USING MYOELECTRIC SIGNALS TO CLASSIFY PREHENSILE PATTERNS

by

Gene R. Shuman
A Dissertation
Submitted to the
Graduate Faculty
of
George Mason University
In Partial fulfillment of
The Requirements for the Degree
of
Doctor of Philosophy
Computer Science

Committee:

____________________ Dr. Zoran Durić, Dissertation Director

____________________ Dr. Lynn H. Gerber, Dissertation Co-director

____________________ Dr. Daniel Barbará, Committee Member

____________________ Dr. Carlotta Domeniconi, Committee Member

____________________ Dr. Sanjeev Setia, Department Chair

____________________ Dr. Kenneth Ball, Dean, The Volgenau School of Engineering

Date: ____________________ Fall Semester 2016
George Mason University
Fairfax, VA

Using Myoelectric Signals to Classify Prehensile Patterns

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at George Mason University

By

Gene R. Shuman
Master of Science
University of Maryland, 1981
Bachelor of Science
Pennsylvania State University, 1975

Director: Dr. Zoran Durić, Associate Professor
Department of Computer Science

Fall Semester 2016
George Mason University
Fairfax, VA

Dedication

This is for Sue, Amanda, and David, whose own accomplishments helped inspire this effort.

Acknowledgments

No work of this size or effort is possible without the help of many different parties. I would like to acknowledge some them here (and apologize to those who deserved to be called out, but were missed). My advisers, Zoran Durić and Lynn Gerber, for diving into the research documented in this dissertation and staying with me for several years, even at times when little progress was being made. My committee members and collaborators at GMU, Daniel Barbará, Carlotta Domeniconi, Jessica Lin, and Kathryn Laskey, for lending their help with the publication of many of the results herein and giving it credibility within the research community. Bonnie Hodsdon, OT, of the National Institutes of Health for helping to identify appropriate grips and movements for inclusion in this study. Fellow GMU graduate students Kenny Daffron and Dave Bagheri, my successful project collaborators during a particularly challenging course. Nikola Dragacevac, for helping visualize some of the signal data that was instrumental in the labelling process. For the five anonymous subjects who gave their time and effort in the data collection process. Finally, my professional colleagues who have shown me over the years what it means to be competent, effective, ethical, dignified, and fair.

Table of Contents

List of Tables

List of Figures

Abstract

USING MYOELECTRIC SIGNALS TO CLASSIFY PREHENSILE PATTERNS

Gene R. Shuman, PhD

George Mason University, 2016

Dissertation Director: Dr. Zoran Durić

People want to live independently, but too often disabilities or advanced age robs them of the ability to do the necessary activities of daily living (ADLs). Finding relationships between electromyograms measured in the arm and movements of the hand and wrist needed to perform ADLs can help address performance deficits and be exploited in designing myoelectrical control systems for prosthetics and computer interfaces.

This dissertation presents the results of applying several machine learning techniques to discover the electromyogram patterns present when performing typical fine motor functional activities used to accomplish ADLs. The primary data in this research is from electromyogram and accelerometer signals collected from the arms and hands of several subjects while they performed typical ADLs involving grips or movements of the hand and wrist. Four approaches were developed and tested. One involved classification of 100 ms individual signal instances. The second and third approaches used a symbolic representation called SAX to approximate signal streams. The second created an affinity matrix approach to model the co-occurrence of SAX symbols and classes to classify based on multiple adjacent signal values. The third used nearest neighbor classification with Dynamic Time Warping (DTW) as a distance measure to classify entire activity segments. A fourth approach used a Hidden Markov Model (HMM) to classify continuous movement segments by applying a 'belief'

calculation that uses that instance's signal reading as the observation model, the belief values of the previous instance's classes, and estimated transition probabilities. Accelerometer data were systematically used to aid in labelling the data since it clearly indicates the start and stop of dynamic movements.

The findings reported here support the view that grips and movements of the hand can be distinguished by combining electrical and mechanical properties of the task to an accuracy of 76.72% for 47 classes in a segmented approach and 75.09% in a continuous movement approach. Converting the signals to a symbolic representation and classifying based on larger portions of the signal stream improves classification accuracy. More precise labelling and applying the belief calculation gave credible results for the more complex continuous movement scenario. Classification errors were in all approaches predominantly concentrated within particular grip family groups. This is both clinically useful and opens the way for an approach to help simulate hand functional activities. With improvements it may also prove useful in real time control applications.

Chapter 1: Introduction

People want to live independently, but too often disabilities or advanced age robs them of the ability to perform basic activities of daily living (ADLs). ADLs are necessary personal functional activities, typically self-care, mobility, feeding, etc. They are largely performed through upper extremity (UE) movements. The hand, being the terminal UE device, is responsible for the detailed performance of ADLs and is essential for their successful completion. It is a complex part of the body that is capable of a nearly infinite number of postures and movements. Understanding the underlying physical mechanisms required for movements of the wrist and hand can help identify deficits in ADL performance with enough specificity to devise effective rehabilitation treatments that would provide many people with options for achieving and prolonging independence. That understanding can also be exploited in control applications, such as driving a prosthetic or robotic hand, and in the design of a "touch-less" computer interface.

1.1 Purpose of the Study

This dissertation presents results of research that explores the use of machine learning pattern recognition techniques to learn and interpret the relationship between a movement and the electrical signals emitted from the muscles that control the movement. The collected signals were used to train a classifier with the aim of creating a software agent that can decide which hand and wrist ADL movement or movements are being performed based on a particular set of signals. The expectation prior to conducting the study was that a large set of grips and movements could be identified using forearm EMG signal patterns to a sufficiently high accuracy to allow for use in control applications.

Four different supervised learning approaches are presented and results of classification training and testing reported. Data from five subjects were collected and used for training and testing. Four of the subjects were in good health and without disability. The intention was to collect data exclusively from non-disabled subjects. However, a fifth subject whose left arm and hand are partly disabled was available and included as a comparator with the non-disabled subjects.

The data were collected while the subjects performed grips or movements of the hand plus the rest position. For two of the four approaches the prediction accuracy of the techniques was improved by using a symbolic representation of the signal stream, incorporating a group of adjacent signals in the stream into the classification decision, and classifying entire activity segments while adopting Dynamic Time Warping (DTW) as a distance measure. Results from earlier tests of the first three approaches for 25 classes (24 grips and movements plus the rest position) were reported in [1].

As the research evolved, the number of activities grew to 47 (46 grips and movements plus rest) to include transition actions to account for continuous movement. The class labelling used accelerometer data as an aid in applying more precise labels for the 47 classes since it is an excellent indicator of the start and stop of movement. The fourth approach was developed with the purpose of being able to classify continuous movement. This approach employed a Hidden Markov Model (HMM) in which a classifier was used to generate class probabilities for the data followed by the application of a 'belief' calculation that used information about the previous state and transition probabilities to arrive at a classification decision.

All four approaches were applied to the 47 class data. Results are reported in Chapter 6 and discussed in Chapter 7.

1.2 Contributions of This Dissertation

The main contributions of this dissertation are:

- Developed and evaluated several machine learning classification techniques to identify

47 fine hand grips and movements used to perform typical Activities of Daily Living (ADLs). Constructed the feature set from electromyogram (EMG) signal data collected from the arms and hands of several subjects while they performed typical ADLs involving grips or movements of the hand and wrist. Collected accelerometer (ACC) signals as a separate data modality to indicate the start and stop of dynamic movements and aid in providing ground truth labels for the grips and movements.

- Designed and executed an experimental protocol to collect EMG and accelerometer data from five subjects. Directed subjects performing a scripted set of eight hand grips and related movements over eight two minute runs that covered the spectrum of grip types: from power to fine. Processed, reduced, and coded the subjects' signal readings to produce labelled classification datasets for the subjects.

- Developed and tested several classification approaches. One classified individual signal instances using Random Forest. In two others converted the signal streams to a symbolic representation called SAX prior to classification. Created an affinity matrix approach to model the co-occurrence of SAX symbols and classes to classify based on multiple adjacent signal values. Developed and tested a nearest neighbor approach that used Dynamic Time Warping (DTW) as a distance measure to classify entire activity segments.

- Developed and tested a Hidden Markov Model (HMM) to classify continuous movement data. Classified signal instances by applying a 'belief' calculation that used an observation model based on the current instance's EMG signal readings, a transition matrix containing probabilities of moving from one action to another in adjoining time periods, and the belief calculation of the previous instance. Measured the improvement in classification accuracy provided by the belief calculation versus Random Forest.

1.3 Document Map

The structure of the dissertation is as follows. Chapter 2 covers the background material used in this research in the areas of anatomy, instrumentation, and automated processes. Chapter 3 reviews related research. Chapter 4 describes the methodology employed, including the specific experimental set-up, instruments, and automated methods used. The latter includes material on how data was collected, activities that were performed and tracked, and how the data were labelled. Chapter 5 is a description of the four supervised learning approaches used to classify the data. Chapter 6 covers the results of the experiments. Finally, Chapters 7 discusses the results, while Chapter 8 suggests potential future work.

Chapter 2: Background

This chapter describes the anatomical, instrumentation, and automated processing areas used in this study.

