af(x) + B)} \quad (4.5)$$

Thus, if  $y_i(w x_i + b) \geq 1$ , the probability that  $y = +1$  is very high. Equally, the probability that  $y = 1$  is very high when  $y_i(w x_i + b) \leq -1$ . That is the reason why we use  $y_i(w x_i + b) \geq 1$  instead of  $y_i(w x_i + b) \geq 0$  in order to strengthen the confidence. We can get the optimal margin classifier from its solution.

In order to solve this problem, we construct the Lagrangian with the Lagrange multiplier

$\lambda \geq 0$ :

$$L(w, b, \lambda) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \lambda_i [y_i (w x_i + b) - 1] \quad (4.6)$$

Then take the partial differential of  $L$  respect to  $w, b$

$$\frac{d}{dw} L(w, b, \lambda) = w - \sum_{i=1}^m \lambda_i y_i x_i = 0 \quad (4.7)$$

We get

$$w = \sum_{i=1}^m \lambda_i y_i x_i \quad (4.8)$$

$$\frac{d}{db} L(w, b, \lambda) = \sum_{i=1}^m \lambda_i y_i = 0 \quad (4.9)$$

Plugging the equation (4.7) and equation (4.8) into equation (4.5) and simplify, we can get the dual problem

$$\max_{\lambda} \sum_{i=1}^n \lambda_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j x_i^T x_j \quad (4.10)$$

Subject to

$$0 \leq \lambda_i \leq C \quad \text{for } i = 1, 2, 3, \dots, n \quad (4.11)$$

$w$  can be got by equation (4.6), since we know the value of  $\lambda$ . Furthermore, the decision function  $f(x)$  would be

$$f(x) = \text{sgn}\left(\sum_{i=1}^m \lambda_i y_i x_i^T x + b\right) \quad (4.12)$$

The  $x$  in equation (4.12) is the input which we want to classify.

### 4.4.3 Kernels

The application of SVM in real-world problems is more complicated than what we discussed before, because real-world would not have such clean data and perfect assumptions. In most cases, as Fig. 4.3 shows, the separating hyperplane is non-linear, since positive and

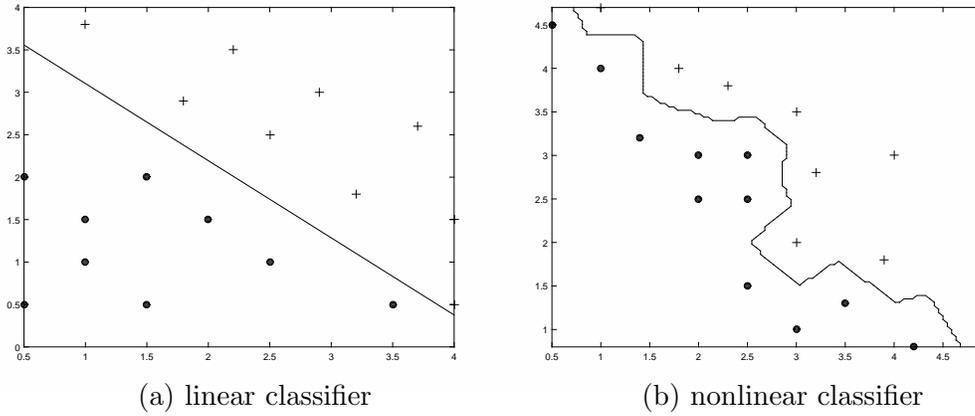


Figure 4.3: Linear and nonlinear classifier.

negative examples have overlaps. In order to solve those problems, a more efficient non-linear classifier, kernels, will be introduced.

There are several kernel functions such as sigmoid kernels, polynomial kernels, and Gaussian radial basis kernels. Gaussian radial basis function (RBF) is powerful and widely used. It is defined as:

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0 \quad (4.13)$$

In addition, if there is a feature  $\Phi$  extracted from the data, the kernel would be

$$K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle \quad (4.14)$$

Replacing the  $x_i^T x_j$  in function (4.12) with  $K(x_i, x_j)$ , we can obtain the non-linear separating hyperplane which cuts data space into two parts. Thus, the final decision function  $f(x)$  would be equation (4.15).

$$f(x) = \text{sgn}\left(\sum_{i=1}^m \lambda_i y_i \Phi(x_i)^T \Phi(x) + b\right) \quad (4.15)$$

