TOWARD IMPROVING THE TRAINING OF PATTERN RECOGNITION BASED MYOELECTRIC CONTROL

by

Kadie Wright

BScEE – University of New Brunswick, 2014

A Thesis Submitted in Partial Fulfillment

of the Requirements for the Degree of

Master of Science in Engineering

in the Graduate Academic Unit of Electrical and Computer Engineering

Supervisors: Erik Scheme, Ph.D., Electrical and Computer Engineering
Kevin Englehart, Ph.D., Electrical and Computer Engineering

Examining Board: Dawn MacIsaac, Ph.D., Electrical and Computer Engineering
Yevgen Biletskiy, Ph.D., Electrical and Computer Engineering
Suprio Ray, Ph.D., Computer Science

This thesis is accepted by the
Dean of Graduate Studies

THE UNIVERSITY OF NEW BRUNSWICK

October 2018

Abstract

Decades of advancements in the development of myoelectric signal processing techniques have made prosthetic devices an effective means of functional replacement for upper limb amputees. One of the control approaches that has been widely researched in this field is pattern recognition (PR) based control using electromyography (EMG) signals, which has only recently become commercially available. There are many opportunities to improve the user experience using PR control. One important issue is optimizing the training of the PR controller, which requires the collection of appropriate data. Although the inclusion of confounding factors (such as varying limb position) in the training data has been shown to significantly improve the performance of the pattern recognition approach, little work has focused on how to actually elicit the training contractions themselves.

This work examined two existing training techniques that are currently being used in the field (ramp contractions, and Velocity Guided Training), and introduces two new alternative training methods; Position Guided Training and an alternate position guided training (Position-Reset Training) approach to mimic the prompts for Prosthesis Guided Training (PGT). The comparison of approaches was motivated by a desire to incorporate more dynamic motion into the training process, which may better reflect the actual use case compared to existing methods. It was hypothesized that the new methods would provide more relevant training data which would result in improvements in real-time performance and usability in a virtual target acquisition task.

Thirteen able-bodied subjects (9 male and 4 female, mean age 24 +/- 2.1 years) completed a Fitts' Law based usability test using controllers trained with each of the training methods. For each method, EMG data representative of five different motions (hand open, hand close, wrist pronation, wrist supination, and no motion) were recorded and used to train the controller, before completing 24 repetitions of the target acquisition task.

Comparison of real-time performance metrics showed no significant difference between the ramp, Position Based Training and Position-Reset Training approaches. Velocity Guided Training, however, the currently employed method of Prosthesis Guided Training, obtained significantly better movement efficiency ($p<0.05$). No significant differences were found in throughput, a Fitts' law summary metric, which combines speed and accuracy into a single measure. These results suggest that, although other training approaches may offer more intuitive training prompts, Velocity Guided Training more effectively informs the training of pattern recognition based myoelectric control. Future work could include consideration of cognitive load and motivation on the part of the user, to help form a more complete picture of training and usability.

Acknowledgments

First and foremost, I would like to express my sincerest gratitude to my supervisors, Drs. Erik Scheme and Kevin Englehart, for their continuous support and the advice they have provided throughout my time as their student. Without their guidance and patience, this thesis would not have been possible.

Next, I would like to thank Jason Robertson and Ahmed Shehata, who have been instrumental in helping me during my classwork and troubleshooting. Jason, thank you for also being there as a friend at a time I needed it most. A big thank you also goes out to anyone that has walked alongside me during the last four years and gave me the encouragement that I needed to make this a reality and a big thank you also goes out to all of the participants who volunteered hours of their time to help me gather data.

Finally - Mom, Dad, Gary, Alex thank you for your continuous love and support throughout this whole process. It's been a long journey and I could not have done it without you.

Table of Contents

List of Tables

List of Figures

List of Symbols and Abbreviations

DOF	Degree of Freedom
PR	Pattern Recognition
EMG	Electromyogram
SGT	Screen Guided Training
PGT	Prosthesis Guided Training
MAV	Mean Absolute Value
ACA	Aligned Cluster Analysis
HACA	Hierarchical Aligned Cluster Analysis
TD	Time Domain
LDA	Linear Discriminant Analysis
ANN	Artificial Neural Network
SVM	Support Vector Machine
HMM	Hidden Markov Model
CA	Classification Accuracy
TAC	Target Achievement Control
VBT	Velocity Based Training
PBT	Position Based Training
PRT	Position-Reset Training
PCA	Principal Components Analysis
MF	Matched Filter
WP	Wrist Pronation
WS	Wrist Supination
HO	Hand Open
HC	Hand Closed
NM	No Motion

Chapter 1. Introduction

Prostheses have been used for centuries to help people restore function and cosmetic appearance that they may have lost due to amputations or are missing due to congenital absence [1]. When properly designed, configured and trained, a prosthetic device can play an important role in a patient's autonomy and self-image.

Although many different forms of prostheses exist, they can generally be categorized into two different options for upper extremity amputees: body powered or externally powered devices. Body-powered prostheses, which are mechanically controlled through shoulder and arm movements, were first introduced in the 19th century. Although this option is considered highly durable, body-powered devices may be obtrusive, often require compensatory motions, and can be physically demanding for some users.

Over the last 50 years, externally powered prostheses have been developed to address some of these problems. These devices use electrodes to detect muscle activity, which is converted into a control signal for the prosthesis. The force and speed of movements of the artificial limb are controlled by varying muscle contraction intensity and estimating the amplitude of the electromyogram (EMG). Typically, EMG electrodes are placed over a pair of agonist/antagonist muscles and the difference in EMG amplitude is used to control a device along a degree of freedom (DOF), such as opening/closing a hand. When a user wants to control more than one degree of freedom (DOF), they must co-contract (contract both muscles simultaneously) in order to switch the mode of the device. This method of control has been termed *conventional* or *direct* control.

While conventional myoelectric control has contributed to significant advances in the field, the control is still limited, and co-contractions remain an unintuitive way to switch between functions [2]. This is a factor in the relatively low adoption rate of these types of devices. One of the more widely researched alternatives to conventional control over the last few decades (and only recently commercialized), is the use of pattern recognition (PR) methods to interpret patterns of EMG signals. Instead of using EMG electrodes over an agonist/antagonist muscle pair, PR typically uses many electrodes placed over the residual limb providing more information about muscle synergies. PR can decode EMG to recognize multiple classes of movement intent, and therefore control more than one degree of freedom (DOF) at a time. If the residual muscles are those used to control the missing functions (e.g. the forearm muscles control most hand and wrist functions), control can be restored in a more physiologically appropriate and intuitive manner. Although this is a compelling option, pattern recognition systems require training data for all of the movements to be recognized. These training exemplars must be collected from the user prior to use to provide a baseline for future activity.

In the literature, various methods have been proposed to collect EMG training data for pattern recognition based myoelectric control: the two most noteworthy categorizations are Screen Guided Training (SGT) and Prosthesis Guided Training (PGT) [3] [4]. SGT, as the name suggests, are approaches wherein the individual is prompted using imagery of the desired muscle contraction displayed on a computer screen. The user is asked to contract the corresponding muscles for a specified amount of time, and the EMG data collected over that period are used to train the pattern recognition classifier. With PGT, the user is untethered from a computer and instead follows the motion of their prosthetic device. The process is initiated by the press of a button on their device, which prompts the training process to begin and to move through a sequence of desired motions. As the prosthesis is moving, the user must follow along, performing the corresponding contractions to the movement of the prosthetic device. Similarly, the data collected during this process are used to train the pattern recognition system.

While both training methods fulfill the requirement of acquiring data to train a classifier, performance is heavily dependent on the nature of the data collected. Both of these methods capture only a limited subset of the full range of dynamics of functional prosthetic control, and so the data collected during these types of training may not necessarily be representative of the data observed during regular use.

The incorporation of dynamic training data within a single class has been shown to improve the usability of prosthetic control [5], however proper labeling (associating an instance of data with the corresponding intended motion) of this training data are necessary to properly train the classifier. This requires segmenting the dynamic training data into epochs which correspond to distinct motion classes. Although the inclusion of dynamic data has been shown to improve usability, it is currently not known what constitutes the optimum boundaries for separating the classes in the training data.

Current training methods are typically class-based, wherein users either maintain a steady constant contraction or gradually increase contraction intensity from rest to the desired class [5]. This latter approach has been shown to improve the robustness of control but necessitates the (somewhat arbitrary) segmentation of the desired class from the rest (no motion) class. The general approach to achieve this has been based on a simple globally calculated amplitude threshold for rest, which doesn't incorporate all of the available information from the EMG data (by comparison, once trained, the classifiers employ information from multiple descriptive features).

It is hypothesized that using strategically selected EMG features to determine the best segmentation of the dynamic EMG to label the training data will improve the usability of these devices.

1.1 Objective

The main objective of this research was to understand the impact of different training modalities on the usability of pattern recognition based myoelectric control. By incorporating dynamic motion into the training process, pattern recognition classifiers are given supplemental information that may be more representative of the actual use case as compared to existing training methods. In order to successfully integrate this added dynamic data, however, a better understanding of the effect of different segmentation methods on the classifier performance in both offline and real-time usability tests is necessary.

1.2 Contributions

The findings from this work help provide a better understanding of which method of training is the most appropriate training data for practical PR EMG control. The investigation of the effects of the additional dynamic training data provides insight into where the classifier may have been lacking information or misinformed during other training methods. In addition to investigating the incorporation of additional data, exploring the different segmentation locations provides significant insights into where the optimal separation location occurs between a single DOF for maximum usability performance.

The results obtained for the classification accuracy of each training method helped determine the optimal location for class segmentation. The usability metrics used on the different training and segmentation techniques yield insights about whether real-time control is improved based on different combinations. It is anticipated that refinements to training protocols, and thereby the performance and robustness of PR systems, may improve the adoption and success rates of PR based myoelectric control.

1.3 Outline

This thesis is organized into six chapters. The first chapter briefly introduces the training for pattern recognition based myoelectric control. Chapter 2 focuses on background information, including descriptions of training techniques and segmentation techniques. Chapter 3 describes preliminary investigations, performed to understand the impact of different training modalities on the usability of PR based myoelectric control. Chapter 4 describes the segmentation and training methods chosen based on the findings of Chapter 3 for the main body of research. Chapter 5 describes the experiment and results using the training and segmentation techniques selected in Chapter 4. Chapter 6 includes the discussion and conclusions of the body of work.

Chapter 2. Background and Related Work

For those with an upper limb amputation, a prosthetic arm can be used to encourage body symmetry and restore some function of the missing limb. Externally powered prosthetic devices typically use the myoelectric signal (or equivalently, the EMG) as a control source [6]. Because of the stochastic nature of EMG, however, some means of decoding movement intent is required to extract a control signal [6].

Conventional approaches to myoelectric control rely on varying the level of the myoelectric signal to modulate the velocity of the terminal device. Two directions along a degree of freedom (DOF) can be realized by using the difference in amplitude between two electrodes placed over antagonist muscle sites (e.g. the flexor and extensor muscles of the forearm) [7]. Although a prosthesis may be comprised of different components (such as a terminal hand and wrist rotator), conventional control approaches are limited to controlling one action at a time due to the lack of independent control sites. To control more than one action/device, users must switch modes by either co-contracting, eliciting a double impulse, or by using some external switch or sensor [8]. The integration of microprocessors in myoelectric control [9] has allowed the development of more sophisticated control schemes. Amongst these, pattern-recognition (PR) based myoelectric control is the most thoroughly researched and has been purported to address many limitations of the conventional control approaches [2].

2.1 Pattern Recognition based Myoelectric Control

Pattern recognition is used in a variety of fields, such as medical sciences, psychology and e-commerce [10] [11], with the aim of finding and recognizing patterns in data using labeled (supervised) or unlabeled (unsupervised) data and assigning them into different classes.

Pattern recognition of biological signals can generally be described by the flowchart shown in Figure 2.1. First, data are collected using a sensor and are then preprocessed and windowed to obtain meaningful segments. Once the data has been collected, relevant and important features are extracted from the raw signals within each window to distinguish these records from one another. The data are then categorized into distinct records based on the intended target during data collection. If the intended target is unknown, an extra classification step is required to separate the data into different records. Finally, a classifier is trained with these features and corresponding labels to provide a baseline comparison for all future data that will be collected. Any new data will be categorized into their respective record based on the features that were extracted. Once the data is assigned a label, various forms of post-processing such as filtering may be required before a final classification is made.

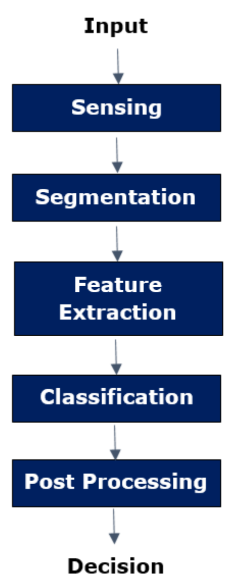

Figure 2.1 - Pattern Recognition Workflow

Pattern recognition based myoelectric control works in a similar way as described above. To distinguish between muscle patterns and properly segment the raw EMG data into different records, training data for all desired movements are collected using electrodes placed on the user to provide a baseline for future activity. Because the target motion is known during data collection, the baseline data can be separated into their respective motions. With this segmented data, relevant features are extracted and used to train a classifier for future use. After training, the classifier is often tested to determine how well it correctly identified the correct movements.

Typically, for PR based myoelectric control, multiple EMG electrodes are used. This helps provide the classifier with more spatial information so that it can be used to control more than one degree of freedom (DOF). Importantly, PR facilitates the use of physiologically appropriate contractions, wherein the user activates the muscles of their residual limb in a way that is consistent with the desired motion of the terminal device. This has the potential to reduce cognitive burden and increase embodiment; by having a device that moves in the same way the muscles are contracted, this offers amputees a more natural, intuitive way of controlling their prosthesis.

These recent advancements in pattern recognition based myoelectric control have resulted in devices that are capable of a greater number of functions than conventional control methods. While this approach to prosthetic control has demonstrated the ability to discriminate between many degrees of freedom and provide more natural muscle contractions to activate motions, it has yet to receive widespread clinical acceptance[12].

A study by Glynn et al. [13] noted that 91% of myoelectric prosthesis rejections resulted from prosthetic related problems involving comfort, lack of function and durability. As noted by Bongers et al. [14], to address the lack of functionality, these devices require more robust myoelectric control, putting a high demand on the training of the users.

While the objective of this work is the improvement of training for pattern recognition based myoelectric controlled prosthetic devices, it is important to first understand the various components that make up such a system.

2.2 Classifier Training Protocols

In order for a pattern recognition based myoelectric control system to estimate which movement a user is trying to elicit, it must first be trained using data acquired from that specific user. This is due to the wide variation between individuals in musculature, tone, electrode location, and the naturally stochastic nature of EMG. The quality of the EMG data acquired during training is, therefore, crucial to the performance and robustness of the control. While quality can refer to signal characteristics (such as signal to noise ratio), here it is used to denote the ability of the training data to inform the classifier of the true class boundaries and the variability that will be encountered during practical use of the device. Toward this goal, we now present and compare the two most commonly used training methods to investigate their effect on the usability of prosthetic devices.

2.2.1 Screen Guided Training

The most common way that data for PR based myoelectric control is collected in a research setting is using some form of graphical computer interface. In this method, referred to here as Screen Guided Training, users are prompted with specific classes of muscle contractions for a desired amount of time. Figure 2.2 demonstrates a screen in the

ACE software package developed at the Institute of Biomedical Engineer [15] that could be used during SGT where typically several different motions are collected to train a classifier.

Figure 2.2 - Screen Guided Training Example

One of the major drawbacks of this training method is the need for an external device such as a computer. If a patient's prosthesis becomes uncalibrated, which is common due to sweating or fatigue, the individual may not be able to recalibrate their device if they are not by a computer.

2.2.2 Prosthesis Guided Training

Prosthesis Guided Training prompts the individual by using the prosthetic device itself. The device moves through a series of motions and the users must contract the corresponding muscles to perform that movement. In order to segment the transient data into different classes, an amplitude threshold is first used to find the active motions (any motion other than no motion). A concern is that this type of training requires the user to follow the velocity of the device, in the same way that they would control it. While velocity control is important for use with multiple sequential degrees of freedom, this has been observed to be somewhat unintuitive as a prompt. A partnership between the Institute of Biomedical Engineering at UNB and Coapt (LLC), a company that offers a commercial PR controller that employs PGT, has been established to determine the needs of their product. Powell *et al.* [16] have shown the importance of being able to train a device from home, and while PGT fits this criterion, the developers at Coapt are concerned that it currently has no error handling or intelligence built in, nor any feedback system. The system and the users have no ability to know if they have correctly followed, and thereby, properly trained their device.

These current training methods collect either static data or transitions between an inactive motion and an active motion [17] [3]. Static methods of training wherein a contraction is held constant for the duration of the prompt, are relatively easy to classify. With the incorporation of dynamic motions, the boundaries for transitions between inactive motions and active motions become somewhat arbitrary. A common method to locate the location for segmentation for this type of training is by incorporating a threshold for activation - if this threshold is reached, data is classified as an active motion, and all other data are considered to correspond to an inactive motion.

So far, training methods that transition between two active motions within a degree of freedom (DOF) have not been explored. The incorporation of the supplemental information created during transitions may be more representative of the actual use case and improve robustness compared to existing training methods. To successfully use a DOF based training method, the proper location for segmentation will need to be found. The use of a threshold for activation would not be able to extend to this type of training as the user may be changing from one class to another in a continuous sense, therefore other types of segmentation will be explored to identify the most suitable locations for segmenting DOF training data.

2.3 Segmentation and Clustering

In many applications, such as EMG PR, data must be split up into segments with common statistical properties in order to classify unlabeled data; this action is called un- or semi-supervised data segmentation. Many different data segmentation techniques have been used in a variety of different fields [18] [19] [20] [21] to characterize patterns in signals. A variety of methods will be investigated in Chapter 3 to assess their efficacy in segmenting EMG signals for PR based control. A natural source of inspiration are the methods for segmentation that have been proposed for other biological signals such as the electrocardiogram [22] [23] and electroencephalogram [24] [25].

While static contractions during Screen Guided Training have an associated target to each record of training data, the introduction of ramp contractions (starting from rest and increasing intensity towards the desired state) [5], necessitates the inclusion of additional segmentation logic. Extending from single-class ramp contractions to the degree of freedom-based training, which includes dynamic transitions between two classes, further complicates this process. To effectively use dynamic training data, segmentation must be performed in order to separate the classes within a degree of freedom. Data segmentation of time series data leverages statistical properties in the data but does not accommodate for user errors (e.g. additional transitions if they were to occur). Therefore, it is also important to consider clustering techniques, which group data into a number of groups where each new data point is assigned to a group that is most similar. As with segmentation, data clustering is broadly used in many applications, but those of interest for this study are those that cluster EMG signals.

