REAL-TIME CLASSIFICATION OF MUSCLE SIGNALS FROM SELF-SELECTED INTENTIONAL MOVEMENTS

by

Kaveh Seyed Momen
B.Sc., Azad University, Iran, 1995

A thesis
presented to Ryerson University
in partial fulfillment of the
requirement for the degree of
Master of Applied Science
in the Program of
Electrical and Computer Engineering.

Toronto, Ontario, Canada, 2005

Author's Declaration

I hereby declare that I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Author's Signature:

[redacted]

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Author's Signature:

[redacted]

Borrower's Page

Ryerson University requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Name	Signature	Address	Date

Abstract

Real-time classification of muscle signals from self-selected intentional movements

©Kaveh Seyed Momen 2005

Master of Applied Science
Department of Electrical and Computer Engineering
Ryerson University

A novel method to automatically differentiate forearm movements has been proposed. The electromyography (EMG) signals were recorded from two muscle sites on the forearm in real-time. Two 2-dimensional feature spaces namely the natural logarithm of root-mean-square values ($Log(RMS)$), and the standard deviation of auto regressive model coefficients ($Stdev(AR)$) were created. The features were calculated within non-overlapping 0.2 second windows in real-time. The feature spaces were clustered using the fuzzy c-means algorithm [1]. The cluster multiplicities were investigated by five different cluster validity indices. Real-time EMG signal classification was achieved by calculating membership values.

$Log(RMS)$ performed superior to the $Stdev(AR)$ feature space. The silhouette validity index provided the best cluster validity index in this study.

On average, the proposed algorithm classified 4 movements with 92.7±3.2% and 5 movements with 79.90% ±16.8% accuracy. The algorithm also revealed the number of repeatable movements. It can also be adapted to daily variations in individual EMG signals.

Acknowledgements

All praise and glory to God, the Compassionate, the Merciful. Thank you God for giving me the knowledge and the courage to do research to help people with disabilities.

I would like to sincerely thank my advisers, Dr. Sridhar Krishnan and Dr. Tom Chau, for providing me with this research opportunity, constant support, encouragement, feedback, and their financial support. They have generously shared their valuable time, knowledge, and ideas during my masters program. I greatly appreciate their support.

I would like to thank Prof. Steve Naumann for providing me with research opportunities at Bloorview MacMillan Children's Centre. His great support and encouragement helped me to start my masters program and pursue my ambitions.

I would like to thank Mr. Alan Morris for his support, encouragement, and supervision during the first year of my masters. I also greatly appreciate his valuable feedback and ideas during the whole program.

I would like to thank the members of the PRISM LAB, Catherine Hutcheson, Jorge Silva, Daniel Cossever, Jesus Miguel De La Rosa Estanol, Stefanie Blain, Natasha Alves, Elaine Biddiss, Scott Young, Jovan Nedeljkovic, and Désiré Ngambia Nbamou for providing a stimulating research environment. Thanks for all their support, laughter and friendship.

I would like to thank the members of the Signal Analysis Research Group with whom I had the opportunity to work, specially, Arunan Ramalingam, Karthikeyan Umapathy, Lam Le, Jiming Yang, April Khademi, and Danoush Hosseinzadeh, for providing valuable feedback and support.

I would like to acknowledge Bloorview MacMillan Children's Hospital Foundation, RE-MAD, Canada Research Chair Program, and the Department of Electrical and Computer Engineering, Ryerson University for providing financial support during my graduate study.

I would like to thank my parents, Nasser and Gitty, for being my first advisers in the university of life. Thanks to all their love, care, encouragement, and their financial support.

Last but not the least, I would like to thank my wife, Maryam, for tolerating the intolerable. Thanks for all her patience, love and care.

Dedication

To my mom and dad, Gitty and Nasser, and my wife, Maryam, for their love, support, and encouragement ...

Contents

List of Figures

List of Tables

Chapter 1

Introduction

It has been reported that in the United States there are approximately 150,000 amputations performed annually. Those numbers are even higher for Mexico and Canada, leaving an estimated 2.5 million amputees living in North America alone [8]. In addition to amputees, people with deformities of congenital defects make up a large portion of those who utilize upper extremity prostheses. A prosthesis is a device, such as an artificial arm , that replaces a part of the body. There are mainly two types of prostheses: Passive prostheses and powered (active) prosthesis.

A passive prosthesis replaces a missing body limb. It may have limited functionality or it may not function at all. Therefore, the amputee may face challenges in performing daily life activities, such as grabbing a glass of water. In contrast, a power prosthesis, which is usually controlled by monitoring muscle signals, adds more functionality to the amputee. The muscle signals are sensed and amplified by electrodes. Then, these signals are fed to a powered prosthesis. For instance, a powered hand prosthesis is capable of opening and closing by responding to the amputee's voluntary forearm muscle contractions.

1.1 Motivation

Although extensive research has been done to improve the design of powered prostheses, there are still some unresolved problems.

- Due in part to a lack of robust real-time classification algorithms, prosthetic hands are yet unable to provide multifunction control. For example, powered prostheses can currently only open and close. Although opening and closing a hand may help an amputee in grasping the objects, the lack of higher degrees of freedom may introduce more problems. For instance, in order to grasp an object, the lack of wrist rotation function in most current prosthetic hands may force the user to contort his forearm in order to orient his prosthetic hand appropriately to complete the task. This may be very frustrating and challenging for amputees with limited functionality.

- Current devices require users to perform activities according to the generally accepted social norms for executing the target activity. For example, flexion of the forearm muscles is typically associated to prosthetic hand closing. However, these normative modes of activity execution may not be natural for the user and often do not make best use of the individual's abilities. As a consequence, effective device usage requires significant user training.

1.2 Objectives

Although there has been extensive research on Electromyographic (EMG) pattern recognition to classify different movements by monitoring EMG signals, most of these methods are done off-line, which make them unsuitable for real life applications. In previous research, the user's ability to produce self-selected functional movements is almost entirely ignored. In most cases, the user is also required to generate a pre-determined set of signals (i.e. wrist flexion and extension) to activate the assistive device, which are meaningful to able-bodied subjects but meaningless to amputees.

The objective of this research is to develop a novel algorithm that can provide reliable real-time classification while allowing the user to select preferred muscle contraction. In realizing this main objective, a number of secondary objectives should be achieved. In particular, the algorithm should aim to:

1. Differentiate different forearm movements based on the user's self-selected intentional

movements.

2. Indicate the most robust and repeatable movements produced over time.
3. Adapt itself overtime to accommodate daily variations in EMG signals, for example, due to fatigue.
4. Be compatible with current clinical practice. This implies that the algorithm should receive EMG signals as the controlling signals from two recording sites.
5. To classify signals without a user-perceivable delay.
6. It is desirable to implement the algorithm in a simple microprocessor based system, rather than expensive DSP chips due to both the cost and complexity. Therefore, it is expected that the algorithm should not use sophisticated signal processing and pattern recognition routines.

1.3 Chapter Roadmap

Chapter 2 presents background information on muscle physiology, past studies, and relevant clustering methods used in this study. Chapter 3 introduces the experimental method, including both the equipment set-up and the protocol used in this study. The results of the experiments are outlined in Chapter 4. Chapter 5 discusses and analyzes the results. Finally, the conclusion, suggested future work, and the summary of the author's contribution in this research are summarized in Chapter 6.

1.4 Summary of the contributions

Figure 1.1 summarizes the contributions of this thesis. These contributions will be explained fully in Section 6.2.

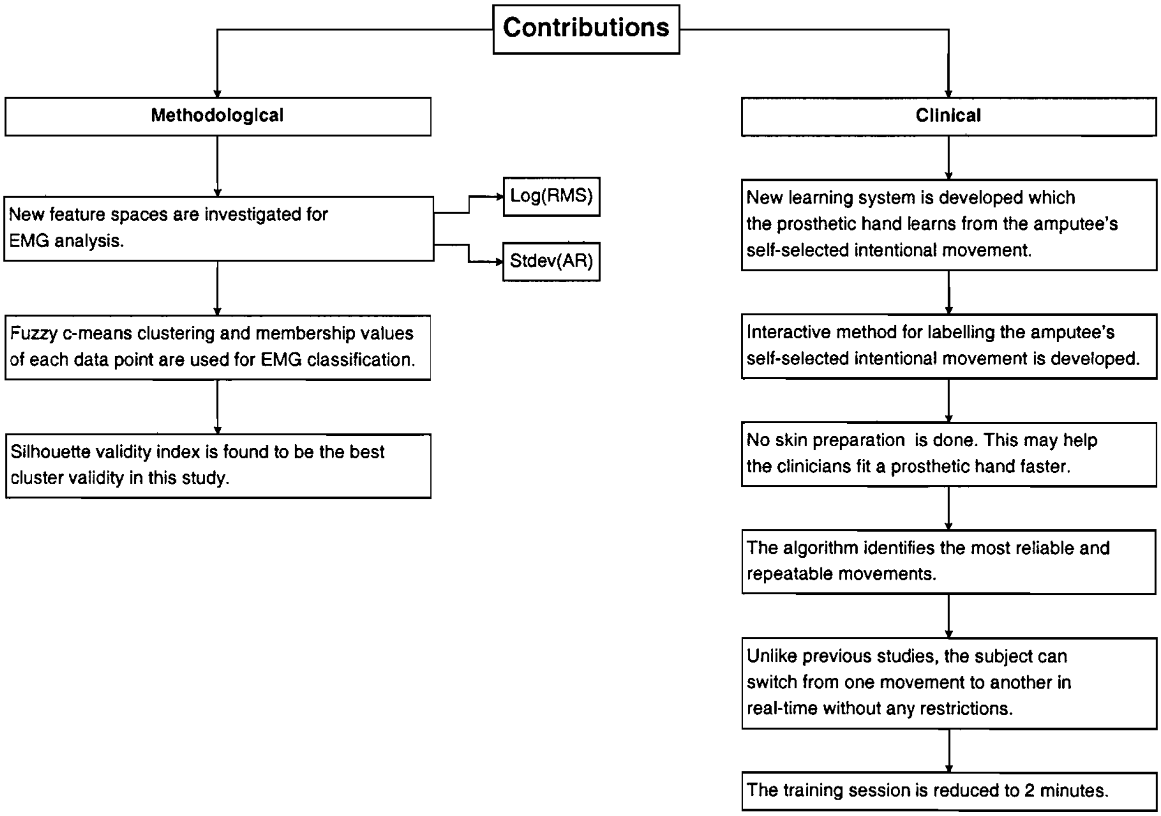

Contributions
Methodological
Clinical
New feature spaces are investigated for EMG analysis.
Log(RMS)
Stdev(AR)
Fuzzy c-means clustering and membership values of each data point are used for EMG classification.
Silhouette validity index is found to be the best cluster validity in this study.
New learning system is developed which the prosthetic hand learns from the amputee's self-selected intentional movement.
Interactive method for labelling the amputee's self-selected intentional movement is developed.
No skin preparation is done. This may help the clinicians fit a prosthetic hand faster.
The algorithm identifies the most reliable and repeatable movements.
Unlike previous studies, the subject can switch from one movement to another in real-time without any restrictions.
The training session is reduced to 2 minutes.

Figure 1.1: Summary of the contributions

Chapter 2

Background

2.1 Introduction

Externally powered prosthetic hands are typically controlled using electromyographic (EMG) signals. These signals originate from the polarization and depolarization of the muscle membrane during voluntary contractions and can be measured at the skin surface using either dry or wet-type electrodes. Dry electrodes are widely used for prosthesis applications. The term 'dry' refers to the fact that the metallic part of the electrode is directly placed on the skin without the need for gels or creams. These types of electrodes usually have a built-in amplifier. Wet electrodes or pre-gelled electrodes use gels as a conductive material between the skin and the metallic part of the electrode. In some other applications such as individual muscle studies, 'needle electrodes' are invasively inserted into muscle. For powered prosthesis control, surface EMG is typically used since its non-invasive character makes it safer and easier to use.

EMG signals can tell us some useful information about the state of the muscle: it is widely used in both medicine (i.e. diagnostics) and engineering (i.e. powered prosthesis). In order to have a better understanding of EMG signals, it is useful to review some background on muscle architecture and how muscle electricity is generated and detected.

This chapter starts with a brief review of muscle physiology. Section 2.2.1 to Section 2.2.3 explain how muscles contract and produce electricity. Section 2.2.4, will discuss the generation and characteristics of EMG signals. Section 2.2.6 briefly introduces the most commonly used technique in EMG recording. In Section 2.3, a summary of the past studies on multi-function EMG classification is reviewed and the major issues will be discussed. This chapter ends with a summary of the clustering, and the cluster validity algorithms used in this study.

2.2 Muscle Physiology

Almost all muscles in our bodies convert chemical energy, which is derived from the food we digest, into mechanical energy. Skeletal muscles, which are attached to adjacent bones via tendons, help us move voluntarily.

Each skeletal muscle consists of smaller, almost cylindrical elements called *muscle fibres.* These muscle fibres are bundled together in parallel to form a muscle. Muscle fibres, in turn, are composed of *myofibrils.* Myofibrils have contractile properties. These myofibrils contain *thin* and *thick* filaments that slide over each other. This combination results in muscle contraction. The myofilaments are arranged in compartments called *sarcomeres.* These sarcomeres are separated from each other by thin disks called *Z-lines.*[6] The composition of muscle elements has been simplified in Figure 2.1 [2].

2.2.1 Motor Unit

Each muscle fibre has a neuron connected to it called a *motoneuron.* The motoneuron sends neural signals to the muscle fibre. The *motor end plate* is where the motoneuron connects to the muscle fibre, usually near the midpoint of each muscle fibre [6]. Motoneurons look like wires which connect the brain to the muscles. Each motoneuron communicates with several muscle fibres at such motor end plates and this number varies from one type of muscle to another. Muscle fibres receive their stimulation via the motor end plates [6]. These electrical stimulation impulses are called *action potentials* that travel from motor cortex in the brain down to the muscle fibre.

The combination of muscle fibre and its motoneuron creates a motor unit. This mechanism is illustrated in Figure 2.2.

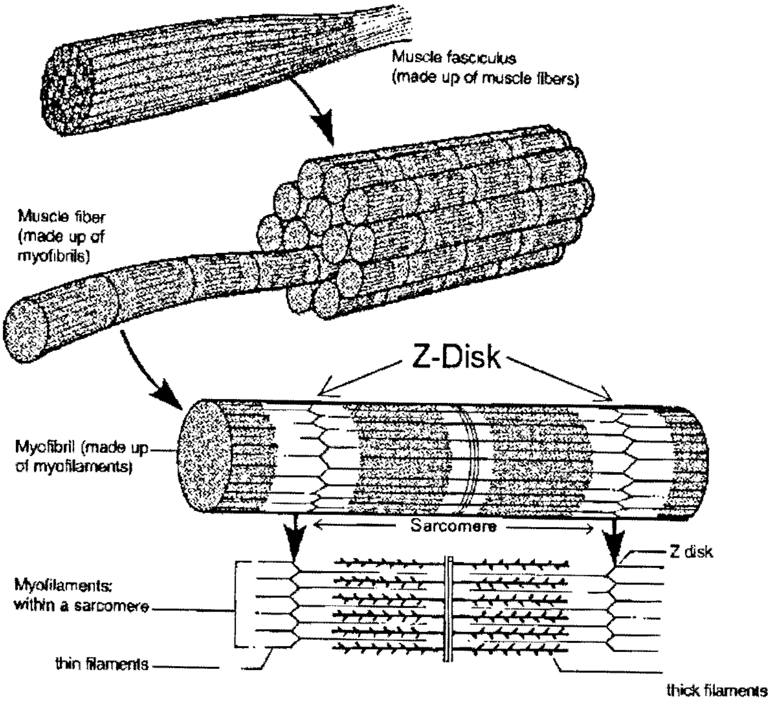


Figure 2.1: The composition of muscle elements (Modified with permission from[2])

2.2.2 Motor Unit Action Potential (MUAP)

As mentioned earlier, an action potential is similar to the flow of electricity down a wire. However, the major difference between a nerve and a wire is that a nerve transmits a single packet of electric charge at a time. An action potential arrives at the end plate of a motoneuron on the muscle fibre [6]. This electrochemical event locally depolarizes the muscle fibre. This releases calcium ions, which subsequently create cross-bridging between thin and thick muscle filaments and as a result, muscle contraction occurs. It should be noted that a single action potential only activates the skeletal muscle fibres for about 0.002 seconds [3]. During this time, the muscle fibres will not be able to generate much force before the action potential decays, calcium is removed from the cell and the fibre relaxes again. A single action

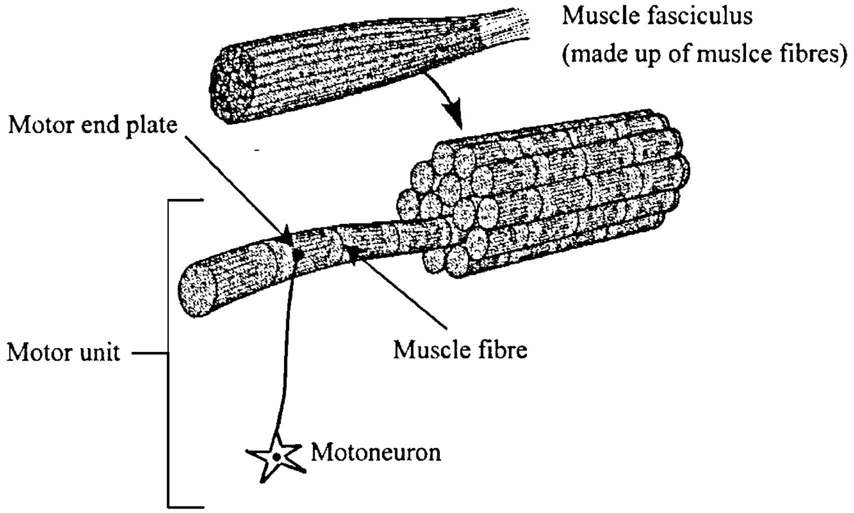


Figure 2.2: A schematic representation of a motor unit and its components. (Modified with permission from [2])

potential will result in a single twitch of the muscle fibres of a similar duration [3].

2.2.3 Motor Unit Action Potential Train (MUAPT)

In order to have a sustained muscle contraction, the muscle needs to be activated repeatedly. Each action potential will cause a twitch, but since the fibre is still partially activated from the previous action potential the subsequent twitch will be greater. Consecutive twitches add together to generate a waveform called the *Motor Unit Action Potential Train* (MUAPT). A sustained muscle contraction happens when a MUAPT is generated [3]. This sequence of events can be shown in Figure 2.3.

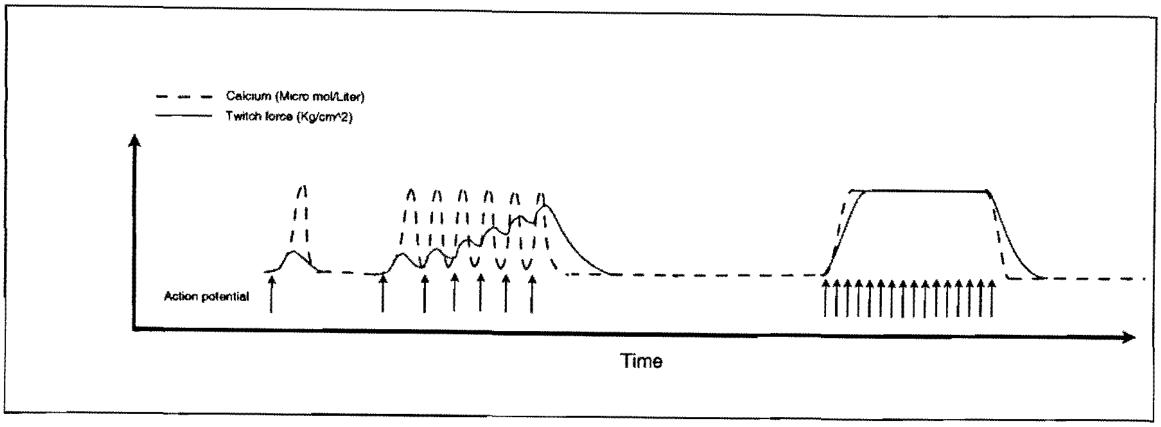


Figure 2.3: The effects of action potentials on muscle. 'Calcium' is usually measured in $\mu mol/Liter$ and 'Twitch force' is usually measured in Kg/cm^2. (Redrawn and modified from [3])

2.2.4 The Origin of EMG

As mentioned in Section 2.2.2, when an action potential arrives at the muscle, it locally depolarizes the muscle fibre. This event generates a potential in the cell membrane which is the basis for the genesis of EMG signal. The actual EMG signal, which propagates through the muscle tissue and can be recorded at the skin surface, is a summation of all these individual cell membrane potentials. Figure 2.5, which has been modified from [4], summarizes how EMG signals are created.