#### 4.4.4 SVM Parameters

Both  $C$  and  $\gamma$  are important parameters which affects the classification accuracy of the SVM model. The parameter  $C$  balances the relative weighting between two goals of ensuring that most examples have functional margin at least 1 and of making the  $\|w\|^2$  small. If the parameter  $C$  is very large, the hypothesis will have a low bias and a high variance, which means the hypothesis will be over-fitting. On the other hand, if the parameter  $C$  is very small, the hypothesis will have a lower variance and a high bias, which means the hypothesis will be under-fitting. For parameter  $\gamma$ , the hypothesis will have a low bias and a high variance and the features are less smooth when  $\gamma$  is large. If the parameter  $\gamma$  is small, the hypothesis will have a low variance and a high bias and the features are more smooth. In this thesis, cross validation and iterative grid search techniques are used in order to reach the best optimal parameter pair  $C$  and  $\gamma$ . In general, all data should be divided into two groups for training and testing. Training data is used to establish SVM model including parameter pair  $C$  and  $\gamma$ . In the search, the values of  $C$  and  $\gamma$  are bounded in the same sets  $\{10^{-4}, 10^{-3}, \dots, 10^2, 10^3\}$ . In order to get the optimal parameter pair  $C$  and  $\gamma$ , we perform the 4-fold cross validation. We cut all training data into 4 groups with equal size. Then use three of them to train the model and use the rest of them to test. Thus, for any combination of  $C$  and  $\gamma$ , there is an accuracy corresponding. The best parameter pair  $C$  and  $\gamma$  are picked with the best accuracy.

## 4.5 Multi-class Classification

All have been discussed above is in binary case whose output is labeled into two classes. However, multi-class SVM is designed to classify three or more categories. Suppose  $k$  categories need to be classified, we train  $k$  SVMs. Each one distinguish  $y = i$  with the rest, for  $i = 1, 2, 3, \dots, k$ . We can get  $w^{(1)}, w^{(2)}, \dots, w^{(k)}$  and pick class  $i$  with largest  $f_i(x) = w^{(i)T}x + b$ .

In the future work, the movement data that have been used above would be separated in 3 groups corresponding to non-activation, thumb and pinky. We will treat the movement of thumb and pinky as activation in order to transfer the problem to binary case.

## 4.6 Application of SVM on Movement Data

Obviously, the movement data is nonlinear. To demonstrate the performance of the proposed method, we will use support vector machine with Gaussian kernels to analyze 667 movement trials collected by the procedure given in the previous section. In our method, the input ( $X$ ) of SVM is the mean frequency and slope of auto-correlation function of energy from the first 3 IMFs among 9 axis vector defined in the previous section,  $X = [X_{xa}, X_{ya}, X_{za}, X_{xg}, X_{yg}, X_{zg}, X_{xm}, X_{ym}, X_{zm}]$ . The output ( $y$ ) is positive ( $y = 1$ ) when the pinky moved. On the other hand,  $y$  is negative ( $y = 0$ ) when the thumb moved. In activation and non-activation case,  $y$  is positive ( $y = 1$ ) when the the type of movement is non-activation and  $y$  is negative ( $y = 0$ ) when the the type of movement is non-activation. We will compare prediction accuracies using this feature with those using different combinations of elements in it to show the proposed feature is superior than the other features.

## Chapter 5

### Experiment Result

#### 5.1 Classification of Pinky and Thumb

We use our feature  $X$  which is a  $1 \times 634$  vector

$$X = [X_{xa}, X_{ya}, X_{za}, X_{xg}, X_{yg}, X_{zg}, X_{xm}, X_{ym}, X_{zm}]$$

as input for SVM. For each axis, the feature is

$$X_l = [m_f(1), R_1(10), \dots, R_1(30), m_f(2), R_2(10), \dots, R_2(30), m_f(3), R_3(10), \dots, R_3(30)]$$

Secondly, we choose 80% data ( $N_{train} = 533$ ) to train model and the rest 20% data ( $N_{test} = 134$ ) to test. For 5-fold cross validation, cutting all data into 5 parts. We use each of them for testing and others for training. That means we did 5 tests whose training data includes 244 thumb trials and 289 pinky trials and testing data includes 62 thumb trials and 72 pinky trials. Also, for the elements from the feature, we use each one of them and then combine them together in order to find average error for model complexity. In the experiment, the training data are randomly selected and the result changed with different feature combinations. Because the vector  $[R_1(10), \dots, R_1(30)]$  represents the slope, we treat it as an element and denote it as  $L_1 = [R_1(10), \dots, R_1(30)]$  which is a  $1 \times 21$  vector. Similarly, we have  $L_2 = [R_2(10), \dots, R_2(30)]$  and  $L_3 = [R_3(10), \dots, R_3(30)]$ . Thus, the whole feature is