2.3.1 Supervised Classification

When using a supervised method of classification for pattern recognition, the target for the training data are already known. This typically applies to applications where some prior knowledge of the system exists, or where some expert can provide a label to a training set. Static SGT is a supervised method of classification since the target and timing of each motion provided to the user is known as the EMG data are captured. Because the aim of this study was to incorporate additional dynamic training data, the knowledge and timing of changes in contractions are far less deterministic. Although a cursor may be used to prompt the user to change motions, there may be inconsistencies in the delay between the prompting and the human response. This uncertainty in the actual data boundaries, therefore, motivates the exploration of data-driven, unsupervised methods.

2.3.2 Unsupervised Classification

Unsupervised methods of classification have unknown targets, which requires data to be clustered into different groups that are statistically separable before being trained. Unsupervised learning is often used as a synonym for clustering [26], which is modeled using statistical similarities. An example of an unsupervised learning application is online advertising. Groups of similar customers are formed by identifying the content of previous web browsing and purchase histories, and then targeted advertisements are displayed depending on which group the customer is assigned [27].

For unsupervised segmentation of times series data with slow changing classes, data clustering based on different measures of temporal similarity or distance can be performed. A common clustering technique for PR is k-means clustering [28]. K-means is a simple unsupervised algorithm that classifies data into a predetermined number of clusters, *k*, by minimizing a distance function. Points are assigned to a cluster based on their proximity to the nearest class center.

One variation of k-means is the fuzzy c-means clustering algorithm [29] which assigns data points to clusters with a "membership" score. This is useful for overlapping data; a perfectly distinct cluster point yields a membership of 1, although data points can be assigned to more than one cluster with smaller membership scores. Kernel k-means, another variation [30], projects the data into a higher dimensionality space before calculating boundaries. This technique is advantageous over the standard k-means, which is limited to locally linear hyper-plane boundaries.

Aligned Cluster Analysis (ACA), an extension of kernel k-means [31], is intended for unsupervised clustering of temporal patterns. The difference between ACA and kernel k-means is that the feature vector contains a variable number of features for each cluster (not a set vector), and a dynamic time warping kernel is used to achieve temporal invariance. Hierarchical Aligned Cluster Analysis (HACA) [32] divides data into clusters over different levels of hierarchy. As an example, in gait segmentation, the first level would be running/walking. The second level takes the first level and divides it into a smaller temporal scale; heel lift, etc.). Another popular unsupervised learning approach is

Hidden Markov Model (HMM) based sequence clustering [33], which builds features such that each sequence of states is represented by a vector of its similarities to a predefined set of reference sequences.

Other adaptive segmentation methods such as modified Varri [34], the generalized likelihood ratio [35], and short-time Fourier transform [36] have also been studied in other fields, but these have been shown to have low efficiencies and/or computational high burden [37] [38] [39] [40] [41].

2.3.3 Manual Segmentation

While unsupervised clustering is an automated way of labeling unknown data, its performance in dynamically varying time series electrophysiological data such as EMG is still relatively unknown. To be able to parse this dynamic data to evaluate the effectiveness of various training approaches, manual segmentation techniques must also be considered. Amplitude-based segmentation may be used to separate data based on a pre-defined baseline amplitude threshold. As in automated methods, once the amplitude of the signal surpasses this threshold, the data are segmented into this respective class (for example, from no motion below the threshold to an active motion above the threshold). In human manual labeling (or visual inspection), the time series and contextual information may be incorporated to inform the selection of local thresholds. Manual segmentation may also be performed in higher dimensional feature space using dimensionality reduction techniques to enable visual inspection of thresholds. While manual/visual segmentation will be used in aspects of this work to establish a baseline for other segmentation approaches, it is important to note that it is both time-consuming to perform and unfeasible for practical use.

2.3.4 Distance and Similarity Measures

For many different segmentation and clustering techniques, similarity or distance measures are needed to determine how similar or different two data points, or groupings of points, are from each other. For unsupervised clustering techniques, since the true target is unknown, the distances separating different data may be used as a guideline. While there are many distance measures, a selection of the most commonly used in the field are identified below.

The Euclidean distance metric [10] measures the multi-dimensional straight-line distance between two points and can be described below in Equation 1.

$$d = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{1}$$

The Euclidean squared metric is the same as the Euclidean distance metric but does not use the square root, which allows for faster clustering than regular Euclidean distance.

Other metrics used for clustering include the correlation metric, which measures the statistical dependence between two different points [10]. This metric is derived from distance variance, distance standard deviation and distance covariance and measures how correlated the two variables are on a scale from -1 to 1. The Bhattacharyya distance measures the similarity of two probability distributions and is used to measure the separability of classes [10]. The Mahalanobis distance [42] is a specific case of the Bhattacharyya distance when the standard deviation of the two classes are the same.

While distance measures can be used for clustering purposes, other similarity measures can also be used to distinguish between patterns in signals. A matched filter can be used by defining a template and then finding the maximum-likelihood of identifying the common elements of this template in the unknown signal [10]. Principal components analysis [43] is a statistical procedure which is used to identify the maximum amount of variance and constructs new characteristics to summarize data; with a reduction in dimensionality, features from the signals are much easier to work with without losing information from the original signal.

2.4 Feature Extraction

Because EMG is a noisy and stochastic signal, patterns in limb movements are most often described by a set of features that represent useful information while discarding irrelevant noise in the signal. The choice of which features to be used has a significant impact on the performance of the classifier as they are being used to discriminate between different classes [44]. In the literature, time domain (TD) features [45] are the most widely used features for EMG since they do not require any form of transformation, and thus require less computational load. In particular, Hudgins' TD feature set: mean absolute value, wavelength, zero crossings, and slope sign change has been the most heavily utilized over the last 15 years [46]. The addition of Wilson amplitude has been shown to further reduce the average classification error [48].

In a study conducted by Phinyomark et al. [49], 50 different features (including time domain, frequency domain, and time-frequency domain) were compared. TD features achieved the highest accuracy with a Linear Discriminant Analysis (LDA) classifier, described in the subsequent section, and also showed a better performance than time-frequency domain features in the classification of both transient and steady-state EMG signals using LDA.

Due to the stochastic nature of myoelectric signal values, windowing of the raw EMG data is required before extracting features. Englehart and Hudgins [50] discovered that overlapping analysis windows in the continuous stream of data improve the decision density while averaging these decisions improves classification accuracy.

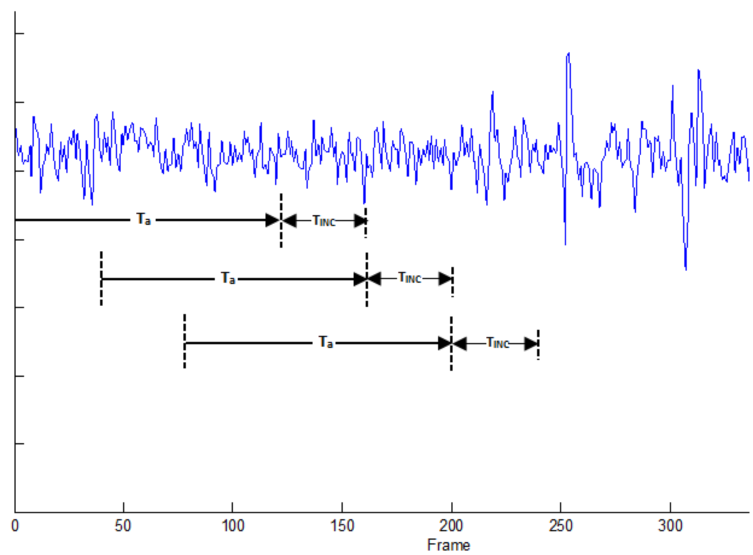

Figure 2.3 – Windowing of Raw Data using Overlapping Windows

As shown in Figure 2.3, the length of the window, T_a, is used to determine the amount of data to be used for feature extraction to produce one class decision, while the frame increment, T_{inc}, increments the N-sample window by a specified frame length. A study by Hargrove et al. [51] investigated the effects of different window lengths on pattern recognition based myoelectric control and found that there is a tradeoff between classification accuracy and controller delay: a too short window increases classification error, however, a too long window produces excessive and noticeable delays. In their work, they found the optimal window length for best performance ranges from 150-250ms as the window length. For each window, the features are then extracted from the training set and provided to a classifier to be used for training.

2.5 Classification

A classifier is used to predict the class membership of input data based on its features. For myoelectric control, it is important for the classifier to make the correct decision for the prosthetic device to make the proper movement. A misinformed or improperly trained classifier that inadvertently opens a prosthetic hand, for example, could be dangerous if a user was holding a hot cup of coffee.

Studies have found there to be little appreciable difference in the classification error of different classifiers such as Artificial Neural Networks (ANN), Hidden Markov models (HMM), Linear Discriminant Analysis (LDA) and Support Vector Machines (SVM) [52] [53]. The LDA is a Bayesian classifier that is simplified by assuming that the data are Gaussian and that all classes share a common covariance. The result is a set of classes boundaries that are linear that yield a maximum likelihood for the selected class. Even if these conditions are not strictly met, the LDA is often almost as accurate in myoelectric control as more complex classifiers [48] [54]. Moreover, because of its good balance between efficiency and speed, LDA is the classifier most commonly used in the field and will be adopted in this work.

Once a classifier is trained, its performance can be evaluated by measuring the Classification Accuracy (CA):

$$CA = \frac{Correct\ classified\ samples}{Total\ number\ of\ samples} * 100\% \quad (2)$$

2.6 Usability Assessment

While offline accuracy has generally been used to describe the metrics for how well the training data has been classified, it has been shown that this is a poor predictor for actual usability [55] [56] and is not a true measure of real-time function. To this end, researchers have investigated other ways of testing the usability of EMG control strategies to examine the clinical robustness and accuracy of pattern recognition control.

There are many functional tests available to assess the usability of myoelectric prostheses [57] [58] [59], but most of these require a physical prosthesis. Virtual, computer-based, environments have consequently been developed and used to efficiently test control schemes using able-bodied and amputee subjects [2] [17] [60]. These virtual environments are used to create a user-friendly and monitored environment where real-time motion testing for control performance can be assessed. They can consist of steering a computer avatar in multiple directions [2], controlling a cursor to hit targets on a computer screen [2], or moving a virtual arm into a target posture [61].

Many different virtual test protocols have been employed to assess the performance of real-time myoelectric control. Kuiken et al. developed the Motion Test [17] and evaluated functional performance by prompting subjects to move a virtual prosthesis through a range of motions for one DOF at a constant speed. A more challenging Target Achievement Control (TAC) test was developed by Simon et al. [61] which incorporated proportional control of the speed of the virtual limb movement based on the intensity of the muscle contraction. This environment involves positioning of a virtual limb in an initial target position and back to a neutral position and can test different difficulties by modifying the number of DOFs required to position the virtual limb as well as the allowable error in position. While this test provides informative performance metrics, it is limited by difficulties in understanding the visual cues required to identify the correct movement to elicit. The TAC test is a close analog to Fitts' Law, which has recently been validated to be suitable in evaluating EMG control [62].

2.6.1 Fitts' Law

Established in 1954, Fitts' law describes a tradeoff between speed and accuracy in ballistic target acquisition tasks by relating the time taken for controlled movements to the difficulty of the intended target. According to Fitts' Law, the time to move to a target with index of difficulty ID can be described as:

$$MT = a + b * ID \tag{3}$$

ID is calculated as:

$$ID = \log_2\left(\frac{2D}{W}\right) \tag{4}$$

where "MT" is the movement time (seconds), ID is the difficulty of the target, "a" and "b" are device dependent constants, "D" is the distance to the center of the target and W is the width of the target. In 2008, the validity of using EMG as a control source was proven by Park et al, and Fitts' law style testing for the evaluation of myoelectric control strategies has since been used extensively [2] [62].

2.6.2 Other Usability Metrics

Quantifying the ability to predict the correct action is necessary to determine how well a classifier has performed. As discussed, however, the offline classification accuracy may not be the best metric to predict usability, as it provides only partial information about the control scheme. Several other metrics are generally used to evaluate the performance of pattern recognition based myoelectric control during virtual tests. The *throughput* describes the control bandwidth in bits/second (information transfer rate) by comparing the movement time and the difficulty of the task, according to Fitts' law. In addition to the throughput, Williams and Kirsch [64] also identified several other metrics that provide more comprehensive information about control performance; these include the path efficiency, completion time, completion rate and overshoot.

The *path efficiency* is the ratio of the shortest possible distance possible to the distance traveled by the user. The *completion time* is the amount of time it took for the cursor to reach the target, while the completion rate is the ratio of completed targets over the total number of targets. The *overshoot* defines the average number of times the cursor exited a target after being acquired. Although these real-time tests provide more information about the usability of the system, they cannot be used in an offline sense to directly compare different algorithms (because the control was influenced by the feedback to the user in the closed-loop system). Furthermore, real-time tests, such as Fitts' law, give the user flexibility in how they approach the target, making it difficult to understand the 'true' class for any given decision. Because offline and real-time performance metrics contain different information, it is therefore important to use both measures when evaluating the performance of a system.

Chapter 3. Training Modalities

The main objective of this research was to understand the impact of different training modalities on the usability of pattern recognition based myoelectric control. It was theorized that, by incorporating dynamic motion into the training process, pattern recognition classifiers would be given supplemental information that may be more representative of the actual use case, resulting in improved performance or robustness. The training methods developed through this work were developed using offline analysis methods and ultimately compared in a usability test to determine whether the addition of dynamic data during training improved the real-time performance.

It has been shown that capturing transition data by way of varying contraction intensity (such as through "ramped" contractions) can be beneficial to the performance of pattern recognition-based myoelectric control [2]. Without this induced variability, as seen in Figure 3.1 (showing the first two dimensions in ULDA projected feature space), constant-force contractions result in highly separable and repeatable classes of motion. Although this may perform well during constant-force contractions, resulting in high offline accuracies, this leaves the classifier unaware of the variability that occurs during transitions from one class to another, as required in real-time myoelectric control.

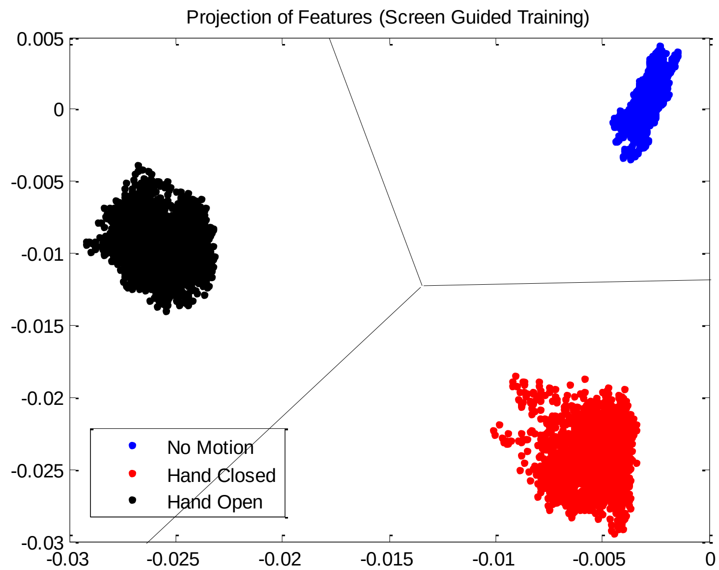

Figure 3.1 - Example of highly separable classes collected using 'offline' static contractions Data are presented as the first two dimensions of a ULDA dimensionality reduction projection.

By comparison, Figure 3.2 shows the same three classes of motion when trained using ramped contractions. The features are less repeatable than with static training, but the classifier is now informed by features generated as the user transitioned from rest to the motion. By including this additional information, a more informed decision can be made by the classifier, ultimately improving the robustness of the controller.

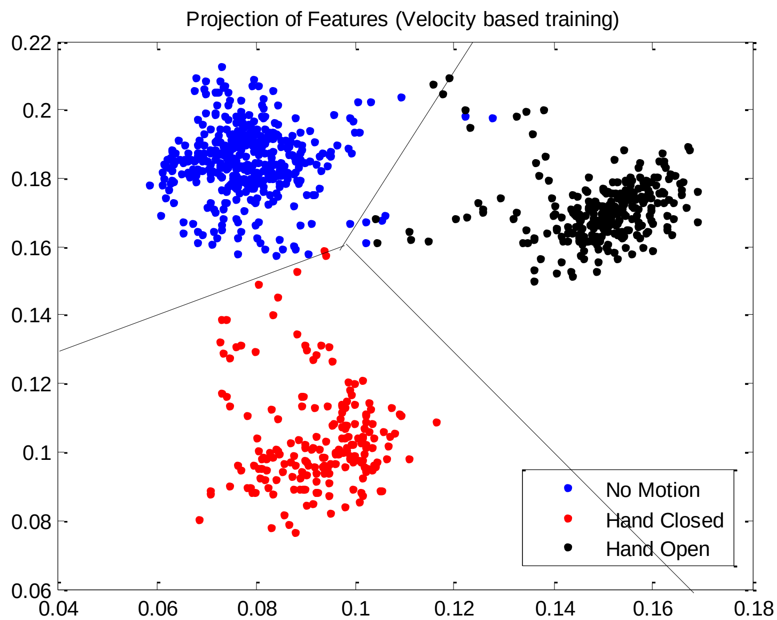

Figure 3.2 - Example of classes collected using 'offline' ramped contractions. Data are presented as the first two dimensions of a ULDA dimensionality reduction projection.

Even with the ramp training method, however, the system is not capturing the full transition between active degrees of freedom, or the return to rest afterward. This has been observed, anecdotally, to lead to inadvertent activations during transitions, and overshoot of virtual targets during screen-based usability testing. The motivation of this work was to better inform the training of classifier boundaries by including these additional transition data that may be more representative of those elicited during functional use and to investigate whether doing so improves the usability of the classifier.

Another important consideration in the collection of training data is the burden on the user and the intuitiveness of the prompting method. User errors, which can occur when a user strays from the prompted motion or timing, can be catastrophic, as they result in mislabeled data being used to inform the learning algorithms. Similarly, errors in the segmentation and labeling of continuous training data, as will be investigated in the coming sections, have a comparable impact. Consequently, it is of interest to quantify the impact that such errors may have on classifier performance.

3.1 Study 1: Simulated Mistakes

Investigating the effects of user error through simulation gives insight into how much mislabeling of training data will impact accuracy. To better understand the effects of improperly labeled data on classifier performance, two different common user errors were simulated using data collected from 15 subjects for five different motions: no motion (NM), wrist pronation (WP), wrist supination (WS), hand open (HO) and hand close (HC).