To move a voluntary muscle, the brain sends a low-level electrical, or myoelectric, signal over the central nervous system to the muscle tissue that causes contraction or relaxation, resulting in the movement. Electromyography is the study of those signals [2]. They can be measured while the muscle is contracting or relaxing and are called electromyograms, or EMGs. EMGs are very low-level — less than 10 mV ($\approx$ .0001 of U.S. household current) — and must be amplified to be measured. An electromyograph amplifies and measures EMGs and has two types of sensors: (1) needles inserted directly into muscle tissue and (2) surface sensors attached externally to the skin as close to the measured muscle as possible. Needle EMGs are inserted into the muscle and target specific areas. Surface EMGs (sEMGs), by contrast, do not distinguish between specific muscles. However, sEMG sensors have been shown to provide as good results as needle-based approaches for pattern recognition applications [3]. Their noninvasive nature and demonstrated good results make them a suitable mechanism for capturing EMG signals and are therefore the choice in this research.

The relationship between a muscle's EMG and the resulting movement is often not obvious. Complex movements such as fine hand and finger movements usually involve several muscles working in concert, often firing sequentially, which makes finding a relationship difficult [4]. One approach to finding the relationship is to use supervised learning, or classification. In this technique a set signals from training instances are associated with an outcome movement to learn a classification model. The model is a function constructed from the training instances that approximates some true underlying relationship. When presented with a similar, but previously unseen instance in the future, the model (the learned function) is used to predict the outcome by translating the set of new signals into the appropriate grip or movement.

2.1 Muscle Actions

Most fine motor tasks not only involve the fingers and wrist, but the forearm, upper arm, and shoulder are also involved for positioning and/or stabilization of the distal segments. Motion about a joint is caused due to force generated by muscle activity. Two kinds of muscle movements can be defined: agonist and antagonist. Agonist muscles are defined as being skeletal muscles that induce motion through the process of its own contraction. An antagonist muscles is one that works in direct opposition to the agonist muscle and is responsible for returning the limb segment to the anatomical position. Extensors and Flexors are antagonist pairs, meaning that if the motion being elicited is of flexion, then the flexor is the agonist and the extensor is the antagonist; vice versa for extension. Forearm muscles that control the wrist and hand are listed in Table 2.1. The location of key muscles is shown in Figures 2.1 and 2.2.

2.2 Anatomy of the Forearm, Wrist and Hand

Since this dissertation focuses on movements of the wrist and hand, an overview of their anatomy is given in this section. The wrist is controlled by three sets of forearm muscles responsible for its flexion, extension, and pronation and supination (twisting inward and outward). Flexion is controlled by the flexor carpi ulnaris and flexor carpi radialis. The palmaris longus is a flexor responsible for tension. Extension is controlled by the extensor carpi radialis longus and brevis and the extensor ulnaris and brevis. Pronation is largely controlled by the pronator teres and brachioradialis.

The hand is controlled by a set of extrinsic muscles located in the forearm and a set of intrinsic muscles that are part of the hand itself. This study focuses on predicting hand grips and motions from forearm sEMGs, so the extrinsic muscles will be described. The extrinsic

Table 2.1: Forearm Muscles Controlling the Wrist and Hand (Adapted from [5])

Muscle	**Action**
flexor carpi ulnaris	flexion of wrist; ulnar deviation of hand
flexor carpi radialis	flexion of wrist; radial deviation of hand
palmaris longus	tension of the palmar fascia
extensor carpi radialis longus and brevis	extension of wrist; radial deviation of hand
extensor carpi ulnaris and brevis	extension of wrist; ulnar deviation of hand
pronator teres	forearm pronation
brachioradialis	pronation or supination of the forearm
flexor digitorum superficialis	flexion of the PIP and MCP joints (fingers)
flexor digitorum profundus	flexion of the DIP, PIP, and MCP joints
flexor pollicis longus	flexion of IP & MCP joints of thumb
extensor pollicis longus	extension of IP & MCP thumb joints; thumb abduction
extensor pollicis brevis	extension of MCP joint of thumb
abductor pollicis longus	abduction of thumb
extensor indicis proprius	extension of index finger
extensor digitorum communis	extension of fingers
extensor digiti quinti proprius	extension of V finger

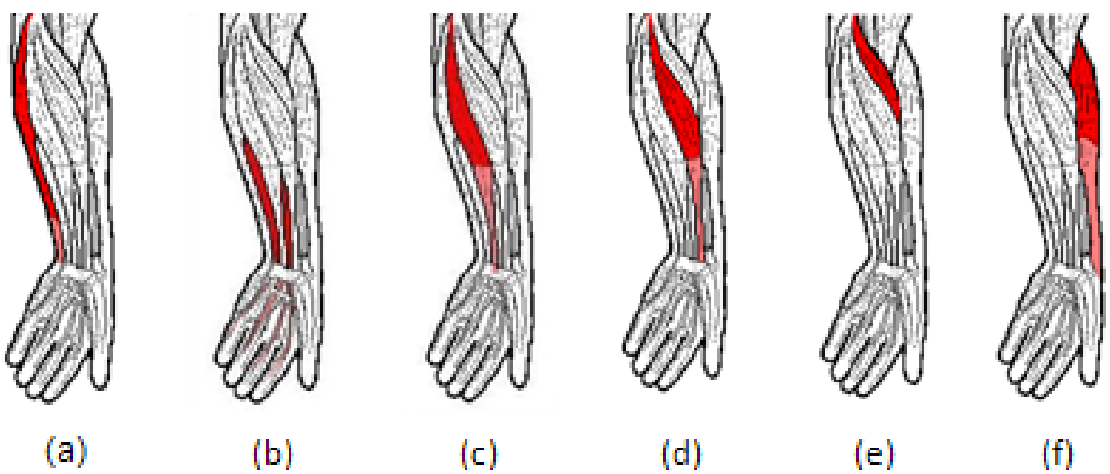


Figure 2.1: Anterior Forearm Muscles Controlling the Hand and Wrist: (a) flexor carpi ulnaris (b) flexor digitorum superficialis (c) palmaris longus (d) flexor carpi radialis (e) pronator teres (f) brachioradialis (Adapted from [6]).

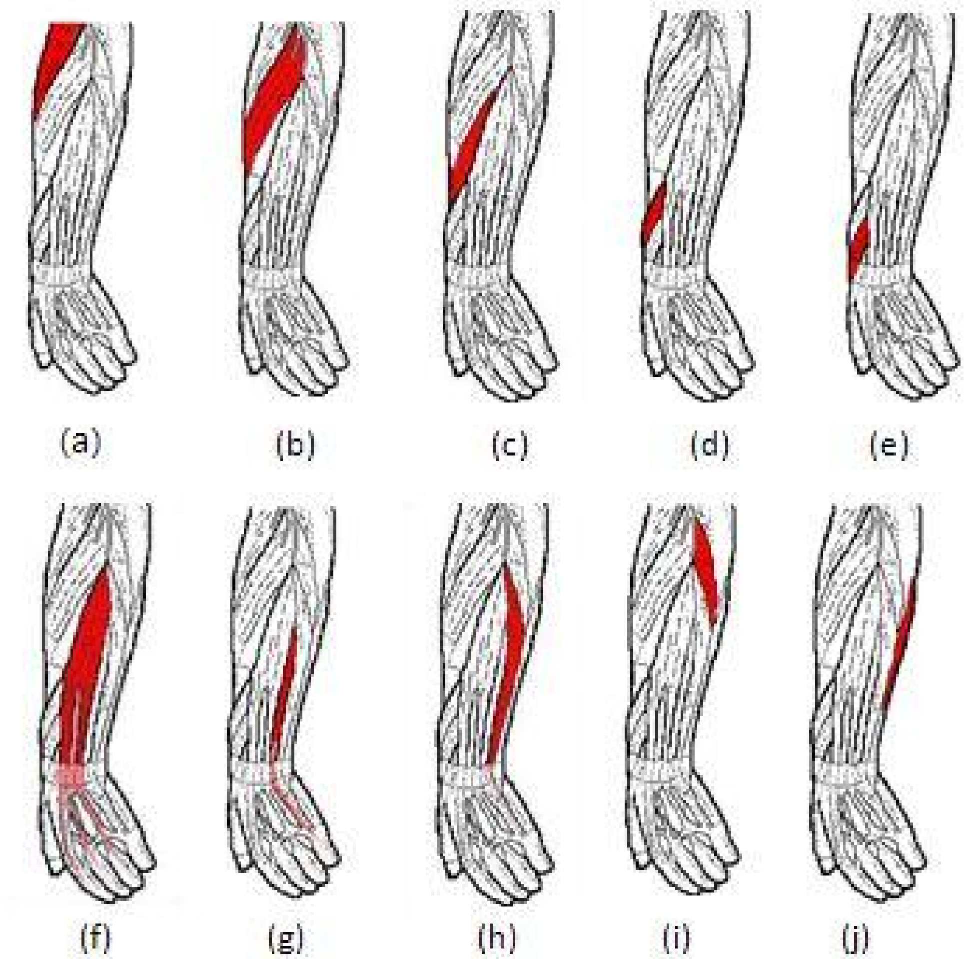


Figure 2.2: Posterior Forearm Muscles Controlling the Hand and Wrist: (a) brachioradialis (b) extensor carpi radialis longus (c) extensor carpi radialis brevis (d) abductor pollicis longus (e) extensor pollicis brevis (f) extensor digitorum (g) extensor digiti minimi (h) extensor carpi ulnaris (i) anconeus (cubitalis rotani) (j) flexor carpi ulnaris (Adapted from [6]).

muscles consists of three sets that control the fingers and thumb: flexors, extensors, and abductors. Flexors control the inward bending of individual fingers and the thumb, and include the flexor digitorum superficialis, flexor digitorum profundus, and flexor pollicus longus. Extensors control the extension of individual fingers and the thumb, and include the extensor pollicis longus and bevis, extensor indicis proprius, and extensor digitorum communis, and extensor digiti quinti proprius. Abduction, the lateral movement of the thumb, is controlled by the abductor pollicis longus. [5].