$$X_l = [m_f(1), L_1, m_f(2), L_2, m_f(3), L_3]$$

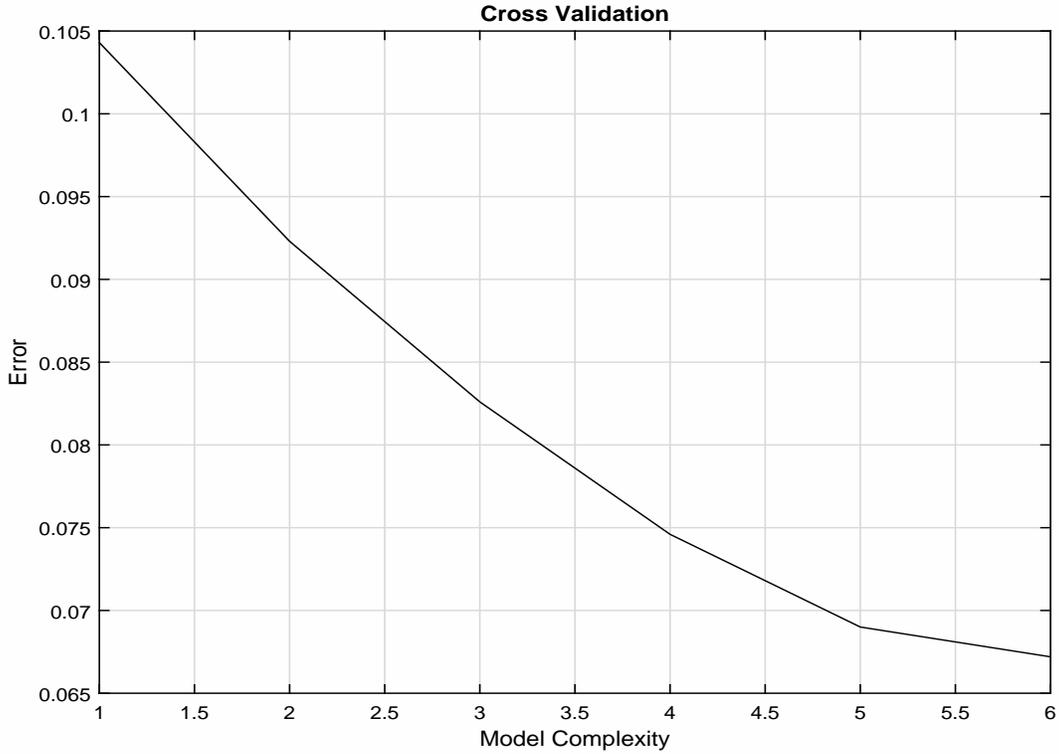


Figure 5.1: Cross Validation

There are 6 elements in the feature which are  $m_f(1)$ ,  $L_1$ ,  $m_f(2)$ ,  $L_2$ ,  $m_f(3)$ , and  $L_3$ , so  $6^2 - 1 = 63$  combinations can be got from them. The result is shown on the Appendix A. In that table, we denote the prediction error rate of pinky as  $e_p$  and the prediction error rate of thumb as  $e_t$ .

Comparing all of the 63 combinations, we found that the accuracies from some combinations are higher than others. Next, we perform best subset selection by calculating accuracy for each possible combination of the 6 elements in the feature. Firstly, if a new feature just has one element, there are 6 possible combinations. Calculate accuracy for all of them and choose the highest one. Secondly, if a new feature has two elements, there are  $\binom{6}{2} = 15$  possible combinations. Calculate accuracy for all of them and choose the highest one, and so forth. The cross validation curve shows on Fig. 5.1

The best accuracy from the simulation results is 93.28% from the feature

$$X_t = [m_f(1), L_1, m_f(2), L_2, m_f(3), L_3]$$

The confusion matrix in this case is shown on Table 5.1.

| Confusion matrix(%) |       | Actual label |       |
|---------------------|-------|--------------|-------|
|                     |       | pinky        | thumb |
| Prediction          | pinky | 93.3         | 5.3   |
|                     | thumb | 7.9          | 94.7  |

Table 5.1: Confusion matrix using all features

Now, we try to compare this result with the feature not using HHT. If we use original data as the feature which is

$$X = [x_a^T(t), y_a^T(t), z_a(t), x_g^T(t), y_g^T(t), z_g^T(t), x_m^T(t), y_m^T(t), z_m^T(t)]$$

The accuracy from simulation result is 76.53% and the confusion matrix is shown on Table 5.2.

| Confusion matrix(%) |       | Actual label |       |
|---------------------|-------|--------------|-------|
|                     |       | pinky        | thumb |
| Prediction          | pinky | 73           | 21    |
|                     | thumb | 27           | 79    |

Table 5.2: Confusion matrix using original data

If we still use the slope from the auto-correlation function as the feature without HHT.

The feature is

$$X_l = [R(10), R(11), \dots, R(29), R(30)]$$

The accuracy from simulation result is 87.32% and the confusion matrix is shown on Table 5.3.

| Confusion matrix(%) |       | Actual label |       |
|---------------------|-------|--------------|-------|
|                     |       | pinky        | thumb |
| Prediction          | pinky | 85           | 11    |
|                     | thumb | 15           | 89    |

Table 5.3: Confusion matrix without HHT

## 5.2 Classification of Activation and Non-activation

Now, we use the same algorithm and method to classify activation and non-activation movement. Due to the fact that the data from different people, we need to analyze separately. The results listed in this section using the data ( $N^1 = 374$ ) from the first candidate. The results from second candidate is similar.