3.1.1 Experimental Methods

Eight surface UNB Smart bipolar electrodes mounted in a flexible cuff, shown in Figure 3.4 (left), were positioned around the dominant forearm of 13 able-bodied participants (9 male and 4 female, mean age 25 +/- 2.2 years). Subjects were prompted to elicit contractions corresponding to 5 different hand/wrist motions, shown in Figure 3.4 (right) using a custom-designed computer interface (an example of Screen Guided Training). Each contraction repetition was held for 4 seconds and repeated 8 times.

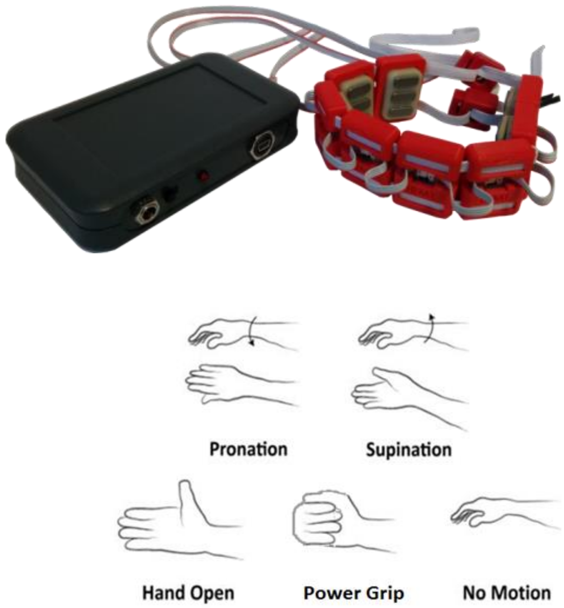

Figure 3.3 - UNB's EMG cuff (left), and classes prompted during Experiments 1 and 2 (right)

All subjects gave informed written consent, and the experiments were performed at the Institute of Biomedical Engineering at the University of New Brunswick. This experiment was approved by the University of New Brunswick's Research Ethics Board (REB 2015-134) in accordance with the Canadian Tri-Council Policy Statement regarding ethical conduct for research involving humans.

Data Analysis

Offline analyses were performed using MATLAB 2009a through a graphical interface developed at UNB [11]. Data were segmented into 160ms windows with a 16ms frame increment. Five EMG time-domain features (mean absolute value, number of zero crossings, waveform length, number of slope sign changes and Willison amplitude) were extracted from each data window from each channel of EMG and were used to train and test an LDA classifier.

3.1.2 User Delay

The first scenario that was investigated was one that simulated the effect of delayed reaction time. The training set features remained the same but were artificially delayed with respect to the targets. An LDA classifier was trained with labels corresponding with these targets and tested with the original unshifted test data set. Figure 3.5 illustrates the original test data with the corresponding targets. Figures 3.6 and 3.7 depict the training set shifted by 50% of a motion (corresponding to 2 seconds) and 100% of a motion (4 seconds) respectively.

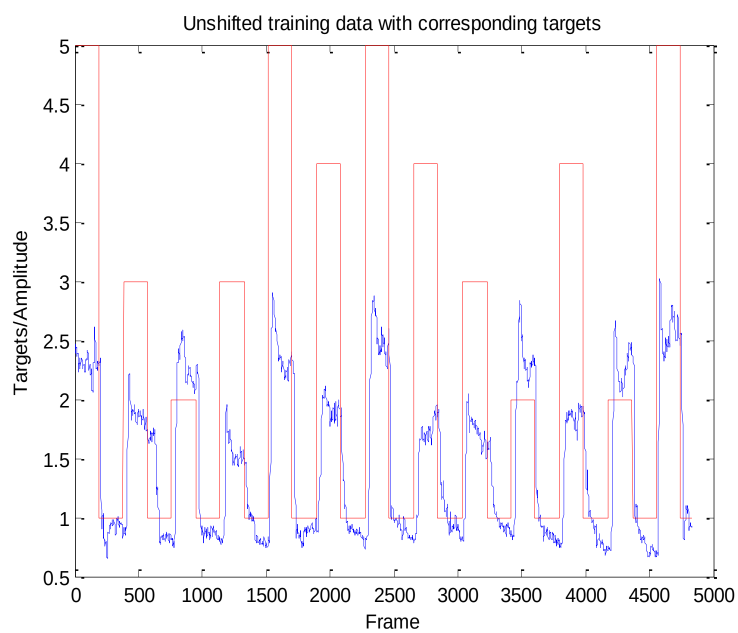

Figure 3.4 - Example of the alignment between the unshifted training targets and user response (Target with value 1: NM; 2: WP; 3: WS; 4: HC; 5: HO)

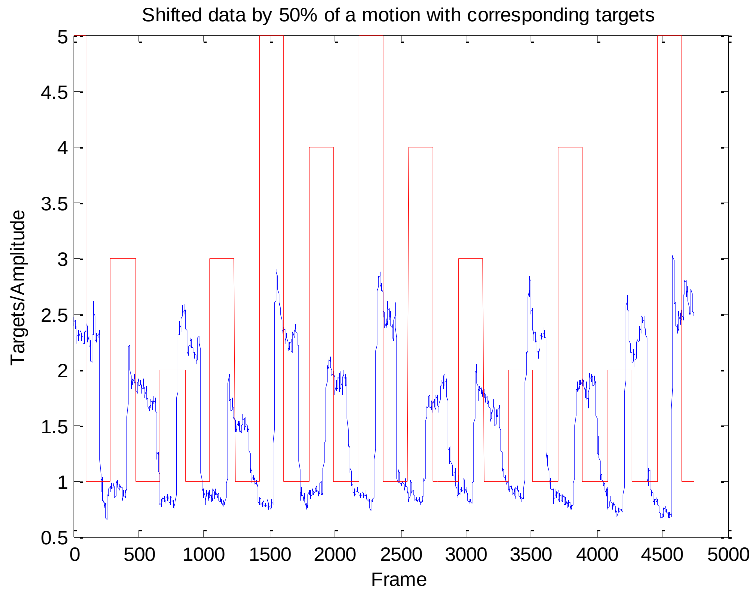

Figure 3.5 - Example of the alignment between the artificially delayed training targets (50%) and user response (Target with value 1: NM; 2: WP; 3: WS; 4: HC; 5: HO)

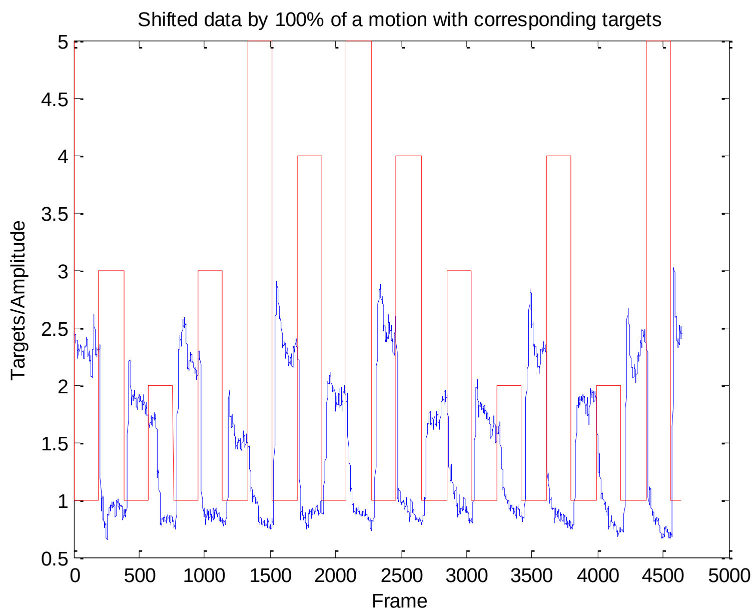

Figure 3.6 - Example of the alignment between the artificially delayed training targets (100%) and user response (Target with value 1: NM; 2: WP; 3: WS; 4: HC; 5: HO)

As shown in Figure 3.7, in the extreme case when shifted by 100%, the active classes are labeled as no motion, and no motion data is labeled as an active motion. This was repeated, training a classifier for each possible shift in the training data (from 0 to 100%), and then tested on a novel test data set. As can be seen in Figure 3.8, the accuracy drops below chance after a shift of 70%, because the labels are now systematically incorrect, not just random.

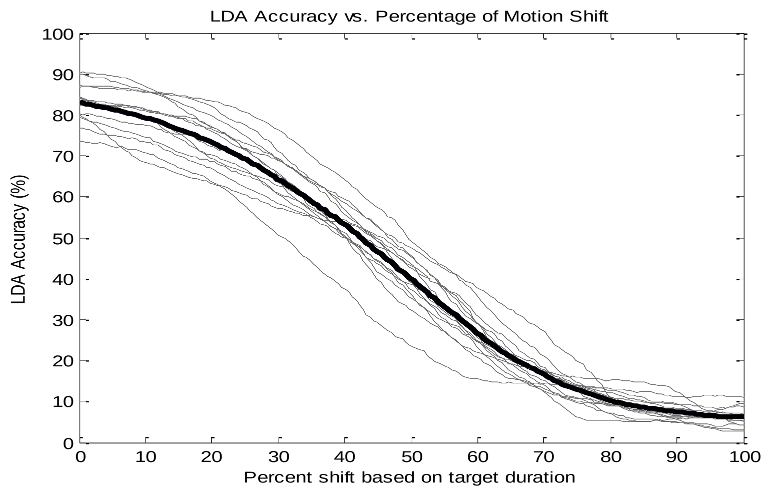

Figure 3.7 - LDA Accuracy as % Shifts (Mean Across Subjects Indicated in Bold)

The results of artificially delaying training data demonstrate the importance of properly aligning training data. As shown above, the correct labeling of targets has a high impact on classification accuracy because of the data-driven approach.

3.1.3 Activation Errors

Another common mistake is when a user accidentally elicits the wrong motion during training. This sometimes occurs when a user tries to anticipate the next motion or their attention wanes. To simulate this error, training set features were replaced with features from a different motion (while keeping the original targets). Figure 3.8 shows data and targets corresponding to three classes: no motion, wrist pronation, and wrist supination. In this offline simulation, one repetition of wrist pronation was completely replaced with features from an extra wrist supination repetition, as shown in Figure 3.9. This would be similar to someone not paying attention to the prompt and eliciting the wrong motion.

Figure 3.8 - Training Repetition with Simulated Incorrect Motion

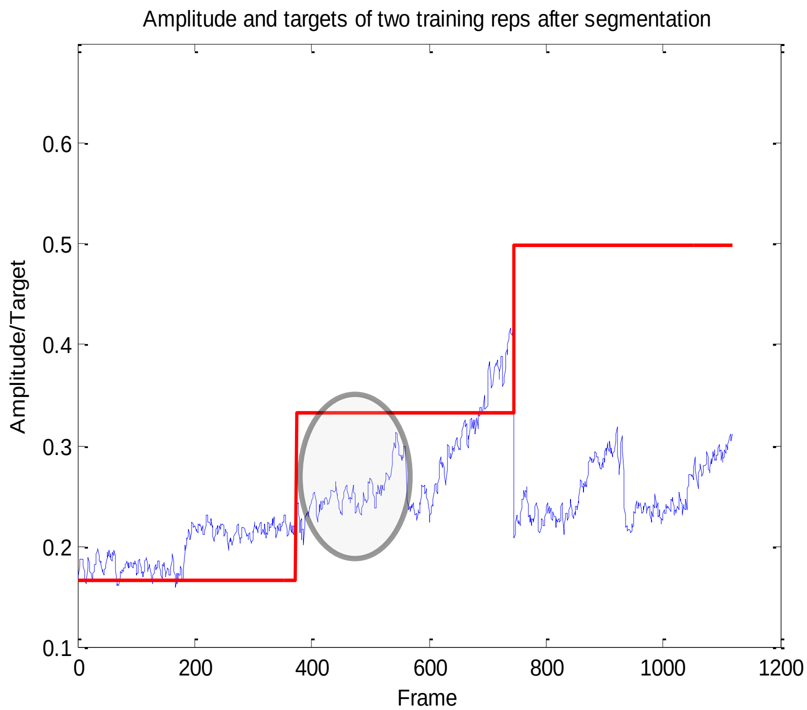

Figure 3.9 - Unaltered (Correct) Training Repetition

When observing the unaffected data (projected down into a two-dimensional feature space for clarity), a clear separation between the three classes can be seen (Figure 3.10). However, once the data have been replaced, the features (as seen in Figure 3.11) are not as separable as they were, and the boundary lines are changed. This change has a significant effect on data classification as demonstrated in Figure 3.12, as different amounts of the motion are swapped out.

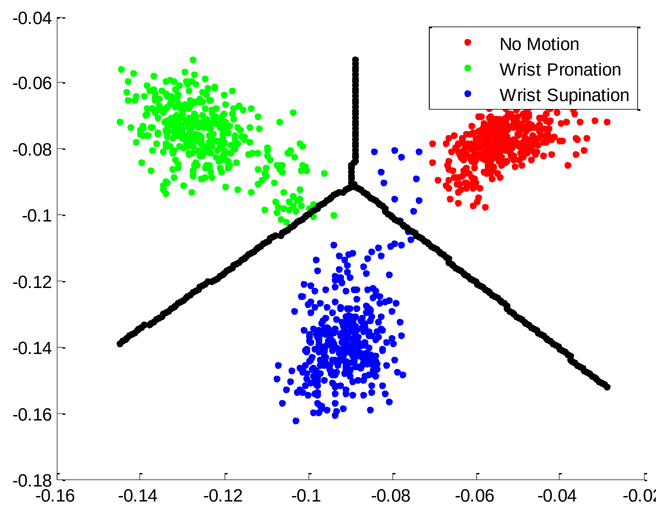

Figure 3.10 - Features and Computed Boundaries for Correct Repetitions

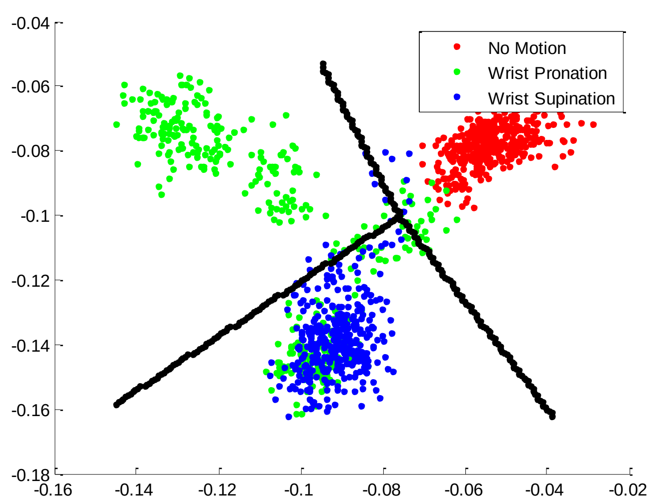

Figure 3.11 - Features and LDA Boundaries with Incorrect Repetition

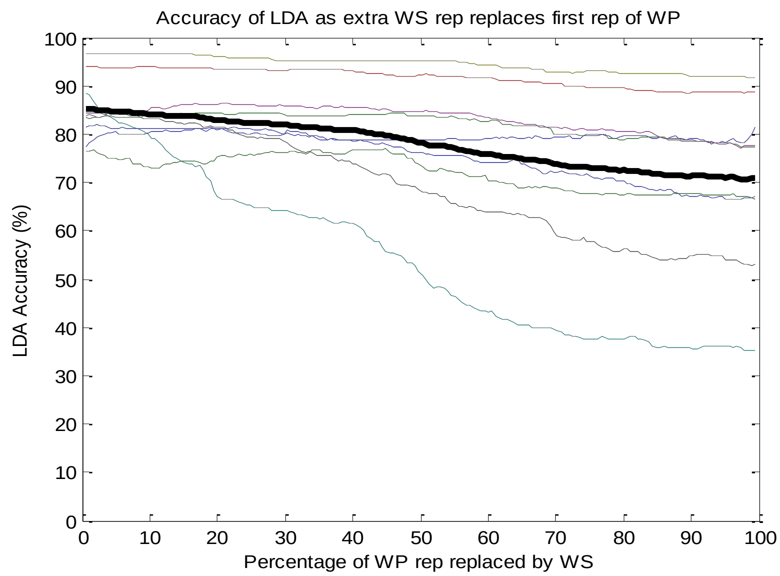

Figure 3.12– Change in LDA Accuracy with Increasing Contraction Type Error (Mean Across Subjects Indicated in Bold)

It is shown that, as the percentage of replacement of a motion increases, the LDA accuracy decreases. Similar results were obtained when observing the hand open/hand close DOF, and performance would degrade even further if more that one rep was replaced.

The results from these erroneous data demonstrate the importance of an intuitive method for training. By having an intuitive training method, the user will be less likely to make common mistakes during training, and the chance of eliciting the improper class would be reduced. This example also demonstrated the weakness in amplitude-based approaches to segmentation; if the wrong motion is elicited, the amplitude remains above the threshold for inactivity and these data will be incorrectly used during training.

3.2 Study 2: Segmentation

As previously investigated, activation errors and user delays have a significant impact on classification accuracy when segmenting based on known targets. To overcome these impacts, it is proposed to use segmentation or clustering techniques based on statistical properties to account for potential user errors that often occur during training. Furthermore, the incorporation of dynamic data during training, which may improve the robustness of the control scheme, also requires segmentation or clustering techniques as the targets for the transition motions are unknown. Multiple segmentation and clustering techniques will, therefore, be compared to determine which approach yields the highest classification accuracy for different methods of training.

In the second study, offline data were collected and were used to evaluate the effect of different segmentation locations on a classifier for four different training methods with the purpose of determining the optimal segmentation locations for classification.

In the remainder of this chapter, a number of segmentation approaches are compared with their relative performance assessed using offline data. This was necessary as a real-time usability study would require repetition of the experiment for each segmentation approach, for many subjects. The offline results were used to inform the "best candidates" for the real-time study described in Chapter 4.

3.2.1 Experimental Methods

The same flexible cuff used from the first study (Simulated Mistakes) was fit on fourteen different able-bodied subjects (9 male and 5 female, mean age 25 +/- 2.1 years, 13 right-handed), to collect EMG data for the same 5 classes of interest.

Four different training sets and one test set corresponding to the SGT and PGT paradigms, as described in further detail below, were acquired from each subject. Each of the methods was used to record multiple repetitions from each of the five classes (wrist pronation, wrist supination, hand open, hand closed and no motion). For all trials, subjects were seated comfortably in a chair with their elbow rested on an armrest. Before data collection began, subjects were given instruction on how to perform each of the five contractions for best separability for each of the following training methods.