The above is summarized in Table 2.1. The locations of key muscles is shown in Figures 2.1 and 2.2.

2.3 Prehensile Movements

Prehensile movements of the hand are those in which an object is seized and held partly or wholly by the hand. These are used in a broad range of activity and involve handling objects of varying shapes and sizes [5]. Napier [7] identfied two distinct patterns of prehensile movement: power grip and precision grip.

The power grip involves grasping an object with flexed fingers and thumb. The wrist is usually slightly flexed to allow it to apply some tension. The precision grip involves manipulating small objects between the thumb and fingers in a finely controlled manner. Wrist position can vary to increase range. There are variations of each of these general types. For the powere grip, these include the coal-hammer, bunched fist, procesion-power, fencing, and jar grips. Precision variations include the scissors, pencil, tip-to-tip pinch, palmar pinch, lateral (key) pinch, and pulp (ulnar) pinch grips. These are representative grips used in ADLs and are explored in this study and are shown in Figures 2.3, 2.4, and 2.5.

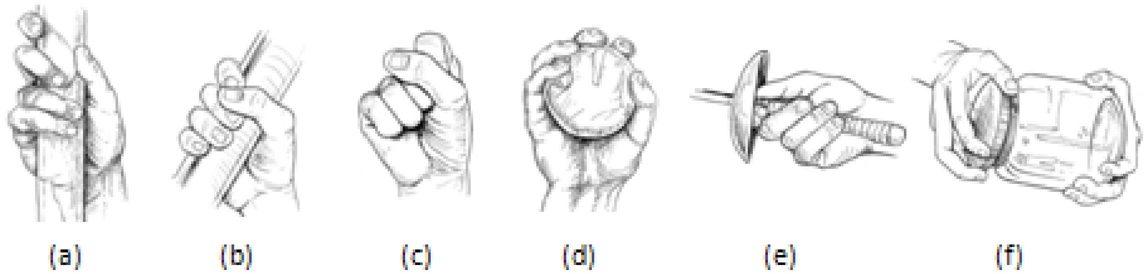


Figure 2.3: Power Grips: (a) typical power grip (b) coal-hammer (non-precision) (c) bunched fist (non-precision) (d) typical precision power grip (e) fencing grip - power grip with element of precision (f) jar grip (Adapted from [5])

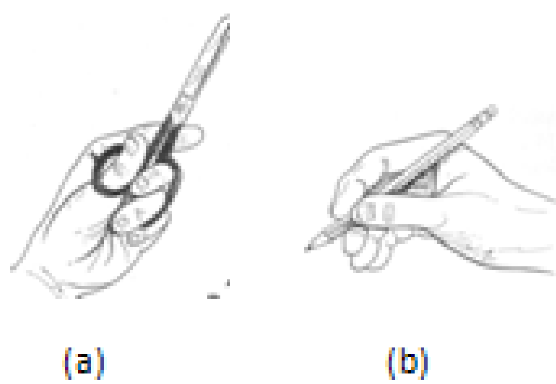


Figure 2.4: Tripod Grips: (a) scissors (b) perncil (Adapted from [5])

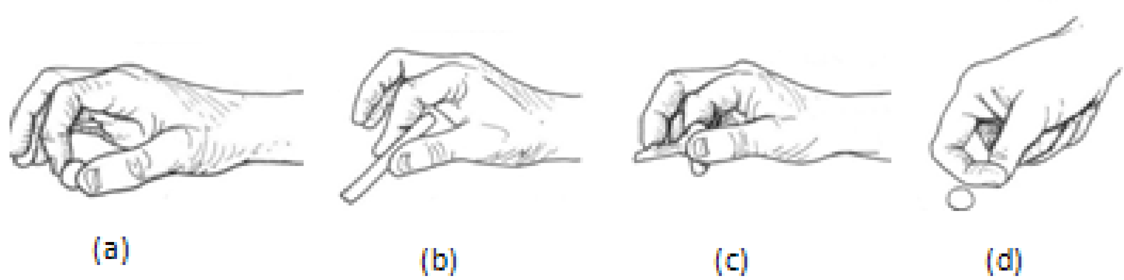


Figure 2.5: Precision Grips (small objects): (a) tip-to-tip pinch (b) palmar pinch (c) lateral (key) pinch (d) pulp (ulnar) pinch (Adapted from [5])

2.4 Electromyogram (EMG) Signals

Surface electromyography (sEMG) is the study of electromyograms that are collected from sensors attached to the skin, as opposed to needle or wire based approaches that have connections placed within the muscle tissue itself. For control applications surface sensors have been just as effective as needles, but are far less intrusive [3, 8]. The electrical signal captured is the composition of all signals generated from the muscle fibers below the sensors. Muscle fibers that are closer to the sensors contribute more to the overall sEMG signal than those further away. The signal becomes stronger as more and more fibers are recruited (signaled by the nervous system).

There are two different types of muscle fibers that produce different electrical signals. Type-I fibers (slow-twitch) have slower contraction velocity, are less prone to fatigue, and are prevalently used in aerobic activities. Type-II fibers (fast-twitch) have a must faster contraction time and are used for high-force, fast-response tasks, but are more prone to fatigue. Different types of fibers will contribute differently to the electrical signal due to their contraction velocities. Fat and other non-muscle tissue (adipose tissue) can also affect the captured signal when using surface sensors. Adipose tissue behaves like an insulator, absorbing some of the electrical activity. Each of the above factors can affect the electrical signal, causing a great deal of variability; therefore there has been significant amount of research to identify key features of the electrical signal that produce useful information. [4] [2]

2.4.1 Equipment (DelSys)

Muscle actions will be captured using the DelSys Trigno Wireless™ sensors and base station for both sEMG and accelerometer (ACC) signal collection [9]. This system was chosen since its sEMG sensors can be placed on the skin allowing for noninvasive data capture. The sensors are placed on the skin above several superficial (closest to the skin) muscles that control the fingers and wrist [2]. The unit is portable and supports up to 16 channels

Figure 2.6: DelSys Trigno kit used for data collection (left, top and bottom) and a subject (right) performing a jar lid turn with ten sensors attached to the action arm.

of sensor input. The base unit communicates with the DelSys EMGWorks^{TM} Acquisition package via a USB interface that, in turn, drives the collection and control of the sensor signals and allows for the real-time monitoring of the signal. sEMG signals in this research were collected at rate the of 2 kHz, ACC signals at 148.1 hz.

The EMGWorks Analysis software package is provided by the vendor to allow for the digitization and processing of signals. The package allows for easy visualization of the signals and provides computation of standard values such as root-mean square (RMS) for a specified time window. [9]

2.5 Classification

Classification is a machine learning technique in the category of supervised learning. An external teacher, or supervisor, provides a set of examples with an associated class label for each example. The goal is to train a software system to recognize the implicit relationships between the examples and the labels. A set of labelled data is given and the classifier is trained to recognize the relationship between the data structure and the class label. Once trained, the system, or classifier, can provide the class label of a previously unseen example with a high degree of accuracy.

Formally, let $\mathbf{x} = \{(\mathbf{x_1},\ y_1),\ (\mathbf{x_2},\ y_2),\ ...,\ (\mathbf{x_n},\ y_n)\}$ be a set of n examples $\mathbf{x_i}$ with associated labels y_i. The vector $\mathbf{x_i} \in \Re^m$ and $y_i \in \mathbf{N}$, where m $\in \mathbf{N}$ represents the number of attributes or features used in the classification training example, and the number of class labels is represented by $j \in \mathbf{N}$. In many classification problems j is a small number, often 2. The goal is to derive a function $f_E(\mathbf{x})$ that approximates the true function f($\mathbf{x}$) as closely as possible. The function $f_E(\mathbf{x})$ represents a separating boundary or decision surface between the classes, y_i, of the various $\mathbf{x_i}$s.

Classification is treated in detail in a number of standard references, including [10], [11], and [12]. These references are the primary sources for the material summarized in the following subsections that describe classifiers.

The primary classifier types considered for used in this study are briefly summarized in the following subsections.

2.5.1 Neural Networks

The Multilayer Perceptron (MLP), sometimes called an Artifical Neural Network (ANN) or just Neural Network, builds on the concept of the Perceptron. The Perceptron respresents a hyperplane that separates the training instances and is relatively fast to train. However, the Perceptron can only find a hyperplane boundary if the training data is linearly separable, which is a serious shortcoming. The MLP, by contrast, can succeed in these cases since it is not limited to linear boundaries.