### 5.2.1 Feature Extraction

We still try to find the significant difference from Hilbert Spectrum between different classes. Fig. 5.2 is the Hilbert Spectrum from non-activation and activation. From those figures, we can get that the frequency (first 3 IMFs) from non-activation is not obviously different from the frequency (first 3 IMFs) from activation. So the mean of frequency from first 3 IMFs

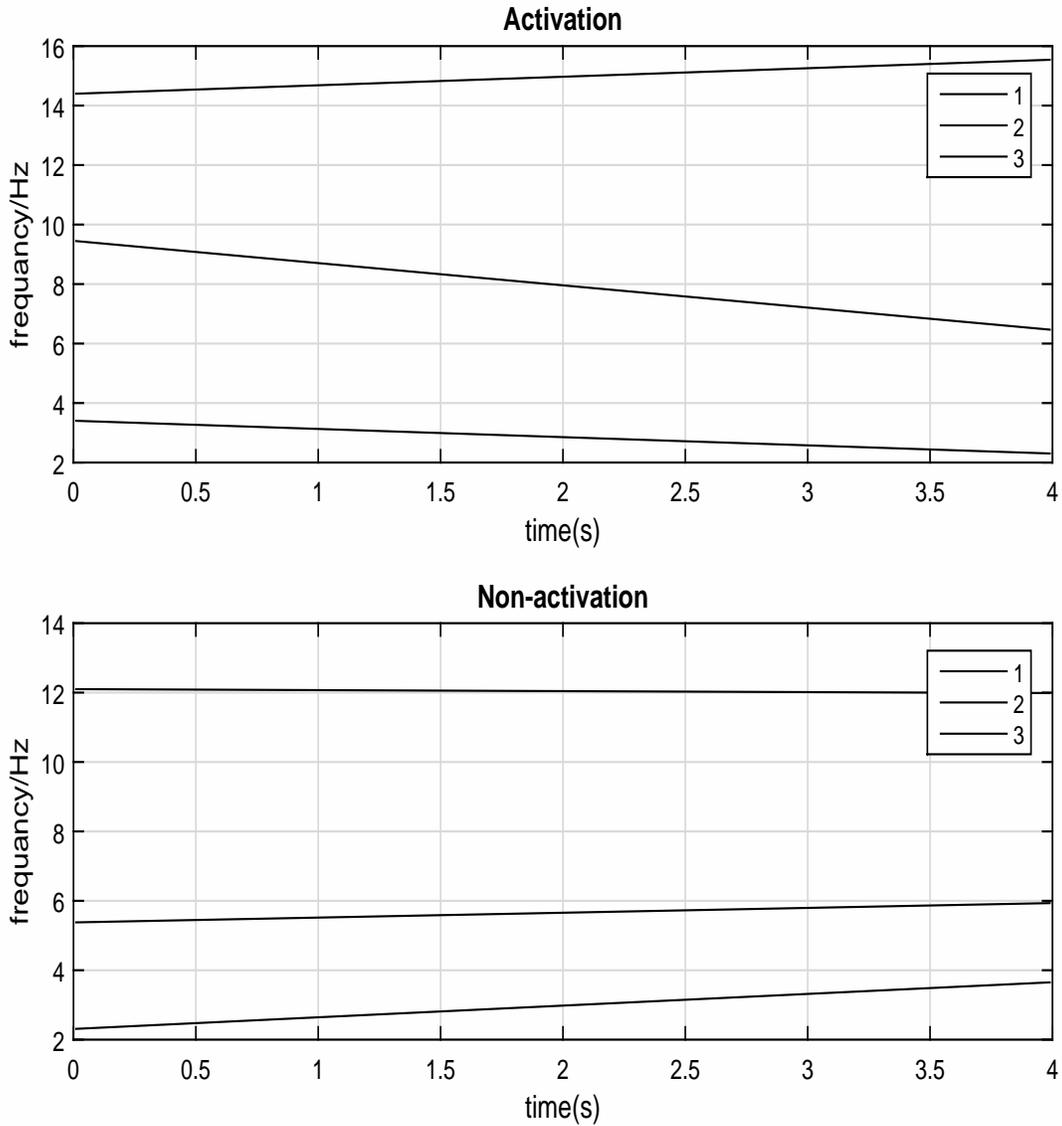


Figure 5.2: Hilbert Spectrum of Movement from Thumb and Pinky

is not a good features in this case. However, we still try computing the auto-correlation function of the energy from first 3 IMFs in different classes.

Fig. 5.3 is the auto-correlation function from activation and non-activation. From those figures, we can get the slope (first 3 IMFs) from non-activation is higher than the slope (first 3 IMFs) from activation. So we decide use the points from R(10) to R(30) from first 3 IMFs,  $X_l = [L_1, L_2, L_3]$  as the feature.