The EMG signals generated at the muscle are different from those detected at the recording site. There are some factors involved in distorting the EMG signal [4]:

- **The location** of the recording electrodes (recording site) is important . EMG signals recorded from a site close to the motor end plates of a muscle fibre are different from EMG signals recorded from locations well away from the same motor end plates.The influence of the electrode location on the amplitude and frequency spectrum of the signal is displayed in Figure 2.4. The preferred location is in the midline of the belly of the muscle between the nearest muscle-nerve junction (innervation zone) and the muscle tendon junction. In this location, the EMG signal with the greatest amplitude is detected [7].

- **Muscle anisotropy:** In reality, the impedance of muscle tissue is not similar in all directions (isotropic). In fact, it is highly direction dependant (anisotropic) [4]. Therefore, the orientation of the detection surfaces of the electrode with respect to the length of the muscle fibre is critical in EMG recording.

- **Tissue filtering:** The muscle tissue can act as a low pass filter and this affects the high frequency component of the generated EMG signals.

- **The size and shape of electrodes:** Referring to Figure 2.5, the detected EMG signal at the recording site is the summation of individual MUAPs. Therefore, the recorded EMG signal is affected by the area and shape of the electrode surface. In particular, the bigger the electrode surface, the higher the number of detected MUAPs, and the higher the EMG amplitude [6].

- **The distance between electrode detection surfaces** determines the bandwidth of the differential electrode configuration [4, 6, 7].

- **Cross-talk** among muscle fibres and the muscle fibre in question affects the recorded EMG signal [6, 4].

- **Noise** The most important source of noise is *environmental noise* such as 60 Hz AC power line. As this noise is common for all recording sites and their recording electrodes, this can be eliminated using differential recording. This technique is widely used for EMG recording as it eliminates the common signal (noise) and amplifies the difference between two electrodes. The interested reader is encouraged to review Part 1, Section 6 of [6] for more information regarding differential EMG signal recording.

2.2.5 EMG properties

An EMG signal can be mathematically expressed as a function of time, t, and force, F, by summing the MUAPTs contributing to the muscle contraction [4]. This simple model can

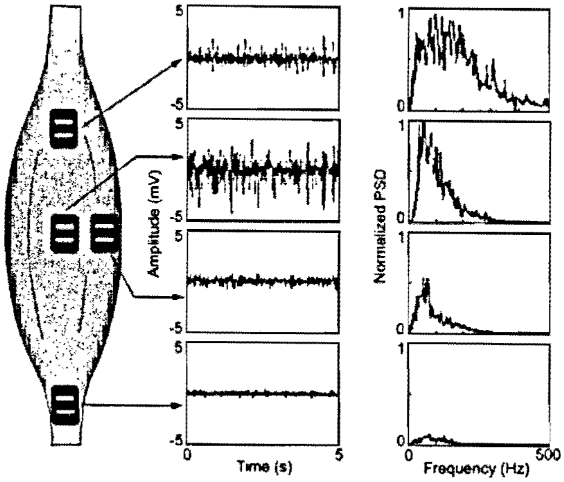


Figure 2.4: The amplitude and frequency spectrum of the EMG signal is affected by the location of the electrode with respect to the innervation zone (top electrode), the myotendonous junction (bottom electrode) and the lateral edge of the muscle (middle right electrode). The preferred location is in the midline of the belly of the muscle between the nearest innervation zone and the myotendonous junction. In this location the EMG signal with the greatest amplitude is detected. (From C.J. DeLuca, 1997, The use of surface electromyography in biomechanics, Journal of Applied Biomechanics 13(2): page 140, Figure 2. ©1997 by Human Kinetics Publishers Inc. Reprinted with permission from Human Kinetics (Champaign, IL))

be shown as [4]:

$$m(t,F) = \sum_{i=1}^{p} u_i(t,F), \tag{2.1}$$

where m is the produced EMG signal at the muscle in question, and u_i is the i^{th} MUAPT contributing to the measured EMG signal. This model has been illustrated in Figure 2.5.

It is well established that the amplitude of the EMG signal is stochastic (random) in nature and can be reasonably represented by a Gaussian distribution function [7]. Due to the random nature of the action potential of the motor unit, the EMG signal is considered stochastic. Raw EMG is biphasic so that it is considered as a zero-mean signal. The amplitude of the signal (peak-to-peak) can be from zero to 10 millivolts. For EMG signals the frequency bandwidth can be from 50 to 500 Hz for some muscles or 300 to 3000 Hz for other muscles [6].

Hogan and Mann [9] have shown that the maximum likelihood (ML) estimation of the

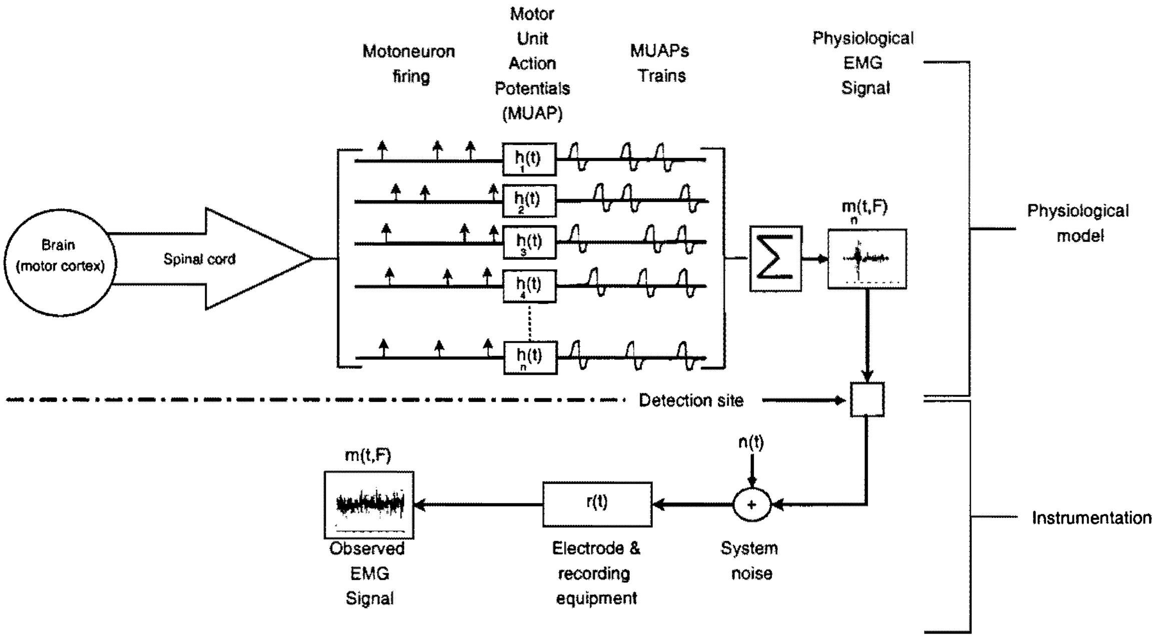


Figure 2.5: Schematic representation of the model for the generation of the EMG signal. (Redrawn and modified from [4])

EMG amplitude based on the Gaussian model is accomplished with the Root-Mean-Square (RMS) processor. The RMS amplitude of EMG signal increases as the voluntary contraction increases [4]. Figure 2.6 shows this relationship. The data were obtained from the biceps, deltoid and First Dorsal Interosseous (FDI) muscles [4]. The RMS feature is the most preferable among other popular time domain features in most applications [4]. It is also recommended for measuring the amplitude of the voluntarily elicited EMG signal because it represents the signal power and thus has a clear physical meaning [7].

Muscle fatigue happens when a contraction is maintained and the muscle fails to produce the desired force. It has been shown that fatigue occurs when the magnitude of the spectrum is shifted toward the lower frequency bands [4].

2.2.6 EMG Recording

As the amplitude of the EMG signal is as small as a few millivolts, any noise with amplitude greater than this will cause interference, and dramatically affect the recorded EMG signal.

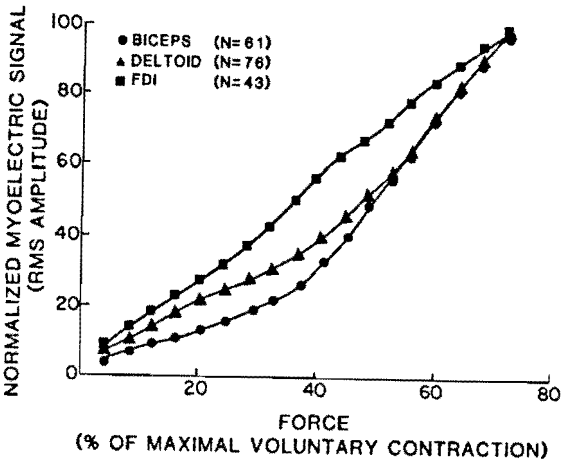


Figure 2.6: Normalized Force / EMG signal relationship for three different muscles. The data have been greatly smoothed, with a window width of 2 seconds. N represents the number of contractions averaged for each muscle. (From C.J. DeLuca, 1997, The use of surface electromyography in biomechanics, Journal of Applied Biomechanics 13(2): page 151, Figure 7. ©1997 by Human Kinetics Publishers Inc. Reprinted with permission from Human Kinetics (Champaign, IL))

Differential recording has been widely used in EMG recording. This technique amplifies the difference between two signals and eliminates the interfering *common* signal. Figure 2.7 illustrates this technique. Many considerations should be made for a clear EMG signal recording. For instance, the input impedance of the recording amplifier should be very high (usually greater than 100 $M\Omega$ [7]), and the wires connected from the electrodes to the amplifier should be as short as possible to minimize contamination due to motion artifact. For complete information regarding EMG recording, the reader is referred to [6, 4, 7]

2.3 Critical Review of Multifunction EMG Classification

2.3.1 Powered prosthesis

Externally powered prosthetic hands are typically controlled using electromyographic (EMG) signals. These signals originate from the polarization and depolarization of the muscle membrane during voluntary contractions and can be measured at the skin surface using either dry

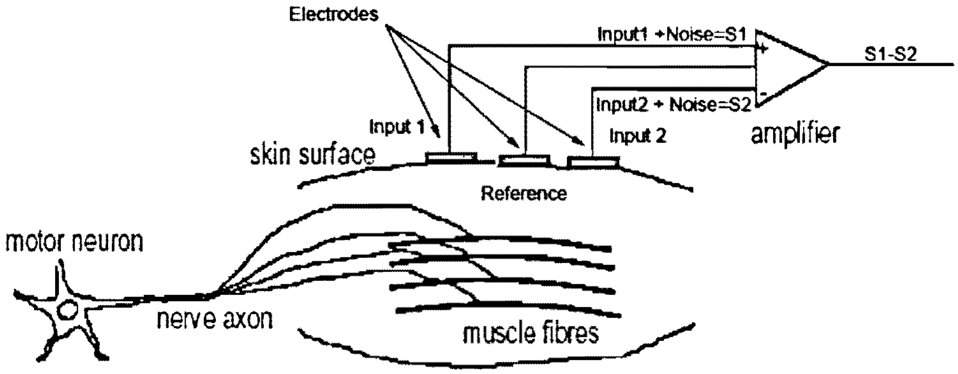


Figure 2.7: Differential EMG recording (Modified from [5] with permission)

or wet-type electrodes. The EMG control signal can be derived from a single site [10, 11, 12] or from multiple sites. Past studies have employed two [13, 14, 15], three [16, 17, 18], four [14, 19, 20, 21, 22, 23, 24], and up to eight [25] recording sites with varying levels of success. Englehart and Hudgins argued that four channels of EMG signals are clearly preferable to two [14]. However, some studies have shown that there is both a practical and theoretical limit to increasing the number of channels [19, 14].

2.3.2 Classification schemes

A much sought after goal in EMG-driven prostheses is to provide the user with multiple limb functions, such as hand opening and closing, and wrist rotation. To control multiple functions, it is necessary to map EMG signals corresponding to different muscle contractions to a variety of prosthetic functions. This mapping is commonly achieved by way of a signal classification scheme. In the past decade, many different EMG classification schemes have been proposed for prosthesis control. These classifiers can be divided into four major categories:

- Linear classifiers [19, 14, 26, 11].
- Artificial neural network classifiers [10, 13, 15, 16, 12, 17, 18, 20, 22, 23].
- Fuzzy and neuro-fuzzy classifiers [10, 27, 28], and

- K-nearest neighbour classifiers [15, 17].

In the majority of these studies, the artificial neural network (ANN) has been the classifier of choice. Classification accuracies ranging from 80% to 100% have been reported using ANN classifiers for 4 to 8 different functions. The number of recording sites varies from 1 to 4 in these studies. Although, most of these methods have only been tested off-line, few real-time implementations have been described. Further, it is unclear how these methods may be adapted to accommodate daily variations in EMG signals, for example, due to fatigue.

2.3.3 Features

A fundamental challenge in building an accurate and reliable EMG signal classifier is the extraction of discriminatory features from EMG signals. The extracted features define the feature space within which the different EMG classifiers can be evaluated. In fact, Englehart et. al. contend that classification performance is more profoundly affected by the choice of the feature set [26] rather than the classifier itself. Different feature sets have been evaluated by various researchers. These feature sets can be categorized into three major categories:

- **Time domain features.** These are extracted from the temporal representation of the signal. Examples include root mean square (RMS), integrated RMS, mean absolute value (MAV), mean absolute value slope (MAVSLP), zero crossing (ZC), waveform length, variance, number of slope sign changes and amplitude histograms [10, 19, 26, 11, 16, 17, 18, 20].

- **Frequency domain features.** These features are derived from a spectral representation of the signal, including such examples as the Fast Fourier Transform (FFT) coefficients, autoregressive (AR) coefficients, and cepstral coefficients [12, 13, 15, 17, 20, 22, 23, 24].

- **Time-frequency features.** The short-time Fourier transform (STFT) coefficients, wavelet coefficients and wavelet packet coefficients [27, 14, 26, 20] are examples of features computed from a joint time and frequency signal representation.

In a recent comprehensive study of 19 potential EMG features, Boostani and Moradi [25] used eight channels of EMG signals to classify 15 different predefined movements. They collected data from 10 amputees. The data analysis appeared to be off-line. They evaluated the 19 different features on the basis of interclass discriminability, noise tolerance and computational complexity. Based on these criteria, they found that the energy of the wavelet coefficients over nine scales along with cepstral coefficients provided the best features.

2.3.4 Real-time dynamic classification

Although an extensive research effort has been devoted to increase the accuracy of EMG-based multifunction myoelectric control, most of these methods have not been implemented in real-time applications. A few successful studies proposed real-time applications [13, 14, 19, 20]. Englehart et. al [14] introduced a wavelet-based "continuous classifier". Although the system achieved on average 0.5% error for four classes of motion, and six classes with 2% error with four channels of steady-state data, there are some issues yet to be resolved:

- No feedback was provided to regulate the force level. Further experiments must be conducted to determine the possibility of implementing velocity control.
- The system must know when to actuate the prosthetic device, and when to suppress actuation. As this system is based on a continuous classifier, with a constant stream of decisions being produced, the actuation must be gated by some means, for example, by introducing an 'inactive' class.

Englehart et. al. [20] used pattern recognition approach to process four channels of myoelectric signals, with the task of discriminating six classes of limb movement. Eleven subjects with intact limbs participated in the study. The subjects were asked to perform six distinct limb motions namely, wrist flexion, wrist extension, supination, pronation, palm open, and palm close. Although they obtained 93.25% accuracy (6.75% error) using a time-domain feature set, with a multilayer perceptron (MLP) artificial neural network classifier, the study relied exclusively on able-bodied individuals to whom the performed movements were perceivable. The next phases of this investigation will have individuals with upper limb amputations assess the control system, and begin clinical trials of an embedded controller.

Nishikawa et. al. [13] proposed a novel on-line learning method for an EMG prosthetic hand control. Although they could discriminate six different hand movements with 89.9% accuracy, the prosthetic hand training was not automatic. The subject had to teach the motions to the prosthetic hand and assign an identifier to each movement. When a subject judged that the system could control a motion intentionally, he taught the hand a new motion, thus directing system learning. On the other hand, when the subject judged that the system was unable to control a motion properly, he had to reteach the motion in question and the system needed to relearn all the target motions.

2.3.5 Able-bodied vs. amputee

Most of the past research efforts have been carried out only with able-bodied subjects [10, 14, 19, 20, 26, 15, 21, 12]. However, we know that there are significant anatomical and physiological differences between the amputated and intact limb. For example, in congenital amputees, there may be partially developed, missing or fused bones. The soft tissue is commonly heavier or thicker than in the intact limb. In the traumatic amputee, the limb may exhibit scarring, grafted skin areas, muscle atrophy and limited range of motion. As a result, the number of available muscle sites and the quality of useful EMG signals may be very different from those of the intact limb, implying that potentially different EMG features and classifiers may be required.

2.3.6 Normative vs. natural

Current devices require users to perform activities according to the generally accepted social norms for executing the target activity. For example, flexion of the forearm muscles is typically associated to prosthetic hand closing. However, these normative modes of activity execution may not be natural for the user and often do not make best use of the individual's abilities. As a consequence, effective device usage requires significant user training.

It appears that in most previous studies, the ability of the subject to express functional intent was ignored. For instance, the subject would be asked to perform different hand movements, i.e. hand close, hand open, finger movement, etc. which are meaningful for able-bodied subjects but meaningless for amputees, especially those that have congenital limb deficiencies.

2.3.7 Supervised vs. unsupervised classifiers

In almost all of the previous experiments, different movements were classified using supervised classifiers. Hence, correct partitioning of the feature space required accurately labelled movements. Such movements could only be obtained according to a strict experimental protocol. In other words, users could not freely generate movement of choice, but had to adhere to the experiment's instruction. Past research often employed pre-recorded EMG databases. These data were recorded with specific electrodes, skin conditions, electrode locations, soft tissue composition, and muscle anatomy. Good performance of the supervised classifiers could only be guaranteed under similar experimental conditions.

Clustering is a technique for automatically grouping data, such as EMG signals. It can be used when the number of classes in the data is unknown or it is not possible to find. As mentioned in Section 1.1, unlike most of the previous studies, we would like to shift the training from the user to the machine. Therefore, the target movements (labels) and the number of movements (number of classes) are not defined. These parameters are expected to be discovered after the user trains the system. As a result, popular supervised classifiers such as artificial neural network classifiers cannot be used. On the other hand, clustering may help us accomplish the objectives of the thesis as outlined in Section 1.2.

2.4 Brief Overview of Relevant Clustering Methods

2.4.1 What is clustering?

Clustering is one of the most important techniques in pattern recognition applications. In clustering, similar objects are grouped together. Each group is called a *cluster.* Different clusters are recognized by the dissimilarities among their members. In clustering, the data are not labelled. Therefore, it is considered as *unsupervised learning.* In contrast, in *supervised learning,* the classes are predefined and the task is to understand the basis for the classification from a set of labelled data (training or learning set). This information is then used to classify future observations. Clustering is pivotal in data mining applications such as scientific data exploration, information retrieval and text mining, spatial database applications, computational biology, and many others where labelling a large set of sample patterns is not possible.

2.4.2 Fuzzy C-Means Clustering

Fuzzy c-means (FCM) has been used in this study as the preferred unsupervised classifier. Fuzzy c-means is a method of clustering which allows one data point to belong to two or more clusters. This method, which was developed by Bezdek in 1981 [1], is frequently used in pattern recognition. It is based on finding the fuzzy membership μ_{ij} and cluster centroid c_i, to minimize the following objective function:

$$J_m = \sum_{i=1}^{C}\sum_{j=1}^{N}(\mu_{ij})^m \left\|x_j - c_i\right\|^2 \;, \quad 1 \leq m \leq \infty, \tag{2.2}$$

where m is the fuzziness index, μ_{ij} is the degree of membership of observation x_j in the cluster i, $x_j(j = 1 \ldots N)$ is the j^{th} d-dimensional data point, c_i is the d-dimensional centre of the cluster, and $\|.\|$ is any norm measuring the similarity between an empirical data point and a cluster centre [1]. Pal and Bezdek [29] showed the best choice for m is probably in the interval [1.5, 2.5], whose mean and midpoint, $m = 2$, have often been the preferred choice for many users of fuzzy c-means. Based on preliminary investigations by the author of this thesis, the chosen value for m was 1.8 in this study.