The MLP is implemented as a feedforward network with input units, output units, and hidden units. Each input unit corresponds to a classifier feature and the output units are determined by the number of classes to be predicted. For a two class problem, one output unit suffices since it can be used to render a yes/no or positive/negative decision. For more than two classes, there are as many output units as classes. The hidden units are the key variable in setting up and tuning an MLP. They can vary in number and in how many layers in which they are arranged. An MLP with a single hidden layer of units can represent most classification problems, so the variable is usually the number of units in the hidden layer. More units means that a more complex decision boundary can be constructed and good accuracy against the training data achieved. However, too many hidden units can lead to over-fitting. In over-fitting, the classifier fits the training data very well, but does not generalize well - that is, does not give as high accuracy against new, previously unseen data instances. The goal is to find enough hidden units to give high accuracy while not specifying so many that over-fitting occurs. [10]

There is no specific way to determine the optimum structure of the MLP network except through experimentation with the number of hidden units and the adjustment of parameters. The path through the network from the input units to the output units requires a transition value from each unit to each of the units in the next layer - from input to hidden units, then from the hidden units to output units. The values are determined by a weighted function of the input value and use of a continuous activation function such as the sigmoid function. The activiation function weights are learned using the backpropagation algorithm. Weights are adjusted using a gradient-descent algorithm and minimizes the squared-error loss value, the difference between the predicted value and the true value, until some minimum threshold is reached. [10] [12]

2.5.2 Support Vector Machines

The Support Vector Machines, or SVM, is a two-class classifier that attempts to find a linear separation boundary between the instances. The optimization criteria is to maximize the distance between the decision boundary and the nearest training instances of both classes. The SVM is often refered to as a maximum margin classifier. The training instances closest to the boundary are called the support vectors and are the only instances needed for classification after classifier training is complete.

To achieve separability of training instances, SVMs preprocess the instances into a higher dimensionality than the original space. A kernel function is used for the transformation that allows for new instances to be quickly and simply classified using a dot-product computation.

SVMs generalize well and do not suffer from overfitting, as do MLPs. They can be adapted to multi-class problems using a one-versus-one or one-versus-all postprocessing approach.

2.5.3 Decision Trees and Random Forests

A Decision Tree (DT) is trained by iteratively selecting features that best separate classes at each node. Criteria such as entropy and Gini purity are used to measure this and drive feature selection. Many DTs are binary, using a single feature at each node, resulting in decision boundaries that are parallel to the feature axes, making them suboptimal. Once trained, however, DTs can classify new instances quickly and the resulting decision path from tree root to class node can be traced and read as a decision rule that can provide information on the structure of the classification problem. Unfortunately, like MLPs, DTs can easily be over-fitted to the training data.

The success of DTs can be improved if more than one tree is generated and applied to the classification. Various trees are randomly generated by selecting different features for each tree, yielding potentially different classification decisions. In so-called Random Forests (RFs), the classifcation results are post-processed using a majority voting scheme for the final decision. RFs can improve the accuracy of DTs [13] and play a significant role in this study.

2.5.4 Nearest Neighbor

Nearest-Neighbor is based on the concept of similarity. The simplest variety is the one nearest-neighbor, or 1-NN. In this case there is no training phase, but rather instances to be classified are compared to the "training" instances at the time classification is needed and the classification decision is the class of the one training instance closest to it. "Closest" is determined by a distance measure selected as a parameter, but is often the Euclidean distance. 1-NN is often used as a benchmark for other classifiers since it appears to provide a reasonable classification performance in most applications [12].

1-NN can be generalized to classifying based on the class label of 'k' nearest neighbors, where $k \geq 2$. K-NN operation is similar to 1-NN except that the majority of the 'k' nearest neighbors determines the class label of the instance being classified. 'k' is specified as a parameter to the algorithm and its optimum value must be experimentally determined.

2.5.5 Detection, Collection, Segmentation, and Processing

MES data is collected from sensors and recorded as a stream (or channel) of digitized values. Analyzing the MES data and preparing it for classification requires the selection of the specific time window of the sample corresponding to the grip or movement of interest. This window covers the time during execution. Because the data is collected as a continuous stream and we wish to track a sequence of grips and movements, the stream must be segmented for processing. A segment is a time slot of data from which a feature or feature set is extracted. Real-time constraints of control systems dictate that a processing delay between selection and classification decision be 300 ms or less [14]. Even though this study is not solely concerned with control systems, the 300 ms limit is kept in mind as a guideline rather than a firm requirement. There are three aspects of data segmentation to be considered: segment length, state of data, and windowing technique [15].

Segment lengths between 32 ms and 250 ms have been found to work well in practice [14], although classifier accuracy degrades with lower segment lengths. Englehart showed [16] that steady-state signal data emanating from a constantly maintained muscle contraction is

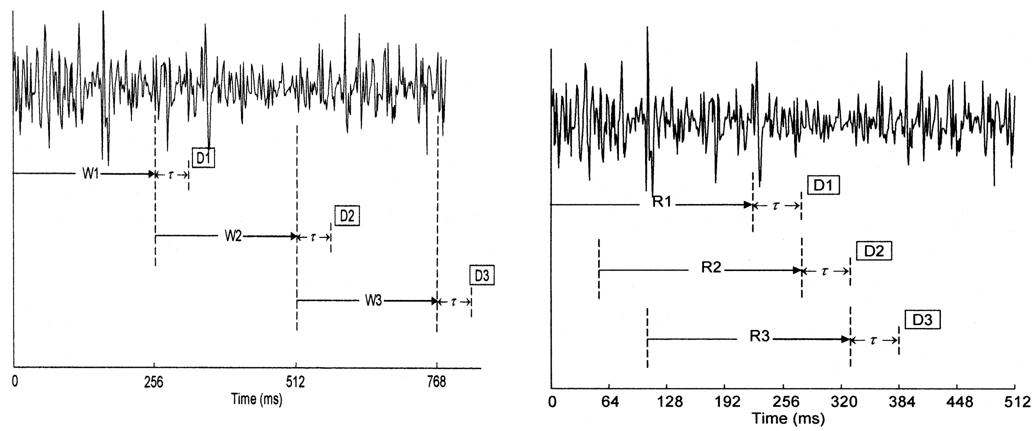


Figure 2.7: Two possibilities for windowing. At left shows 256 ms windows with no overlap, at right shows the same window length but with a 32 ms overlap. (Adapted from [14])

classified more accurately than transient data which emanates at the onset of a contraction. An MES has an undetermined state during transition between contraction levels and most errors occur when switching between classes. This is consistent with the results reported later in this study. Therefore, classifying steady-state data with a segment length of ≈ 128 ms will yield fast, accurate results. [15]

Windowing can either mean processing time-adjacent segments of data or overlapping segments. Overlapping leads to a denser input stream and when combined with post-processing can result in high accuracy. Englehart [14] showed that processing overlapping windows as small as 32 ms with post-processing of segments using a majority voting scheme works well. Figure 2.7 illustrates the concept for both overlapping and non-overlapping windows.

2.5.6 Feature Generation

Feature selection is a key part of creating a classifier and strongly influences the classification results. The sEMG signals are the features of interest in Myoelectric Systems (MES) studies and can be used directly or combined or transformed so the classes of interest can be easily separated by a classifier.

Hudgins [17] identified several time-domain features that he used for prosthesis studies. Mean absolute value (MAV) or average rectified value (ARV) and root mean squared value (RMS) are very prevalent time-domain features that are commonly used, and are given by:

$$\bar{X}_i = \frac{1}{L}\sum_{k=1}^{L}|x_k|,$$

$$RMS(i) = \sqrt{\frac{\sum_{k=1}^{L} x_k^2}{L}},$$

where $\bar{X}_i$ is the MAV of the i^{th} window of size L, and x_k is the k^{th} electrical signal sample within the given window.

Difference ARV is another measure characterized by the difference in ARV between windows:

$$\Delta\bar{X}_i = \bar{X}_{i+1} - \bar{X}_i,$$

Hudgins used a simple frequency measure given by the number of zero crossings (ZC). Due to the considerable artifacts that can affect the signal, a threshold must be used to ease the effect of noisy data. Therefore a zero crossing is detected if the difference between subsequent samples is greater than a defined threshold T:

$$ZC(k) = \begin{cases} 1 & \text{if } |x_{k+1} + x_k| > T; \\ 0 & \text{otherwise.} \end{cases}$$

$$ZC_{Window}(i) = \sum_{k=1}^{L} ZC(k)$$

Another simplified means to collect information about the frequency of the electrical signal, is the number of sign slope changes (SSC) between three consecutive samples, where

x_k is a relative max or min.

$$SSC(k) = \begin{cases} 1 & \text{if } (x_k > max(x_{k-1}, x_{k+1}) \text{ or } x_k > min(x_{k-1}, x_{k+1})); \\ & \text{and } max(|x_k - x_{k+1}|, |x_k - x_{k-1}|) > T|; \\ 0 & \text{otherwise.} \end{cases}$$

$$SSC_{Window}(i) = \sum_{k=1}^{L} SSC(k)$$

Finally, Hudgins examined the total length of the signal over the window, which is calculated as the sum of absolute voltage differences between each sample within the window:

$$WL(i) = \sum_{k=2}^{L} |x_k - x_{k-1}|$$

These time-domain features have been shown to have high classification accuracy among steady states (i.e. muscle is in tension to hold the segment stationary) [17,18]. Time-domain features work well for steady states, and the electromyogram is a quasi-stationary signal, therefore it is promising to use sliding windows to capture these features over time.