Screen Guided Training

The screen guided training method prompted users with an image of the desired muscle contraction for four seconds, as seen in Figure 3.13. During this training session, participants were instructed to slowly activate the desired contraction while the system was recording. This process was repeated for each of the five classes, and a total of six repetitions of data was collected for each of the users.

Figure 3.13 - SGT Sample Interface

This process was then repeated with constant static contractions during the recording of another six repetitions. This test set was collected for use in the calculation of the classification accuracy for all training sets. Since static contractions are in theory perfectly segmented into their desired motions, these data were used to test and compare the proposed segmentation locations.

Prosthesis Guided Training

To emulate the velocity-based training (VBT) employed by PGT, a cursor was shown on a computer screen (represented by a horizontal bar, seen in Figure 3.14) for the participants to follow. The users were asked to follow the *velocity* of the cursor as it moved left and right according to a sinusoidal target. An important consideration is that whenever the bar stopped, the user had to stop contracting, regardless of where the position stopped. This was found by many subjects be counterintuitive, especially when the cursor slowed down at the limits of its excursion.

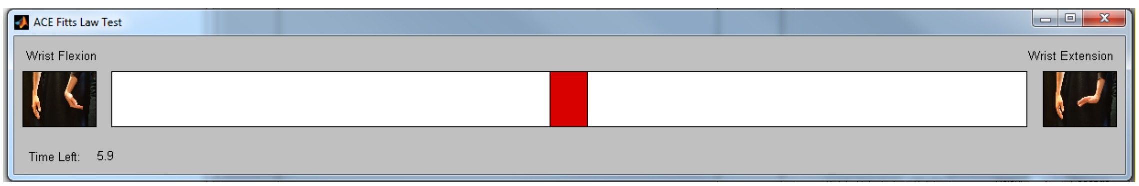

Figure 3.14 - Example of the PGT emulation interface

Novel Method: Position-Based Training

A novel training approach (position-based training; PBT) was employed using the same graphical interface as the previous training method. The only distinction was that users were asked to follow the *position* of the cursor. Although this differs from the eventual usage scenario (myoelectric control employs velocity-based control), it was theorized that the users would be more able to intuitively mimic the position of the cursor with their limb, possibly reducing user error. By having the users follow the position of the bar, they could also be prompted to elicit the full range of transition data, as opposed to going solely between active classes and no motion.

Novel Method: Position-Reset Training

The final training method was a position-based training method with a return to zero. In this training, Position-Reset Training (PRT), the same cursor as the two previous methods appeared on the screen, and the users were asked to follow its *position*. In this case, after the cursor reached the extremities, it immediately returned to the center, which prompted the participants to rest their muscles. This training set was designed to effectively capture the same data as with Prosthesis Guided Training, however, participants were able to think about following the position of the cursor as opposed to its velocity.

3.3 Segmentation

Because each of the training approaches included continuous transitions, the training data required segmentation prior to being used to train a classifier. A number of segmentation approaches were compared and evaluated using offline classification accuracy. The results from these segmentation techniques were used in a future experiment to further evaluate the training methods through a usability test.

3.3.1 Amplitude Segmentation at the Valleys

The average Mean Absolute Value (MAV), the mean of the mean absolute values of windows of all eight EMG channels, monotonically increases and decreases with contraction intensity [65], with slow transitions between active contractions typically exhibiting a brief transition through the no motion class. Correspondingly, this method of amplitude segmentation assumes that transitions between classes occur at the valleys (the lowest amplitude between the peaks of the active motions). Figure 3.15 shows an example of unsegmented average MAV, while Figure 3.16 demonstrates how these data were segmented through a visual segmentation process completed by a human observer. Note that any sections of the data that fell below a simple amplitude-based activation threshold derived from the no motion class (by calculating the mean and adding two standard deviations of the first 1.9 seconds of motion data collected) were also labeled as no motion.

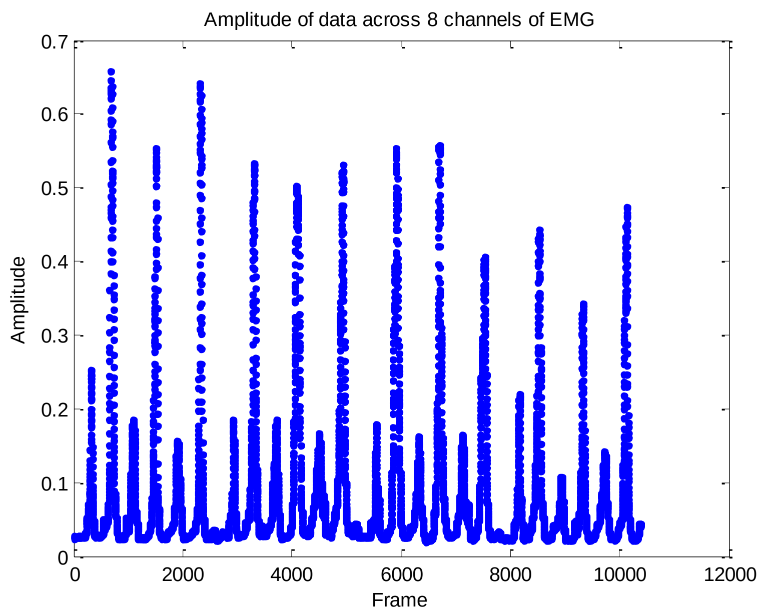

Figure 3.15- Unsegmented Average MAV of 8 channels

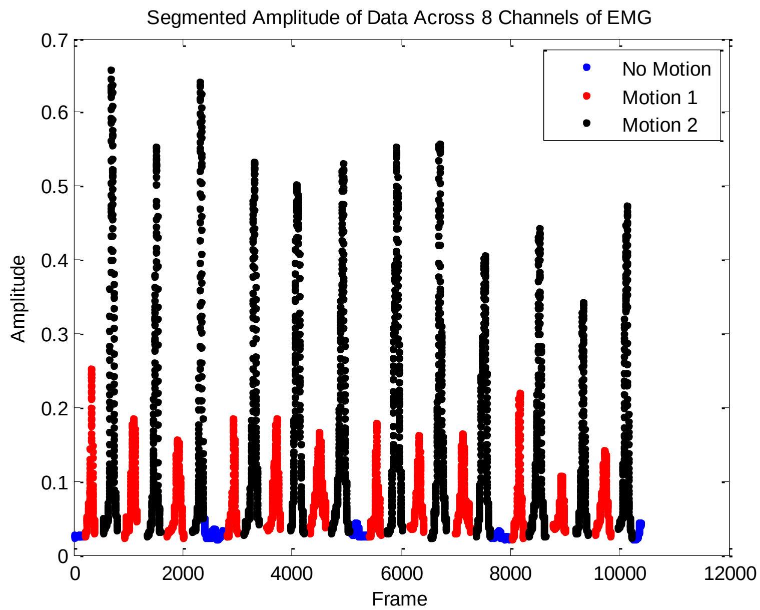

Figure 3.16 - Raw MAV Data, Segmented into 3 motions

While inspection of the data suggests that segmenting this way may prove insightful for research purposes, it is a tedious and unrealistic process to manually select the appropriate valleys in every repetition of every trial, for every user. Consequently, several amplitude-based thresholding techniques were explored in the hopes of implementing an automatic segmentation algorithm. Inconsistencies in amplitude, especially for wrist pronation and wrist supination, however, proved to be challenging. It was also observed that if an automated algorithm mislabeled one transition, the remainder of the data would subsequently be incorrectly classified.

3.3.2 Feature Space Segmentation

The amplitude information contained in the above figures represents only a portion of the available feature information. For example, one complete cycle of a DOF segmented manually based on amplitude can be seen in Figure 3.17.

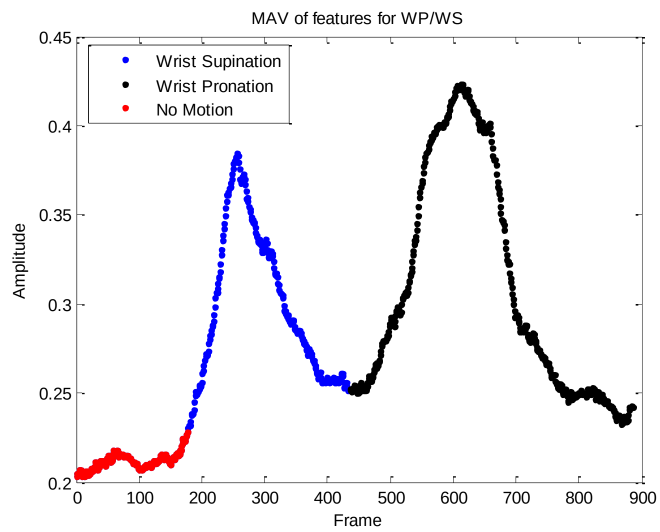

Figure 3.17 – Manual Segmentation based on Amplitude using MAV

When projecting this set of data into 2-dimensional feature space (using ULDA), the segmentation border between the active motions and no motion become much clearer (Figure 3.18). By manually segmenting this data now in the feature space, and then observing it in the time domain (as in Figure 3.19), slight differences in the labeling can be seen.

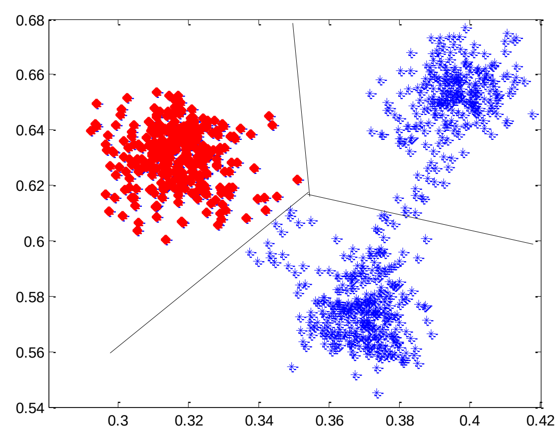

Figure 3.18- MAV of Features Projected in Feature Domain

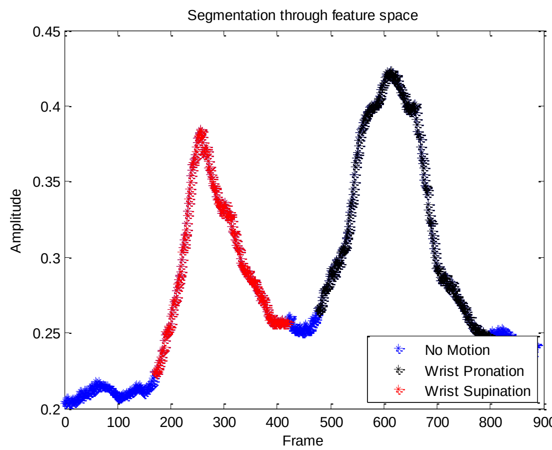

Figure 3.19- Feature Domain Segmentation Observed in Time Domain

Differences observed between the time-domain amplitude segmentation and feature domain segmentation, as highlighted in Figure 3.20, show that more of the low amplitude and transition data gets classified as no motion during the feature space segmentation. This difference motivates the need to find a more sophisticated segmentation method than amplitude for transient motions.

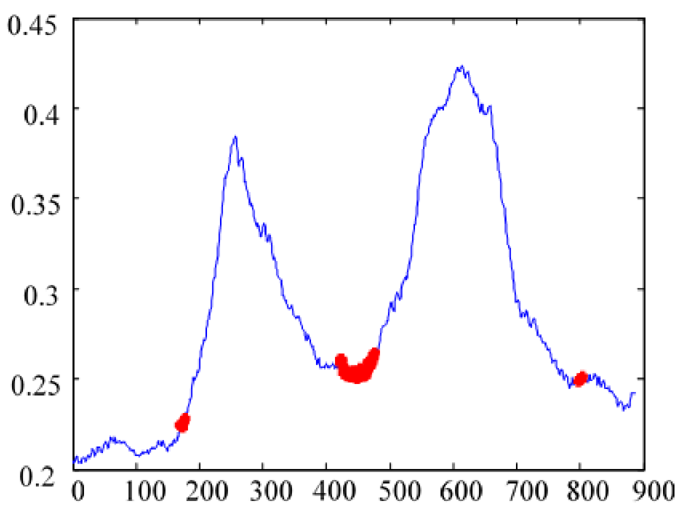

Figure 3.20 - Differences between Time Domain and Feature Domain Segmentation

3.3.3 Segmentation using Local Maximum Bhattacharyya Distance

The Bhattacharyya distance is a measure of similarity between two discrete or continuous probability distributions that takes the covariance of the data into account. By applying a sliding window of approximately the length of one full cycle, the Bhattacharyya distance was calculated between the data in the first and second halves of the window. This Bhattacharyya distance computation window was slid along the full data set, and points of maximum Bhattacharyya distance between active motions were chosen as segmentation locations. Figure 3.21 below shows one cycle of a DOF (top), along with the Bhattacharyya distance calculated along this cycle (second from top). The third subplot shows the features segmented based on the valley of the MAV of data (chosen manually) and the last shows the features segmented based on the maximum Bhattacharyya distance chosen from the second subplot.

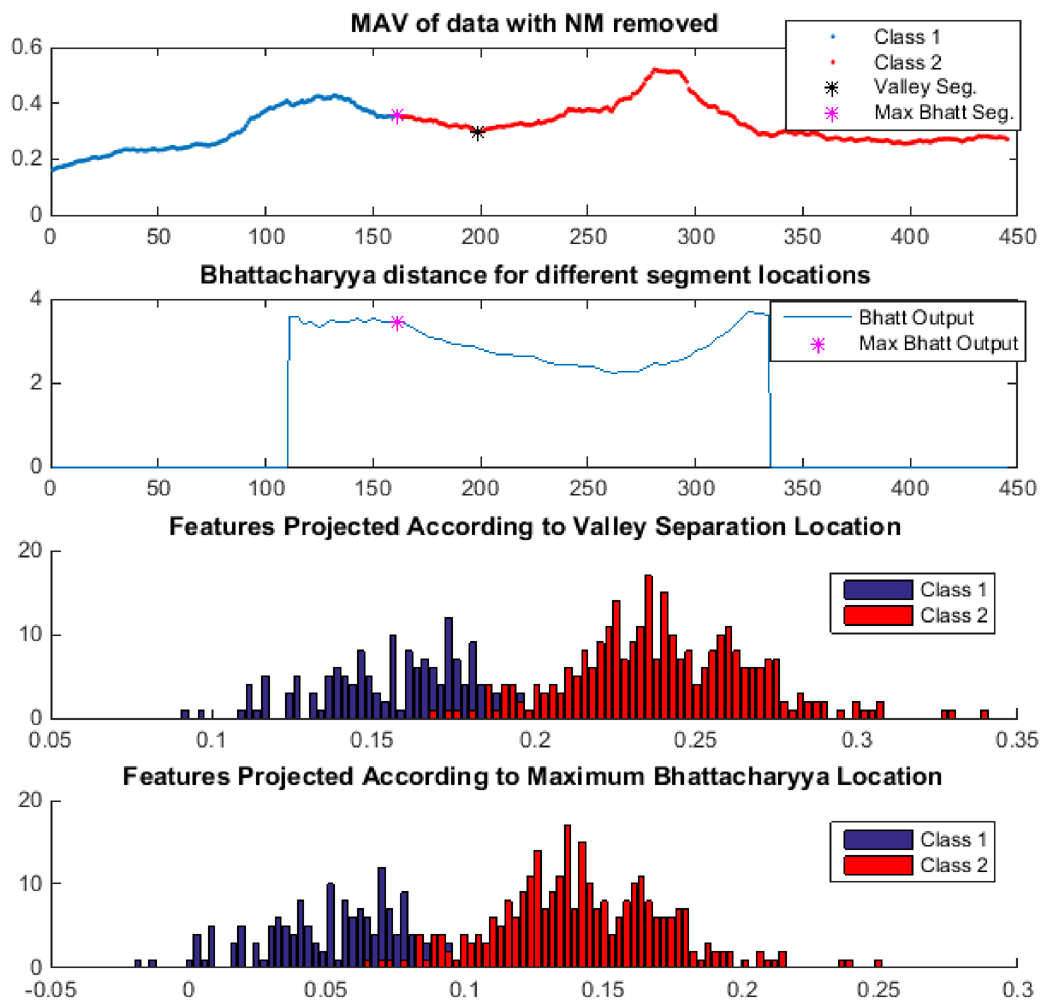

Figure 3.21 - Bhattacharyya Distance for one cycle of a DOF

As seen in the figures above, the Bhattacharyya distance is maximum directly after a contraction and results in a slightly better separation of features when projected down into feature space compared to segmenting at the valleys.

Although for the example provided, improvements can be seen in the feature separation compared to segmenting in the valleys, this did not extend across all subjects and trials. As before, a single mislabeled point in a time series would result in the remainder of the data being incorrectly classified. To better leverage the temporal aspects of the oscillations during training, a minimum distance between successive Bhattacharyya Distance maximums could be implemented. This would, however, constrain the approach to the specific timing of the experiment used here and may not translate to a real environment where conditions may change.

As this investigation was focussed on the impact of the training approach, and not necessarily the segmentation technique, any mislabeling of data during segmentation was unwanted. Consequently, this promising, but imperfect, technique was used in combination with human observer segmentation. For each manual location selected, the Bhattacharyya distance was used to refine the locations that yield the highest separation of features.

It should be noted that although this semi automated method was found to yield higher classification accuracies for most data sets, it could not identify incorrect movements that may have been elicited during training. Even if a participant was to elicit the incorrect motion for a short period of time during a cycle, the Bhattacharyya distance would be maximized during this incorrect motion and the segmentation location for the given cycle would be mislabeled. Although this would not affect the entire set of data with an incorrect segmentation location, it is also not an ideal training method for unsupervised use.

3.3.4 Segmentation using K-means Clustering

K-means clustering works by partitioning data into k clusters; k=3 in this case, as each DOF has two different classes, plus the no motion class. The algorithm initially chooses three random observations of features from the data set and uses these as the initial means for each cluster. Each observation is then partitioned into one of these clusters based on the least within-cluster sum of squares (squared Euclidean distance). The new cluster means are then calculated, and this process is repeated until all of the data has been partitioned. Since the initial clusters are chosen randomly, the algorithm is replicated ten times in order to ensure that the clustering results are accurate and not limited by local minima. An example of the MAV of all features after k-means clustering can be seen below in Figure 3.22.