Fuzzy partitioning is carried out through an iterative optimization of the objective function (2.2), with the update of membership μ_{ij} and the cluster centres c_i by: [1]

$$\mu_{ij} = \frac{\left[\frac{1}{\|x_j - c_i\|^2}\right]^{\frac{1}{m-1}}}{\sum_{t=1}^{C} \left[\frac{1}{\|x_j - c_t\|^2}\right]^{\frac{1}{m-1}}} \tag{2.3}$$

$$c_i = \frac{\sum_{j=1}^{N} (\mu_{ij})^m x_j}{\sum_{j=1}^{N} (\mu_{ij})^m} \tag{2.4}$$

Iterative optimization will stop when

$$\left\|\mu_{ij}^{(k+1)} - \mu_{ij}^{k}\right\| < \epsilon, \tag{2.5}$$

where $0 < \epsilon < 1$ is a termination criterion, and k is the iteration step. This procedure always converges to strict local minimum of J_m [1] starting from an initial guess of μ_{ij}, but different choices of initial μ_{ij} might lead to different local minima.

Referring to Equation (2.2), it can be seen that in order to minimize the function J_m, the number of clusters (C) should be specified. Therefore, fuzzy c-means clustering is considered a *partially supervised* classification algorithm.

Specifying the number of clusters can be a subjective exercise. Therefore, several methods have been developed to automatically validate the number of clusters.

2.4.3 Cluster Validity Criteria

Cluster validity refers to the process of validating the number of specified clusters. Validation reveals how well the data points are clustered. In general, indices of cluster validity fall into one of three categories:

- Some indices measure partition validity by evaluating the properties of the crisp structure imposed on the data by the clustering algorithm, such as Dunn's validity index [30], and Davies-Bouldin validity index [31]

- In the case of fuzzy clustering algorithms, some validity indices such as partition coefficient [1] and classification entropy [1] use only the information of fuzzy membership grades to evaluate clustering results.

- The third category consists of validity indices that not only make use of fuzzy membership grades but also the structure of the data. The partition index [32], Xie-Beni index [33], and Silhouette validation method [34] are examples. The Silhouette validation method has been widely used with k-means and fuzzy c-means algorithms [35, 36, 37, 38]. The main advantage of using the silhouette index among other cluster validity indices is that regardless of the data dimension, cluster definitions can be visually illustrated in a 2-dimensional plot.

Since only fuzzy partitioning is used in this research, cluster validity indices for crisp structures are not considered in this study. It should also be noted that none of the available cluster validity indices are perfect by itself for all applications. Therefore, in this study, several prominent cluster validity criteria are studied. This would help to choose the best cluster validity index for this application.

Below we briefly explain the aforementioned cluster validity criteria used in this thesis.

Partition Coefficient (PC)

Bezdek [1] designed the partition coefficient to measure the amount of *overlap* between clusters. He defined the partition coefficient as follows:

$$PC(C) = \frac{1}{N}\sum_{i=1}^{C}\sum_{j=1}^{N}(\mu_{ij})^2, \tag{2.6}$$

where μ_{ij} $(j = 1, 2, \dots N)$ and $(i = 1, 2, \dots C)$ is the membership of data point j in cluster i. The disadvantage of PC is lack of direct connection to some property of the data themselves.

The optimal number of clusters is the value of C which maximizes Equation (2.6).

Classification Entropy (CE)

CE measures the fuzziness of the cluster partition only, proposed by Bezdek [1], and it is similar to the partition coefficient.

$$CE(C) = -\frac{1}{N}\sum_{i=1}^{C}\sum_{j=1}^{N}(\mu_{ij})log(\mu_{ij}) \tag{2.7}$$

Partition Index (SC)

Partition index (SC) is the ratio of the sum of compactness and separation of the clusters. It is a sum of individual cluster validity measures normalized through division by the fuzzy cardinality of each cluster [32].

$$SC(C) = \sum_{i=1}^{C} \frac{\sum_{j=1}^{N}(\mu_{ij})^m \left\|x_j - c_i\right\|^2}{n_i \sum_{k=1}^{C} \left\|c_i - c_k\right\|^2}, \tag{2.8}$$

where

$$n_i = \sum_{j=1}^{N} \mu_{ij} \tag{2.9}$$

is called the *fuzzy cardinality* of cluster i. A lower value of SC indicates a better partition.

Xie-Beni separation index

Xie-Beni validity index [33] is obtained by dividing the objective function described in Equation (2.2) by the number of data points and the square of the minimum distances among the cluster centres. The separation index $S(C)$ is defined as

$$S(C) = \frac{\sum_{i=1}^{C}\sum_{j=1}^{N}(\mu_{ij})^m \left\|x_j - c_i\right\|^2}{N \times min_{i,k} \left\|c_i - c_k\right\|^2}, \tag{2.10}$$

where $min_{i,k} \|c_i - c_k\|$ is the minimum Euclidean distance between cluster centres. The above index can be interpreted as the average compactness over all data points divided by the minimum separation. The farther apart the clusters are, the larger the value of $min_{i,k} \|c_i - c_k\|$, and the smaller $S(C)$. Thus, the smallest $S(C)$ indeed indicates a valid optimal partition.

Silhouette Validation Method

The silhouette value [34] is a measure of the classification quality. The silhouette values show how close the elements of a cluster are together and how far they are from the next closest cluster. Therefore, average silhouette values can be applied for evaluating clustering validity. The algorithm consists of the following steps:

For the i^{th} object, we denote by A the cluster to which it belongs, and compute

$$a(i) = \frac{1}{|A|-1} \sum_{j \in A, j \neq i} d(i,j), \tag{2.11}$$

where $a(i)$ is an average dissimilarity of the i^{th} object to all other objects of A.

Now consider a second cluster C different from A and write

$$d(i,C) = \frac{1}{|C|} \sum_{j \in C} d(i,j), \tag{2.12}$$

where $d(i,C)$ is an average dissimilarity of the i^{th} object to all objects of C.

After computing $d(i,C)$ for all C we take the smallest $d(i,C)$:

$$b(i) = \min_{C \neq A} d(i,C) \tag{2.13}$$

The cluster B which attains this minimum [that is, $d(i,B) = b(i)$] is called the *neighbour of object i*. This is the second-best cluster for object i.

The silhouette value $s(i)$ of the object i is defined as:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{2.14}$$

Clearly $s(i)$ always lies between -1 and 1. The value $s(i)$ may be interpreted as follows:

- $s(i) \approx 1 \Rightarrow$ object i is well classified in A.
- $s(i) \approx 0 \Rightarrow$ object i lies intermediate between two clusters (B and A)
- $s(i) \approx -1 \Rightarrow$ object i is badly classified (closer to B than to A).

The average of all $s(i)$ is called the *average silhouette value*, and is used to justify the total number of clusters.

2.5 Chapter Summary

In order to understand the origin of the EMG signal, the reader should have some basic background on muscle physiology which was given at the beginning of this chapter. EMG properties and a popular EMG recording technique were reviewed in this chapter. This was followed by a review of past studies on EMG signal classification and relevant pattern recognition techniques. In this research, unlike most previous studies, an unsupervised learning system is used. The fuzzy c-means algorithm was introduced as the preferred clustering technique in this research. The main advantage of using a fuzzy clustering versus crisp clustering (i.e. k-means) is that there is not a hard boundary between the clusters. Instead, each data point (feature) has a membership degree in each of the clusters. With the fuzzy c-means algorithm, the number of clusters should be specified a priori. Unfortunately, in most applications, the number of clusters is not known a priori. Several cluster validity indices were reviewed. The next chapter will introduce the method used in this study which is based on the fundamental material reviewed in this chapter.

Chapter 3

Experimental method

3.1 Introduction

The first section of this chapter outlines the equipment set-up and the signal recording method used in this study. The next section introduces the feature extraction and real-time classification technique, as well as the experimental protocol.

3.2 Equipment set-up and data recording

3.2.1 The ethics

It was required both by Ryerson University and Bloorview MacMillan Children's Centre that all graduate student research involving human subjects be reviewed by the relevant research ethics board before work began. A project proposal for this research was prepared and successfully passed both the scientific review and the ethical review by the responsible research ethics board at Bloorview MacMillan Children's Centre.

3.2.2 Participants

The number of participants in previous studies varied from 1 to 12 persons. Most of the studies were done on able-bodied people [10, 19, 14, 26, 15, 21, 12]. There were only a few amputees (mostly only one amputee subject) who participated in previous studies. In this study, we recruited a convenience sample of 8 able-bodied adults and 1 below elbow amputee.

Due to the lack of standardized measures in EMG classification studies, a rigorous power analysis could not be conducted for sample size estimation, at the time of this writing.

Able-bodied subjects consisted of a convenience sample of adults aged 20-30 years with no upper limb deficiencies. An amputee was recruited from Bloorview MacMillan Children's Centre. It was particularly desirable to choose an amputee who was not a powered prosthesis user since he or she would most likely be unfamiliar with forearm muscle control. This lack of prior training would present a nontrivial challenge to the automatic separation of EMG signals.

3.2.3 The electrodes and skin preparation

Pre-gelled Ag/AgCl disposable wet-type electrodes made by Grass (model: F-E8SD) were used in this study. These electrodes are less expensive than active electrodes (e.g. Otto Bock) and can be applied to the subject's skin easily and securely. No skin preparation was performed to minimize the total experiment duration. The whole experiment on each subject was expected to take about 30 minutes per subject. The proposed algorithm was expected to learn from the user and adapt itself to the user's variations. Therefore, performing the experiment on unprepared skin actually provided information about system robustness to low quality signals recorded from unprepared skin.

3.2.4 Recording site

In the normal limb, at least two opposing muscles are required to control a joint [4]. The interaction of agonist[1] and antagonist[2] and often synergist muscles determines the action of the joint. Two site control is a first step towards utilizing these natural interactions [39]. Therefore, two muscle sites were chosen in this study.

It is believed that the bulk of the muscle produces higher amounts of EMG signals. Thus, it was desirable to put the electrodes on the bulk of the muscles [7]. The electrodes were placed in pairs with one reference electrode at each site, over the group of agonist and

[1] A muscle responsible for producing a specific movement. This muscle contracts to move the limb.

[2] A muscle that relaxes while the agonist contracts.

antagonist muscles on the forearm, as illustrated in Figure 3.1. The location of the attached electrodes on the forearm was photographed for future reference.

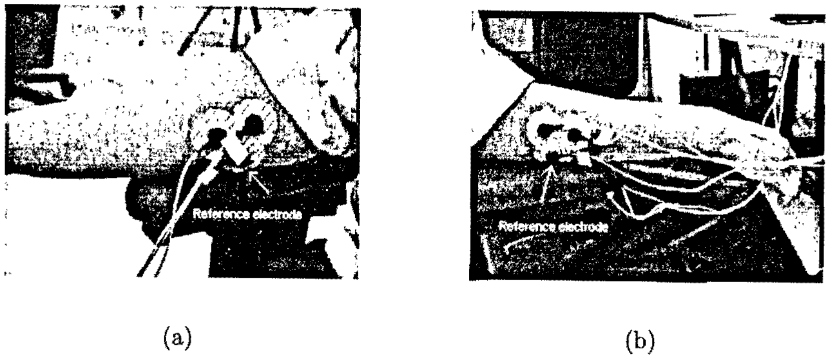


Figure 3.1: The Electrode locations: The electrodes were placed in pairs with one reference electrode (the marked electrode). Figure 3.1(a) shows the location of the electrodes placed on forearm flexor muscles. Figure 3.1(b) shows the location of the electrodes placed on forearm extensor muscles.

3.2.5 Data acquisition

The raw EMG signals from two muscle sites on the forearm, were amplified by a GRASS TELEFACTOR 15A54 amplifier. This amplifier conforms to Underwriters Laboratories (UL) 2601-1 safety standard, and certified to Canadian Standard Associations (CSA) C22.2 #601-1 safety standard . This amplifier optically isolated the subject from the amplifier so that it was safe for direct recording from a human subject. The low cutoff frequency (-6 dB) on this amplifier can be set at: 0.01, 0.1, 0.3, 1, 3, 10, 30 and 100 Hz and high cutoff frequency (-6 dB) can be set at 30,100, 300, 1000 and 6000 Hz. The amplifier also has a line notch-type filter (50/60 Hz), jumper selectable, to remove the power line noise. The amplification was adjustable from 50 to 200,000.

In this study, the gain of the amplifier was set at 5000 and the signals were band-passed inside the amplifier. Although much literature has suggested the usable band for EMG signals is from 20 Hz to 500 Hz [4, 7], due to limitations on the amplifier, the chosen band for this experiment was set as 30 Hz to 1000 Hz. The low frequency cutoff filter was used to avoid motion artifact [7]. In order to eliminate the power line 60 Hz noise, the built-in line notch filter was used in addition to the primary filters. The amplified signals then were fed to a personal computer (P4- 2.8 GHz, 1GB RAM) equipped with a National Instrument, PCI-6014 data acquisition board. The signals were sampled at 2000 samples per seconds to avoid anti-aliasing. Figure 3.2 illustrates the power spectrum density calculated for Subject5 after the training session. This confirms that although the high cutoff filter was set at 1000 Hz, there is not much information beyond 500.

The necessary system program responsible for initializing the data acquisition board and recording the data was written in MATLAB® 7.1. Figure 3.3 shows the equipment setup.

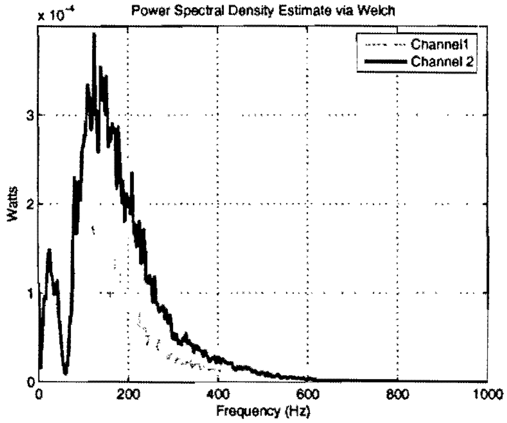


Figure 3.2: Power spectral density (PSD) calculated over the training session for Subject5. The line notch filter eliminated the power line 60 Hz noise. It can be seen that the usable band for the recorded EMG signals in this experiment, is from 30 Hz to 500 Hz which agrees with the findings of previous studies. [4, 6, 7]. The PSD recorded from flexor muscles and extensor muscles on the subject's forearm are designated as channel1 and channel2, respectively.

3.3 The Protocol

The experimental protocol has been summarized in Figure 3.4. The algorithm simply extracted the features from the raw EMG signals recorded at two sites, and clustered the feature space. The membership value for each data point was calculated and used for classification. A testing session at the end, estimated the performance of the algorithm.

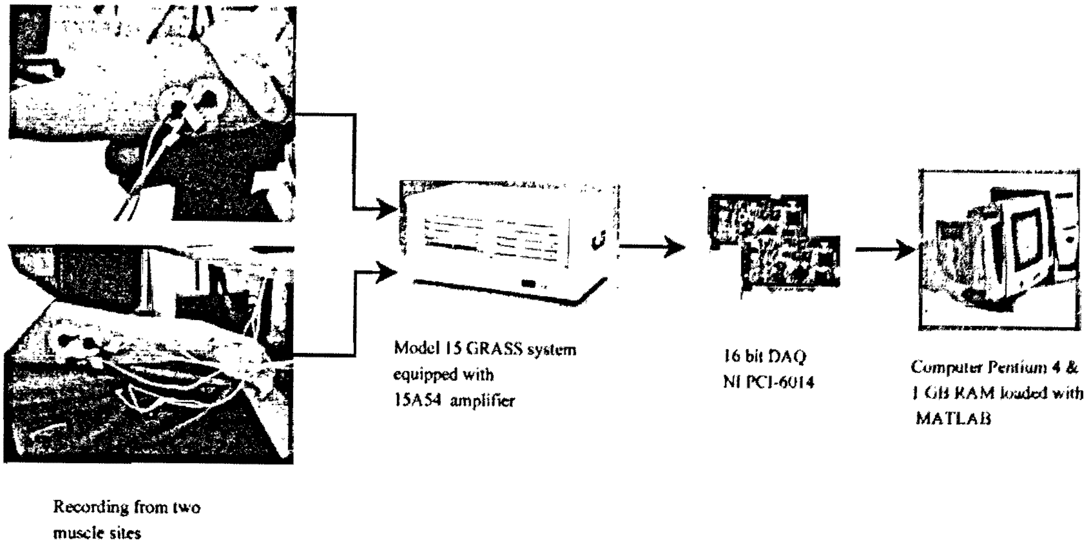


Figure 3.3: The equipment setup

3.3.1 Training

As mentioned previously, the objective of this project is to automatically define different preselected intentional movements. Therefore, the subject was not restricted to any predefined sitting position. However, in order to establish a well-defined and repeatable resting position for each individual, each subject was asked to sit on a chair, to relax and to rest his or her arm in the most comfortable position. The subject was not required to perform any predefined activities. Instead, the participant was asked to perform self-selected forearm movements, that were deemed both perceivable and repeatable, as soon as the computer prompted. The computer started to extract the features every 0.2 seconds in real-time, as illustrated in Figure 3.5. All necessary data were saved in both the computer memory and the hard drive. This session was considered as *training session* and it ended after recording of two minutes worth of data. The subject could rest at any time while the data were being recorded.

3.3.2 Feature Extraction (Step 1)

The traditional goal of the feature extractor is to characterize an object to be recognized by measurements whose values are very similar for objects in the same category, and very different for objects in different categories [40]. Referring to section 2.2.5, it has been widely reported that the amount of force that a muscle generates correlates with the root-mean-square (RMS) value of the EMG signal of the corresponding contracting muscle. Therefore, RMS of the EMG signals could constitute the feature space. On the other hand, many researchers have used autoregressive (AR) coefficients as a possible feature space. In this study, both RMS feature space and AR coefficients were studied.

RMS feature space

The RMS values could be calculated as:

$$RMS(x_1, x_2, \ldots, x_n) = \sqrt{\frac{x_1^2 + x_2^2 + \cdots + x_n^2}{n}}, \tag{3.1}$$

where $x_1, x_2, \ldots, x_n$ are the data points.

In this study, the RMS feature of each channel was calculated to create a 2-dimensional feature vector. It was argued that the response time for the control system should not introduce a delay perceivable by the user. This threshold is generally regarded to be roughly 300 milliseconds [19, 14]. In this study we calculated the RMS features of each channel within a 200 millisecond window. This ensured that the algorithm made the decision in less than 300 milliseconds.

Preliminary studies showed that this feature space was not suitable for clustering, since the data did not appear to naturally segment into groups. For instance, the fuzzy clustering algorithm used in this study [1] was not able to differentiate clusters close to the origin. Figure 3.7(a) illustrates one of the preliminary results based on the calculated RMS features from the extensor and flexor muscles, for each 200 millisecond window. These results were obtained from an able-bodied subject, over a 2-minute training session.

It is shown in Figure 3.7(a) that data points are concentrated around the origin and they are spread far across the axes. In order to spread the concentrated data points away from the origin of the plot and make the data points across the axes more clustered, a special function was needed. This function should be able to amplify data points around the origin more, in order to spread them, and amplify farther data point from the origin less, to concentrate them. One of the functions that met these requirements was the *natural logarithm* function. Figure 3.6 shows the general plot of this function. It can be seen this function amplifies the points around the origin with a high gain and then gradually lessens amplification of the more distant data points, nonlinearly. It should be noted that only positive numbers have natural logarithm values.

RMS values of EMG signals are always positive. Therefore, the natural logarithm of the RMS features were calculated and plotted, to form the feature space as follow:

$$F_{x_j,y_j} = \left(\log\left(RMS(x_{1j}, x_{2j}, \ldots, x_{nj})\right), \log\left(RMS(y_{1j}, y_{2j}, \ldots, y_{nj})\right)\right), \tag{3.2}$$

where F_{x_j,y_j} is the calculated feature for the $j^{th}(j = 1, 2, \ldots, N)$ feature. N is the length of the feature vector. $x_{1j}, x_{2j}, \ldots, x_{nj}$ are sample data points, recorded from flexor muscles (channel1), and $y_{1j}, y_{2j}, \ldots, y_{nj}$ are sample data points, recorded from extensor muscles (channel2). n is the length of the window over which the features were calculated. In this study, as mentioned earlier in this section, the length of the window should be 200 milliseconds. Since the sampling frequency was set at 2000 samples per seconds, the length of the sequence would be: $n = 400$

Figure 3.7(b) illustrates the effect of the natural logarithm function on the feature space.