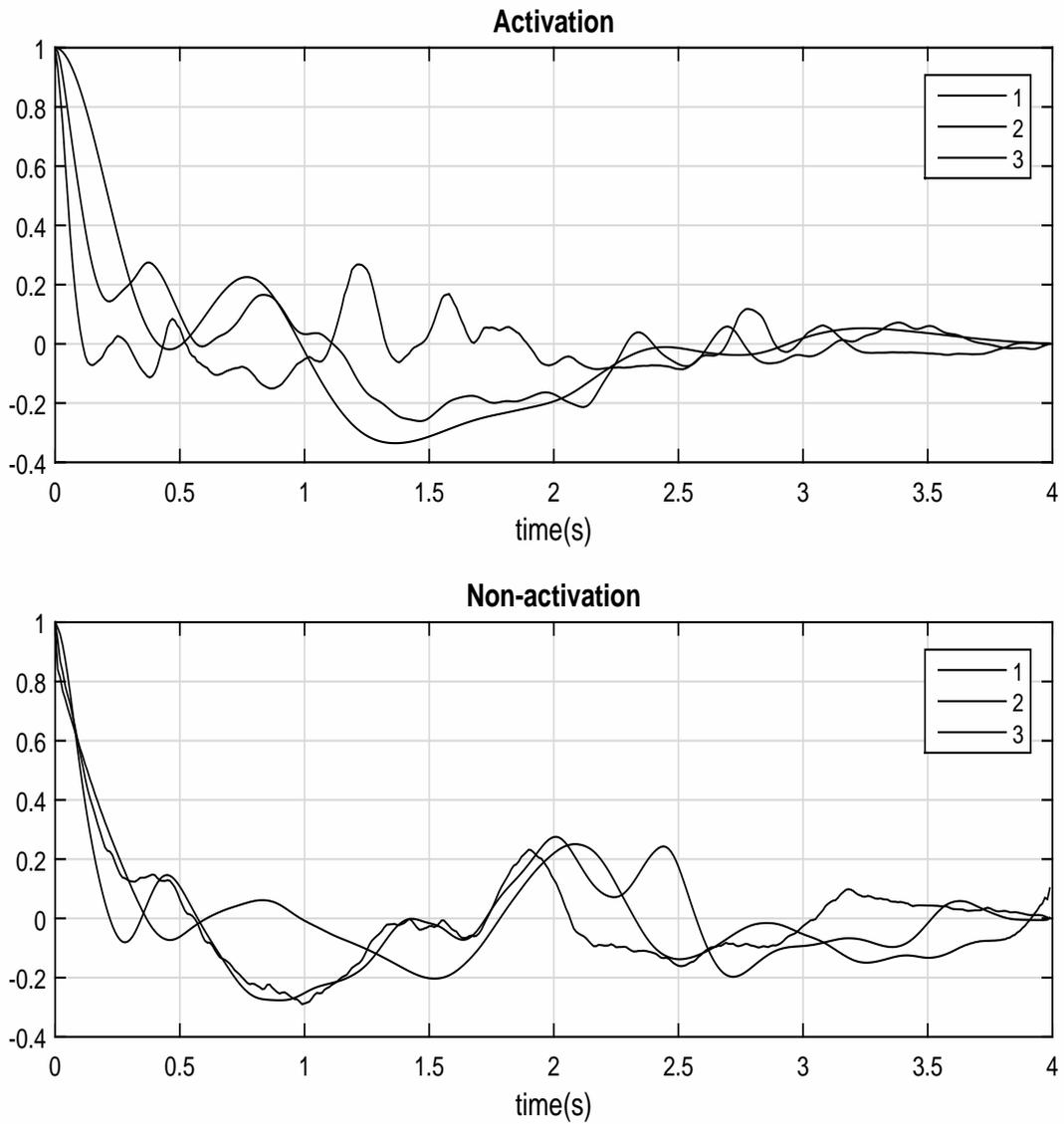


Figure 5.3: Auto-correlation of Energy of Movement from Thumb and Pinky

### 5.2.2 Experiment Result

Now we get the feature  $X$  which is a  $1 \times 63$  vector

$$X_l = [L_1, L_2, L_3]$$

as our final feature.

Now, we use our feature  $X$  as input for SVM, and we use 80% data to train model and rest 20% data to test. For 5-fold cross validation, cutting all data into 5 parts. We use each of them for testing and others for training. Also, for the elements from the feature, we use each one of them and then combine them together in order to find average error for model complexity. In the experiment, the training data are randomly selected and the result changed with different feature combinations. There are 3 elements in the feature which are  $L_1$ ,  $L_2$ , and  $L_3$ , so 7 combinations can be got from them. The result is shown on the Appendix B.