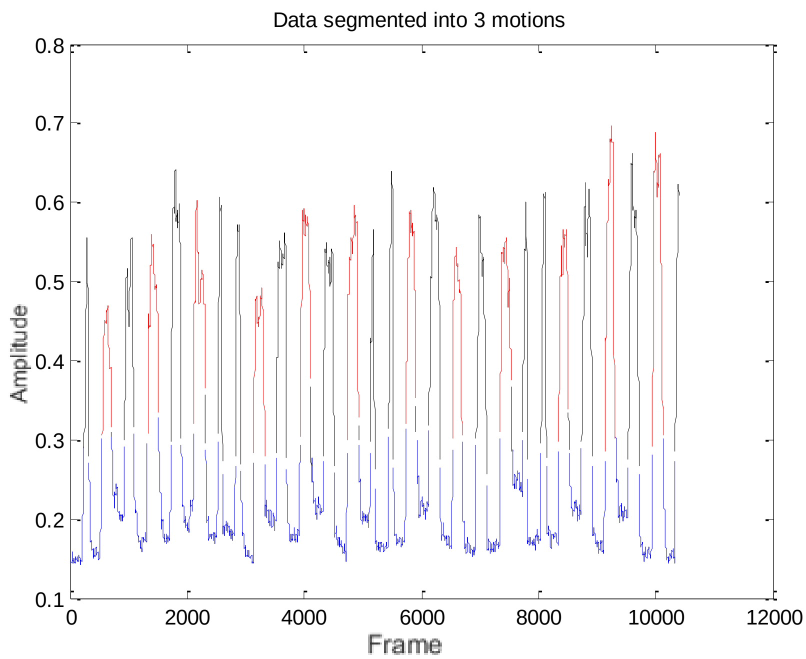

Figure 3.22- Data Properly Segmented into 3 Motions using k-means

While these results look promising, this method was not always consistent either. In some instances, k-means partitioned the data based on amplitude alone, as shown in Figure 3.23. The corresponding data projected into a two-dimensional feature space, in Figure 3.24, indicates there was a high separability of these features, but the classified motions were incorrect.

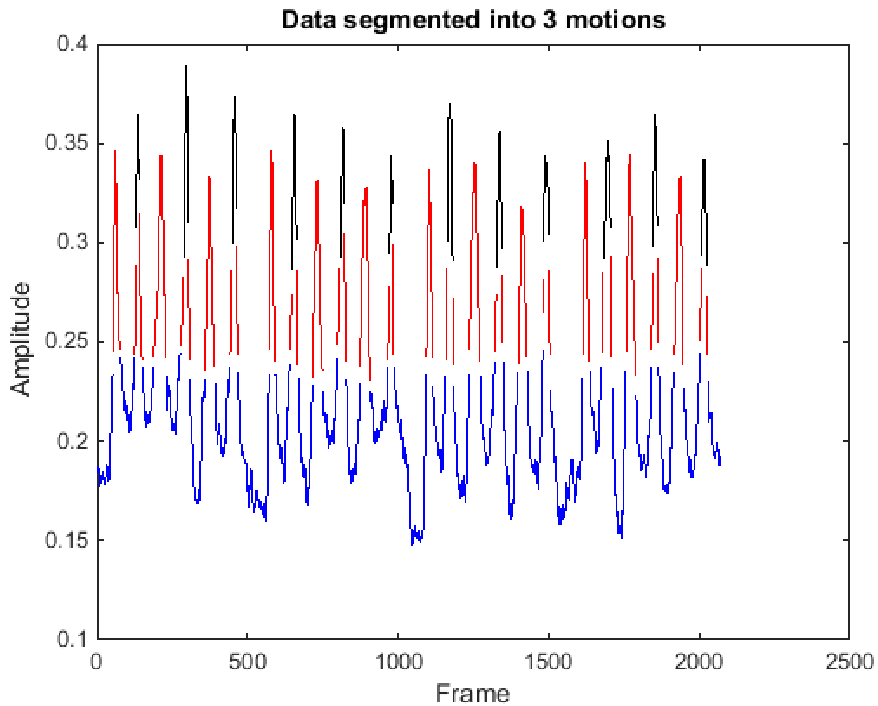

Figure 3.23 - Data Improperly Segmented into 3 Motions using k-means

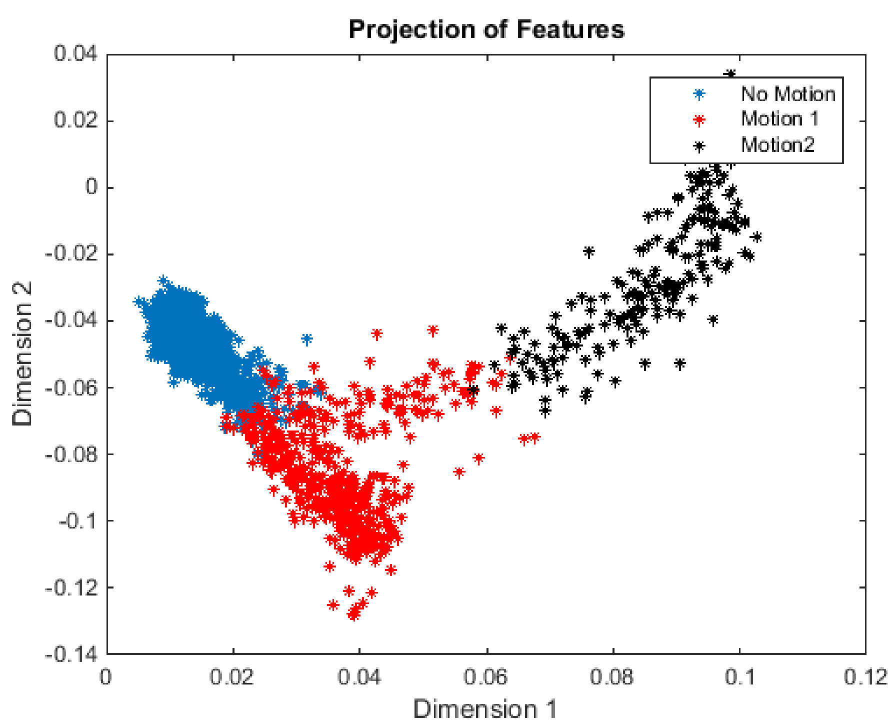

Figure 3.24 - Projection of Features of Improperly Segmented k-means Data

In the rare instance that this amplitude-focussed k-means segmentation occurred, there was nothing automatic that could be done to change the outcome. In these cases, the 10 iterations of k-means (or more) converged to the same result, as did normalizing the data per feature or assigning purposely selected start locations.

3.3.5 Segmentation using Principal Components Analysis (PCA)

Principal Components Analysis can be used for dimensionality reduction by projecting data onto dimensions that explain as much variation as possible. Because three different motions were to be segmented, it was hypothesized that the top three principal components may capture the inter-class variations. An example of the PCA results in Figure 3.25 shows the amplitude of the top three PCA components across the 8 EMG channels in different colours of a full repetition. Classes were segmented by analyzing the maximum of the top three components for each frame along the repetition and classifying the data accordingly.

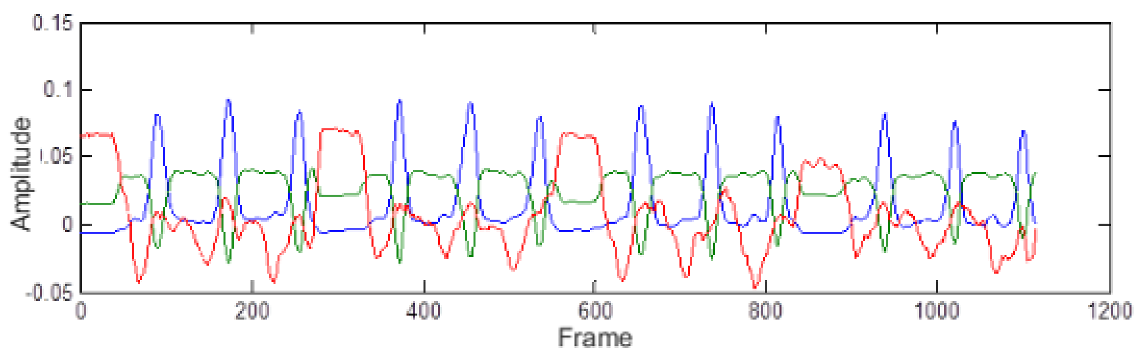

Figure 3.25 - Amplitude of the top 3 PCA components over a full repetition

Again, for this particular example, selecting the points of transition between the maximum PCA component resulted in good separation. However, this did not always extend across all data sets. Similar to k-means, if the PCA components failed (as seen in

Figure 3.26), it was seldom possible to mitigate or easily fix the proper segmentation locations.

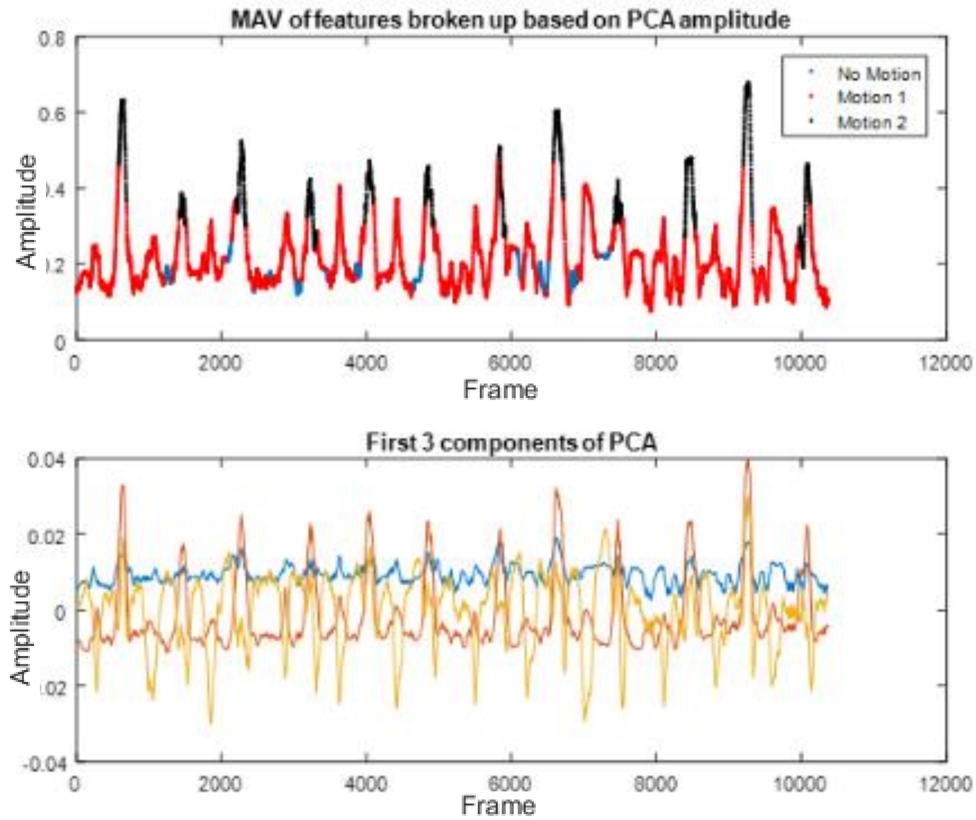

Figure 3.26 - Improper PCA Segmentation

3.3.6 Other Segmentation and Clustering Techniques/Combinations

Many different combinations and techniques were tested in order to improve the robustness of the previous automatic segmentation techniques. The local Mahalanobis distance, k-means clustering using PCA, k-means clustering using PCA and Bhattacharyya, Euclidean angles and distances, and seeding the PCA with different numbers of components were all explored, but all yielded inconsistent results. When one of these methods would fail to provide the correct segmentation, the entirety of the data was misclassified, resulting in very low accuracies, as opposed to only a few frames being misclassified and the classifier being able to adjust. A major limitation of these approaches was their inability to leverage the temporal evolution of the signal.

3.3.7 Segmentation using Matched Filtering

Because the data for each dynamic training method are cyclic in nature, a method was sought that exploited this temporal structure. A *matched filter (see Appendix A)* involves correlating a known signal, or *template,* with an unknown signal to detect the presence of the template in the unknown signal. It was hypothesized that a matched filter would be able to identify the transitions between each cycle if it was given an initial cycle as the input. For this segmentation technique, therefore, the first cycle of the first repetition was manually segmented (to obtain the template for the matched filter), and then this filter was passed through the remainder of the data. In order for this segmentation technique to be successful, the initial cycle that was used as the matched filter template was assumed to be a perfect contraction. The results from the matched filter output should provide a maximum when the full cycle template is matched in the data, which should occur when a new cycle is about to commence. As shown in Figure 3.27, the matched filter output is also maximum at motion changes, which is ideal for determining segmentation locations to break the data into the 3 motions of interest.

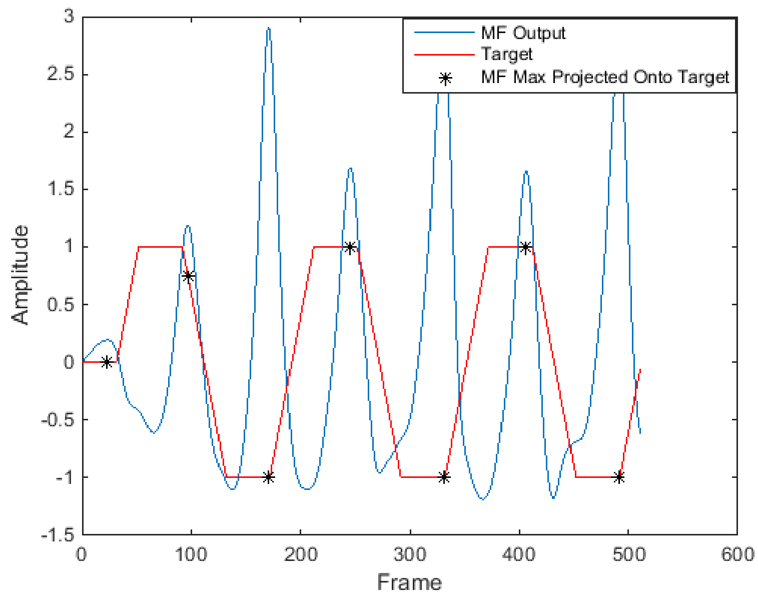

Figure 3.27 - Matched Filter Output and Corresponding Targets

As seen in many of the other segmentation techniques, determining the segmentation locations by amplitude was difficult as not all data sets have perfect peaks. However, the matched filter approach yielded much more consistent results than the previous methods investigated. With a semi-supervised approach to ensure the matching was working properly, this method provided the potential for real-time usability testing (although it would not be ideal for clinical applications as the supervision would still be needed).

3.3.8 Segmentation using PCA and Matched Filtering

As one final evaluation, the PCA and matched filter approaches were combined. This segmentation technique was similar to that of the matched filter, except that it used the PCA components derived from the raw data as inputs to the matched filter as opposed to the TD features. Its results are summarized along with the others in the following sections.

3.4 Segmentation Performance

The offline accuracy of a classifier represents how well it was able to extrapolate to a new set of test data. The accuracy is determined by dividing the correct number of classifications by the total number of classifications possible, as explained in Equation 2. Although this measure does not strictly relate to the usability of a control scheme, it provides a rough guideline for candidates for further real-time testing.

The training methods and different segmentation/clustering techniques were compared by calculating the mean classification accuracy across all participants for both DOFs using the static data collected during Screen Guided Training as the testing data. A summary of results can be seen below in Tables 1 (for wrist pronation/supination) and 2 (for hand open/close, HO/HC).

Table 1 - Summary of Classification Accuracies (WP/WS)

		Velocity Based Training	***Position Based Training***	***Altered Position Based Training***
Manual	Accuracy # Failed Trials	80.80 +/- 15.61% N/A	73.56 +/- 24.29% N/A	73.59 +/- 15.46% N/A
Manual + Bhattacharyya	Accuracy # Failed Trials	81.83 +/- 15.95% N/A	70.87 +/- 22.54% N/A	70.65 +/- 15.77% N/A
K-means	Accuracy # Failed Trials	75.81 +/- 21.53% 4	64.90 +/- 21.76% 6	73.50 +/- 17.93% 5
PCA+k-means	Accuracy # Failed Trials	77.84 +/- 18.45% 11	68.13 +/- 21.83% 11	75.38 +/- 17.70% 13
PCA	Accuracy # Failed Trials	76.68 +/- 11.89% 6	69.54 +/- 15.06% 4	72.37 +/- 11.28% 5
Matched Filter	Accuracy # Failed Trials	91.02 +/- 4.98 % N/A	86.06 +/- 12.55% N/A	88.85 +/- 9.37% N/A
PCA + Matched Filter	Accuracy # Failed Trials	90.36 +/- 5.58% N/A	83.98 +/- 13.43% N/A	89.35 +/- 8.18% N/A

For the WP/WS segmentation the automatic segmentation methods (PCA and k-means) yielded low accuracies due to the instances of complete failure during segmentation. The manual segmentation techniques are more consistent than the automatic methods, however, the segmentation based on amplitude lead to lower efficiency than the matched filter methods that segment based on feature similarities.

Table 2 - Summary of Classification Accuracies (HO/HC)

		Velocity Based Training	***Position Based Training***	***Altered Position Based Training***
Manual	Accuracy # Failed Trials	81.35 +/- 14.33% N/A	79.78 +/- 17.15% N/A	80.53 +/- 14.83% N/A
Manual + Bhattacharyya	Accuracy # Failed Trials	82.28 +/- 12.81% N/A	80.06 +/- 19.24% N/A	79.61 +/- 12.95% N/A
K-means	Accuracy # Failed Trials	86.68 +/- 15.85% 2	75.27 +/- 20.23% 5	83.18 +/- 12.95% 4
PCA+k-means	Accuracy # Failed Trials	83.82+/-17.74% 8	74.79+/- 20.07% 6	83.37+/-12.96% 7
PCA	Accuracy # Failed Trials	73.95 +/- 13.37% 5	73.88 +/- 11.40% 3	77.81 +/- 7.38% 6
Matched Filter	Accuracy # Failed Trials	91.35 +/- 9.99% N/A	91.05 +/- 8.27% N/A	92.51 +/- 6.89% N/A
PCA + Matched Filter	Accuracy # Failed Trials	90.74 +/- 10.76% N/A	90.69 +/- 8.88% N/A	93.24 +/- 6.27% N/A

The HO/HC DOF segmentation results were similar to those of WP/WS. The matched filter methods, which segment based on feature similarities, outperformed the automatic segmentation methods and the methods that rely on solely on the MAV features.

Chapter 4. Selection of Segmentation Locations

The contents of Chapter 3 demonstrate that an automatic segmentation of transient training data may not always yield consistent results. Manual segmentation methods that rely solely on the MAV or instantaneous features lead to lower efficiency rates than those that leverage temporal information. Although manual, or semi-supervised segmentation is not practical in a clinical context, it was employed in this work because the purpose of this work was to evaluate the dynamic training methods.