Frequency domain features are typically used to study muscle fatigue and infer changes in muscle recruitment [15]. This is because the muscles that commonly fatigue are composed of Type-II, which have faster firing rates which will be present in frequency analysis. Common features extracted include mean (MNF) and median (MDF) frequency:

$$MNF(i) = \frac{\sum_{j=1}^{J} I_j F_j}{\sum_{j=1}^{J} I_j}$$

$$MDF(i) = F_k; \operatorname*{argmin}_k \left\{ \sum_{j=1}^{k} I_j > \frac{\sum_{j=1}^{J} I_j}{2} \right\}$$

where the frequency F_j is computed by the Fast Fourier Transform, and I_j is the amplitude at the frequeny F_j.

Parametric methods have also been used for frequency analysis of the muscle action; typically an autoregressive (AR) model is used [18], and represents random processes. The Akaike Information Criterion is commonly used to select the order of the model. Sample x_k is modeled by:

$$x_k = c + \sum_{q=1}^{p} \phi_q x_{k-q} + \epsilon_k,$$

where is c is a constant factor, p is the order of the model, ϵ_k is white noise, and ϕ_q are the autoregressive parameters that estimated and used to describe the signal.

When processing the MNF and MDF, information regarding the time-domain is lost, therefore the Short-Time Fourier Transform (STFT) is used to slide a window through the data and observe the frequency changes over time. Wavelet transforms are also used to examine the time-frequency domain. The definition of a continuous wavelet transform (CWT) follows:

$$X_{CWT}(s, \tau) = \int \frac{x(t)\psi^*(t)\left(\frac{s-\tau}{a}\right)}{\sqrt{a}} dt,$$

where s is a scale factor, τ is the translational factor, and ψ is the wavelet function to use. * denotes the complex conjugate.

A final step after the feature creation process is dimensionality reduction, or reducing the total number of features used in classification. Features vary in their strength, or contribution to the classifier's accuracy. Selecting a subset of the strongest can lead to higher accuracy while at the same time reducing the time needed to train the model and perform individual classification decisions. Since the features used in this study are based on physical signals collected from a variety of sensor locations, signal strength and quality varies. Gauging the contributions of individual features and their associated sensors can provide information on muscular contributions to the various grips and movements used to perform ADLs.

Three methods, Principle Component Analysis (PCA) [16], uniformisation [19], and ANOVA [13], have been shown to be useful in past studies. However, this study employed the MAV (ARV) of eight signal channels as the feature set and attempts at reducing the number in the set did not yield improved results. Since eight is not a high number of features, no further reduction was performed.

2.5.7 Validating Classification Models

A good classifier is one whose predictive accuracy on unseen input examples is high. Predictive accuracy, or just accuracy, is the percentage of correctly classified instances. Error rate, which is the percentage of misclassified instances, is an equivalent measure. A problem arises, however, in measuring accuracy against unseen data while the classifier is being trained and before using it in a real-life production mode. A good method is to divide the input data into training and validation sets, using the classifier's accuracy when processing the validation set as the metric. However, this method reduces the amount of data available for training and risks skewing the training or validation set so neither is representative of the mix of the total set.

Cross-validation avoids these problems. It is a fast and robust classifier evaluation technique that is useful when training data is limited. In the case of 10-fold cross-validation, one of the more commonly used schemes, 10 separate training and classification runs are conducted [10] [11]. For each run 90% of the training instances are used in training and 10% are held for evaluation. The runs are constructed so that each of the instances is held out one time, while participating in training 9 times. The classifier accuracy is measured by summing up the number of correct and incorrect classifications for the held-out 10% over the 10 separate runs.

2.6 SAX

SAX (Symbolic Aggregate approXimation) is a method of representing a time series using a set of symbols assigned based on a discrete range of the sensor values [20]. It was used for part of this research and is described in this section. Since the signal stream coming from the performance of a grip or movement represents a time series, the SAX approach is appropriate here. Only the idea of symbolic representation of the signal rather than real number values was borrowed from SAX, an approach that includes other concepts not used here.

In this implementation of n-symbol SAX the range of possible signal values is divided into n intervals in such a manner that all symbols are equally probable. As will be seen later, there are eight signal sensor values, each of which is represented by an MAV value of a 100 ms signal slice, with each slice overlapped by 50 ms. The result is 20 SAX 8-tuples per second. The eight MAV values are converted to an 8-tuple symbol for use as a feature. While the feature dimension remains at eight, the total feature space of possible values is reduced to a finite number determined by the size of the alphabet of symbols. For each channel the signal probability is estimated using the histograms of signal values. An example of applying this method to discretize signal data is shown in Fig. 2.8. Each SAX window covers 100 ms, one MAV segment per window.

In this research the alphabet size n was varied between 5 and 15. Since each 8-tuple signal value is replaced by an 8-tuple n-value symbol, a total of n^8 different 8-tuples are possible. MATLAB® code was developed to implement the above as well as the second and third learning approaches described in the following two sections. The conversion of the eight signal channels to SAX 8-tuples was done separately for each subject. The symbol ranges were established separately for each subject as well.

For an $n = 5$ size alphabet, examples of SAX 8-tuples are 'AAAAAAAA', 'ABDEDBAC', 'CDEEDCDE', and 'ABCDEEDC'. In the remainder of this dissertation a SAX 8-tuple of symbols is understood to represent one 100 ms MAV signal segment and will be referred to as a word.

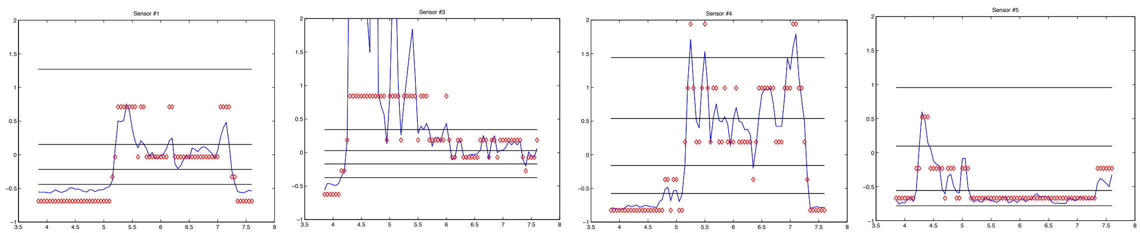


Figure 2.8: Assigning symbols to the five sensor streams using an alphabet size of five. Graphs for four of the eight signal channels are shown for a 3.5 to 8 second segment, corresponding to the end of the neutral/rest (NR) activity and the entire hammer grip (HG), for the average of all sensor values for the training runs. The SAX window size is 100 ms. The horizontal lines show the cut-off boundaries for the five symbol alphabet, A through E. The selected graphs show the diversity of cut-off values that vary for each of the eight sensors.

2.7 Toolsets

This section describes the two primary tools used to perform classification: Weka and MATLAB.

2.7.1 Weka

Weka (Waikato Environment for Knowledge Analysis) [21] is a collection of machine learning software developed at the University of Waikato, New Zealand. It's a java based open source suite issued under the GNU General Public License.

The main interface is the Explorer panel that consists of Preprocess, Classify, Associate, Cluster, Select Attribute, and Visualize panels. To build a classifier using Weka, training data is first put into the ARFF (Attribute-Relation File Format) format and imported using the Prepocessor. Once in the Weka ARFF format, it can be processed and used to rapidly train a variety of classifiers. The data can be visualised using bar graphs, attributes selected or deleted, and dimensionality reduction performed.

A variety of classification algorithms are implemented, including the Artificial Neural Network Multi-Layered Percceptron (ANN/MLP), Support Vector Machine (Sequential

Minimization Optimization), Decision Trees, Random Forest, Nearest Neighbor, and others. Weka classifier training involves building a model and, if specified, performing cross-validation to gauge performance. After the cross-validation is complete, details of the model are output, including the time needed to build the model, and a confusion matrix is displayed showing how many instances were correctly and incorrectly classified. The accuracy, number of correctly classified instances divided by the total, is displayed and is the main performance metric.

Weka can be invoked from a java routine, allowing it to be imbedded in a larger process. Various plug-ins are supported, including LIBSVM.

2.7.2 MATLAB

MATLAB® is a proprietary programming language developed and sold by Mathworks [22] as an environment for numerical computations. It allows for fast matrix processing, plotting of functions, implementation of algorithms, and interfacing with other languages.

Custom MATLAB code was developed in support of this research for Approaches 2, 3, and 4. The MATLAB *Treebagger* class and *predict* method were essential parts of Random Forest classification in Approach 4.

Chapter 3: Related work

Relating EMGs to movement has two basic use cases. One involves control applications in which the EMG readings are used to drive a prosthesis, robotic hand, or a touchless computer interface. These tend to be real time applications that require that the signals be acquired and processed, and a related activity initiated in a very short time frame, often a fraction of a second. The second involves using EMGs to assess motor and sensory signals. Abnormalities of the signal may contribute criteria that may assist in making diagnoses or tracking recovery. Additionally, research has provided data about the relationships between signal amplitude and muscle strength. The relationships are complex, but may provide some clinically relevant information [23]. This second use case is usually done as a batch process that allows for the complete collection of a set of EMG data that can be read and interpreted at a later time.