Comparing all the 7 combinations, we found that the accuracies are almost 100% in most cases. The best accuracy from that simulation results is 100% and the false alarm  $P_{FA} = 0$ ,  $P_D = 1$  from the feature

$$X_l = [L_1, L_2, L_3]$$

The confusion matrix is shown on Table 5.4.

| Confusion matrix(%) |                | Actual label |                |
|---------------------|----------------|--------------|----------------|
|                     |                | activation   | non-activation |
| Prediction          | activation     | 100          | 0              |
|                     | non-activation | 0            | 100            |

Table 5.4: Confusion matrix using all features

Now, we try to compare this result with the feature not using HHT.

If we use original data as the feature which is

$$X = [x_a^T(t), y_a^T(t), z_a(t), x_g^T(t), y_g^T(t), z_g^T(t), x_m^T(t), y_m^T(t), z_m^T(t)]$$

The accuracy from simulation result is 81.63% and the false alarm  $P_{FA} = 0.29$ ,  $P_D = 0.99$ .

The confusion matrix is shown on Table 5.5.

| Confusion matrix(%) |                | Actual label |                |
|---------------------|----------------|--------------|----------------|
|                     |                | activation   | non-activation |
| Prediction          | activation     | 99           | 29             |
|                     | non-activation | 1            | 71             |

Table 5.5: Confusion matrix using original data

If we still use the slope from the auto-correlation function as the feature without HHT.

The feature is

$$X_l = [R(10), R(11), \dots, R(29), R(30)]$$

The accuracy from simulation result is 98% and the false alarm  $P_{FA} = 0.02$ ,  $P_D = 1$ . The confusion matrix is shown on Table 5.6.

| Confusion matrix(%) |                | Actual label |                |
|---------------------|----------------|--------------|----------------|
|                     |                | activation   | non-activation |
| Prediction          | activation     | 100          | 2              |
|                     | non-activation | 0            | 98             |

Table 5.6: Confusion matrix without HHT

## Chapter 6

### Conclusions and Future Work

This chapter will discuss some conclusions from the results and give some advice for future work.

#### 6.1 Conclusions

Up to now, the main object of this project is to extract feature from data and build the algorithm to classify the two types of finger movement. According to the non-linearity and non-stationary of movement signals, we presented an automatic classification technique using Hilbert-Huang Transform. Hilbert-Huang transform includes the empirical mode decomposition method and the associated Hilbert spectral analysis. The feature which was used get accuracy 93% classification of thumb and pinky and accuracy 100% classification of non-activation and activation with a low false alarm. The result is much better than extracting the feature without using HHT. Thus, the Hilbert-Huang transform based method can be used as an effective movement classification. The process of empirical mode decomposition (EMD) can mine data deeply. It can analyze the correlation and the feature hidden among the data.

Moreover, the classification method, support vector machine (SVM) with Gaussian kernel is an effective way to realize binary classification. From the cross validation curve, we can figure out the error decreases as the model complexity increases. That means more elements

in feature, more accuracy we will get.

## **6.2 Future Work**

Due to the result we got previously, the expectation is that future work on finger movement classification would obtain better performances using more advanced algorithm such as deep learning to achieve higher accuracy and classify more than two types of movement with significant features. This way, we can obtain more information just from finger movements helping severe disability people communicate with others in their daily life.

## Chapter 7

### Reference

- [1] Williams, Ethan Storm. Design of an Assistive Technology Adaptive Switch using an Inertial Measurement Unit. Diss. University of Arkansas, 2016.
- [2] Huang, Norden E., et al. "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis." Proceedings of the Royal Society of London A: mathematical, physical and engineering sciences. Vol. 454. No. 1971. The Royal Society, 1998.
- [3] Li, Yi, et al. "Sleep stage classification based on EEG Hilbert-Huang transform." Industrial Electronics and Applications, 2009. ICIEA 2009. 4th IEEE Conference on. IEEE, 2009.
- [4] Ding, Pei, Chongchong Yu, and Xiaoyu He. "A study and application of HHT in vibration signal analysis of bridge structural health monitoring system." Image and Signal Processing, 2009. CISP'09. 2nd International Congress on. IEEE, 2009.
- [5] Li, Dechang Yang Yong, Christian Rehtanz, and Rongkun Xiu. "Analysis of low frequency oscillations in power system based on HHT technique." Environment and Electrical Engineering (EEEIC), 2010 9th International Conference on. IEEE, 2010.
- [6] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. (1992). A Training Algorithm for Optimal Margin Classifiers. ACM New York, NY, USA, 144-152.
- [7] Drish, J. (1998). Obtaining Calibrated Probability Estimates from Support Vector Machine.

- [8] Guo, Qing. Improving Golf Putt Performance with Statistical Learning of EEG Signals. University of Arkansas, 2014.
  
- [9] Ojutiku, Alli Ayoola. Text-Independent Speaker Identification using Statistical Learning. Diss. University of Arkansas, 2015.