Because the matched filter approach with human supervision yielded the highest classification accuracies, it was adopted for further study as follows: the first three PCA components were extracted from the raw training data as collected in Chapter 3 from 8 EMG channels. The first cycle of the first repetition was then manually selected within ACE by the experimenter by looking at the MAV of the features across the 8 channels, seen in Figure 4.1. The PCA components derived from this manually extracted segment were then compared to the remainder of the data, and the output from the matched filter was evaluated visually to ensure peaks were occurring during transitions.

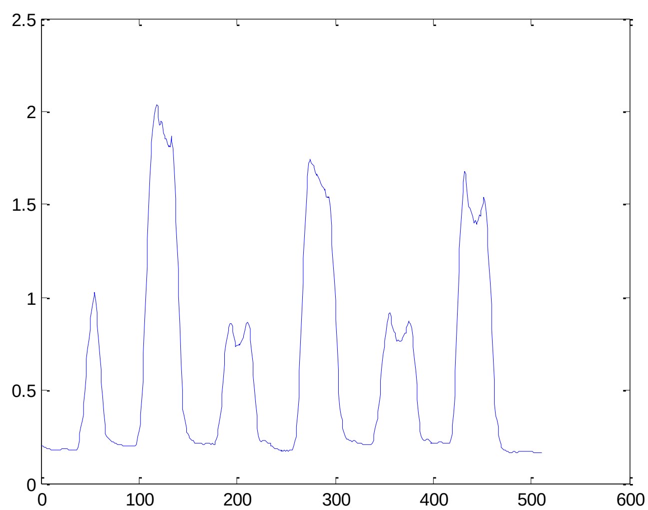

Figure 4.1– MAV for one repetition of WP/WS Data

The matched filter output corresponding to one repetition is shown below in Figure 4.2. The peaks were automatically detected and shown on the figure for the experimenter to approve or change before proceeding to the following repetition. Since the training targets for all training sets were known, the experimenter had knowledge of the approximate locations of these transition points. If the peaks were misplaced, or if there were any missing, the experimenter was able to manually select or change them.

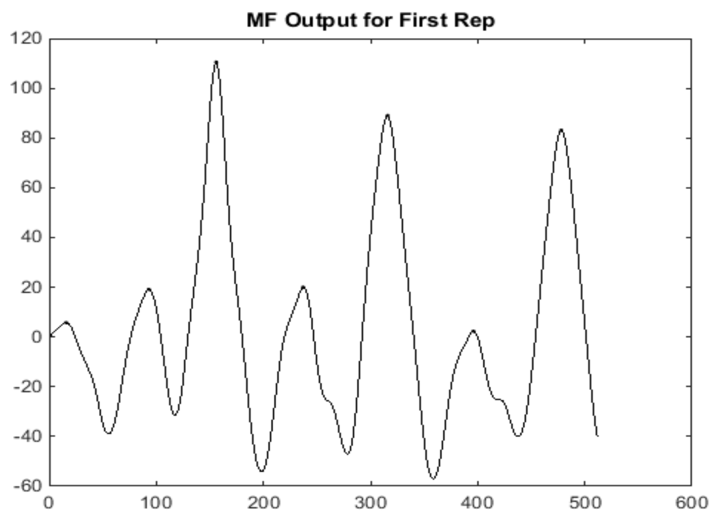

Figure 4.2 MF Output of WP/WS Data for 1 Repetition

Because the exact location in a dynamic contraction when the change of classes occurs was unknown, three different data segmentation locations were examined for future implementation during our usability study, as explained below.

4.1 Matched Filter Output

The first data segmentation technique used the maximum matched filter output location, as detailed above. As shown in Figure 4.3, the matched filter locations that yielded the highest amplitude occurred during the low amplitude portion of data collection, as expected (after the user releases one contraction but before they initiate the next).

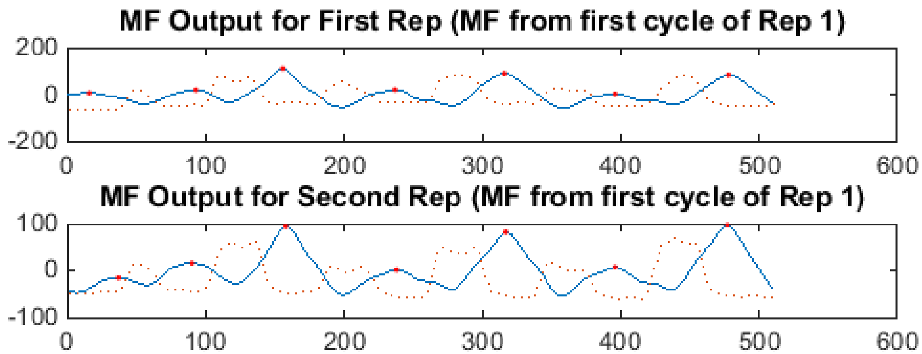

Figure 4.3 - Matched Filter Segmentation Locations Superimposed on MAV of Features (blue, matched filter; dotted red, MAV; red dot, segmentation location)

4.2 Matched Filter Output with Bhattacharyya Refinement

Based on the findings in the second preliminary experiment, the Bhattacharyya distance was found to be an effective distance measure that maximizes the distance between features. The second segmentation technique therefore extended the matched filter approach by incorporating the Bhattacharyya distance to refine the selected location. Figure 4.4 illustrates the maximum Bhattacharyya distances using the same training data as figure 4.3 and shows that the Bhattacharyya refinement slightly modifies the segmentation locations compared to the matched filter output alone. This method, in general, segments the data earlier in the time series, while the signal is still reducing from its peak.

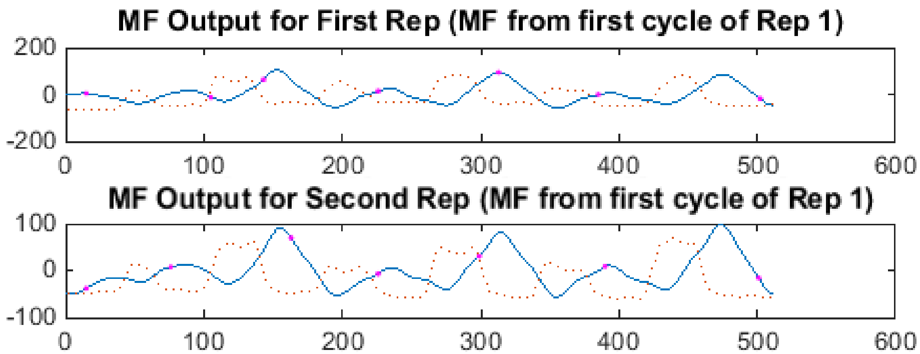

Figure 4.4 - Matched Filter Output with Bhattacharyya Refinement (blue, matched filter; dotted red, MAV; red dot, segmentation location)

4.3 Matched Filter Output with Shift

The third segmentation location algorithm incorporated a constant shift from the peaks found by the matched filter approach. When testing the offline accuracy of the LDA classifier, it was found (as shown in Figure 4.5) that a constant shift between -15%-15% of the matched filter output produced similar LDA classification accuracies, however a shift of -15% of the cycle length offered a slightly better accuracy, it was evaluated as a third location option. The corresponding segmentation locations can be seen in Figure 4.6.

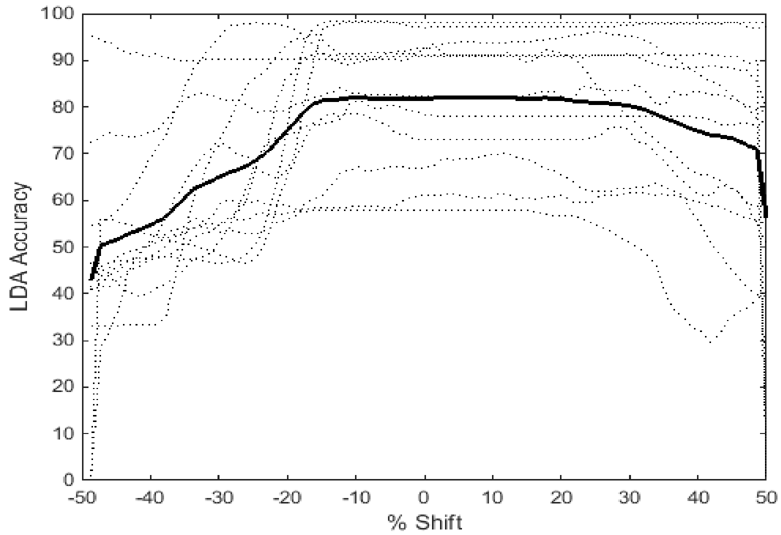

Figure 4.5 - Matched Filter Output with % Shift (Mean Across Subjects Indicated in Bold)

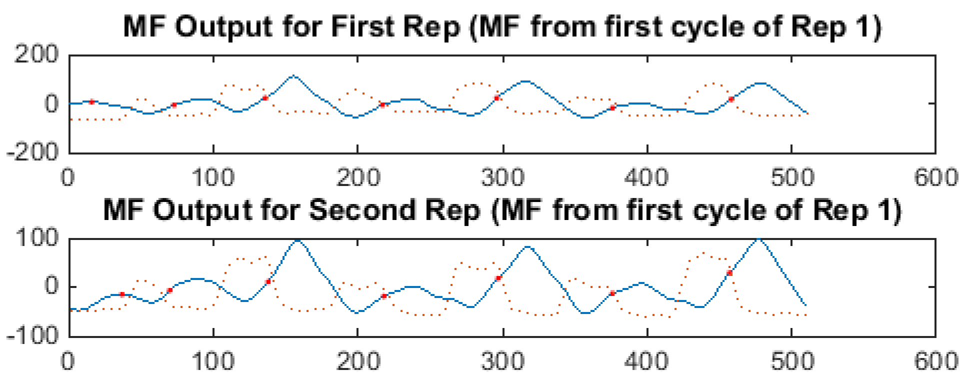

Figure 4.6 - Segmentation using Matched Filter Output with 15% Shift (blue, matched filter; dotted red, MAV; red dot, segmentation location)

Chapter 5. Real-time Usability Testing

Upon selection of the segmentation method and locations outlined in Chapter 4, a real-time target acquisition experiment was conducted to assess the usability performance of ten different combinations of training and segmentation techniques. The target acquisition test in this experiment was as a Fitts' law style test, as described in [2].

5.1 Protocol

Thirteen able-bodied subjects (9 male and 4 female, mean age 24 +/- 2.1 years, 12 right-handed) participated in this experiment, using the same experimental setup as described in Chapter 3.2.1. Nine of the subjects reported having had some previous experience with myoelectric control. The ten permutations of training and segmentation consisted of all combinations of the position, velocity and position-reset training methods with the three different segmentation approaches (as described in Chapter 4), and a baseline comparison classifier trained using ramp contractions and segmented based on amplitude.

5.2 Data Collection

An 8-electrode cuff was placed around the subject's dominant forearm and training was performed using the custom graphical interface described in Chapter 3. Before beginning the experiment, each subject learned each of the five motions, and the electrode activity was monitored until the electrodes settled so there was negligible amount of noise remaining on all channels. Each subject was trained using the four different training methods described in Chapter 3 and were guided through one familiarization session of each of the different training methods.

5.3 Target Acquisition Test

A Fitts' law-based acquisition test was conducted to evaluate the usability of the ten different training sets. For each trial, after training, the subject was prompted with two targets and a cursor on the screen. A trial consisted of 24 different target locations as described in Table 5.1. The aim of this test was to accurately move the cursor into two required targets, rotational and vertical targets, within the required time by eliciting the corresponding contractions.

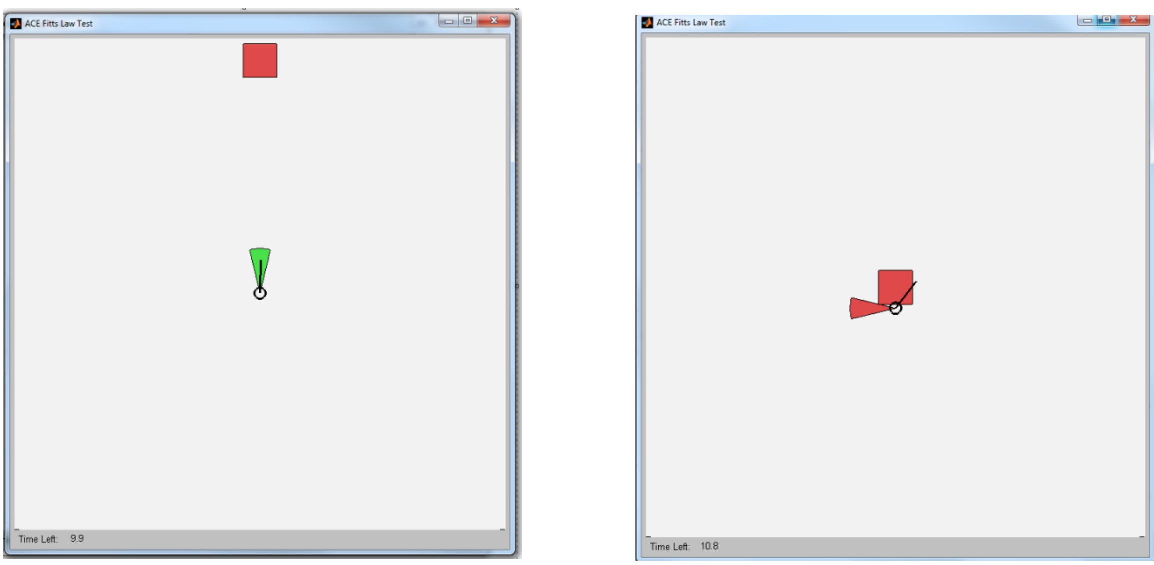

Figure 5.1 - Fitts' Law Target Acquisition Test; (left) rotation target achieved, vertical target not achieved, (right) neither target achieved

As seen in Figure 5.1 (left), the red boxes indicate the targets for the black cursor. Once the cursor is within the target, it was turned green. The test was considered complete once the cursor remained within both targets for a full one second dwell time.

The movement of the cursor was controlled by the class output of the classifier, and the velocity by a proportional speed control determined by the mean of the MAV of all eight channels [62]. Performing no motion (a resting state) resulted in the cursor not moving. Pronation/supination class outputs controlled the rotation of the cursor 'dial', while hand open/close controlled the up/down movement of the cursor. If the task was not completed within 12 seconds, the target timed out and was considered unsuccessful. Each trial included 24 consecutive targets with randomized combinations of three different widths and two different distances, resulting in six different indices of difficulty. The two distances were 25% and 100% of the full trajectory, while the widths (W) were 10%, 15%, and 25%. The index of difficulty (ID) was calculated by:

$$ID = \log_2\left(\frac{2D}{W}\right) \tag{5}$$

where D corresponds to the target distance, W is the width of the target, and ID is the Index of Difficulty. The target widths and distances along with the IDs are summarized in the table below.

Table 3 - Target Locations and Widths

Distance	Width	ID
100	10	3.46
100	15	2.94
100	25	2.32
25	10	1.81
25	15	1.42
25	25	1.00

Because classification is sequential, only one class could be activated at a time. A two second rest time was provided between each target testing attempt during which the cursor was returned to the neutral position. The trial of 24 targets was repeated for each of the 10 training/segmentation schemes, presented to each participant in a random and blinded order. After each test, subjects were asked to take a break if needed, and a mandatory break was given after the third and seventh trials.

5.4 Performance Metrics

A set of performance metrics used in many prior experiments to investigate the usability of myoelectric control [2] [3] [62] were employed to provide information about the control quality during the usability test.

Table 4 - Performance Metrics

Metric	*Description*
Path Efficiency	Quality of control; a ratio of the shortest possible distance to the target and the actual distance traveled.
Throughput	Index of performance; a combination of speed and accuracy expressed as a single measure of information bandwidth [61]
Overshoot	Target acquisition; the number of times the target was acquired and then lost before completing the dwell time over the total number of targets possible.

5.5 Results

5.5.1 Participant Feedback

Quantitative and qualitative information was collected from each subject at the beginning and end of the experiment. Each subject provided their age, handedness, and level of prior EMG experience before the experiment began, and at the end of the experiment each subject was asked to rank the training protocols from 1-4, where 1 corresponded to the most intuitive protocol, and 4 belonged to the least. As shown in Figure 5.2, most participants found velocity-based training (VBT), as employed by PGT, to be the least intuitive training method. Conversely, the majority of subjects preferred the position-based training as the most intuitive.

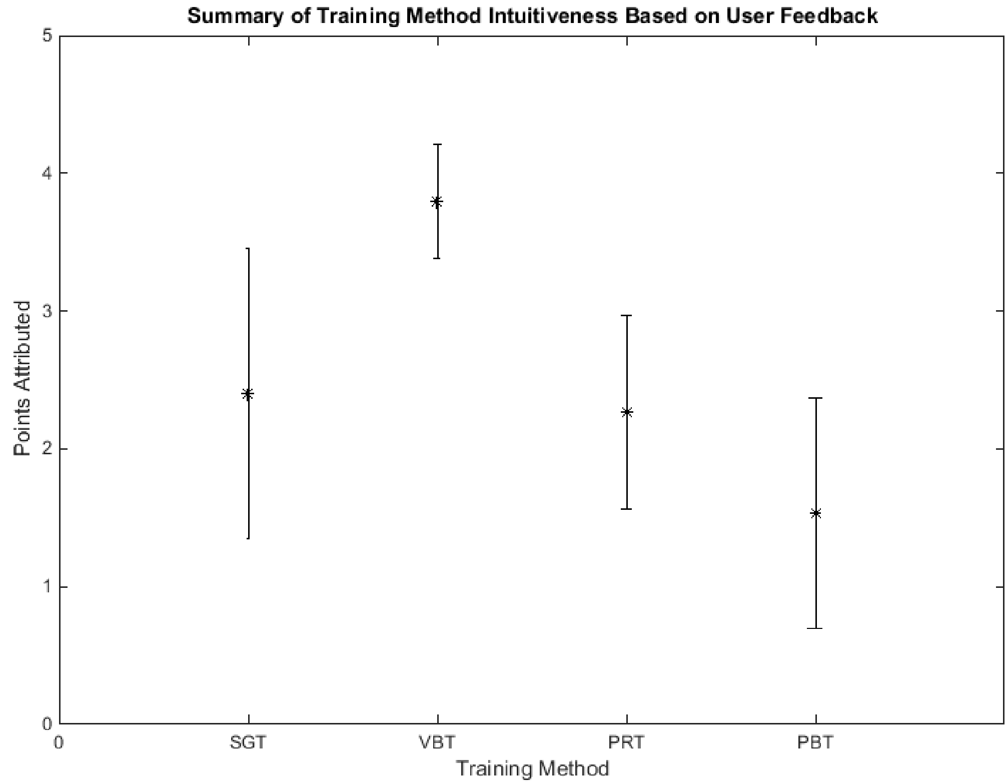

Figure 5.2 - Participant Feedback: Intuitiveness of Training Methods (Mean and Standard Deviations of the Average Points Attributed Across All Participants)

5.5.2 Performance Metrics

To evaluate the performance of each of the training and segmentation methods, a two-tailed paired t-test was used with a significance level of 0.05 to compare the efficiency, overshoot and throughput metrics. Two main effects were evaluated: 1) the segmentation approach, and 2) the training method. For the ramp training method, only the training method was evaluated as this data was not segmented in any way.