EMG measurement began in the early 1800s with the invention of the galvanometer. In the 1920s the newly invented cathode ray oscilloscope was used to give a visual representation of EMGs. As instrumentation improved in the 1930s and onward, researchers began to use sEMGs more widely for the study of normal and abnormal muscle function, dynamic movement, and for the treatment of emotional and functional disorders. Biomedical engineers introduced the differential amplifier during the 1950s, eliminating the need for "copper rooms" and moving sEMG into the realm of clinicians [2].

During the 1960s Basmajian, the "father" of surface electromyography, worked on single motor unit training, leading to research on biofeedback. Clinical use of sEMG for the treatment of various disorders began in the 1960s. These included teaching students not to sub-vocalize during silent reading, retraining patients with various neuromuscular conditions, and the restoration of function of hemiplegic patients [2].

In the 1980s a handheld scanning device was used in a clinical setting and a normative database of patient data developed. Suitable small and lightweight instrumentation was developed, making it widely available for the first time [2].

3.1 Control Systems

Much of the related work involves the first use case: exploiting EMG patterns in myoelectric control system (MES) applications [15], especially those needed to drive a prosthetic hand or arm. EMGs contain rich information from which a user's intention in the form of muscle contractions can be detected using surface electrodes. This information can be exploited to drive control systems of various kinds. sEMG information has been applied to the problem of controlling a multifunction prosthesis, wheelchair control, gait generation, grasping control, virtual keyboards, gesture-based interfaces, virtual worlds, and diagnosis and clinical applications [15].

The first viable EMG controlled prosthesis appeared in the 1960s. Progress in developing control systems roughly occurred in three generations: (1) ON/OFF control with a single speed; (2) state machine control, threshold manipulation, signal amplification, and some proportional control; and (3) microprocessor control with an infinite range of adjustments [15].

3.2 Pattern Recognition-driven Control

Since the early 1990s there has been an increasing amount of study of the use of sEMG in control systems, mostly powered prosthetics, but also rehabilitation devices, tele-operation of robotic limbs, and human-computer interfaces. The individual's intention is determined by the control system through the detected muscular electrical signal (EMG). To that end machine learning techniques have been used, primarily through classification algorithms.

While the research has varied, a common framework has emerged that can be used to understand the various studies. Almost all the studies involve extracting sEMG signals from one or more human subjects while tracking some number of movements. The types and number of movements the control system attempts to predict varies, as does the number of sEMG sensors employed and their location on the subject. The sEMG signals are collected as a continuous stream of electrical voltage values and are usually processed off-line in batch fashion at a later time. Features can simply be created from the raw signal values, or by using more sophisticated techniques such as computing statistics from selected elapsed time window sizes or using time frequency transforms. Classifier algorithms vary, but the most frequently occurring ones are Artificial Neural Network Multi-Layer Perceptron (ANN/MLP), Linear Discriminant Analysis (LDA), and Support Vector Machines (SVM). The control system outputs the predicted movement and sometimes force and torque [15].

3.2.1 Movements of the Wrist, Forearm, and Shoulder

Hudgins, Parker, and Scott [17] performed experiments in driving an upper extremity prosthesis in which features were extracted from a transient state (emanating from a burst of fibers, moving a limb from a resting state) starting roughly 100 ms after movement onset. They performed a series of five experiments involving 3 to 18 subjects, including 6 amputees. Four forearm and wrist movements were predicted and signals gathered using two electrodes - one on the biceps and one on the triceps. The continuous signal streams were divided into 100 ms segments and five statistics computed for each window. The five are mean absolute value (MAV), mean absolute value slope (MAVS), zero crossings (ZC), slope sign changes (SSC), and waveform length (WL). The statistics were frequently used as features in subsequent studies and referred to as the Hudgins Time Domain (TD) statistics. The experiment used an MLP neural network with one hidden layer consisting of 4 to 12 nodes, and trained using back propagation. Average accuracy of movement prediction ranged from 85.5% for the amputees to 91.2% for normally-limbed subjects. The results were encouraging and inspired additional research.

Building on the work of Hudgins, et al, Englehart, Hudgins, and Parker [18] performed a similar experiment using 16 subjects, two electrodes, and performing four forearm and wrist movements. Signals were sampled as a continuous steady state set (emanating from a constantly maintained contraction) and time-scale features generated using a short-time Fourier transform (STFT), wavelet transform (WT), and the wavelet packet transform (WPT), as well as the above cited Time Domain statistics for comparison. Principle Component Analysis (PCA) was used to reduce the feature set to 20 and LDA used for classification. The WPT features with PCA applied achieved 93% movement recognition accuracy. They continued the study [16] with 11 normally-limbed subjects, using six hand and wrist movements, two and four forearm sensors, both transient and steady-state signal sampling, and an LDA classifier with PCA feature reduction. The size of the signal window used to generate features and perform classification was examined. Accuracies up to 98% were achieved using WPT features with PCA and an LDA classifier. The study concluded that steady-state data is classified more accurately than transient data, four sensors yield higher accuracy than two, larger window sizes lead to more accurate classification but longer response times in an on-line application (i.e. driving a prosthesis), but degradation in smaller window size is less in the steady-state case. Classifying a steady-state signal stream means continuous classification of movement is possible and allows it to be more readily used in real world applications.

Englehart and Hudgins [14] extended the above in a study that involved 12 normally-limbed subjects, four hand and wrist movements, four forearm sensors, with continuous steady-state sampling. They examined various window sizes and a set of four TD features (MAV, ZC, SSC, and WL). They also examined windows of varying sizes and an overlapping processing window scheme that achieved a dense, continuous set of classifications. An LDA classifier was used with majority vote (MV) post processing of the set of most recent classification decisions. MV post-processing improved accuracy, especially during movement transition (where most classification errors occur), by allowing spurious decisions to be ignored. They concluded that the TD feature set outperformed the time-frequency features for processing continuous data, achieving accuracies in the 95% range. They are computationally less expensive and therefore better suited for on-line applications. Also, a processing window overlap of 32 ms was found to give a dense signal stream and provide high accuracy while affording a fast response time to the control system.

Generative models have also been explored for use in sEMG control system studies. Huang, Englehart, Hudgins, and Chan [24] developed a means of using Gaussian Mixture Models (GMM) for classification, which was built from existing work of Chan and Englehart [25] where it was shown that feature distributions can be well approximated by GMMs. Huang, et al, using the same data collected in [14] performed several experiments to determine the best combination of features to use from the selection of time-domain, root-mean squared, and autoregressive coefficients, and the number of Gaussians to use. Data was processed in continuous steady-state 256 ms overlapping window segments. They found that six autoregressive coefficients and root mean squared (RMS) features (7 total) together and three Gaussians for the mixture model provided accuracy of 95.6% with MV post-processing. This compared favorably with three commonly used classifiers: LDA (95.6%), Linear perceptron (95.6%), and MLP (95.4%), indicating that probabilistic techniques are viable for this application.

Hidden Markov Models have also been studied by Chan and Englehart[26], in which a six-state fully connected HMM was trained on overlapping 256 ms observation windows, spaced 32 ms apart. Six wrist and forearm movements were tracked using four forearm sensors. Features were RMS plus six autoregressive coefficients. Single mixture Gaussian observation densities were used on each state and its results were compared to the results from using MLP. Accuracy of 94.6% was achieved and statistical differences were shown in which the HMM method outperformed the MLP method. However, transition classification decisions were deemed less reliable and discarded.

Au and Kirsch [27] extended the neural network into a time-delayed artificial neural network (TDANN) that processed raw signals from six sensors on the shoulder, pectoral, and upper arm as features. The eight subjects included six able-bodied and two with C5 tetraplegia. They tracked joint angles for the shoulder and elbow and showed that it could predict the kinematic variables of the shoulder and elbow with an accuracy range of 76.6%

to 90.8%. They demonstrated that the EMG signals reflect the underlying force dynamics relatively well.

3.2.2 Movements of the Hand and Fingers

Bitzer and van der Smagt [28] concentrated on predicting six thumb and finger movements, specifically flexion and extension of the thumb, index finger, and the middle, ring, and little finger simultaneously. They used 10 forearm sensors to collect EMG signals in an interrupt-driven fashion - recording whenever the value of one or more sensors changes. An SVM classifier was used that achieved 94% accuracy for an arm in a relaxed position, and 90% when pronated. The control system was connected to an external robotic hand that moved when the subject moved, with the classifier instructing the robotic hand to perform one of the six same moves being performed by the human subject. The authors carefully chose the placement of the sensors on the forearm to be sensitive to the specific muscles involved in the six movements.

Tsenov, Zeghbib, Palis, Shoylev, and Mladenov [29] experimented with four hand movements: thumb, index finger, and middle finger flexion, and hand close. They used both two and four forearm sensors, STFT and TD features, and an MLP classifier with 10 nodes in a signal hidden layer. Predictive accuracy was 94% with two sensors, and 98% with four.