## Chapter 8

### Appendix

#### Appendix A: Results of different combination of feature

| Feature                | Accuracy(%) | $e_p$ (%) | $e_t$ (%) |
|------------------------|-------------|-----------|-----------|
| $X = [m_f(1)]$         | 86.3        | 12.3      | 14.4      |
| $X = [L_1]$            | 87.2        | 13.8      | 11.5      |
| $X = [m_f(2)]$         | 89.5        | 13.7      | 8.7       |
| $X = [L_2]$            | 88.8        | 12.5      | 9.4       |
| $X = [m_f(3)]$         | 86.4        | 13.8      | 14.2      |
| $X = [L_3]$            | 87.3        | 8.2       | 18.1      |
| $X = [m_f(1), L_1]$    | 90.8        | 6.9       | 11.5      |
| $X = [m_f(1), m_f(2)]$ | 88.5        | 12.7      | 10.9      |
| $X = [m_f(1), L_2]$    | 88.2        | 11.7      | 9.6       |
| $X = [m_f(1), m_f(3)]$ | 89.1        | 12.5      | 9.0       |
| $X = [m_f(1), L_3]$    | 89.5        | 13.7      | 8.7       |
| $X = [L_1, m_f(2)]$    | 88.9        | 13.7      | 8.1       |
| $X = [L_1, L_2]$       | 88.7        | 12.5      | 9.0       |
| $X = [L_1, m_f(3)]$    | 90.2        | 12.5      | 92.3      |
| $X = [L_1, L_3]$       | 88.9        | 12.8      | 8.9       |

### Appendix A (Cont.)

| Feature                        | Accuracy(%) | $e_p$ (%) | $e_t$ (%) |
|--------------------------------|-------------|-----------|-----------|
| $X = [m_f(2), L_2]$            | 87.2        | 13.8      | 11.5      |
| $X = [m_f(2), m_f(3)]$         | 88.5        | 10.6      | 89.2      |
| $X = [m_f(2), L_3]$            | 87.3        | 8.2       | 18.1      |
| $X = [L_2, m_f(3)]$            | 89.7        | 12.5      | 7.8       |
| $X = [L_2, L_3]$               | 88.8        | 12.5      | 9.4       |
| $X = [m_f(3), L_3]$            | 89.5        | 12.7      | 9.1       |
| $X = [m_f(1), L_1, m_f(2)]$    | 90.1        | 11.3      | 8.3       |
| $X = [m_f(1), L_1, L_2]$       | 90.8        | 6.9       | 11.5      |
| $X = [m_f(1), L_1, m_f(3)]$    | 88.9        | 12.7      | 9.1       |
| $X = [m_f(1), L_1, L_3]$       | 88.5        | 13.7      | 9         |
| $X = [m_f(1), m_f(2), L_2]$    | 91.0        | 8.2       | 9.9       |
| $X = [m_f(1), m_f(2), m_f(3)]$ | 90.5        | 7.3       | 11.8      |
| $X = [m_f(1), m_f(2), L_3]$    | 89.5        | 11.1      | 9.9       |
| $X = [m_f(1), m_f(2), m_f(3)]$ | 91.7        | 6.8       | 9.9       |
| $X = [m_f(1), L_2, m_f(3)]$    | 89.7        | 12.5      | 7.8       |
| $X = [m_f(1), L_2, L_3]$       | 90.1        | 11.3      | 8.3       |
| $X = [m_f(1), m_f(3), L_3]$    | 89.5        | 12.7      | 9.1       |
| $X = [L_1, m_f(2), L_2]$       | 90.9        | 8.3       | 9.9       |

### Appendix A (Cont.)

| Feature                             | Accuracy(%) | $e_p$ (%) | $e_t$ (%) |
|-------------------------------------|-------------|-----------|-----------|
| $X = [L_1, m_f(2), m_f(3)]$         | 90.9        | 8.3       | 9.8       |
| $X = [L_1, m_f(2), L_3]$            | 91.0        | 8.2       | 9.9       |
| $X = [L_1, m_f(2), L_3]$            | 90.2        | 4.2       | 16.4      |
| $X = [L_1, L_2, m_f(3)]$            | 89.5        | 12.5      | 8.2       |
| $X = [L_1, L_2, L_3]$               | 90.3        | 12.3      | 6.6       |
| $X = [L_1, m_f(3), L_3]$            | 91.3        | 8.3       | 9.2       |
| $X = [m_f(2), L_2, m_f(3)]$         | 89.6        | 11.3      | 8.3       |
| $X = [m_f(2), m_f(3), L_3]$         | 89.7        | 12.5      | 7.8       |
| $X = [L_2, m_f(3), L_3]$            | 90.1        | 11.3      | 8.3       |
| $X = [m_f(1), L_1, m_f(2), L_2]$    | 91.9        | 8.2       | 8.1       |
| $X = [m_f(1), L_1, m_f(2), m_f(3)]$ | 87.3        | 10.9      | 14.8      |
| $X = [m_f(1), L_1, m_f(2), L_3]$    | 91.7        | 6.9       | 9.9       |
| $X = [m_f(1), L_1, L_2, m_f(3)]$    | 91.0        | 8.2       | 9.9       |
| $X = [m_f(1), L_1, L_2, L_3]$       | 92.0        | 8.2       | 8.1       |
| $X = [m_f(1), L_1, m_f(3), L_3]$    | 89.5        | 10.3      | 10.7      |
| $X = [m_f(1), m_f(2), L_2, m_f(3)]$ | 91.7        | 6.9       | 9.9       |
| $X = [m_f(1), m_f(2), L_2, L_3]$    | 90.3        | 10.8      | 8.5       |