Autoregressive (AR) Coefficients feature space

The AR model of the current sample of the signal $x(j)$ is described as a linear combination of previous samples plus an error term $e(j)$ which is independent of past samples:

$$x(j) = -\sum_{k=1}^{p} a_k x(j-k) + e(j), \qquad j = 0, 1, \ldots, n-1, \tag{3.3}$$

where $x(j)$ are the samples of the modelled signal, a_k are the AR coefficients, p is the model's order, $e(j)$ is the time series of residual values (prediction errors), and n is the length of the window over which the features were calculated. The model can be interpreted as a linear system with $e(j)$ as its input (white noise) and $x(j)$ as its output. The transfer function $H(z)$ for the AR process is:

$$H(z) = \frac{X(z)}{E(z)} = \frac{1}{1 + \sum_{k=1}^{p} a_k z^{-k}} \tag{3.4}$$

The spectrum of the model can be estimated from Equation (3.4) as:

$$P_{AR} = \frac{1}{\left| 1 + \sum_{k=1}^{p} a_k e^{-jwk} \right|^2} \tag{3.5}$$

It has been shown that the EMG spectrum changes with muscle contraction state, resulting in a change in AR coefficients [41]. To define the order of the model, various experimental and theoretical approaches have been proposed [41]. The results of both approaches show that a model order of 4 is adequate for AR time series modelling of EMG signals [12, 17, 24, 41, 42].

There are many ways to compute AR coefficients. The Burg algorithm [43] is probably the most widely known AR procedure. In this study, MATLAB® built-in function 'arburg' was used to calculate the 4^{th} order AR coefficients on each recording site, to form an 8-dimensional feature space. Unfortunately, the higher the dimension of the feature space, the more the number of samples required for AR coefficient estimation. This is known as '*the curse of dimensionality*' and was proposed by Bellman in 1961 to describe the exponential growth in combinatorial optimization as the dimension increases [44].

Many researchers have used different methods such as *Principal component analysis (PCA)* or *Genetic algorithm (GA)* to project a higher dimensional feature space onto a lower dimension space [14, 15, 23, 22].

In this study an alternative and simpler method was investigated to reduce the dimensionality of the feature space. We define vector A_j, created by 4^{th} order coefficients of the flexor muscle group (channel1 recording site), and B_j, created by 4^{th} order coefficients of the extensor muscle group (channel2 recording site), both calculated for the j^{th} 0.2-second window (Figure 3.5),

$$A_j = \{a_{1j}, a_{2j}, a_{3j}, a_{4j}\} \tag{3.6}$$

$$B_j = \{b_{1j}, b_{2j}, b_{3j}, b_{4j}\} \tag{3.7}$$

In order to investigate how AR coefficients obtained from each channel vary over the training session, a feature vector, F_{A_j,B_j} was created and plotted:

$$F_{A_j,B_j} = \{(a_{1j}, b_{1j}), (a_{2j}, b_{2j}), (a_{3j}, b_{3j}), (a_{4j}, b_{4j})\} \tag{3.8}$$

Figure 3.8(a) illustrates the created feature space based on Equation (3.8). Each region shows the bivariation of corresponding AR coefficients, calculated from 2 channels $((a_{ij}, b_{ij}), i = 1, 2, 3, 4, j = 1, 2, \ldots N)$. The 'Resting position', which is shown with black symbols on the plot, occurred when the forearm was in a neutral position and no movement was performed. The cluster arising from the first two seconds of the resting position (from a preliminary trial) is shown by the black symbols in Figure 3.8(a). In the real-time preliminary trials we observed that the four detected clusters based on AR coefficients of the flexor and extensor channels, vary from one movement to another, away from the resting cluster. As a result, it was expected that the *standard deviation* among the AR coefficients for each channel, could yield a meaningful feature space.

The standard unbiased deviation estimate, S, of a data vector x is given by,

$$S = \left(\frac{1}{N-1} \sum_{j=1}^{N} (x_j - \bar{x})^2 \right)^{\frac{1}{2}} , \tag{3.9}$$

where

$$\bar{x} = \frac{1}{N} \sum_{j=1}^{N} x_j \tag{3.10}$$

and N is the number of elements in the sample. In this study, MATLAB® built-in function $std(x)$, using Equation (3.9) was used to create a new 2-dimensional feature space as follows,

$$F_{Stdev(A_j),Stdev(B_j)} = (Stdev(a_{1j}, a_{2j}, a_{3j}, a_{4j}), Stdev(b_{1j}, b_{2j}, b_{3j}, b_{4j})) , \tag{3.11}$$

where *Stdev* is the standard deviation using Equation (3.9).

Definition

From now on, we refer to the feature space created using Equation (3.11) as the *Stdev*(AR) feature space and the feature space created using Equation (3.2) as the *Log*(RMS) feature space. Figure 3.8(b) shows a 2-dimensional feature space created and plotted based on Equation (3.11).

The raw data as well as the aforementioned *Log*(RMS) and *Stdev*(AR) features of each channel were recorded and calculated simultaneously in real-time and saved in MATLAB® format, in the hard drive, for further analysis.

Nonstationary conditions and AR features

AR modelling requires that the fitted signal be stationary over the given interval. The EMG signal, due to its random nature, is considered to be nonstationary. However, it has been shown that an AR model can be applied to a nonstationary process provided that the process can be segmented as locally stationary [45].

Signal stationarity over short epochs can be tested with different approaches , but are rarely applied in the case of EMG signals [46].

Reverse arrangement test

In this study, the reverse arrangement test was used to validate the stationarity of the recorded EMG signal within 0.2-second windows.

The reverse arrangement test can detect a monotonic trend in a time series $x(j = 1, 2, \ldots, n)$. The method is based on the computation of the number of times that $x(j) > x(k)$ with $j < k$ (this inequality is called a 'reverse arrangement') for all j. If the sequence of $x(j)$ are independent observations of the same random variable, then the number of reverse arrangement is a random variable with

$$Mean = \frac{n(n-1)}{4} \tag{3.12}$$

and

$$Variance = \frac{n(2n+5)(n-1)}{72} \tag{3.13}$$

An observed number of reverse arrangements significantly different from the mean (Equation (3.12)) indicates nonstationarity because of the possible presence of a trend in $x(j)$. The results of this test showed that, on average, more than 85% of all recorded signals within a 0.2-second window, passed the stationary criterion at a significance level of $\alpha = 0.05$.

The effect of windowing on EMG signal

It has been shown [47, 46] that the particular window applied to estimates of spectral parameters of EMG signals has a negligible effect. The rectangular window (used in this study - Section 3.3.1) is acceptable and even preferable in EMG spectral estimation [47].

3.3.3 Clustering (Step 2)

At the end of the training session, fuzzy c-means clustering algorithm [1] was applied to the proposed feature spaces, to find the different clusters of activities associated to different movements. Each subject estimated the number of different performed movements. This number was taken as the number of clusters and applied to the fuzzy c-means clustering algorithm. Unlike most applications, in this study, since the user defines the number of self-selected movements, the cluster validity is not expected to find the optimum number of clusters. In this study cluster validity was used to validate the following.

- The performance of the fuzzy c-means algorithm. Cluster validity ensured that the clustering algorithm converged properly to its minimum, so that the data points were assigned correctly to the cluster to which they belong.
- The optimum number of clusters. This number should be found automatically. Different cluster validity algorithms were being investigated in this study, as described in Section 2.4.3. Since the number of performed movements were strictly defined by the user, the performance of these cluster validity algorithms for finding the optimum number of clusters can be investigated.

The generated clusters along with the centres of the clusters was shown to the user on the computer screen. Figure 3.10(a) shows an example of a clustered $Log(RMS)$ feature space using fuzzy c-means algorithm. Figure 3.10(b) shows an example of fuzzy c-means clustering of $Stdev(AR)$ feature space. The centre of the clusters have been magnified in black.

It was assumed that each movement should have its own unique associated cluster in both $Log(RMS)$ and $Stdev(AR)$ feature spaces. Therefore, at the end of the training session (Section 3.3.1), both $Log(RMS)$ and $Stdev(AR)$ feature spaces were clustered using the fuzzy c-means algorithm and the results were saved in a file for further analysis.

3.3.4 Cluster validity (Step 3)

Preliminary studies showed that the number of movements expressed by the user, was best estimated by the silhouette [34] validation method. For instance, the subject who created the feature space plotted in Figure 3.9(a) expressed the number of self-selected movements as 6. In this example, the silhouette validation method was the only cluster validity index that estimated the number of clusters correctly. Figure 3.9 shows an example of the preliminary results of the fuzzy c-means clustering of a feature space based on the natural logarithm of

RMS values, and the performance of the silhouette validity index. The number of clusters varies from 2 to 10 and the maximum mean of the silhouette value occurs when the number of clusters is set to 6.

Other cluster validity indices such as, Partition coefficient, Classification entropy, Partition index, and Xie-Beni separation index were computed as well. These results were saved in the computer hard drive for subsequent analysis.

3.3.5 Membership values for classification (Step 4)

For classifying new data after the training session, the fuzzy membership values based on Equation (2.3) were calculated for each new point and in turn, each point was assigned to the cluster to which it had the highest membership.

Let F_x^j, F_y^j represent the calculated features for the j^{th} sampling window. As mentioned previously, these two features were formed to create a 2-dimensional feature $F_{(x,y)}^j$. Since the fuzzy clusters have already been created in the feature space, (Section 3.3.3), the maximum membership value for each new data point, $\mu_{C_i}^j$, assigns this point to the corresponding cluster, C_i. This process repeats every 0.2 seconds. Therefore, for every 0.2 second, we have a classified feature in real-time.

The visual feedback

In order to create a more user friendly environment, the software program displayed a window consisting of separated vertical blue bars. Each bar was associated with a specific cluster. The bars were labelled arbitrarily as *Movement_1* to *Movement_n*, where n was the n^{th} cluster. Referring to Figure 2.6, since the amount of force produced by a muscle is a nonlinear function of the RMS value of the EMG signal, it was preferable to take the $Log(RMS)$ feature space as the primary feature space, and then compare the results with the $Stdev(AR)$ feature space. By visually comparing Figures 3.7(b) and 3.8(b), it appears that for the same empirical data set, the $Log(RMS)$ feature space is better clustered than $Stdev(AR)$ feature space[3]. Therefore, in this study, the visual feedback was associated to the $Log(RMS)$ feature space. Figure 3.10 shows an example of the aforementioned clustered feature spaces and the user interface for 6 movements associated with 6 different clusters (forearm movements).

For each new data point, the software program highlighted the corresponding vertical bar in red, as soon as the maximum membership value of the new point, $\mu^j_{C_i}$, was discovered. Therefore, $\mu^j_{C_i}$ served as a highlight in the visual feedback to the user, to indicate the movement selected by the user.

Figure 3.10(d) shows an example of a user who is using the visual feedback as a response to his forearm movements. In this example, the user is doing wrist flexion and the computer responds to the user by highlighting the first bar from the left on the computer screen.

3.3.6 Labelling the clusters (Step 5)

The user was given a chance to highlight the bars by performing the same forearm movements, performed during training, for another two minutes. This helped the subject find the relationship among the movements and the highlighted bars.

At the end of this session, if the user felt that the system responded to the movements accordingly, the software would ask the subject to label the movements. Otherwise the system would ask the subject to retrain the system. The system may not respond to the user due to the following two main reasons:

1. The subject did not train the system properly and did not provide repeatable natural movements during the training session.

2. The fuzzy c-means algorithm converged to a local minimum such that the data points were not properly assigned to the clusters.[4]

When the subjects felt that the system followed their movements properly, they were asked to label the vertical bars, which corresponded to their forearm movements. They

[3] It should be noted that visual inspection of a data set does not guaranty the cluster separability. It is also unknown if $Log(RMS)$ features make a well separated feature space for other subjects or other trials.

[4] This happened only once. In this case, the user did not re-train the system. Instead, fuzzy c-means algorithm was re-ran , at which point the algorithm converged properly.

could label the movements in any language or with any words which would help them to remember the movement. As will be discussed in section 3.3.8, the computer cued the user to create movements in random sequence. Therefore, labelling the movements was necessary to estimate the performance of the system.

For each movement, the user was given five seconds to choose any of the unlabelled bars (movements) to label. As mentioned in Section 3.3.5, every 0.2 seconds a new data point with the maximum membership value $\mu^j_{C_i}$ was assigned to the i^{th} cluster. This process was hidden from the user. The user was provided with visual feedback in form of vertical blue bars, highlighted in response to the calculated maximum membership value $\mu^j_{C_i}$. At the end of the 5-second window allowed for labelling one of the movements, the last membership value highlighted the corresponding vertical bar. It was assumed that at the end of the 5-second window, the user would have selected the desired movement (bar) to label. The user was given a chance to verify his or her choice and could repeat the selection process as desired. Let $\mu^j_{C_k}$ be the last calculated maximum membership value. Therefore, the label that the user chose, was assigned to the k^{th} cluster, C_k. This process was done simultaneously on both $Log(RMS)$ and $Stdev(AR)$ feature spaces. This means that if $\mu^j_{C_k}$ is the last calculated maximum membership value in $Log(RMS)$ feature space, there is a similar maximum membership value, i.e. $\hat{\mu}^j_{C_k}$, in the $Stdev(AR)$ feature space. It should be noted that cluster C_k in the $Log(RMS)$ feature space is not necessarily cluster C_k in $Stdev(AR)$ feature space. $\hat{\mu}^j_{C_k}$ may point to cluster C_m, where $m \neq k$.

In order to compare the performance of $Log(RMS)$ and $Stdev(AR)$ formulations, it was necessary to find the one-to-one relationship between the clusters in these two feature spaces. Therefore, when cluster C_k was labelled by the user in the $Log(RMS)$ feature space, the corresponding cluster in the $Stdev(AR)$ feature space (i.e. C_m) was *virtually* assigned the same label. The one-to-one relationship between the clusters in the mentioned feature spaces, were calculated and saved in a file along with other information obtained in this session. Figure 3.11 illustrates the labelling process.

An example of six different movements and their corresponding labelled vertical bars are illustrated in Figure 3.12. In this example, the subject labelled the movements in Spanish.

3.3.7 Familiarizing with new labels (Step 6)

When the discovered movements were labelled, the subjects were given another minute to become accustomed to the labelled movements. They were asked to choose any of the movements randomly and to try to highlight the corresponding vertical bars by performing the same forearm activities with which they trained the system.

3.3.8 Testing (Step 7)

For error estimation, the user was required to perform one of the labelled movements randomly. Therefore, a random sequence representing the movement indices had to be generated.

A uniform random sequence of length of N, consisting of indices representing movements, was generated. The length of the sequence was 10 times as long as the number of generated movements. This would assure that each movement was randomly requested exactly 10 times during the testing procedure. For creating this sequence of length of N, a smaller sequence of indices from 1 to n was generated. Each index of the sequence represented a movement, where n was the maximum number of discovered movements. This sequence was *permuted* to rearrange the indices randomly. This can be mathematically expressed as follow:

Let $\{a_i\}$ be a unique set of positive integer indices:

$$a_i \in Z \quad a_i \neq a_j, \ \forall i,j \ \ , i = 1 \ldots n, \ j = 1 \ldots n, \ i \neq j \tag{3.14}$$

where Z is the set of positive integers.

We define $P(\{a_i\}) \in Z^{1 \times n}$ as permutation of a_i

We define B as the uniformly distributed sequence of random indices

$$B = \{P_1(\{a_i\}) \ldots P_k(\{a_i\})\}, \ \ B \in Z^{1 \times N}, \ \ N = kn \tag{3.15}$$

where k is the repetitions of each movement, n is the number of detected movements, and $P_j(\{a_i\})$ is the j^{th} permutation of $\{a_i\}$, $i = 1 \ldots n$ and $j = 1 \ldots k$

It should be noted that larger k results in better error estimation, but due to the limitation of the duration of the study, $k = 10$ was chosen. This ensured that each index (movement) in the longer testing sequence was repeated 10 times.

Based on the uniformly distributed sequence B, which represented the *target movements*, the computer randomly selected one of the movements and highlighted the corresponding vertical bar in yellow. This highlight served as a prompt for the user to perform the requested movement. As mentioned in Section 3.3.5, the visual feedback was associated with the *Log*(RMS) feature space. It is clear that the generated random sequence B, was the sequence of random indices associated with the clusters in the *Log*(RMS) feature space. Clearly, the cluster indices in the *Stdev*(AR) feature space are not the same as the cluster indices in the *Log*(RMS) features. Therefore, it was needed to find the *target sequence* for *Stdev*(AR) feature space. Based on the calculated one-to-one relationship among clusters in these two feature spaces, which was described in Section 3.3.6, the corresponding target sequence for the *Stdev*(AR) feature space was created. This sequence, let's say $\hat{B}$, held the target movements in *Stdev*(AR) feature space and helped the computer to monitor the user's performance based on *Stdev*(AR) feature space in the background.

We introduced time T_d as the *dwell time*. T_d was set at 3 seconds. Therefore, the user's response time (Ts) was valid as long as $T_s \leq T_d$.

Preliminary results showed that if T_d were set at two seconds the users would not have enough time to respond to the computer and generate the requested movement. A three second time window for T_d was estimated from preliminary tests on the author and it was found that most users could comfortably respond to the computer within a three-second window.

3.4 Summary

In this chapter the experimental method and the setup protocol were discussed. Two different 2-dimensional feature spaces were introduced as *Log*(RMS) and *Stdev*(AR). The fuzzy c-means algorithm was introduced as the clustering algorithm of choice. Different cluster validity indices were also applied to the detected fuzzy clusters to estimate the optimum number of clusters. The membership values were used to assign each data point to the appropriate cluster. The classification was done in real-time by calculating the membership values of the new data points. Chapter 4 presents the results obtained based on the proposed method.

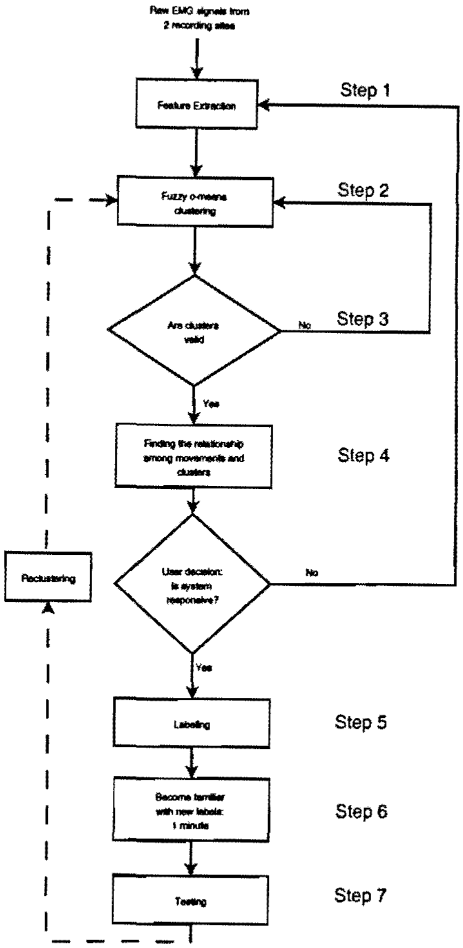

Raw EMG signals from 2 recording sites
Feature Extraction
Step 1
Fuzzy c-means clustering
Step 2
Are clusters valid
No
Step 3
Yes
Finding the relationship among movements and clusters
Step 4
Reclustering
User decision: is system responsive?
No
Yes
Labeling
Step 5
Become familiar with new labels: 1 minute
Step 6
Testing
Step 7

Figure 3.4: The experimental protocol. Steps 1, 4, 5, 6 and 7, required real-time feature extraction. This technique was described in Section 3.3.1. The reclustering method is considered for future development.