Saponas, Tan, Morris, and Balackrishnan [30] studied the application of sEMG in the building of a Human-Computer Interface (HCI). Their study involved 13 subjects, 18 individual finger movements (extension and flexion of each, plus light and hard tapping), and eight evenly spaced forearm sensors arranged in a narrow band. Signals were sampled in 250 ms blocks and 74 features built using the EMG window's RMS, RMS ratios, frequency energy, and average phase ratios without feature reduction. Predictive accuracy for same-subject data - the data is tested against a training model using data from the same subject - ranged from 78% to 95%. Eliminating the "light tapping" movement, which the authors believe is hard to distinguish, raises the accuracy. Cross-user trials - building a model using all subject data then testing each against it - decreased accuracy to 57%. The authors also showed that decreasing the amount of training data reduces classifier accuracy by up to 20%.

Tenore, Ramos, Fahmy, Acharya, Etienne-Cummings, and Thakor [31] examined 12 finger movements: flexion and extension of the thumb and each finger, plus flexion and extension of the middle, ring, and little fingers as a group. The six subjects included one transradial amputee. Normally limbed subjects had 28 sensors placed on their forearms and four just above the elbow arranged in five levels. The amputee only had 19 sensors placed on the upper, non-amputated part of the forearm. Features were extracted from a 200 ms continuous steady-state signal window every 25 ms, meaning the windows overlapped. Four time domain statistics were created for each sensor channel: MAV, variance, WL, and the Willison amplitude, for a total of 128 features. Classification was a MLP with one hidden layer with 64 to 512 nodes. Accuracy for the able-bodied participants when using all 32 sensors ranged from 84.9% to 99.7%, and from 81.6% to 99.1% when using only the 12 clustered around the elbow. The amputee's accuracy, using only the 12 sensors around the elbow, ranged from 82.3% to 87.8%. There was a statistical difference between the able-bodied and amputee accuracy, but not between the level of sensors (i.e. the 12 clustered around the elbow yielded equivalent results to using the full set of 32).

Shuman [13] performed a one subject experiment to classify the performance of six hand gestures: moving from a fist to holding one to five fingers, and taking no action. Features were created from the RMS from five forearm sensors of a two second window corresponding to the period during which the gesture was made. The sensors were placed on the forearm over muscles believed to contribute to the formation of the gestures. MLP, SVM, Decision Tree, Random Forest, Decision Tree using boosting, and K-nearest neighbor were trained. Accuracies of 93% were achieved using an MLP with one hidden layer.

More recently the Ninapro project [32, 33] used classification to recognize up to 52 grips and finger postures with the aim of driving a prosthetic hand. That effort employed 12 sensors, eight placed uniformly just below the elbow, the remainder on the extensors, flexors, and biceps. Overall accuracy of the classifier was in the 50-75% range. The collected data are publicly available.

Shuman, et al, [34] used several standard classifiers as well as a symbolic representation of the signal stream called SAX, a probabilistic approach, and Dynamic Time Warping as a distance measure to classify 15 grips and movements (14 plus rest) to an accuracy of 77%. That research was based on data from one subject. The techniques were modified to handle 25 grips and movements and use data collected from three subjects. Results are reported in [1]. These techniques were also used in this dissertation and applied to more subjects and for 47 activities. They are described in detail later.

3.2.3 Movements of the Hand and Fingers with Force Estimation

Castellini and van der Smagt [19] experimented with five grip motions: thumb to index finger, thumb to middle finger, thumb to ring finger, thumb to all fingers, and no action. Their study only involved a single subject, used 10 sensors (six on the forearm, four on the upper), and used the continuous raw signals as features. The continuous collection of signals provided too many input points, so they used a process called uniformisation to reduce the input stream. The process compared consecutive input points and only accepted new ones that differed from the previous by a certain threshold distance. Euclidean distance was used and found to give acceptable results, since a more sophisticated measure like the Mahalanobis distance is computationally expensive and impractical to use in real time applications. In addition to predicting the grip, the grip force was also estimated using the sEMG signals. The grip type was predicted using an MLP and SVM classifier, and force was estimated using Locally Weighted Projection Regression (LWPR). The MLP and SVM classifiers both achieved 90% accuracy in predicting the grip type, and the force accuracy was predicted with 0.80 correlation (7.89+/- .09N). Uniformisation significantly decreased the size of the input set while suffering only modest predictive degradation. The results were demonstrated by having the control system drive a robotic hand in real time from input sensors on a human subject's hand.

Castellini, Fiorilla, and Sandini [35] examined three grips: precision - thumb to index finger, precision - thumb to all fingers, and power (as in gripping a hammer). Ten able-bodied subjects participated, each holding a force sensor using one of the three grips for a period of time over several minutes. Seven sensors were carefully placed on the forearm over muscles believed to contribute to holding the grips. Features were created by continuously calculating the RMS for a 500 ms window from each channel for use in classification, and for a 100 ms window for the force-estimation regression task. An SVM was used for classification and regression. The continuous stream of input - captured at 2000 hz - was reduced using the uniformisation process described above, which dramatically reduced the number of input features while not degrading accuracy. Subjects performed the series of grips in an arm stationary position, and then while walking around to simulate a daily life activity scenario. Average accuracy for the stationary scenario was 97%, and 95% when walking-around. Cross-subject classification accuracy range was 51.7% to 54%. Force estimation for stationary averaged a .93 correlation factor, and .90 when walking around. The results are encouraging, especially for the walking-around scenario. Classifying grip and hand movements, and estimating force are possible in real-time, real-world activity settings.

Khokhar, Xiao, and Menon [36] experimented with eight subjects performing four movements: flexion and extension of the wrist, ulnar deviation, and radial deviation. Besides attempting to classify those four movements, they trained the classifier to recognize up to five levels of torque applied to them: 10%, 20%, 30%, 40%, and 50% of maximum voluntary contraction (MVC) torque for the flexion and extension, and 10% through 40% for the deviations. The torque estimation was built into the predicted class and expressed as a discrete rather than real value. An SVM was trained using RMS, AR coefficients, and wavelength (WL) as features extracted from four carefully chosen forearm sensors and 250 ms window. Accuracy was 88% for the full 19 class set, and 96% for a reduced class set of 13. The results indicate that movement and a torque category can be predicted using classification.

Table 3.1 summarizes the various studies cited in this section.

Table 3.1: Summary of related work.

Reference	**Application**	**# Motions**	**Channel**	**Classifier**	**Features**
Hudgens et al. [1993]	Upper limb prosthesis	4-class	2 channel	MLP NN	MAV, MAVS, ZC, SSC, WL
Englehart et al [1999]	Upper limb prosthesis	4-class	2-channel	LDA/PCA	STFT, WT, WPT
Englehart et al. [2001]	Upper limb prosthesis	6-class	4-channel	PCA/LDA	STFT, WT, WPT
Huang et al. [2005]	Upper limb prosthesis	6-class	4-channel	GMM/MV	RMS, AR
Englehart et al [2003]	Upper limb prosthesis	4-class	4-channel	LDA w/MV	MAV, ZC, SSC, WL
Chan et al [20405]	Upper limb prosthesis	6-class	4-channel	HMM	RMS + 7 AR coeff
Bitzen [2006]	Hand grips	6-class	10-channel	SVM	Not stated
Tsenov [2006]	Hand grips	4-class	4-channel	MLP	STFT + TD feats.
Saponas et al [2008]	Finger gestures	18-class	8-channel	SVM	74 from raw signal
Castellini [2008]	Hand grips & force est.	5-class	10-channel	MLP, SVM, LWPR	10 - raw signal
Shuman [2009]	Hand gestures	6-class	5-channel	MLP, SVM, RF, KNN	5 - RMS
Tenore [2009]	12 finger movements	12-class	32-channel	MLP	4 TDs/channel (128 total)
Castellini [2009]	Hand grips & force est.	4-class	7-channel	SVM	7 - RMS window
Khokhar et al. [2010]	Wrist movement	13 & 19-class	4-channel	SVM	RMS, AR, WL
Au and Kirsch [2000]	Neuromuscular stimul.	8-class	6-channel	TDANN	
Kuzorskij [2012]	hand grips	52-class	10-channel	Multiple	RMS
Atzori [2014]	hand grips & movements	50-class	12-channel	Multiple	RMS, Hudgins TD
Shuman et al. [2015]	hand grips & movements	15-class	8-channel	RF, NN, Affinity	MAV
Shuman et al. [2016]	hand grips & movements	25-class	8-channel	RF, NN, Affinity	MAV

3.3 Differences from Previous Research

The research reported in this dissertation differs from previous efforts in that it attempts to recognize a relatively large set of 47 fine motor movements of the hand needed to perform typical ADLs. It uses a moderate number of sensors targeted to specific muscle areas. It reports on results using four different learning approaches. The first involves classification using several well-known classification techniques. In the second, a symbolic representation scheme for the sensor data is employed and the concept of an Affinity Matrix is introduced to construct a learning model using adjacent signals in the stream and perform classification. The symbolic representation is also used in the third. In that one, Dynamic Time Warping (DTW) is used as a distance measure in a nearest-neighbor classification scheme. In the fourth approach, the sEMG signal stream is divided into segments of continuous movement and classification testing is performed in chronological order for those segments. A Hidden Markov Model (HMM) classifier was developed that that takes into account the previous instance and the likelihood of transitioning to other specific states.