Efficiency

A summary of the results for the efficiency metric is shown in Tables 5 and 6. The tables can be read by comparing the row training method with the column method – for example, within Segmentation 1, VBT had 10% higher efficiency than PBT. Notable results are indicated for the VBT training method which had a significantly higher ($p<0.05$) efficiency than all three other training methods when using Segmentation 1. VBT also outperformed PBT and Ramp training methods with Segmentation #3. When comparing the training methods using Segmentation 2, no statistical significances were found.

Table 5 - Differences in Mean Efficiencies by Segmentation Type for the Different Training Methods

		PBT	PRT	Ramp
Segmentation 1	**VBT**	10.0*	5.5*	5.1*
	PBT		-4.5	-4.9
	PRT			-0.38
Segmentation 2	**VBT**	8.5	-0.26	0.92
	PBT		-8.7	-7.6
	PRT			1.2
Segmentation 3	**VBT**	7.6*	6.2	5.2*
	PBT		-1.4	-2.3
	PRT			-0.97

* Denotes statistically significant difference ($p<0.05$)

When comparing the efficiencies of the various segmentations methods for each training method, it was observed that for VBT, Segmentation 1 and Segmentation 3 outperformed the ramp baseline. For PBT, Segmentation 3 and the ramp baseline had a significantly higher efficiency than Segmentation 2. No statistical significances were found between the segmentation methods of PRT.

Table 6 - Mean Efficiencies for Different Segmentation Methods for VBT, PBT and PRT

		Segmentation 2	Segmentation 3	Ramp
VBT	**Segmentation 1**	4.2	-0.10	5.1*
	Segmentation 2		-4.3	0.92
	Segmentation 3			5.2*
PBT	**Segmentation 1**	2.7	-2.6	-4.9
	Segmentation 2		-5.2*	-7.6*
	Segmentation 3			-2.3
PRT	**Segmentation 1**	-1.5	0.59	-0.38
	Segmentation 2		2.1	1.2
	Segmentation 3			-0.97

* Denotes statistically significant difference (p<0.05)

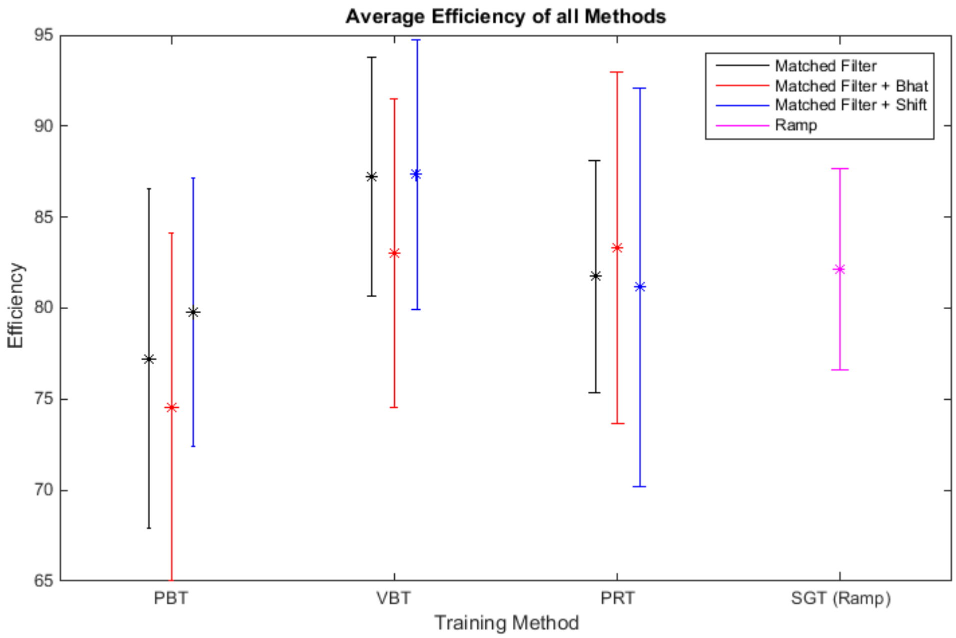

Figure 5.3 - Average Efficiencies across all Training and Segmentation Methods

Throughput

No significant differences in throughput were found between any of the combinations of training or segmentation schemes, as evident in Figure 5.5.

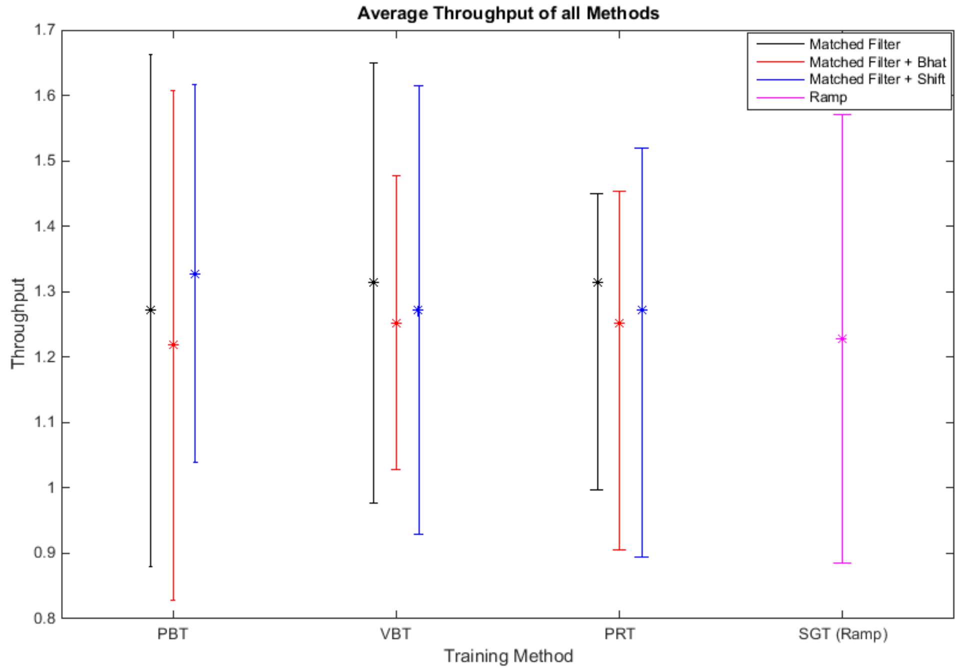

Figure 5.4 - Average Throughput of all Training and Segmentation Methods

Overshoot

The only significant difference in overshoot between segmentation methods was with the fixed shift segmentation technique. The average overshoot for VBT was 0.2 less than PBT ($p < 0.05$) and 0.16 less than PRT ($p < 0.05$). The only significant differences between segmentation methods occurred with PRT training when the matched filter output segmentation overshoot was 1.2 less than that of the shifted segmentation method ($p<0.05$). Both methods yielded an overshoot of 1 less than ramp ($p<0.05$).

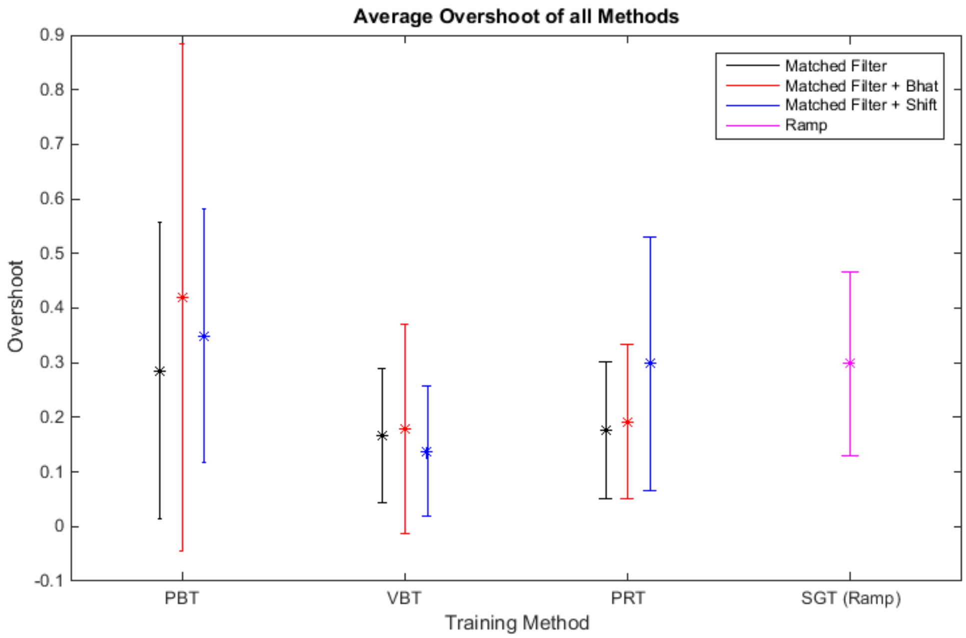

Figure 5.5 - Average Overshoot of all Training and Segmentation Methods

Chapter 6. Discussion and Conclusions

6.1 Discussion

The overall goal of this research was to evaluate the efficacy of different training for pattern recognition based myoelectric control. This was motivated by the fact that current methods of training fail to incorporate the transitions required during actual usage. While the addition of extra data may seem advantageous, its inclusion necessitates an additional step of segmentation which has proven to be difficult to implement automatically.

This thesis looked at four different training methods, 2 of which were newly proposed methods, and 2 that are commonly used, to determine which was the most reliable for real-time use. As shown in the results, although the prosthesis guided training (PGT) approach (which follows the VBT scheme) may have been deemed to be the least intuitive training by the subjects, it still led to the best performance when coupled with the fixed shift segmentation technique.

To facilitate the comparison of dynamic training approaches, a variety of segmentation approaches were explored. While amplitude-based methods seemed intuitive, it was observed that amplitudes of the wrist pronation/supination DOF are not always distinguishable. Automatic clustering algorithms also suffered in some instances with no identifiable method of recovery. Ultimately, a matched filter, which took temporal information into account, was combined with a semi-supervised approach to yield a testable system. Since the matched filter approach naturally segmented movements at the valleys of the transitions, variations of this MF output location were also explored.

Although the different segmentation methods yielded varying levels of success during initial offline analyses, no consistently significant differences were found during the usability test. The matched filter segmentation approach, which segmented classes in the valleys yielded significantly higher efficiency than all 3 other training methods when paired with VBT. While subjects found this to be the least intuitive training protocol to follow, the difficulty of following this prompt may have led participants to pay closer attention during this training.

The PBT training method was found to be the most difficult to segment, likely due to greater similarities in features from one motion to another. Because there was no "no motion" period to distinctly break up the DOF, it was a lot more difficult to determine the transition location, particularly for the wrist pronation/supination DOF. During usability testing, participants had the most trouble achieving the wrist rotation targets, no matter the training method, indicating that the proper segmentation location for the WP/WS DOF may be different than the location for the HO/HC DOF.

Overall, this work confirmed the moderate superiority of the VBT training approach employed by the current commercial PGT system as compared to simple ramp training approaches.

6.2 Contributions

The key contributions of this work included:

1) an analysis of the differences between multiple training methods in a usability test.

2) the evaluation of various automatic segmentation techniques for continuous and dynamic myoelectric signals
3) a semi-supervised method of training for PR myoelectric control.

The first contribution is somewhat lackluster in the sense that there was no one training or segmentation approach that consistently outperformed all other methods. Nevertheless, as a result, the need for better segmentation techniques was uncovered, and further improvement may provide added clarity in the outcomes of this work. The second result is useful for future studies trying to segment additional dynamically transitioning time-series data. A better understanding on different segmentation and clustering techniques for myoelectric control were obtained, and a matched filter-based approach using time domain features proved to have higher classification accuracies than pure amplitude techniques. The third contribution includes a functional semi-supervised method of training for PR myoelectric control which can be applied to many different training methods used in the field.

6.3 Future work

Although this work focussed on the impact of different training approaches, it required a substantial amount of work on segmentation. A number of approaches were able to successfully segment dynamic training data in most cases; however, none worked sufficiently well to avoid the need for manual input from the experimenter was included during the usability test. In a real-life scenario, this is impractical, and thus more work on automatic segmentation approach is still warrented. As research continues and the technology and understanding of pattern recognition grows, an automatic implementation of a segmentation technique may be within reach and could improve the robustness of myoelectric pattern recognition systems. A possible approach may include a fusion of multiple segmentation approaches.

The consideration of cognitive load and user motivation during training must be taken into consideration to help form a more complete picture of training and usability. Subjects reported that training with Prosthesis Guided Training required a lot more attention than all other training methods as it was not as intuitive as a prompt as the others. This may have led to better compliance from users, as it required them to pay careful attention. Alternatively, it may also lead to user error when used in a less constrained and supervised setting. Future experiments could be conducted to include motivations, such as a point system, to keep the participant's level of engagement the same throughout all training methods.

Although the order of the training and segmentation combinations was randomized, it was also observed that participants seemed to lose motivation as the experiment wore on, lasting nearly two hours causing the overall quality of data to diminish. In certain instances, the timeout period during the usability test was reached without the participant successfully achieving the targets. As mentioned previously, the WP/WS DOF seemed to be the DOF which caused the most issues during testing; however, a lack of motivation could have also played a factor in the user not trying their best to reach the target. The implementation of a fully automatic segmentation algorithm, adding a motivation factor to the experiment, and testing fewer combinations, could help to reduce the total time of the experiment and increase the motivation to properly train and reach the targets within the Fitts' Law study.

6.4 Conclusions

The goal of this work was to better understand the impact of different training modalities on the usability of pattern recognition based myoelectric control. This was achieved by incorporating dynamic motion into two existing training methods and 2 new methods, then observing the classification performance for different segmentation locations in both offline and real-time usability tests.

As shown by Scheme and Englehart [5], the incorporation of dynamic training data within a single class has been shown to improve the usability of prosthetic control. This work has shown that segmentation required when including additional training data that contains boundaries between two active motions is challenging. Moreover, even when this confounding factor is addressed, the inclusion of these data may not further improve the usability of pattern recognition based myoelectric control. This may be a result of the natural and increased variability in dynamic multi-class training contractions, which may not translate well to real-time control.

References

[1] K. M. Norton, "A Brief History of Prosthetics," 2007. [Online]. Available: http://www.amputee-coalition.org/resources/a-brief-history-of-prosthetics/.

[2] L. Hargrove and S. Wurth, "A real-time comparison between direct control, sequential pattern recognition control and simultaneous pattern recognition control using a Fitts' law style assessment procedure," *Journal of NeuroEngineering and Rehabilitation,* 2014.

[3] B. A. Lock, A. M. Simon, K. Stubblefield, and L. J. Hargrove, "Prosthesis-Guided Training For Practical Use Of Pattern Recognition Control Of Prostheses," in *MEC Symposium*, Fredericton, 2011.

[4] C. Chicoine, A. M. Simon and L. J. Hargrove, "Prosthesis-guided training of pattern recognition-controlled myoelectric prosthesis," 2012.

[5] E. J. Scheme and K. Englehart, "Training Strategies for Mitigating the Effect of Proportional Control on Classification in Pattern Recognition Based Myoelectric Control," *Journal of Prosthetics and Orthotics,* 2013.

[6] P. Parker, K. Englehart and B. Hudgins, "Myoelectric signal processing for control of powered limb prostheses," *Journal of Electromyography and Kinesiology,* vol. 16, pp. 541-548, 2006.

[7] R. Meier and D. Atkins, "Functional Restoration of Adults and Children with Upper

Extremity Amputation," *Demos Medical Publishing*, pp. 207-224, 2004.

[8] P. Fang, Z. Wei, Y. Geng and F. Yao, "Using speech for mode selection in control of multifunctional myoelectric prosthesis," in *IEEE Eng. Med. Biol. Soc.*, 2013.

[9] M. Atzori and H. Müller, "Control Capabilities of Myoelectric Robotic Prostheses by Hand Amputees: A Scientific Research and Market Overview," *Frontiers in Systems Neuroscience*, 2015.

[10] K. Fukunaga, Introduction to Statistical Pattern Recognition, San Diego, 1990.

[11] N. Wang, L. Yang, Q. Zhang and M. Chen, "A Novel E-Commerce Recommendation System Model based on the Pattern Recognition and User Behavior Preference Analysis," in *Information Science and Industrial Applications*, 2016.

[12] E. Scheme and K. Englehart, "Electromyogram pattern recognition for control of powered upper-limb prosthesis: State of the art and challenges for clinical use," *Journal of Rehabilitation Research & Development*, vol. 48, pp. 643-660, 2011.

[13] M. Glynn, H. Galway, G. Hunter, and W. Sauter, "Management of the upper-limb-deficient child with a powered prosthetic device," *Clin Orthop*, pp. 202-205, 1986.

[14] B. Terlaak, H. Bouwsema, C. K. v. d. Sluis and R. M. Bongers, "Virtual Training of the Myosignal," *PLoS ONE*, vol. 10, no. 9, 2015.

[15] E. J. Scheme and K. Englehart, *A flexible user interface for rapid prototyping of advanced real-time myoelectric control schemes,* Fredericton: Myoelectric

Controls/Powered Prosthetics Symposium, 2008.

[16] N. V. Thakor and M. A. Powell, "A training strategy for learning pattern recognition control for myoelectric prostheses," *J. Prosthet. Orthot.*, pp. 30-41, 2013.

[17] T. A. Kuiken, G. Li, B. A. Lock, R. Lipschutz, L. Miller, K. A. Stubblefield, and K. B. Englehart, "Targeted Muscle Reinnervation for Real-Time Myoelectric Control of Multifunction Artificial Arms," *JAMA*, vol. 301, no. 6, 2009.

[18] B. Douillard, J. Underwood, N. Kuntz, V. Vlaskine, A. Quadros, P. Morton, and A. Frenkel, "On the segmentation of 3D lidar point clouds," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2011.