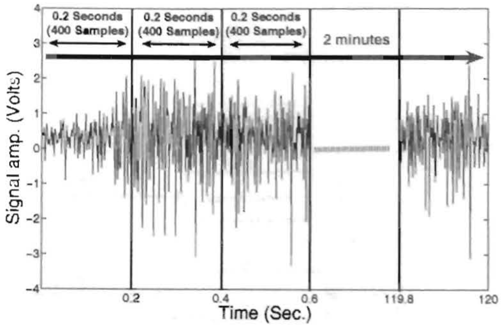


Figure 3.5: Real-time feature extraction of raw EMG for training session. The computer extracts the natural logarithm of RMS values, recorded from each recording site, for every 0.2 seconds in real-time. The training session ended after 2 minutes. This real-time feature extraction technique was used in this study as needed. For example, it was used in Steps 1, 4, 5, 6 and 7 in Figure 3.4.

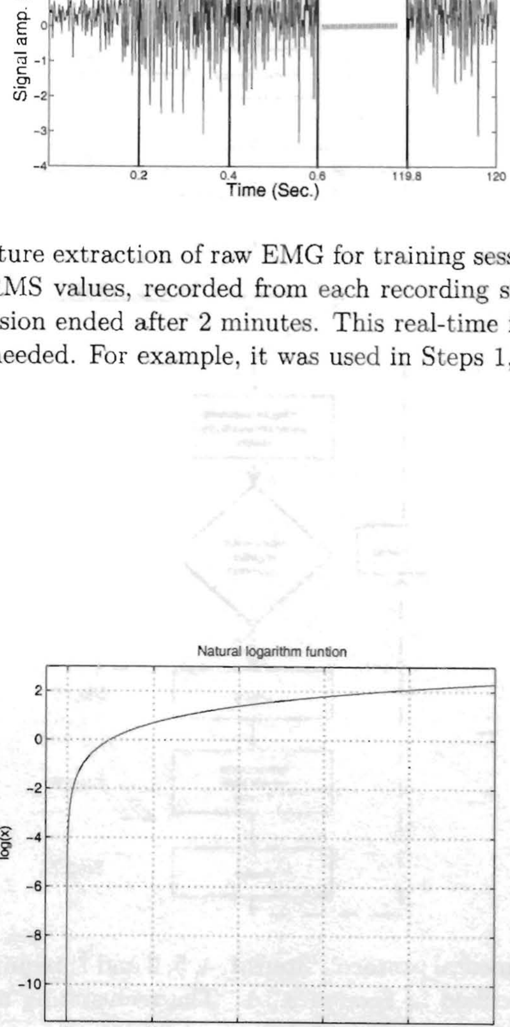
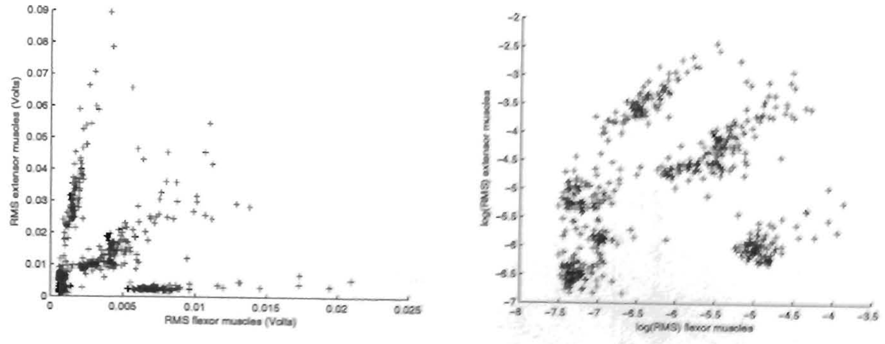
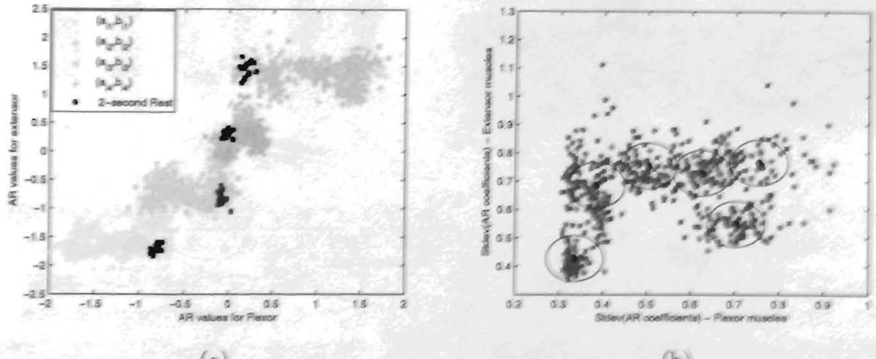


Figure 3.6: The natural logarithm function.



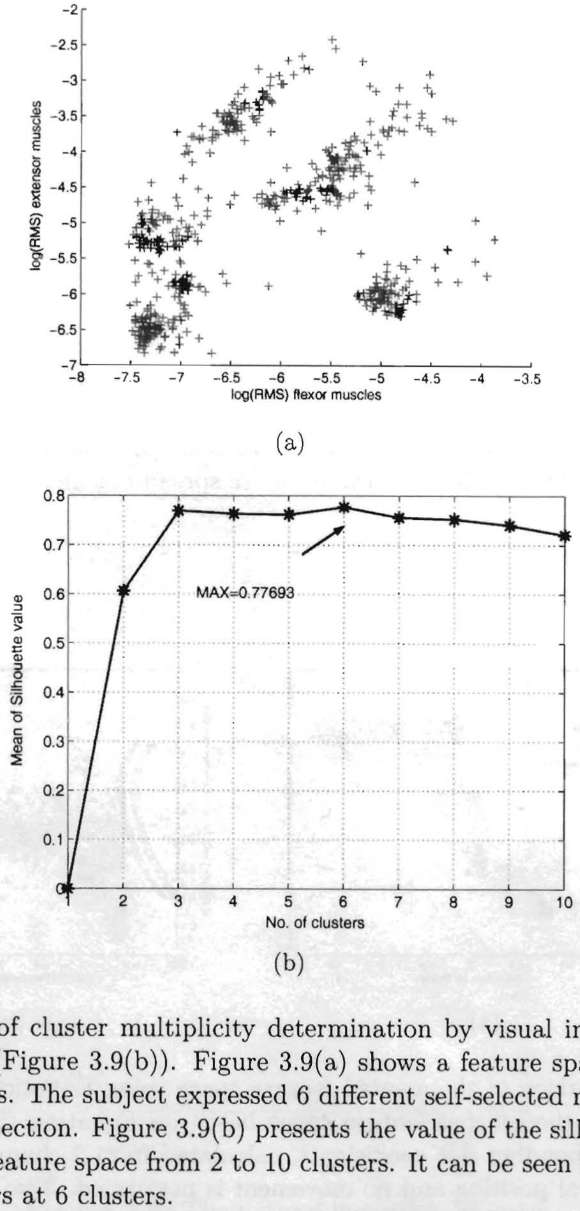
(a) (b)

Figure 3.7: The calculated features from extensor muscles vs. the calculated features from flexor muscles. Each point is calculated over a 200 millisecond window. Figure. 3.7(a) illustrates the calculated features based on RMS values. Figure. 3.7(b) illustrates the calculated features based on natural logarithm of the RMS values. It can be seen that the clusters in the feature space based on the natural logarithm of RMS values, are more spread out and well clustered.



(a) (b)

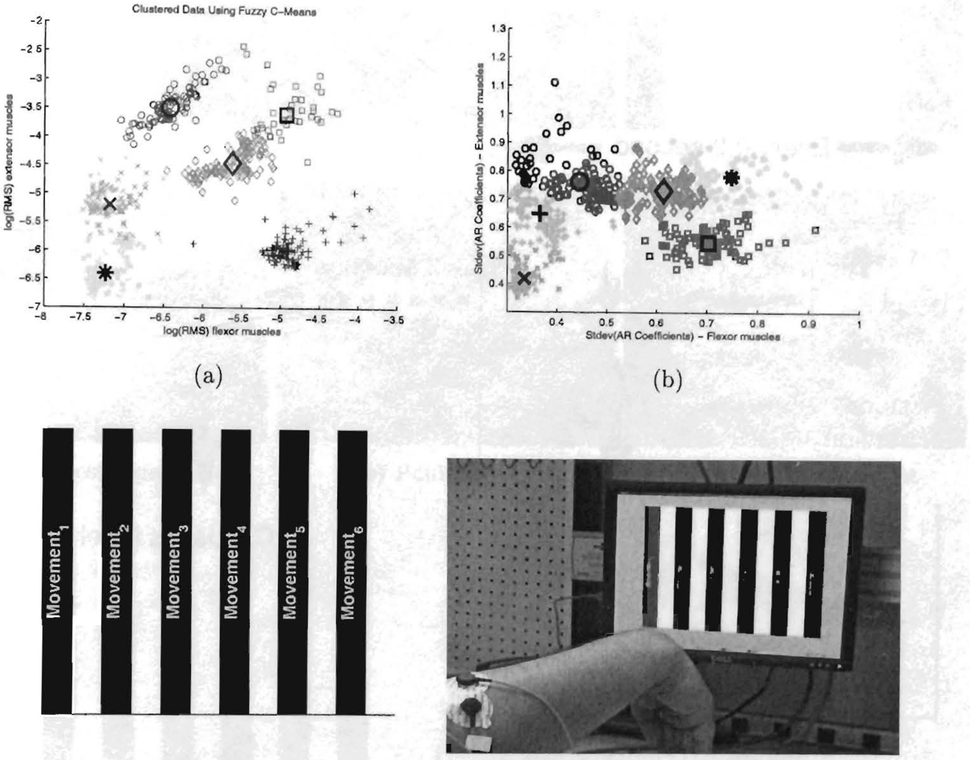
Figure 3.8: Comparison of the created feature space using Equation (3.8) and Equation (3.11). Figure 3.8(a) shows the created feature space based on Equation (3.8). Each region shows the bivariation of corresponding AR coefficients, calculated from 2 channels. The 'Rest' is when the forearm is in a neutral position and no movement is performed. Two seconds of feature values for 'Rest' has been shown in black symbols. Figure 3.8(b) shows the created feature space based on Equation (3.11) (dimension reduction). By visual inspection of this feature space, one may estimate 6 possible clusters. These clusters are shown by circular regions in this plot.



(a)

(b)

Figure 3.9: Example of cluster multiplicity determination by visual inspection (Figure 3.9(a)) and silhouette method (Figure 3.9(b)). Figure 3.9(a) shows a feature space based on the natural logarithm of RMS values. The subject expressed 6 different self-selected movements, which can be confirmed by visual inspection. Figure 3.9(b) presents the value of the silhouette validity index for fuzzy clustering of the feature space from 2 to 10 clusters. It can be seen that the maximum mean of silhouette value occurs at 6 clusters.



(a)

(b)

(c)

(d)

Figure 3.10: Clustered feature spaces using fuzzy c-means algorithm. Figure 3.10(a) shows an example of the clustered $Log(RMS)$ feature space using fuzzy c-means along with the centre of the clusters magnified in black. Figure 3.10(b) shows an example of fuzzy c-means clustering of $Stdev(AR)$ feature space, using Equation (3.11), along with the centre of clusters. Figure 3.10(c) shows an example of the mapping of the clusters on vertical bars for real-time visual feedback during the experiment. Each bar is associated with a forearm movement. In our experiment, the labels were arbitrary and set by the user when the relationship among the forearm movements and the bars was discovered by the user. Figure 3.10(d) shows the user who is using the visual feedback as a response to his forearm movements. In this example, the user is doing wrist flexion and the computer responses to the user by highlighting the first bar from the left on the computer screen.

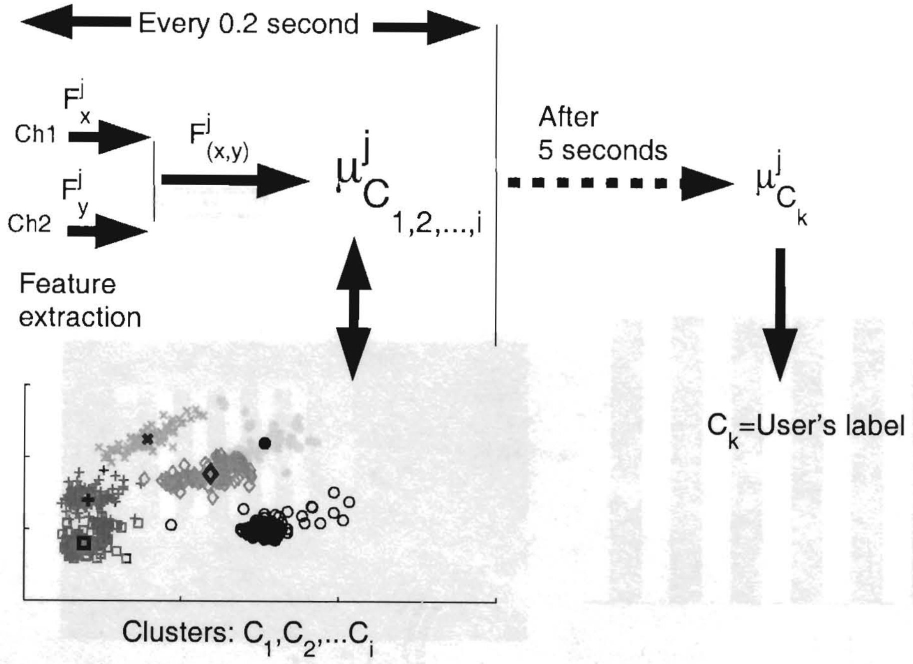
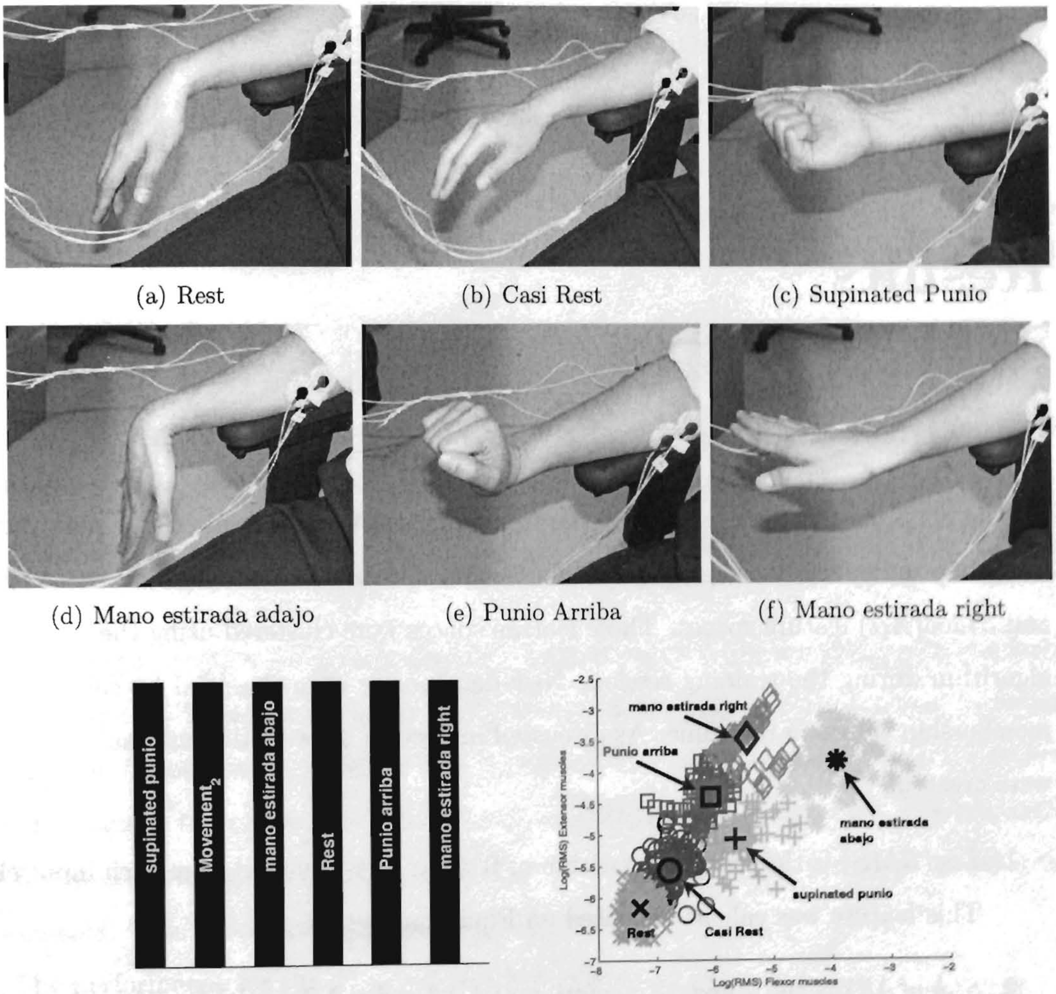


Figure 3.11: Labelling the clusters. The EMG signals are recorded from two Channels (Ch1 and Ch2). The features from each channel (F^j_x, F^j_y) are calculated and combined to form the feature vector, $F^j_{(x,y)}$, which yields a 2-dimensional feature space. $\mu^j_{C_i}$ is the maximum membership value of the current calculated feature which assigns it to the i^{th} cluster. This process is repeated every 0.2 seconds. $\mu^j_{C_k}$ is the last maximum membership value, calculated at the end of the 5-second window for labelling one of the movements. Therefore, the label that the user chooses is assigned to the k^{th} cluster, which is C_k.



(a) Rest

(b) Casi Rest

(c) Supinated Punio

(d) Mano estirada adajo

(e) Punio Arriba

(f) Mano estirada right

(g) Labelled vertical bars (Visual feedback)

(h) Clustered Log(RMS) feature space using fuzzy c-means. The centre of clusters are magnified in black.

Figure 3.12: An example of a subject who could create six different movements which were both perceivable and repeatable for himself over time (Subject5). After training the system, the subject labelled the movements in a way which best facilitated recall. He chose Spanish-English words for this purpose. Figures 3.12(a), 3.12(b), 3.12(c), 3.12(d), 3.12(e), 3.12(f), show the movements and corresponding labels. Figure 3.12(g) shows the visual feedback, provided to the subject. Each bar corresponds to a specific movement (cluster) and it is labelled entirely by the user. In this example, the subject is in the process of labelling the sixth movement which he called 'Casi Rest', by highlighting the corresponding bar. Figure 3.12(h) shows the corresponding clustered feature space along with the chosen labels for the clusters

Chapter 4

Results

4.1 Introduction

In this chapter, the performance of the individual subjects as well as the performance of the group under study are presented. These results were obtained based on the $Log(RMS)$ and $Stdev(AR)$ feature spaces. These feature spaces were clustered using the fuzzy c-means algorithm during the training session. New data points were classified by calculating their membership values in real-time. As discussed in Section 3, two 2-dimensional feature spaces were created:

1. $Log(RMS)$ is the natural logarithm of RMS values, recorded from each input channel. This feature was calculated based on Equation (3.2)

2. $Stdev(AR)$ is the standard deviation of a 4^{th} order AR model for each input channel. This feature was calculated using Equation (3.11).

Since the subjects were asked to perform any self-selected, natural and repeatable movements, both the number of movements and the type of the movement vary from one subject to another. Therefore, the performance for each subject has been calculated based on the number and type of the movement that the specific subject created in the training session. The training session was covered in Section 3.3.1.

In this chapter, the error estimation technique is discussed, followed by the individual and group performance results. The complete results for all individual subjects are included in Appendix A. In the last part of this chapter, the performance of the different cluster validity algorithms, which were used in this study, are compared.

4.2 Error estimation

To estimate the error, a movement was considered as an intentional self-selected movement if the subject could maintain the contraction for five consecutive sampling windows. This could be mathematically explained as follows:

Let $F^j_{X,Y}$ be the feature vector calculated in the j^{th} sampling window.

We define μ^j_F as the maximum membership value of the data in the j^{th} window. Therefore, a movement is considered intentional if:

$$\mu^j_F = \mu^{j+1}_F = \mu^{j+2}_F = \mu^{j+3}_F = \mu^{j+4}_F \tag{4.1}$$

Referring to Section 3.3.2, each sampling window was 0.2 seconds over which the features were calculated. This means that the subject should maintain the contraction (movement) for 1 second (5 consecutive movements × 0.2-second window=1 second). Then, the computer would recognize the contraction (movement) as an intentional movement. The requirement for 5 consecutive classifications ensures that the algorithm only hones in on intentional movements. This procedure is illustrated in Figure 4.1.