### Appendix A (Cont.)

| Feature                                       | Accuracy(%) | $e_p$ (%) | $e_t$ (%) |
|---|-------------|-----------|-----------|
| $X = [m_f(1), m_f(2), m_f(3), L_3]$           | 91.7        | 6.9       | 9.9       |
| $X = [m_f(1), L_2, m_f(3), L_3]$              | 90.4        | 9.9       | 9.4       |
| $X = [L_1, m_f(2), L_2, m_f(3)]$              | 90.5        | 7.3       | 11.8      |
| $X = [L_1, m_f(2), L_2, L_3]$                 | 92.5        | 8.5       | 6.2       |
| $X = [L_1, m_f(2), m_f(3), L_3]$              | 89.3        | 12.5      | 7.8       |
| $X = [L_1, L_2, m_f(3), L_3]$                 | 91.8        | 6.8       | 9.9       |
| $X = [m_f(2), L_2, m_f(3), L_3]$              | 91.7        | 6.9       | 9.9       |
| $X = [m_f(1), L_1, m_f(2), L_2, m_f(3)]$      | 93.2        | 8.2       | 4.2       |
| $X = [m_f(1), L_1, m_f(2), L_2, L_3]$         | 92.0        | 9.1       | 6.8       |
| $X = [m_f(1), L_1, m_f(2), m_f(3), L_3]$      | 91.1        | 8.3       | 9.6       |
| $X = [m_f(1), L_1, L_2, m_f(3), L_3]$         | 93.2        | 9.5       | 4.3       |
| $X = [m_f(1), m_f(2), L_2, m_f(3), L_3]$      | 93.2        | 8.2       | 4.2       |
| $X = [L_1, m_f(2), L_2, m_f(3), L_3]$         | 93.1        | 8.1       | 5.7       |
| $X = [m_f(1), L_1, m_f(2), L_2, m_f(3), L_3]$ | 93.3        | 7.9       | 5.3       |

### Appendix B: Results of different combination of feature

| Feature               | Accuracy(%) | P <sub>FA</sub> (%) | P <sub>D</sub> (%) |
|-----------------------|-------------|---------------------|--------------------|
| $X = [L_1]$           | 98.7        | 2.8                 | 1                  |
| $X = [L_2]$           | 99.1        | 1.7                 | 1                  |
| $X = [L_3]$           | 98.9        | 2.1                 | 1                  |
| $X = [L_1, L_2]$      | 1           | 0                   | 1                  |
| $X = [L_1, L_3]$      | 1           | 0                   | 1                  |
| $X = [L_2, L_3]$      | 1           | 0                   | 1                  |
| $X = [L_1, L_2, L_3]$ | 1           | 0                   | 1                  |

## Appendix C: IRB Approved Letter



Office of Research Compliance  
Institutional Review Board

April 12, 2017

### MEMORANDUM

TO: Kaleb Crow  
Shanqin Sun  
Andrew Greek  
Justin Ryan  
Jingxian Wu

FROM: Ro Windwalker  
IRB Coordinator

RE: New Protocol Approval

IRB Protocol #: 17-02-499

Protocol Title: *Wireless Movement Sensing System for People with Severe Disabilities*

Review Type:  EXEMPT  EXPEDITED  FULL IRB

Approved Project Period: Start Date: 04/11/2017 Expiration Date: 03/20/2018

Your protocol has been approved by the IRB. Protocols are approved for a maximum period of one year. If you wish to continue the project past the approved project period (see above), you must submit a request, using the form *Continuing Review for IRB Approved Projects*, prior to the expiration date. This form is available from the IRB Coordinator or on the Research Compliance website (<https://vpred.uark.edu/units/rscp/index.php>). As a courtesy, you will be sent a reminder two months in advance of that date. However, failure to receive a reminder does not negate your obligation to make the request in sufficient time for review and approval. Federal regulations prohibit retroactive approval of continuation. Failure to receive approval to continue the project prior to the expiration date will result in Termination of the protocol approval. The IRB Coordinator can give you guidance on submission times.

**This protocol has been approved for 10 participants.** If you wish to make *any* modifications in the approved protocol, including enrolling more than this number, you must seek approval *prior to* implementing those changes. All modifications should be requested in writing (email is acceptable) and must provide sufficient detail to assess the impact of the change.

If you have questions or need any assistance from the IRB, please contact me at 109 MLKG Building, 5-2208, or [irb@uark.edu](mailto:irb@uark.edu).