[19] G. Kaur, A. Arora, and V. Jain, "Comparison of the techniques used for segmentation of EMG signals.," in *11th WSEAS International Conference on MATHEMATICAL and COMPUTATIONAL METHODS in SCIENCE and ENGINEERING*, 2009.

[20] G. Okeyo, L. Chen, H. Wang, and R. Sterritt, "Dynamic sensor data segmentation for real-time knowledge-driven activity recognition," in *Pervasive and Mobile Computing*, pp. 155-172.

[21] S. Rossignol, X. Rodet, J. Soumagne, J. Collette and P. Depalle, "Automatic Characterisation of Musical Signals : Feature Extraction and Temporal Segmentation," *Journal of New Music Research*, vol. 28, no. 4, pp. 281-295, 1999.

[22] H. Azami, A. Khosravi, M. Malekzadeh and S. Sanei, "Based on Hiaguchi’s Fractal

Dimension," pp. 152-159, 2012.

[23] H. Azami, S. Sanei and K. Mohammadi, "A Novel Signal Segmentation Method based on Standard Deviation and Variable Threshold," *International Journal of Computer Applications*, vol. 34, no. 2, pp. 27-34, 2011.

[24] H. Praetorius, G. Bodenstein, and O. Creutzfeldt, "Adaptive segmentation of EEG records: a new approach to automatic EEG analysis," *Electroencephalogr Clin Neurophysiol,* vol. 42, no. 1, pp. 84-94, 1977.

[25] H. Azami, H. Hassanpour, J. Escudero and S. Sanei, "An intelligent approach for variable size segmentation of non-stationary signals," *Journal of Advanced Research,* vol. 6, no. 5, pp. 687-698, 2015.

[26] K. J. Cios, R. W. Swiniarski, W. Pedrycz, and L. A. Kurgan, "Unsupervised learning: clustering," in *Data Mining*, Boston, Springer, 2007, pp. 257-288.

[27] C. Perlich, B. Dalessandro, O. Stitelman, T. Raeder and F. Provost, "Machine Learning for Targeted Display Advertising: Transfer Learning in Action," *Machine Learning,* vol. 95, no. 1, 2013.

[28] A. Likasa, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," *Pattern Recognition,* vol. 36, no. 2, pp. 451-461, 2003.

[29] M. Oskoei and H. Hu, "Myoelectric control systems - A survey," in *Biomedical Signal Processing and Control 2*, 2007.

[30] I. S. Dhillon, Y. Guan and B. Kulis, "Kernel k-means, Spectral Clustering, and

Normalized Cuts," in *International Conference on Knowledge Discovery and Data Mining*.

[31] F. Zhou, F. D. l. Torre and J. K. Hodgins, "Aligned Cluster Analysis for Temporal Segmentation of Human Motion," in *8th IEEE International Conference on Automatic Face & Gesture Recognition*, 2008.

[32] F. Zhou, F. D. l. T. Frade and J. K. Hodgins, "Hierarchical Aligned Cluster Analysis for Temporal Clustering of Human Motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 35, no. 3, pp. 582-596, 2013.

[33] K. Englehart, B. Hudgins, M. Stevenson, and P. Parker, "A dynamic feedforward neural network for subset classification of myoelectric signal patterns," *Conf. Proc. IEEE Eng. Med. Biol. Soc.*, vol. 1, pp. 819-820, 1995.

[34] H. Azami, K. Mohammadi and B. Bozorgtabar, "An Improved Signal Segmentation Using Moving Average and Savitzky-Golay Filter," *Journal of Signal and Information Processing*, pp. 39-44, 2012.

[35] C. Fancourt and J. C. Principe, "On the use of neural networks in the generalized likelihood ratio test for detecting abrupt changes in signals," in *IEEE-INNS-ENNS International Joint Conference*, 2000.

[36] B. Hannaford and S. Lehmen, "Short Time Fourier Analysis of the Electromyogram: Fast Movements and Constant Contraction," *IEEE Transactions on Biomedical Engineering*, Vols. BME-33, no. 12, 1986.

[37] S. M. Anisheh and H. Hassanpour, "Designing an adaptive approach for segmenting non-stationary signals," *International Journal of Electronics*, vol. 98, no. 8, pp. 1091-1102, 2011.

[38] W. El Falou, J. Duchêne, D. Hewson, M. Khalil, M. Grabisch and F. Lino, "A segmentation approach to long duration surface EMG recordings," *Journal of Electromyography and Kinesiology*, vol. 15, no. 1, pp. 119-119, 2005.

[39] F. D. Farfán, J. C. Politti and C. J. Felice, "Evaluation of EMG processing techniques using Information Theory," *Biomedical Engineering Online*, 2010.

[40] S. H. Nawab, S. S. Chang, and C. J. De Luca, "High-yield decomposition of surface EMG signals," *Clinical Neurophysiology*, vol. 121, no. 10, pp. 1602-1615, 2010.

[41] I. Olier, J. Amengual, and A. Vellido, "Segmentation of EMG time series using a variational Bayesian approach for the robust estimation of cortical silent periods.," in *8th European Symposium on Artificial Neural Networks*, 2010.

[42] H. M. Al-Angari, G. Kanitz, S. Tarantino and C. Cipriani, "Distance and mutual information methods for EMG feature and channel subset selection for classification of hand movements," *Biomedical Signal Processing and Control*, pp. 24-31, 2016.

[43] D. Staudenmann, I. Kingma, A. Daffertshofer, D. Stegeman and J. Dieen, "Improving EMG-Based Muscle Force Estimation by Using a High-Density EMG Grid and Principal Component Analysis," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 4, 2006.

[44] A. Phinyomark, C. Limsakul, and P. Phukpattaranont, "A Novel Feature Extraction for Robust EMG Pattern Recognition," *Journal of Computing,* vol. 1, no. 1, 2009.

[45] A. Phinyomark, P. Phukpattanont, and C. Limsakul, "Feature reduction and selection for EMG signal classification," *Expert Systems with Applications,* vol. 39, no. 8, pp. 7420-7431, 2012.

[46] B. Hudgins, P. Parker, and R. Scott, "A new strategy for multifunction myoelectric control," *IEEE Trans. Biomed. Eng.*, vol. 40, no. 1, pp. 82-94, 1993.

[47] E. Scheme and A. Radmand, "Investigation of optimum pattern recognition methods for robust myoelectric control during dynamic limb movement," in *Canadian Med. Biol. Eng. Soc.*, 2013.

[48] A. Radmand and E. Scheme, "Investigation of optimum pattern recognition methods for robust myoelectric control during dynamic limb movement," in *Canadian Med. Biol. Eng. Soc.*, 2013.

[49] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau, "EMG feature evaluation for improving myoelectric pattern recognition robustness," *Expert Systems with Applications,* no. 40, pp. 4832-4840, 2013.

[50] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE Trans. Biomed. Eng.,* pp. 848-854, 2003.

[51] L. Smith, L. Hargrove, B. Lock, and T. Kuiken, "Determining the optimal window length for pattern recognition-based myoelectric control: balancing the competing effects of classification error and controller delay," *IEEE Trans Neural Syst Rehabil*

Eng., vol. 19, no. 2, 2011.

[52] L. Hargrove, K. Englehart, and B. Hudgins, "A Comparison of Surface and Intramuscular Myoelectric Signal Classification," *IEEE Transactions on Biomedical Engineering,* vol. 54, no. 5, 2007.

[53] N. Nazmi, M. A. A. Rahman, S.-I. Y. S. A. Ahmad, H. Zamzuri and S. A. Mazlan, "A Review of Classification Techniques of EMG Signals during Isotonic and Isometric Contractions," *Sensors*, vol. 16, no. 8, 2016.

[54] F. A. Liu and G. Lui, "Analysis of extracted forearm sEMG signal using LDA, QDA, K-NN classification algorithms," *The Open Automation and Control Systems Journal*, vol. 6, no. 1, pp. 108-116, 2014.

[55] B. A. Lock, K. Englehart, and B. Hudgins, "Real-time myoelectric control in a virtual environment to relate usability vs. accuracy," in *MyoElectric Controls/Powered Prosthetics Sump.*, Fredericton, 2005.

[56] E. J. Scheme, K. B. Englehart and B. S. Hudgins, "Selective classification for improved robustness of myoelectric control under nonideal conditions," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 6, pp. 1698-1705, 2011.

[57] A. Soares, A. Andrade, and R. Carrijo, "The development of a virtual myoelectric controlled by an EMG pattern recognition system based on neural networks," *J. Intll. Inf. Syst.*, vol. 21, no. 2, pp. 127-141, 2003.

[58] A. Soares, E. Junior, A. Andrade, and A. Cardoso, "Virtual and augmented reality: A new approach to aid users of myoelectric prostheses," *Computational Intelligence in*

Electromyography Analysis - A Perspective on Current Applications and Future Challenges, 2012.

[59] E. Lopes, A. Lamounier, A. Cardoso and A. Soares, "Using augmented reality techniches to simulate myoelectric upper limb prostheses," *J. Bioeng. Biomed. Sci.*, 2013.

[60] D. Blana, T. Kyriacou, J. M. Lambrecht and E. K. Chadwicka, "Feasibility of using combined EMG and kinematic signals for prosthesis control: A simulation study using a virtual reality environment," *Journal of Electromyography and Kinesiology*, vol. 29, pp. 21-27, 2016.

[61] A. Simon, L. Hargrove, B. Lock and T. Kuiken, "The Target Achievement Control Test: Evaluating real-time myoelectric pattern recognition control of a multifunctional upper-limb prosthesis," *J. Rehabil Res. Dev.*, vol. 48, no. 6, pp. 619-627, 2011.

[62] E. J. Scheme and K. Englehart, "Validation of a Selective Ensemble-Based Classification Scheme for Myoelectric Control Using a Three-Dimensional Fitts' Law Test," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2013.

[63] S. Wurth and L. Hargrove, "Real-time comparison of conventional direct control and pattern recognition myoelectric control in a two-dimensional Fitts' law style test," 2013.

[64] M. Williams and R. Kirsch, "Evaluation of Head Orientation and Neck Muscle EMG

Signals as Command Inputs to a Human-Computer Interface for Individuals with High Tetraplegia," *IEEE Trans Neural Syst Rehabil Eng,*, vol. 16, no. 5, pp. 485-496, 2008.

[65] M. Zecca, S. Micera, M. C. Carrozza and P. Dario, "Control of Multifunctional Prosthetic Hands by Processing the Myoelectric Signal," *Critical Reviews in Biomedical Engineering*, vol. 30, no. 4-6, pp. 459-485, 2002.

Appendix A

Matched Filter

A *matched filter* involves correlating a known signal, or *template*, with an unknown signal to detect the presence of the template in the unknown signal. Since the data collected is cycling in nature, it is possible to use one full cycle as a template, then detecting the output to understand where there is a high match of features. Note that two separate matched filters, corresponding to both motions within a degree of freedom, were also observed, however it was found that they resembled so closely in features that they provided high outputs even for the opposite motion. By using a full cycle, this helped provide the template with additional data and still provided the output with high peaks between both classes, which is what was needed.

The raw data collected from the participant is scaled by the overall standard deviation of the whole data set (all four repetitions), then the five features are extracted from this data. The matched filter segmentation approach was implemented by considering the first cycle of the first repetition during training as the matched filter – this cycle was considered to be a perfect contraction. Figure B1 illustrates the prompt for the experimenter to select the matched filter start and end – this selection is a manual process, however, the experimenter should select the location which occurs near the beginning of the start of the first transition and near the end of the second transition.

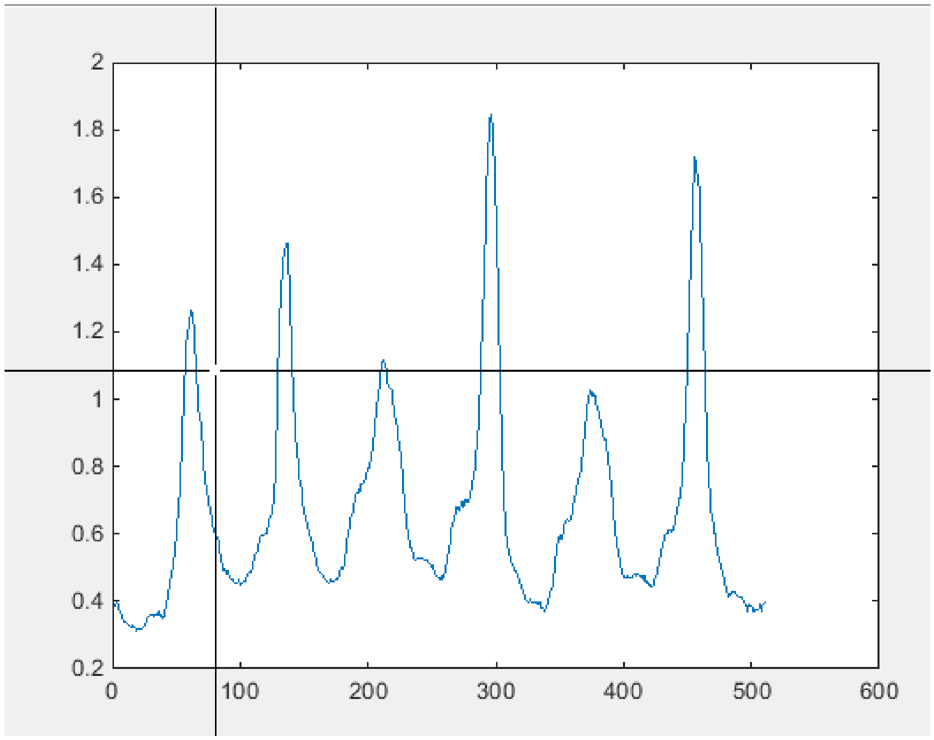

Figure A1– Selection Prompt for Matched Filter

Then, the first four PCA features of this section of data and of each repetition collected are extracted and considered to be the matched filter features used for the remainder of the DOF data. An example of a selected matched filter can be seen in Figure B2.

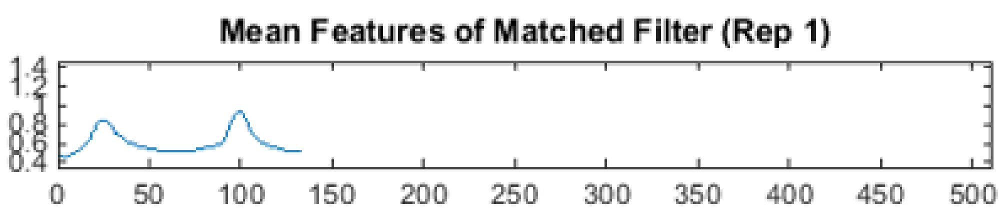

Figure A2 – Example of MAV of Matched Filter Features

Once the matched filter features have been extracted, they are run through the entirety of the PCA features of each repetition separately. Figure B3 demonstrates the output of the matched filter for the first repetition of data collected during training. Since the matched filter output is highest when features are more closely similar, this indicates it should be highest at a point of transition (as the matched filter was selected before and after a cycle transition).

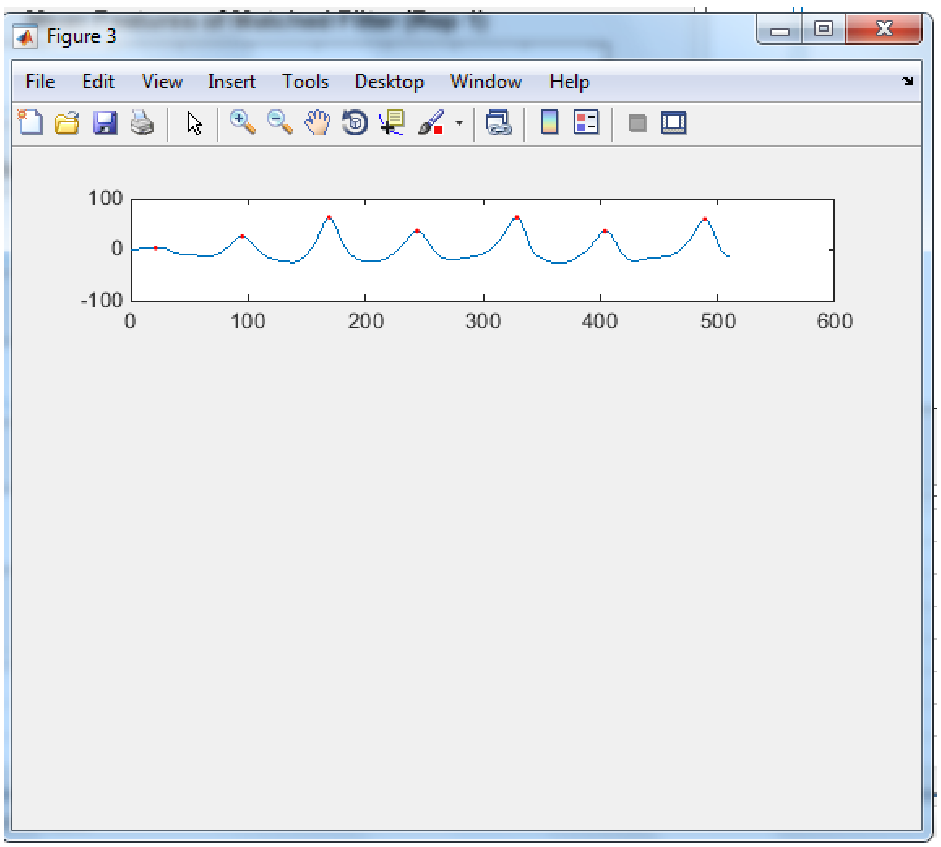

Figure A3- Experimental Prompt for Repetition #1 to Select Segmentation Locations

The peaks are automatically detected using an algorithm created in MATLAB, however, there were times when the peaks were not correctly labeled, therefore a prompt was given to the experimenter to indicate whether the peaks were properly found. If the experimenter indicated they weren't correct, 2 different options could be chosen:

1) Reselect all of the peaks manually

2) Keep all of the identified peaks, but add another selection

Once the new selection of peaks was made, or if they were determined to be correct initially, the same logic followed through for the remaining 3 training repetitions and these peaks were eventually used as segmentation locations for classifier training.

Curriculum Vitae

Kadie Wright

BScEE, University of New Brunswick 2014

Conference Presentations:

1. Wright, K., Englehart, K., Scheme, E., "A Comparison of Training Approaches for Pattern Recognition based Myoelectric Control," *Myoelectric Controls Symposium MEC '17,* Fredericton NB, Aug 15-18, 2017.
2. Wright, K., Englehart, K., Scheme, E., "Towards Improving the Training of Pattern Recognition Based Myoelectric Control", *21th Congress of the International Society of Electrophysiology and Kinesiology*, Chicago, July 5–8, 2016

Publications: None