The performance for the i^{th} movement was calculated by:

$$R_i = \frac{M_i}{k}\ , \quad i = 1 \ldots n, \tag{4.2}$$

where R_i is the calculated performance for the i^{th} movement, M_i is the number of correct movements for the i^{th} movement, n is the total number of different movements and k is the number of repetitions for each requested movement by the computer.

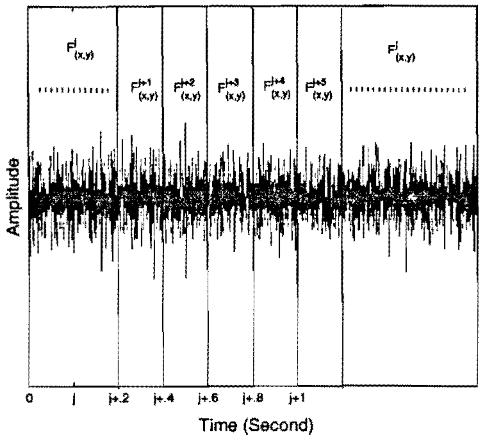


Figure 4.1: The estimation of the correct intentional movement. $F^{j}_{X,Y}$ is the feature vector calculated in the j^{th} sampling window. μ^{j}_{F} is defined as the maximum membership value of the data point in the j^{th} window. If $\mu^{j}_{F} = \mu^{j+1}_{F} = \mu^{j+2}_{F} = \mu^{j+3}_{F} = \mu^{j+4}_{F}$, then the current movement is an intentional movement. Amplitude is in arbitrary units.

4.3 Classification results

An example of a subjects's performance results is shown in Table 4.1. These results have been calculated for Subject5. In this example, the subject created 6 different movements that were perceivable to himself. The movements were named mostly in Spanish. Each movement has a corresponding cluster in both the *Log*(*RMS*) and *Stdev*(*AR*) feature spaces. The computer recognizes each cluster by its cluster index. As mentioned earlier, the cluster indices may differ from one feature space to another. It is also possible that two different movements have two different clusters in one feature space, but share a cluster in another feature space. In this case, these two movements are not separable in the latter feature space. For example, in Table 4.1, the corresponding cluster indices for movements 'supinated punio' and 'Casi Rest' are cluster #5 and cluster #6 in *Log*(*RMS*) feature space, respectively. These two movements share cluster # 5 in *Stdev*(*AR*) feature space. Therefore, these movements are not separable in *Stdev*(*AR*) feature space, which indeed affects the average performance. The accuracy of each movement was calculated using Equation (4.2), where $k = 10$. That is, each movement was repeated 10 times. Therefore, if the subject fails to perform one movement, the accuracy is 90% for this movement.

The subject's accuracy was considered as the average accuracy over all movements. The average accuracy for Subject5 over 6 movements was 85% ± 17.08%. The performance for all other subjects can be found in Appendix A

The average accuracy for all participants in this study are compared in Table 4.2. The results[1] were obtained based on *Log*(RMS) feature space. Subject9 was the only amputee who participated in this study and never used a powered prosthesis before.

Table 4.3 shows the same group results based on *Stdev*(AR) features.[1] Comparing Table 4.1 and Table 4.2 clearly shows that using the *Log*(RMS) feature space, leads to a better performance.

4.4 Cluster validity analysis results

At the end of each trial, the performance of each cluster validity algorithm was calculated. This test was done to find what algorithm could estimate the optimum number of clusters (movements) with minimal error. Table 4.4 compares the performance among different cluster validity indices for Subject1. In this example, at the end of the training session, Subject1 claimed that he performed 6 different movements. Therefore, this number was taken as the base number for comparing cluster validity indices. The results shown in Table 4.4 are calculated based on the *Log*(RMS) feature space. For the 'Partition' and 'Xie-Beni' indices, the optimum number of clusters is where the score attains minimum. For 'Silhouette', 'Partition coefficient' and 'Classification entropy' the optimum number of clusters, occurs when the score is maximized. Figure 4.2 is the visual representation of Table 4.4. It can be seen that in this example, only the Silhouette cluster validity index correctly estimates the actual number of clusters.

Similar scores were obtained for other subjects. The complete cluster validity results for individual subjects based on the *Log*(RMS) and *Stdev*(AR) feature spaces can be found in Appendices B, and C, respectively.

[1] The results obtained from Subject2 were eliminated as he could not comply with experimental protocol.

4.5 The best cluster validity index candidate

The results obtained in Section 4.4 for all subjects, suggests that the cluster validity indices generally do not estimate the actual number of movements, as specified by the users, for each trial. Therefore, for estimating the best score, the cluster validity averaged over all trials was considered.

The cluster validity error is simply defined as:

$$S_c = |C_{act} - C_{est}|, \tag{4.3}$$

where S_c is the cluster validity error for the index in question, C_{act} is the actual number of clusters defined by the user, and C_{est} is the number of clusters estimated by the cluster validity algorithm in question. To have a better understanding of Equation (4.3), consider Table 4.4. The number of different movements claimed by Subject1 (C_{act}) was 6. Therefore, the cluster validity errors for Subject1 for all the tested indices are:

- $S_c = |C_{act} - 9| = 3$ (Partition Index)
- $S_c = |C_{act} - 7| = 1$ (Xie-Beni Index)
- $S_c = |C_{act} - 6| = 0$ (Silhouette Index)
- $S_c = |C_{act} - 2| = 4$ (Partition Coefficient)
- $S_c = |C_{act} - 10| = 4$ (Classification entropy)

Now define $\bar{S}$ as the average cluster validity performance for the index in question, calculated for over all subjects as follows,

$$\bar{S} = \frac{\sum_{k=1}^{N_T} S_i}{N_T}, \tag{4.4}$$

where S_i is the cluster validity error for the index in question calculated for the i^{th} subject, and N_T is the total number of trials. In this study, since we had 7 able-bodied subjects and two trials of an amputee, the total number of trials was set at $N_T = 9$.

Table 4.5 compares the calculated average cluster validity performance for all five studied indices. *The best candidate for the cluster validity index* is the index that has the minimum deviation from the actual claimed number of clusters. Referring to Table 4.5, it can be seen that the 'Silhouette validity index', on average, could predict the number of clusters with least error. The second best candidate for this application, is the 'Xie-Beni index', with performance equal to that of the Silhouette index, in $Stdev(AR)$ feature space.

4.6 Summary

In this chapter individual and group performance were presented. It was shown that the proposed algorithm performs better in the $Log(RMS)$ feature space. The performance of different cluster validity algorithms were compared and it was found that on average, the 'Silhouette index' estimates the actual number of performed movements with less error in the $Log(RMS)$ feature space. It was also shown that both 'Silhouette' and 'Xie-Beni' indices perform comparably in the $Stdev(AR)$ feature space.

Chapter 5 will discuss the results in more details.

Table 4.1: The results obtained from Subject5. The subject labelled his performed movements mostly in Spanish. Each movement has a corresponding cluster in both the *Log*(*RMS*) and *Stdev*(*AR*) feature spaces. *Cluster index* is an integer number assigned to each cluster in either of the feature spaces. The computer program internally refers to each cluster by this index. *Accuracy1* is calculated based on the clustered data in the *Stdev*(*AR*) feature space. This calculation assumes that each cluster in the *Log*(*RMS*) feature space has a corresponding cluster in the *Stdev*(*AR*) feature space, and the number of clusters are equal in both feature spaces. Unlike *Accuracy1*, *Accuracy2* takes the exact number of clusters in the *Stdev*(*AR*) feature space. In this case, if two clusters have the same index number in the *Stdev*(*AR*) feature space, it means the feature space has one cluster less than the *Log*(*RMS*) feature space. In this case, the group mean accuracy is calculated based on the exact number of clusters in the *Stdev*(*AR*) feature space. In this example, it is inferred that *supinated punio* and *Casi Rest* clusters are the same movement, have the same cluster in the *Stdev*(*AR*) feature space, and share the same accuracy.

Name of the movement	Cluster index for Log(RMS) features	Accuracy for Log(RMS) features (%)	Cluster index for Stdev(AR) features	Accuracy1 (%)	Accuracy2 (%)
supinated punio	5	100.00	5	20.00	20.00
Casi Rest	6	80.00	5	0.00	
mano estirada abajo	1	90.00	1	90.00	90.00
Rest	3	100.00	4	90.00	90.00
Punio arriba	4	50.00	3	90.00	90.00
mano estirada right	2	90.00	2	70.00	70.00
Average among movements	N/A	**85.00 ± 17.08**	N/A	**60.00 ± 36.51**	**72.00 ± 27.13**

Table 4.2: Experimental results based on the $Log(RMS)$ features. For each subject, the accuracy was calculated by averaging the score over all movements. The score for each movement was calculated using Equation (4.2)

Subject	**Accuracy mean among movements (%)**	**No. of Performed movements**	**No. of Performed movements ≥80%**	**Accuracy for movements ≥80%**
Subject1	75.00 ± 18.93	6	3	90.00 ± 8.16
Subject3[2]	90.00 ± 12.65	5	4	95.00 ± 8.66
Subject4	94.00 ± 4.90	5	5	94.00 ± 4.90
Subject5	85.00 ± 17.08	6	5	92.00 ± 7.48
Subject6	48.89 ± 39.85	9	4	92.50 ± 4.33
Subject7	96.67 ± 4.71	6	6	96.67 ± 4.71
Subject8	86.00 ± 8.00	5	5	86.00 ± 8.00
Subject9[3]	56.00 ± 37.20	5	2	95.00 ± 5.00
Subject9[3]	87.50 ± 12.99	4	3	93.33 ± 9.43
Group Average	**79.90% ±16.81%**	**5.67 ±1.41**	**4.11 ±1.27**	**92.72% ±3.18%**

[2] Subject3 repeated the test due to misunderstanding of the protocol.

[3] Subject9 (amputee) was asked to participate in the study twice to investigate the possible improvement from the first session to the second session.

Table 4.3: Experimental results based on *Stdev*(AR) features. For each subject, the accuracy was calculated by averaging the score over all movements. The score for each movement was calculated using Equation (4.2). *Expected No. of Movements* is the number of movements in the *Log*(RMS) feature space. This assumes that if the *Stdev*(AR) feature space is as qualified as *Log*(RMS), the same number of clusters should be found. *Actual No. of Performed Movements* is the actual number of clusters that the user 'virtually' labelled in the *Stdev*(AR) feature space. (For more information refer to Section 3.3.6)

Name	Expected No. of Movements	Accuracy for Expected No. of Movements	Actual No. of Performed Movements	Accuracy for Actual No. of Performed Movements	No. of Performed Movements ≥ 80%	Accuracey for No. of Performed Movements ≥ 80%
Subject1	6	26.67 ± 24.27	4	40.00 ± 18.71	0	N/A
Subject3[2]	5	51.00 ± 41.52	4	63.75 ± 36.64	2	100.00 ± 0.00
Subject4	5	38.00 ± 39.19	3	63.33 ± 30.91	2	85.00 ± 5.00
Subject5	6	60.00 ± 36.51	5	72.00 ± 27.13	3	90.00 ± 0.00
Subject6	9	18.33 ± 28.28	5	33.00 ± 30.92	1	90.00 ± 0.00
Subject7	6	71.67 ± 34.84	6	71.67 ± 34.84	4	92.50 ± 8.29
Subject8	5	65.00 ± 34.93	4	81.25 ± 14.31	3	88.33 ± 8.50
Subject9[3]	5	29.00 ± 34.70	3	48.33 ± 32.74	1	90.00 ± 0.00
Subject9[3]	4	52.50 ± 35.62	3	70.00 ± 21.60	1	100.00 ± 0.00
Group Average	**5.67 ±1.41**	**45.80 ±18.62**	**4.11 ±1.05**	**60.37 ± 16.28**	**1.89 ± 1.27**	**91.98 ± 5.38**

[2] Subject3 repeated the test due to misunderstanding of the protocol.

[3] Subject9 (amputee) was asked to participate in the study twice to investigate the possible improvement from the first session to the second session.

Table 4.4: Comparing the scores of different cluster validity indices obtained form Subject1. For 'Partition' and 'Xie-Beni' indices, the optimum number of clusters is where the score is the minimum. For 'Silhouette', 'Partition coefficient' and 'Classification entropy' the optimum number of clusters, occurs at the maximum score.

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.65064	8.126673	0.612376	0.734102	0.419002
3	0.314355	9.415495	0.59488	0.675246	0.582835
4	0.293784	6.736207	0.606054	0.65235	0.676572
5	0.240109	8.591402	0.70219	0.658036	0.71245
6	0.196282	8.944959	0.706429	0.660865	0.737112
7	0.167794	6.481326	0.686248	0.644219	0.789852
8	0.155124	9.870567	0.649376	0.621625	0.850926
9	0.13806	9.189331	0.640582	0.610087	0.893465
10	0.164747	8.492805	0.64035	0.585975	0.970473

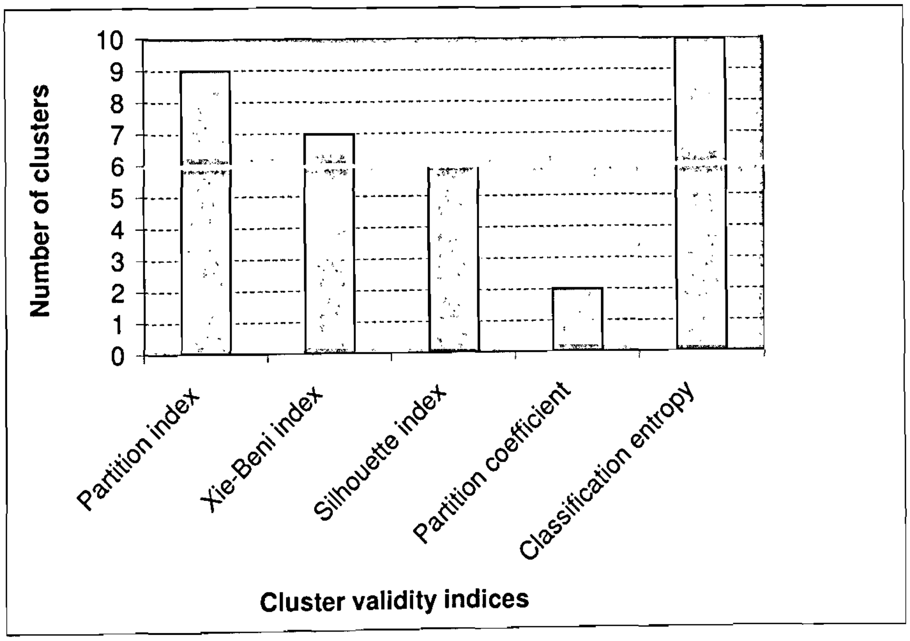


Figure 4.2: The visual representation of Table 4.4. It can be seen that, only silhouette index was successful at estimating the optimum number of clusters, as expressed by the subject.

Table 4.4: Comparing the scores of different cluster validity indices obtained form Subject1. For 'Partition' and 'Xie-Beni' indices, the optimum number of clusters is where the score is the minimum. For 'Silhouette', 'Partition coefficient' and 'Classification entropy' the optimum number of clusters, occurs at the maximum score.

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.65064	8.126673	0.612376	0.734102	0.419002
3	0.314355	9.415495	0.59488	0.675246	0.582835
4	0.293784	6.736207	0.606054	0.65235	0.676572
5	0.240109	8.591402	0.70219	0.658036	0.71245
6	0.196282	8.944959	0.706429	0.660865	0.737112
7	0.167794	6.481326	0.686248	0.644219	0.789852
8	0.155124	9.870567	0.649376	0.621625	0.850926
9	0.13806	9.189331	0.640582	0.610087	0.893465
10	0.164747	8.492805	0.64035	0.585975	0.970473

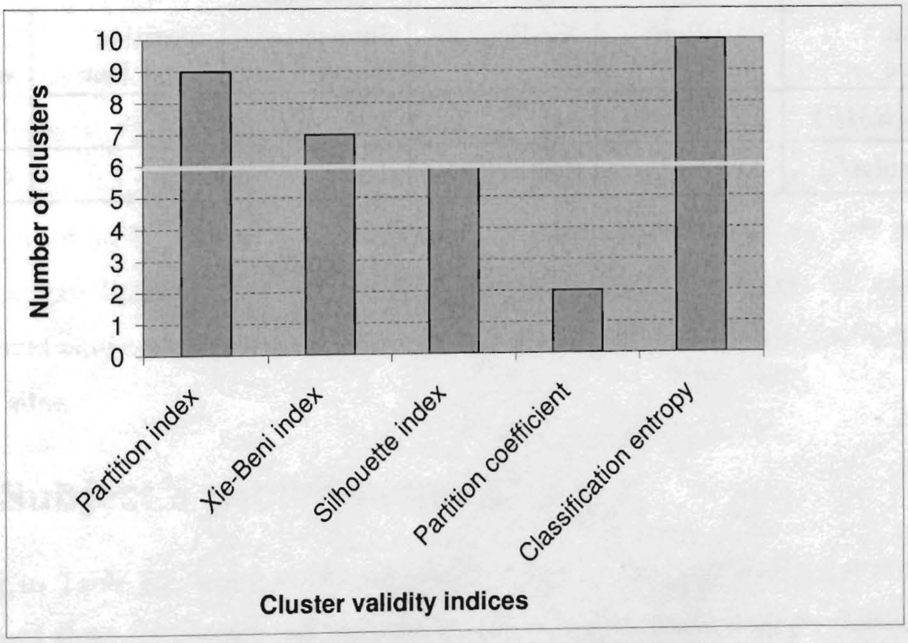


Figure 4.2: The visual representation of Table 4.4. It can be seen that, only silhouette index was successful at estimating the optimum number of clusters, as expressed by the subject.

Table 4.5: Average cluster validity performance. Each number shows the average offset between the actual number of clusters determined by the subject and the estimated number of clusters. For example, in the $Log(RMS)$ feature space, on average, the Xie-Beni index, estimated the number of the clusters with ±3.00 clusters offset from the actual number of clusters.

Feature Space	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
$Log(RMS)$	4.00	3.00	2.44	3.67	4.11
$Stdev(AR)$	4.00	3.33	3.33	3.67	4.33

Chapter 5

Discussion

5.1 Introduction

Before opening a discussion on the results, it should be noted that the performance results obtained in this study, may not be comparable to those of previous studies. Previous researchers used different approaches to classify forearm muscle signals. Nevertheless, each approach has its own pros and cons. For example, as mentioned in Section 2.3, in the previous studies, the number of recording sites for recording EMG signals, varies from a single site up to 8 channels. In the past studies, a great deal of different features in different domains (time domain, frequency domain, time-frequency domain) were investigated and different classifiers were used. The majority of the systems, although yielding very high accuracies, were investigated off-line. Therefore, the author believes that the results obtained in this study should only be compared with studies that demonstrated real-time classification from 2 muscle sites.

5.2 Subject's performance

Referring to Table 4.2, which shows the results based on the $Log(RMS)$ feature space, it can be seen that, on average, all subjects except Subject6 could produce between 4 to 6 movements, with better than 75% accuracy. Although Subject6 provided 9 different movements at the training session, he failed to produce the same type of movements later in the testing session. Subject6 later revealed that not all 9 movements felt natural. Therefore, he could not remember how he generated those movements. Althought, this poor performance affected the group average performace, the algorithm suggested that Subject6 could perform only four repeatable self-selected movements with 92.5±4.33% accuracy. Subject2 performed combinational movements (i.e. rotating a wrist while opening and closing fingers). Therefore, the results for this subject were eliminated from the test. This may suggest this algorithm may not work for all individuals. It is expected that the subjects should understand how to use the system and how to train it. Otherwise, the system may fail to learn from the user.

5.3 $Log(RMS)$ or $Stdev(AR)$ feature space?

Comparing the results in Table 4.2 to Table 4.3 shows that the $Log(RMS)$ feature space performs much better than the $Stdev(AR)$ feature space. The number of repeatable movements as well as the accuracy is higher based on the $Log(RMS)$ feature space. This may be due to the fact that RMS is a nonlinear function of force (Section 2.2.5, Figure 2.6). $Log(RMS)$ may reflect the nonlinear relationship observed between the measured RMS value and contraction force. In contrast, referring to Section 3.3.2, $Stdev(AR)$ feature space was expected to perform as well as $Log(RMS)$, but it failed to produce comparable results. This maybe do to the fact that in AR modelling, the signal is assumed to be locally stationary. It also assumes that the system is linear-time-invariant. Although the results obtained from 'reverse arrangement test', described in Section 3.3.2 showed that more than 85% of all recorded signals within a 0.2-second window, passed the stationarity criterion at a significance level of $\alpha = 0.05$, the rest of the windowed signals which did not pass the test, may have caused the calculated AR coefficients to be invalid. Therfore, this may have caused $Stdev(AR)$ feature space to be poorly clustered. It is also possible that the 4^{th} order chosen for AR modeling is not sufficient for modelling signal within the sampling window. Alternately, perhaps the standard deviation of AR coefficients is not a suitable dimension reduction function in this study. This portion needs further investigation and study.

5.4 The user error

It appears that the accuracy of the system was affected by user error. Although the system asked users to label the movements themselves during the testing session, when the system asked the user to perform a specific movement, the user may have performed the wrong movement. Several subjects expressed nervousness in completing the test. This state of stress likely accounted for heightened user error in some instances. In spite of expressing nervousness in responding to the computer prompt at the testing session, most users found the control system to be very responsive and easy to switch from one movement to another.

5.5 Movements repeatability

Preliminary tests showed that the system was very sensitive about the resting point. For example, if the user rests his or her forearm on the arm chair with his or her hand closed, this resting point is different from when he or she rests with the hand open. This is exactly what happens in real life applications because although these two are considered as similar resting positions from the user's point of view, physiologically speaking, these two are different movements. (i.e. rest with the hand close is different from the rest with the hand open). This problem was also observed while the subjects performed movements other then 'Rest'. Although, the subjects were asked to concentrate on their movements and their forearm positions to avoid this type of error, monitoring the subjects during the experiment revealed that some of the subjects did not perform exactly the movements with which they had trained the system. Some of the errors may have been due to this problem. Subject3 volunteered to perform the second trial because of this misunderstanding. Referring to Table 4.2 it can be seen that Subject3 had one of the highest scores among users. This proves that the subject should train the system by performing repeatable natural movements so that he or she can produce them again after training session.

It is expected that for increasing the accuracy of the system, more trials are needed. Unfortunately, due to limited time and resources in this study, it was not possible to perform several trials on every participant and investigate the performance accuracy over different trials. However, Subject3 and Subject9 performed 2 different trials and the results showed significant increase in accuracy. This may be due to the fact that the subject learns how to manipulate the system over time. Although 2 trials alone are not sufficient to conclude that the accuracy increases with the number of trials, further investigation is warranted.

The surprisingly high accuracy for the lone amputee participant in this study (Subject9) could be due to the lack of fingers. Lack of finger movements and finger positions during the experience for Subject9, may have helped the subject to perform the forearm movements without being concerned about the finger position. For example, 'Rest' position for this subject had only one possible meaning, regardless of the position of the fingers.

To be admissible, a movement had to surpass an 80% accuracy threshold. Such movements were deemed repeatable movements. This definition of repeatable movements may help clinicians to identify the most reliable signals to map to a prosthetic hand with multiple degrees of freedom. For example, as mentioned earlier, although Subject6 claimed 9 performed movements, the algorithm suggested that only 4 of them could be used reliably (with $92.5\% \pm 4.33$ accuracy) as control signals.

5.6 The system's delay

Currently, the delay time for error estimation is 1 second (0.2 seconds × 5 consecutive calculated features) within a 3-second window, which was earlier introduced as T_d. The algorithm should be assured that the requested movement was not performed by chance. Therefore, it waited for five consecutive movements (one second) before assigning the performed movements into one of the obtained categories. In real life applications, this delay may also be eliminated as all the movements are intentionally initiated by the subjects instead of being cued by a computer. As a consequence, T_d would be eliminated because in real life applications, the user would perform self-selected movements spontaneously. Therefore, there would no need to introduce the delay, T_d into the system.

5.7 The best cluster validity index in this application

As mentioned in Section 2.4.2, the number of expected clusters should be passed to the fuzzy c-means clustering algorithm before clustering the feature space. The objective of using a cluster validity index is to estimate the optimum number of clusters available in the feature space. Fortunately, in this study, the number of clusters was declared by the subject. If the subject felt that he or she produced 5 different movements, this number was taken as the number of expected clusters in the feature space. The validity of the clusters was confirmed by Step 4 of the protocol. Referring to Figure 3.4, Step 4 is when the subject was given the chance to test the system's responsiveness. If the subject felt that the system was responsive, the feature space was well clustered and the number of defined clusters was valid. This step could *subjectively* validate the clusters in the feature space. In order to validate the clusters *objectively*, as mentioned earlier, the performance of five different cluster validity indices was compared to the subjective results.

Referring to Table 4.5, although the results look somewhat disappointing, the 'Silhouette cluster validity index' in the $Log(RMS)$ feature space seemed to estimate the optimum number of clusters with minimum error. On the other hand, in the $Stdev(AR)$ feature space, both 'Xie-Beni index' and 'Silhouette index' performed better than the other cluster validity indices. However, in a fully automatic system where the number of movements are expected to be discovered entirely by the algorithm, more reliable cluster validity algorithms, or possibly a combination of these indices should be considered.

5.8 Limitations

As the system uses wet-type electrodes and performs amplification distally, the system is susceptible to motion artifact. Some of the misclassified features may be due to this phenomenon. This can be eliminated by embedding both the amplifier and the filters inside the electrode at the skin surface, as in conventional Otto Bock electrodes.

The software program was written in MATLAB®. It has been reported that the

MATLAB® environment is not suitable for real-time applications. For instance, the system had to acquire data, stop data acquisition, and then calculate the features. In order to calculate the features for the next sampling window (0.2 seconds later), this process had to be repeated. This 'loop' introduced a delay in the experiment. It is believed that by implementing the algorithm into an embedded system, this delay can be minimized.

The clustering is based on the fuzzy c-means algorithm which is to optimize Equation (2.2). Due to the nature of all minimization algorithms, the process may not always converge to a global minimum and as a result, the feature space may be poorly clustered. Therefore, new points maybe misclassified and assigned to a wrong cluster. This error can be reduced by setting different starting points and performing the fuzzy c-means accordingly to find the optimum solution to Equation (2.2).

5.9 Summary

This chapter reviewed the comparison between the two proposed feature spaces based on the subject's performances. It was found that the $Log(RMS)$ feature set exhibits superior performance. Several factors were mentioned as possible causes of lowering the performance of the $Stdev(AR)$ feature space. The 'Silhouette validity index' was confirmed to be the best candidate for this application and limitations of the proposed system were identified.

Chapter 6

Conclusions

6.1 Conclusion

A novel method has been proposed for automatically clustering and classifying muscle signals from self-selected intentional movements in real-time. This algorithm used a very simple feature set (natural logarithm of RMS values) to classify the muscle signals, which may suit real-time application where the computational complexity should be considered. The classification was done by the fuzzy c-means clustering algorithm, which is an unsupervised classification technique. Unlike the previous studies, in this study the user was not constrained to perform predefined movements. The subject trained the system based on his or her self-selected intentional movements. Therefore, the system could be adapted to his or her ability to produce different forearm movements. The performance of different cluster validity indices were compared and it was found that the 'Silhouette cluster validity index' was the best among all the tested indices in terms of agreement with the user specified number of movements. Since no skin preparation was done, results suggest that this algorithm could be robust to any skin condition.

On average, most people were able to produce 4 different movements with 92.72% ±3.18% accuracy, and 5 movements with 79.90% ±16.81% accuracy, which seems to be reliable enough for prosthetic control. The visual feedback helped the participants to see the responsiveness of the computer to their forearm movements. In spite of expressing nervousness in responding to the computer prompt during the testing session, most users found the control system to be very responsive and were able to switch from one movement to another with ease. The system could also suggest the number of repeatable movements. This might help the clinicians to identify the most reliable control signals, derived from forearm movements, for powered prosthesis.

6.2 Summary of the contributions

The contributions of this thesis can be summarized into two different categories:

6.2.1 Methodological Contributions

- Two new feature spaces, the natural logarithm of the root-mean-square ($Log(RMS)$) and the standard deviation of autoregressive coefficients ($Stdev(AR)$), were introduced for EMG pattern recognition. These had not been used in any of the past studies. The novel choice of the 2-dimensional $Log(RMS)$ feature space, was particularly conducive to clustering.

- The fuzzy c-means algorithm was proposed as the means to cluster the novel feature spaces. Real-time classification was done by calculating the membership value of each data point. The author believed that none of the reviewed literature at the time of writing this thesis had used this unsupervised approach to classify forearm EMG signals.

- The Silhouette index was shown to yield the best estimation of cluster multiplicities (i.e. the number of different movements). Optimality was measured in terms of minimum classification error for the maximum number of movements.

6.2.2 Clinical Contributions

- A new learning system was proposed. Unlike previous studies, the subject was not restricted to any predefined movements. The system was trained entirely on each subject's self-selected intentional movements. Therefore, learning was shifted from the subject to the machine.

- A new interactive method was proposed to help the subject graphically label the created self-selected movements.
- In unprecedented fashion, the proposed algorithm suggested the number of reliable and repeatable movements a subject had produced. This feature might help clinicians to identify the most reliable control signals for powered prosthesis.
- Unlike previous studies, the subject could switch from one movement to another in real-time without any restrictions.
- Since no skin preparation was done in this study, the results suggested that this system could learn and adapt itself to the current skin condition of the subject. This capability may help clinicians to fit multi-function prosthetics, faster and with less problems.
- It may take up to several weeks to train an amputee to control a power prosthetic hand. The method developed in this thesis required minimal user training. In fact, in the present study, users only had 2 minutes of training. It appeared that the results of this study significantly contributed in reducing the training period for the amputee subjects, thereby potentially lowering the cost and the resources associated with amputee training.

6.3 Future work

Most of the previous work that used other approaches such as artificial neural network , were not adaptive over time. Once the network is trained, it is not possible to retrain the network in real-time based on the newly recorded EMG data. In contrast, the proposed method may overcome this problem in real-time applications as it keeps finding new clusters based on the previous data. This, will be investigated by implementing the algorithm into a real life application.

Although no skin preparation was done in this study, the system was still able to classify different hand movements regardless of the signal quality. The system calculated the features based on the skin condition at the time of training. If the skin condition changed over time, due to the change in recorded EMG signal, the clusters moved accordingly. But as the system keeps finding new clusters based on the recorded data, new clusters will replace the older ones. This could be considered as a slow transition from older clusters to the new clusters in the feature space. Robustness of the algorithm to different skin conditions will be investigated in the future.

The author is hoping this algorithm may contribute to the design of new upper extremity prostheses, which are user-independent and can operate on the basis of a user's self-selected functional intent.

Appendix A

Detailed results for all subjects

In this appendix the detailed results obtained from all subjects are shown. The subjects labelled the performed movements that were perceivable to themselves. Each movement has corresponding clusters in both *Log*(RMS) and *Stdev*(AR) feature spaces.*Cluster index* is an integer number assigned to each cluster in either of the feature spaces. .The computer program internally refers to each cluster by this index. *Accuracy1* is calculated based on the clustered data in the *Stdev*(AR) feature space. This calculation assumes that each cluster in the *Log*(RMS) feature space has a corresponding cluster in the *Stdev*(AR) feature space, and the number of clusters are equal in both feature spaces. Unlike *Accuracy1*, *Accuracy2* takes the exact number of clusters in the *Stdev*(AR) feature space. In this case, if two clusters have the same index number in the *Stdev*(AR) feature space, it means the feature space has one cluster less than the *Log*(RMS) feature space. In this case, the group mean accuracy is calculated based on the exact number of clusters in the *Stdev*(AR) feature space.

Subject3 repeated the test due to misunderstanding of the protocol. Subject9 (amputee) was asked to participate in the study twice to investigate the possible improvement from the first session to the second session.

The results obtained from Subject2 were eliminated as he could not comply with experimental protocol.

Name of the movement	Cluster index for Log(RMS) features	Accuracy for Log(RMS) features (%)	Cluster index for Stdev(AR) features	Accuracy1 (%)	Accuracy2 (%)
Up	1	90.00	4	60.00	60.00
Palm Up	2	70.00	5	0.00	10
Split Finger	5	40.00	5	10.00	
Fist	4	70.00	6	50.00	50.00
Rest	6	100.00	6	0.00	
Down	3	80.00	3	40.00	40.00
Average among movements	**N/A**	**75.00± 18.93**	**N/A**	**26.67± 24.27**	**40.00 ± 18.71**

Table A.1: The performance results from Subject1

Name of the movement	Cluster index for Log(RMS) features	Accuracy for Log(RMS) features (%)	Cluster index for Stdev(AR) features	Accuracy1 (%)	Accuracy2 (%)
Down	5	100.00	1	100.00	100.00
Fist - Bit-UP	4	80.00	4	20.00	20.00
Straight UP	3	100.00	2	35.00	35
Rest	2	100.00	2	0.00	
Palm UP	1	70.00	3	100.00	100.00
Average among movements	**N/A**	**90.00± 12.65**	**N/A**	**51.00± 41.52**	**63.75 ± 36.64**

Table A.2: The performance results from Subject3

Name of the movement	Cluster index for Log(RMS) features	Accuracy for Log(RMS) features (%)	Cluster index for Stdev(AR) features	Accuracy1 (%)	Accuracy2 (%)
Palm	2	90.00	2	80.00	80.00
Rest	3	100.00	1	90.00	90.00
UP	1	100.00	3	20.00	20.00
Down	5	90.00	4	0.00	0.00
Fist	4	90.00	5	0.00	0.00
Average among movements	**N/A**	**94.00± 4.9**	**N/A**	**38.00± 39.19**	**38.00± 39.19**

Table A.3: The performance results from Subject4

Name of the movement	Cluster index for Log(RMS) features	Accuracy for Log(RMS) features (%)	Cluster index for Stdev(AR) features	Accuracy1 (%)	Accuracy2 (%)
supinated punio	5	100.00	5	20.00	20.00
Casi Rest	6	80.00	5	0.00	
mano estirada abajo	1	90.00	1	90.00	90.00
Rest	3	100.00	4	90.00	90.00
Punio arriba	4	50.00	3	90.00	90.00
mano estirada right	2	90.00	2	70.00	70.00
Average among movements	**N/A**	**85.00 ± 17.08**	**N/A**	**60.00 ± 36.51**	**72.00 ± 27.13**

Table A.4: The performance results from Subject5

Name of the movement	Cluster index for Log(RMS) features	Accuracy for Log(RMS) features (%)	Cluster index for Stdev(AR) features	Accuracy1 (%)	Accuracy2 (%)
Rest	1	100.00	2	90.00	90.00
Rotate right	9	30.00	9	40.00	40
Flex Left	7	10.00	9	0.00	
Flex Right	6	20.00	5	10.00	10
Straight Up	5	90.00	5	0.00	
Straight Down	3	90.00	5	0.00	
Rotate left	2	0.00	8	20.00	20.00
Straight Fingers	4	10.00	4	5.00	5.00
Fist	8	90.00	4	0.00	
Average among movements	**N/A**	**48.89±39.85**	**N/A**	**18.33± 28.88**	**33.00± 30.92**

Table A.5: The performance results from Subject6

Name of the movement	Cluster index for Log(RMS) features	Accuracy for Log(RMS) features (%)	Cluster index for Stdev(AR) features	Accuracy1 (%)	Accuracy2 (%)
Palm	2	100.00	5	100.00	100.00
Rest	4	100.00	2	100.00	100.00
Soft Fist	1	100.00	4	90.00	90.00
Tight Fist	3	100.00	3	80.00	80.00
Down	5	90.00	6	0.00	0.00
Fist UP	6	90.00	1	60.00	60.00
Average among movements	**N/A**	**96.67±4.71**	**N/A**	**71.67± 34.84**	**71.67± 34.84**

Table A.6: The performance results from Subject7

Name of the movement	Cluster index for Log(RMS) features	Accuracy for Log(RMS) features (%)	Cluster index for Stdev(AR) features	Accuracy1 (%)	Accuracy2 (%)
Rest	1	100.00	2	60.00	60.00
Fist	5	80.00	4	80.00	80.00
Flexion	3	80.00	1	85.00	85.00
Extension	2	80.00	1	0.00	
Flat	4	90.00	5	100.00	100.00
Average among movements	**N/A**	**86.00±8.00**	**N/A**	**65.00± 34.93**	**81.25± 14.31**

Table A.7: The performance results from Subject8

Name of the movement	Cluster index for Log(RMS) features	Accuracy for Log(RMS) features (%)	Cluster index for Stdev(AR) features	Accuracy1 (%)	Accuracy2 (%)
Rest	1	100.00	1	90.00	90.00
Wrist right	2	30.00	2	0.00	10.00
Arm twist to the left	5	60.00	2	10.00	
Arm twist to the right	4	0.00	5	45.00	45.00
Wrist down	3	90.00	5	0.00	
Average among movements	N/A	**56.00±37.20**	N/A	**29.00± 34.70**	**48.33± 32.74**

Table A.8: The performance results from Subject9 - Trial #1

Name of the movement	Cluster index for Log(RMS) features	Accuracy for Log(RMS) features (%)	Cluster index for Stdev(AR) features	Accuracy1 (%)	Accuracy2 (%)
Wrist right	3	80.00	2	100.00	100.00
Wrist down	2	100.00	3	60.00	60.00
Twist right	4	70.00	3	0.00	
Rest	1	100.00	1	50.00	50.00
Average among movements	N/A	**87.50±12.99**	N/A	**52.50± 35.62**	**70.00± 21.60**

Table A.9: The performance results from Subject9 - Trial #2

Appendix B

Detailed results of different cluster validity indices based on $Log(RMS)$ feature space

In this appendix, detailed results of different cluster validity indices applied to $Log(RMS)$ feature space for each individual is reviewed. The actual number of movements claimed by the subject is a highlighted row in the table. The estimated optimum number of clusters based on different indices are also highlighted in the corresponding column. It should be noted that for 'Partition' and 'Xie-Beni' indices, the optimum number of clusters is where the score is the minimum. For 'Silhouette', 'Partition coefficient' and 'Classification entropy' the optimum number of clusters, occurs at the maximum score.

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.65064	8.126673	0.612376	0.734102	0.419002
3	0.314355	9.415495	0.59488	0.675246	0.582835
4	0.293784	6.736207	0.606054	0.65235	0.676572
5	0.240109	8.591402	0.70219	0.658036	0.71245
6	0.196282	8.944959	0.706429	0.660865	0.737112
7	0.167794	6.481326	0.686248	0.644219	0.789852
8	0.155124	9.870567	0.649376	0.621625	0.850926
9	0.13806	9.189331	0.640582	0.610087	0.893465
10	0.164747	8.492805	0.64035	0.585975	0.970473

Table B.1: The results from cluster validity test for Subject1

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.442855	17.275022	0.670362	0.779907	0.354988
3	0.302975	10.906927	0.686647	0.708998	0.534151
4	0.213069	13.825942	0.682793	0.69624	0.594205
5	0.207981	7.651438	0.632415	0.657257	0.696598
6	0.178931	18.512948	0.582722	0.626858	0.779159
7	0.188656	8.773921	0.577969	0.609826	0.843401
8	0.19308	4.833199	0.546979	0.574936	0.935225
9	0.170477	5.731126	0.559709	0.565149	0.977137
10	0.147213	4.781307	0.515152	0.534642	1.053964

Table B.2: The results from cluster validity test for Subject3

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	1.193147	9.468989	0.537111	0.697116	0.464836
3	0.419385	7.929489	0.614999	0.661235	0.607476
4	0.277711	9.42425	0.601553	0.635307	0.705256
5	0.25394	17.636872	0.562421	0.59637	0.810815
6	0.245411	5.305926	0.615576	0.590336	0.871144
7	0.266697	6.092545	0.559936	0.56077	0.965571
8	0.216861	9.152159	0.544343	0.552302	1.002512
9	0.233067	5.086164	0.547911	0.532733	1.075175
10	0.21589	5.233357	0.580739	0.535687	1.092867

Table B.3: The results from cluster validity test for Subject4

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.273862	18.381967	0.689767	0.800724	0.322204
3	0.189523	8.094421	0.72172	0.773236	0.421339
4	0.192465	15.917488	0.70713	0.721704	0.548184
5	0.200919	14.576765	0.638542	0.658979	0.690415
6	0.154886	6.295869	0.581236	0.618011	0.783442
7	0.147862	6.087663	0.584796	0.604343	0.83303
8	0.143025	4.76202	0.587115	0.588241	0.891242
9	0.145683	3.487917	0.52914	0.545734	0.993829
10	0.127667	5.895475	0.534961	0.542513	1.020324

Table B.4: The results from cluster validity test for Subject5

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.591228	149.07279	0.71479	0.808181	0.321715
3	0.406249	12.425423	0.688414	0.734783	0.490202
4	0.195783	17.171468	0.753798	0.73695	0.524609
5	0.201715	10.326327	0.726148	0.697353	0.624923
6	0.129893	16.246895	0.74311	0.730785	0.582111
7	0.141956	10.580136	0.70787	0.714192	0.635601
8	0.108656	5.99321	0.711724	0.74189	0.582921
9	0.123686	5.470089	0.664947	0.713408	0.655504
10	0.104755	12.268958	0.726765	0.746857	0.59283

Table B.5: The results from cluster validity test for Subject6

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.20669	19.987621	0.792983	0.871226	0.213684
3	0.212084	10.437215	0.784946	0.816644	0.340489
4	0.204587	11.004662	0.792923	0.791776	0.416357
5	0.195701	8.149356	0.766118	0.758728	0.499417
6	0.163942	26.453084	0.674137	0.716466	0.589518
7	0.157065	10.815532	0.688178	0.704912	0.631883
8	0.143749	23.398877	0.681923	0.697684	0.661479
9	0.146759	10.842847	0.590513	0.638236	0.790802
10	0.126562	10.374092	0.601905	0.654829	0.768479

Table B.6: The results from cluster validity test for Subject7

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.596888	452.7753	0.588082	0.741199	0.409268
3	0.277367	6.506738	0.637198	0.690495	0.558276
4	0.208738	6.182448	0.596298	0.644058	0.681795
5	0.176825	8.114561	0.584736	0.623887	0.755959
6	0.183304	5.07942	0.557079	0.582166	0.870028
7	0.183769	6.552859	0.548527	0.560457	0.948236
8	0.139621	7.648326	0.535701	0.557106	0.974043
9	0.146695	6.716973	0.538791	0.538147	1.042589
10	0.151399	4.819369	0.557596	0.520321	1.109927

Table B.7: The results from cluster validity test for Subject8

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.30978	9.287381	0.750831	0.833868	0.27987
3	0.271324	12.319618	0.657939	0.738013	0.468336
4	0.243956	5.158272	0.697432	0.711287	0.565306
5	0.235008	8.605832	0.647133	0.66692	0.677551
6	0.222819	8.138481	0.6697	0.648453	0.746924
7	0.184999	5.329139	0.604592	0.617256	0.817567
8	0.182984	4.541947	0.580786	0.596245	0.880761
9	0.187549	5.923449	0.592341	0.58626	0.92653
10	0.175043	5.662729	0.555865	0.564997	0.986399

Table B.8: The results from cluster validity test for Subject9 (Trial #1)

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.271423	16.701643	0.828785	0.876343	0.216761
3	0.270071	13.659609	0.746767	0.778315	0.40253
4	0.303731	21.318801	0.706893	0.722449	0.537478
5	0.254113	11.525047	0.649096	0.686477	0.628478
6	0.24921	13.135648	0.653272	0.666768	0.698646
7	0.245762	6.892603	0.637422	0.642898	0.769503
8	0.261888	11.296902	0.571421	0.591928	0.891781
9	0.216983	5.712739	0.566348	0.582676	0.91684
10	0.214295	7.061058	0.539088	0.561583	0.976243

Table B.9: The results from cluster validity test for Subject9 (Trial #2)

Appendix C

Detailed results of different cluster validity indices based on *Stdev*(*AR*) feature space

In this appendix, detailed results of different cluster validity indices applied to *Stdev*(*AR*) feature space for each individual is reviewed. The actual number of movements claimed by the subject is a highlighted row in the table (highlighted in yellow). The estimated optimum number of clusters based on different indices are also highlighted in the corresponding column (highlighted in green). It should be noted that for 'Partition' and 'Xie-Beni' indices, the optimum number of clusters is where the score is the minimum. For 'Silhouette', 'Partition coefficient' and 'Classification entropy' the optimum number of clusters, occurs at the maximum score.

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	1.918308	20.2236	0.737194	0.7341023	0.419002
3	1.920054	11.861537	0.624506	0.6752457	0.582835
4	1.423545	13.188563	0.534372	0.6523501	0.676572
5	1.32238	4.677331	0.515741	0.6580361	0.71245
6	1.287681	8.357447	0.463223	0.6608651	0.737112
7	1.143205	8.131436	0.436759	0.6442192	0.789852
8	1.142941	5.273848	0.468223	0.6216248	0.850926
9	1.085639	4.473085	0.484071	0.6100869	0.893465
10	1.02944	3.246107	0.460313	0.5859745	0.970473

Table C.1: The results from cluster validity test for Subject1

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	2.095321	30.064534	0.663232	0.7799066	0.354988
3	1.709251	15.14686	0.562784	0.7089983	0.534151
4	1.556335	8.979462	0.559334	0.6962399	0.594205
5	1.273236	17.849045	0.542133	0.6572568	0.696598
6	1.184577	7.542964	0.520364	0.6268584	0.779159
7	1.184947	5.567849	0.481448	0.6098256	0.843401
8	1.229178	3.182005	0.504784	0.5749363	0.935225
9	1.164396	4.192892	0.47953	0.5651492	0.977137
10	1.010628	3.615247	0.469441	0.5346417	1.053964

Table C.2: The results from cluster validity test for Subject3

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	5.353902	18.728344	0.504596	0.6971157	0.464836
3	2.533631	9.875792	0.56874	0.6612347	0.607476
4	1.40192	8.104487	0.58801	0.6353069	0.705256
5	1.404859	4.941367	0.519378	0.5963701	0.810815
6	1.409015	4.517685	0.50148	0.5903364	0.871144
7	1.284481	4.575663	0.473183	0.56077	0.965571
8	1.265288	3.830895	0.464306	0.552302	1.002512
9	1.241802	4.966214	0.490992	0.5327332	1.075175
10	1.218941	3.227717	0.494399	0.5356866	1.092867

Table C.3: The results from cluster validity test for Subject4

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	1.623981	9.603393	0.688002	0.8007241	0.322204
3	1.48042	10.604762	0.560069	0.773236	0.421339
4	1.224036	12.428556	0.614735	0.7217035	0.548184
5	1.064484	8.974024	0.609454	0.658979	0.690415
6	0.908951	3.85959	0.540857	0.6180114	0.783442
7	0.825214	6.173928	0.540634	0.6043433	0.83303
8	0.737346	3.812036	0.529635	0.5882406	0.891242
9	0.729579	3.723168	0.542417	0.5457338	0.993829
10	0.747594	4.651802	0.537331	0.5425132	1.020324

Table C.4: The results from cluster validity test for Subject5

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	1.718306	19.599801	0.700312	0.8081815	0.321715
3	0.663844	6.740262	0.801677	0.7347833	0.490202
4	0.513696	13.394371	0.796665	0.7369499	0.524609
5	0.45552	6.027654	0.703255	0.6973534	0.624923
6	0.432291	5.805754	0.673561	0.7307845	0.582111
7	0.421557	9.695696	0.6582	0.7141916	0.635601
8	0.398685	11.93615	0.634474	0.7418896	0.582921
9	0.424023	8.145222	0.631639	0.7134084	0.655504
10	0.437115	7.962195	0.588825	0.7468575	0.59283

Table C.5: The results from cluster validity test for Subject6

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	0.627204	19.220594	0.860898	0.8712257	0.213684
3	0.806804	9.568371	0.760025	0.8166438	0.340489
4	0.65295	12.608071	0.642451	0.791776	0.416357
5	0.605865	7.880822	0.610948	0.7587277	0.499417
6	0.673964	8.323339	0.606083	0.716466	0.589518
7	0.686497	5.155336	0.584894	0.7049125	0.631883
8	0.648245	5.524454	0.589968	0.6976835	0.661479
9	0.597411	7.347014	0.546189	0.6382356	0.790802
10	0.59755	5.767824	0.536315	0.6548288	0.768479

Table C.6: The results from cluster validity test for Subject7

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	2.381912	71.098963	0.617552	0.7411993	0.409268
3	1.619139	8.773492	0.532496	0.6904948	0.558276
4	1.233158	5.151035	0.545659	0.644058	0.681795
5	1.22386	7.575612	0.436911	0.6238872	0.755959
6	0.973674	5.724714	0.512377	0.5821662	0.870028
7	0.968917	6.426531	0.499195	0.5604565	0.948236
8	0.961064	6.820682	0.467921	0.5571063	0.974043
9	0.921622	4.188363	0.463121	0.5381465	1.042589
10	0.875758	4.610532	0.448709	0.5203207	1.109927

Table C.7: The results from cluster validity test for Subject8

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	3.813039	19.671554	0.570698	0.833868	0.27987
3	2.924254	6.48129	0.533172	0.7380128	0.468336
4	2.802523	4.93337	0.47208	0.7112874	0.565306
5	1.961616	7.222247	0.483688	0.6669203	0.677551
6	1.756749	8.712028	0.466498	0.6484526	0.746924
7	1.700364	6.362208	0.49746	0.6172563	0.817567
8	1.70819	4.526776	0.472099	0.5962454	0.880761
9	1.670011	3.430397	0.460878	0.5862603	0.92653
10	1.612629	6.680348	0.466195	0.5649969	0.986399

Table C.8: The results from cluster validity test for Subject9 (Trial #1)

No. of Clusters	Partition index	Xie-Beni index	Silhouette index	Partition coefficient	Classification entropy
2	2.754286	21.286159	0.679215	0.8763432	0.216761
3	2.657722	8.895187	0.559302	0.7783153	0.40253
4	2.395907	9.697809	0.607502	0.7224493	0.537478
5	1.818515	5.85742	0.556708	0.6864771	0.628478
6	1.696182	6.288385	0.526756	0.6667677	0.698646
7	1.613887	4.193458	0.51758	0.6428978	0.769503
8	1.585083	5.973523	0.506427	0.5919281	0.891781
9	1.578307	12.228796	0.524382	0.5826757	0.91684
10	1.453954	4.779107	0.497273	0.5615835	0.976243

Table C.9: The results from cluster validity test for Subject9 (Trial #2)

Appendix D

List of Publications

In this section, we list the publications resulted from our research work for the thesis.

Refereed Conference Paper

- Momen, K., Krishnan, S., Beal, D., Bouffet, E., Kavanagh, B., Chau, T., "Self-organization of the communication space based on user range-of-motion: a framework for configuring non-contact augmentative communication devices", Proceedings of the 28 th Canadian Medical And Biological Engineering Society, Quebec City, Quebec, September 9-11, 2004.

Journal

- Momen, K., Krishnan, S., Chau, T., "Real-time Classification of Muscle Signals from Self-selected Intentional Movements" (Under submission).

Bibliography

[1] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum Press, New York, 1981.

[2] Michael W. King. Muscle biochemistry. Website, Oct 2004. isu.indstate.edu/mwking/muscle.html.

[3] Robert M. Berne and Matehew N Levy. *Physiology.* Mosby Publishing, New York, forth edition, 1998.

[4] John V. Basmajian and Carlo J. Deluca. *Muscles Alive, their functions revealed by electromyography.* Williams & Wilkins, fifth edition, 1985.

[5] Kevin Englehart. *Signal Representation for Classification of the Transient Myoelectric Signal.* PhD thesis, University of New Brunswick, 1998.

[6] Gerald E. Loeb and Carl Gans. *Electromyography for Experimentalists.* University of Chicago Press, 1986.

[7] Carlo J. DeLuca. The use of surface electromyography in biomechanics. *Journal of applied Biomechanics*, 13:135–163, 1997.

[8] Brown Biomed. (Brown University's Division of Biology and Medicine). Amputation and limb deficiency. Website, July 2005. biomed.brown.edu/Courses/BI108/BI108_2003_Groups/Athletic_Prosthetics/Bkgd_text.htm.

[9] N. Hogan and R. W. Mann. Myoelectric signal processing: Optimal estimation applied to electromyography Part I: Derivation of the optimal myoprocessor. *IEEE Transactions on Biomedical Engineering*, 27(7):382–395, July 1980.

[10] Francis H. Y. Chan, Yong-Sheng Yang, F. K. Lam, Yuan-Ting Zhang, and Philip A. Parker. Fuzzy EMG classification for prosthesis control. *IEEE Transactions On Rehabilitation Engineering*, 8(3):305–311, September 2000.

[11] S. M. ElBasiouny, A. M. El-Bialy, M. F. Taher, A. H. Kandil, and M. E. Rasmy. A myoelectric prosthesis controller. In *Proceedings of IEEE 29th Annual Bioengineering Conference*, pages 140 – 141, March 2003.

[12] Bekir Karlik, M. Osman Tokhi, and Musa Alci. A fuzzy clustering neural network architecture for multifunction upper-limb prosthesis. *IEEE Transactions on Biomedical Engineering*, 50(11):1255– 1261, November 2003.

[13] Daisuke Nishikawa, Wenwei Yu, Hiroshi Yokoi, and Yukinori Kakazu. On-line learning method for EMG prosthetic hand control. *Electronics and Communications in Japan, Part 3*, 84(10):1510–1519, November 2001.

[14] Kevin Englehart, Bernard Hudgins, and Philip A. Parker. A wavelet-based continuous classification scheme for multifunction myoelectric control. *IEEE Transactions on Biomedical Engineering*, 48(3), March 2001.

[15] Dori Peleg, Eyal Braiman, Elad Yom-Tov, and Gideon F. Inbar. Classification of finger activation for use in a robotic prosthesis arm. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 10(4):290–293, December 2002.

[16] Ning Ma, D K Kumar, and Nemuel Pah. Classification of hand direction using multichannel electromyography by neural network. In *The Seventh Australian and New Zealand 2001 Intelligent Information Systems Conference*, pages 405 – 410, November 2001.

[17] Han-Pang Huang, Yi-Hung Liu, and Chun-Shin Wong. Automatic EMG feature evaluation for controlling a prosthetic hand using supervised feature mining method: an intelligent approach. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, volume 1, pages 220 – 225, Taipei, Taiwan, September 2003.

[18] Han-Pang Huang, Yi-Hung Liu, Li-Wei Liu, and Chun-Shin Wong. EMG classification for prehensile postures using cascaded architecture of neural networks with self-organizing maps. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, volume 1, pages 1497 – 1502, Taipei,Taiwan, September 2003.

[19] Kevin Englehart and Bernard Hudgins. A robust, real-time control scheme for multifunction myoelectric control. *IEEE Transactions on Biomedical Engineering*, 50(7):848–854, July 2003.

[20] Kevin Englehart, Bernard Hudgins, and Adrian D.C. Chan. Continuous multifunction myoelectric control using pattern recognition. *Technology and Disability*, 15:95–103, 2003.

[21] Marko Vuskovic and Sijiang Du. Classification of prehensile EMG patterns with simplified fuzzy ARTMAP networks. In *Proceedings of the 2002 International Joint Conference on Neural Networks, IJCNN '02*, volume 3, pages 2539–2544, Honolulu, HI, May 2002.

[22] Yuuki Yazama, Minoru Fukumi, Yasue Mitsukura, and Norio Akamatsu. Feature analysis for the EMG signals based on the class distance. In *Proceedings of the 2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, volume 2, pages 860 – 863, Kobe, Japan, July 2003.

[23] Yuji Matsumura, Yasue Mitsukura, Minoru Fukumi, and Norio Akamatsu. Recognition of EMG signal patterns by neural networks. In *Proceedings of the 9th International Conference on Neural Information Processing*, volume 2, pages 750 – 754, November 2002.

[24] M.C. Santa-Cruz, R. Riso, and F. Sepulveda. Optimal selection of time series coefficients for wrist myoelectric control based on intramuscular recordings. In *Proceedings of the 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 2, pages 1384 – 1387, Istanbul,Turkey, October 2001.

[25] Reza Boostani and Mohammad Hassan Moradi. Evaluation of the forearm EMG signal features for the control of a prosthetic hand. *Physiological Measurement*, 24(2):309–319, May 2003.

[26] K. Englehart, B. Hudgins, P.A. Parker, and M. Stevenson. Classification of the myoelectric signal using time-frequency based representations. *Medical Engineering & Physics*, 21(6-7):431–438, July 1999.

[27] Xiaowen Zhang, Yupu Yang, Xiaoming Xu, and Ming Zhang. Wavelet based neuro-fuzzy classification for EMG control. In *IEEE 2002 International Conference on Communications, Circuits and Systems and West Sino Expositions*, volume 2, pages 1087–1089, July 2002.

[28] Sherif E. Hussein and Malcolm H. Granat. Intention detection using a neuro-fuzzy EMG classifier. *IEEE Engineering in Medicine and Biology Magazine*, 21(6):123–129, November-December 2002.

[29] Nikhil R. Pal and James C. Bezdek. On cluster validity for the fuzzy c-means model. *IEEE Transactions on Fuzzy Systems*, 3(3):370–379, Aug 1995.

[30] J.C. Dunn. Well separated clusters and optimal fuzzy partitions. *Journal of Cybernet*, 4:95–104, 1974.

[31] D.L. Davies and D.W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(4):224–227, 1979.

[32] Amine M. Bensaid, Lawrence 0. Hall, James C. Bezdek, Laurence P. Clarke, Martin L. Silbiger, John A. Arrington, and Reed F. Murtagh. Validity-guided (re)clustering with

applications to image segmentation. *IEEE Transactions on Fuzzy Systems*, 4(2):112–123, May 1996.

[33] X. L. Xie and G. Beni. A validity measure for fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):841–847, 1991.

[34] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[35] L. Lovmar, A. Ahlford, M. Jonsson, and AC Syvanen. Silhouette scores for assessment of SNP genotype clusters. *BMC Genomics*, 6(35), MAR 2005.

[36] R. Lleti, MC. Ortiz, LA. Sarabia, and MS. Sanchez. Selecting variables for k-means cluster analysis by using a genetic algorithm that optimises the silhouettes. *Analytica Chimica Acta*, 515(1):87–100, July 2004.

[37] Mark J. Van der Laan, Katherine S. Pollard, and Jennifer Bryan. A new partitioning around medoids algorithm. *Journal of Statistical Computation and Simulation*, 73(8):575–584, August 2003.

[38] P. Barbieri, G. Adami, A. Favretto, A. Lutman, W.Avoscan, and E. Reisenhofer. Robust cluster analysis for detecting physico-chemical typologies of freshwater from wells of the plain of friuli (northeastern italy). *Analytica Chimica Acta*, 440(2):161–170, August 2001.

[39] Tom Chau. Pattern recognition of processed EMG signals for two-site myoelectric control. Master's thesis, University of Toronto, August 1994.

[40] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification.* John Wiley & Sons Inc., second edition, 2001.

[41] Mahyar Zardoshti-Kermani, Bruce C. Wheeler, Kambiz Badie, and Reza M. Hashemi. EMG feature evaluation for movement control of upper extremity prostheses. *IEEE Transactions on Rehabilitation Engineering*, 3(4):324 – 333, Dec 1995.

[42] Ainishet Asres, Huifang Dou, Zhaoying Zhou, Yuli Zhang, and Sencun Zhu. A combination of AR and neural network technique for EMG pattern identification. In *Proceedings of the 18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Bridging Disciplines for Biomedicine.*, volume 4, pages 1464 – 1465, Amsterdam, October-November 1996.

[43] John G. Proakis and Dimitris G. Manolakis. *Digital Signal Processing: Principles, Algorithms and Applications.* Prentice Hall, third edition, 1996.

[44] David W. Scott. *Multivariate Density Estimation.* John Wiley & Sons Inc., New York, 1992.

[45] Constantinos S. Pattichis and Andreas G. Elia. Autoregressive and cepstral analyses of motor unit action potentials. *Medical Engineering and Physics*, 21(6-7):405–419, July-September 1999.

[46] Dario Farina and Roberto Merletti. Comparison of algorithms for estimation of EMG variables during voluntary isometric contractions. *Journal of Electromyography and Kinesiology*, 10:337–349, 2000.

[47] R. Merletti, G. Balestra, and M. Knaflitz. Effect of FFT based algorithms on estimation of myoelectric signal spectral parameters. In *Proceedings of the 11th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 1989. Images of the Twenty-First*, volume 3, pages 1022 – 1023, Seattle, WA, November-December 1989.