Chapter 4: Data Collection, Preparation, and Feature Creation

sEMGs were recorded while a subject performed upper extremity (UE) movements used in a selected set of ADLs. The grips and movements selected for this research focused on those of the hand and wrist executed in a short time span (five seconds or less). They involve fine motor movements required to perform typical activities of daily living and include several types of grips and associated movements: lateral (key) grip (gripping and turning a key), power or hammer grip, door knob grip and turn, jar lid grip and turn, scissors grip and open/close, 3-jaw chuck grip and tip pinch grip.[5].

The research described in this dissertation involves four different learning approaches to classify sEMG signals into one of a set of hand grips or movements. Data were collected one time from five subjects and used throughout. The subjects performed selected activities in eight different "grip families", including the 46 specific grips and movements and the neutral/rest position listed in Table 4.3. The table includes their description, codes (used to label the activity) and action group or grip family. The eight grip families are determined by the base hand grip that must be engaged before the related follow-on actions are performed.

The collection and processing were done as a first step in this investigation and involved data collected from the five subjects. Results from an earlier stage of the research for 25 classes are reported in [1].

The goals for of all approaches include exploring the ability to recognize the activities from their EMG signal patterns, determining relationships between grip signals and their follow-on movements, and discovering relationships among the various action groups' EMG signals. In the remainder of this section the instrumentation and collection protocol are described in detail.

4.1 Instrumentation

The DelSys Trigno Wireless^{TM} sensors and base station were used for sEMG and accelerometer (ACC) signal collection[9]. The sensors each contain a rechargeable battery that communicates with the base station at a range of up to 40 m. The base unit communicates with the DelSys EMGWorks^{TM} Acquisition package via a USB interface that, in turn, drives the collection and control of the sensor signals and allows for the real-time monitoring of the signal. sEMG signals were collected at rate the of 2 kHz, ACC signals at 148.1 hz.

4.2 Sensor placement

Ten sensors were used in the data collection, each attached to the skin surface of the subject's hand and arm used in performing the actions. The sensors were secured using adhesive skin interfaces provided by DelSys for the purpose. Eight were sEMG sensors located over the arm muscles believed to contribute to the grip or movement. Seven were located on the extrinsic muscles in the forearm that control the hand and wrist, and one on the biceps. A sensor was placed on the biceps in an attempt to capture the contributions of the upper arm to the activities in Table 4.3, especially those involving raising an object and turns of the hand and wrist. Trials conducted during previous research [34] indicated measurable EMGs from the biceps but not the triceps during the listed activities. Guidance from [2] indicated good candidate areas of the arm contributing to hand and wrist movements. One (on the extensor indicis) was configured to also collect ACC data for possible later use. EMG data from the eight sensors are used as classification features.

Two additional sensors were attached on the active hand's posterior, just below the base of the thumb and below the little finger. They were configured to collect ACC data as an aid in labelling the movement actions. Changes in the ACC data from the hand sensors indicate that a movement has started and the associated EMG data instances can be appropriately labelled. The location on both sides of the hand was chosen to maximize the detection, while locating them on the hand's posterior minimized interference with

Table 4.1: Location of the ten sensors and data collected.

Sensor#	Muscle Location	Data Collected
1	extensor digitorum (ED)	EMG
2	extensor indicis (EI)	EMG and ACC
3	flexor carpi radialis (FCR)	EMG
4	flexor digitorum superficialis (FDS)	EMG
5	flexor carpi ulnaris (FCU)	EMG
6	pronator quadratus (PQ)	EMG
7	brachioradialis (Bra)	EMG
8	biceps brachii (Bic)	EMG
9	base of the thumb (posterior)	ACC
10	base of little finger (posterior)	ACC

the subjects' performance of the actions. Data from these two sensors are not used as classification features, but were used as in aid in labelling.

Sensor placement is shown in Table 4.1 and was not altered during the trials. Fig 2.6 shows the Trigno kit and a subject turning a jar lid with ten sensors attached to the action arm.

4.3 Data collection protocol and feature creation

Data were collected from five subjects as described below. The subjects included one middle aged male, one middle aged female, one male in his 20s, one female in her 20s, and one male in his early 30s. The first four were in good physical condition, without disability of any kind, and able to perform all grips and actions without difficulty. All four were naturally right handed and used their right hand to perform the activities. The male in his 30s had a partially disabled left side and used that hand to perform the activities. The original intention of the study was to collect data exclusively from non-disabled subjects. However, when the fifth subject became available he was included as a comparator with the non-disabled subjects.

Table 4.2 summarizes the subjects involved in the study whose captured data were used for analysis. Note that the data collection protocol and activities performed were identical

Table 4.2: Summary of the five subjects from whom data were collected.

Subj#	**M/F**	**Age range**	**Date collected**	**Hand used**	**Ability**
1	M	50-70	9/23/2015	right	fully
2	F	50-70	10/9/2015	right	fully
3	M	20-30	12/15/2015	right	fully
4	F	20-30	2/23/2016	right	fully
5	M	30-40	3/28/2016	left	partly disabled

for all subjects. The collection followed a protocol approved by the George Mason University Institutional Review Board, Reference number 8672.

The subjects performed a series of eight two minute data collection runs. The activities for a single action group, or grip family, were performed in one run. Note that there are eight grip families, one per run. The subjects followed a timed script in performing 12 repetitions of the action family's grip and movements, each repetition lasting ten seconds. Table 4.3 shows breakdown for the 47 actions.

Each two minute run starts with the subject maintaining their hand and arm in a neutral or rest (NR) posture for the first five seconds. At second five the repetitions begin. First, the grip for the family was engaged and held for three seconds. At the eight second mark actions requiring the specific grip are performed until second 11, at which point the subject transitions back to the neutral/rest position until the start of the next repetition.

The subjects were instructed to begin each grip with the hand in proximity to, but not touching the object to be grasped, ensuring the act of gripping was captured. For the hammer, ball, scissors, and jar lid action groups, the object was placed in the non-active hand between repetitions, with the active hand approximately ten centimeters from the object. For the two grip families involving turning, the door knob and key, the subject was instructed to release the object after the first turn movement and resume the grip before proceeding with the second turn, essentially inserting a brief pause, or neutral/rest, between the two turns. For the fine movement families, tip pinch and 3-jaw chuck, the subject was instructed to grasp the objects, a U.S. quarter dollar coin for tip pinch and golf ball for the

Table 4.3: The 47 activities (grips and associated movements) used in the analysis of the captured data.

Activity#	**Grip group#**	**Grip**	**Code**	**Activity Description**
1	-	none	NR	neutral/rest
2	1	hammer	HGIN	hammer grip - transition in
3	1	hammer	HG	hammer grip
4	1	hammer	HR	hammer raise
5	1	hammer	HGR	hammer grip - raised pos.
6	1	hammer	HL	hammer lower
7	1	hammer	HLOUT	hammer Lower - transition out
8	2	jar lid	JLGIN	jar lid grip - transition in
9	2	jar lid	JLG	jar lid grip
10	2	jar lid	JLP	jar lid turn - pronation
11	2	jar lid	JLRP	jar lid - rest/pause
12	2	jar lid	JLS	jar lid turn - supination
13	2	jar lid	JLOUT	jar lid - transition out
14	3	ball	BGIN	ball grip - transition in
15	3	ball	BG	ball grip
16	3	ball	BSQ	ball squeeze
17	3	ball	BSQOUT	ball squeeze - transition out
18	4	door knob	DKGIN	door knob grip - transition in
19	4	door knob	DKG	door knob grip
20	4	door knob	DKTS	door knob turn - supination
21	4	door knob	DKTR	door knob turn - rest/pause
22	4	door knob	DKTP	door knob turn - pronation
23	4	door knob	DKTOUT	door knob turn - transition out
24	5	key	KGIN	key grip - transition in
25	5	key	KG	key grip
26	5	key	KTS	key turn - supination
27	5	key	KGTR	key grip turn - rest/pause
28	5	key	KTP	key turn - pronation
29	5	key	KGOUT	key grip = transition out
30	6	scissors	SCGIN	scissors grip - transition in
31	6	scissors	SCG	scissors grip
32	6	scissors	SCO	scissors open
33	6	scissors	SCGO	scissors grip - open position
34	6	scissors	SCC	scissors close
35	6	scissors	SCOUT	scissors grip - transition out
36	7	3-jaw chuck	3JCGIN	3-jaw chuck grip - transition in
37	7	3-jaw chuck	3JCG	3-jaw chuck grip
38	7	3-jaw chuck	3JCR	3-jaw chuck raise
39	7	3-jaw chuck	3JCGR	3-jaw chuck grip - raised position
40	7	3-jaw chuck	3JCL	3-jaw chuck lower
41	7	3-jaw chuck	3JCOUT	3-jaw chuck grip - transition out
42	8	tip pinch	TPGIN	tip pinch grip - transition in
43	8	tip pinch	TPG	tip pinch grip
44	8	tip pinch	TPR	tip pinch raise
45	8	tip pinch	TPGR	tip pinch grip - raised position
46	8	tip pinch	TPL	tip pinch lower
47	8	tip pinch	TPLOUT	tip pinch grip - transition out

Table 4.4: The elapsed times shown in the table are only approximate since labelling used accelerometer data to accurately label the 100 ms 8-tuples. The text gives more details.

	One 10 second repetition							
Grip family	$s_{x-1}4$	s_x5	s_x6	s_x7	s_x8	s_x9	s_x10	s_x11–14
hammer	NR/HGIN	HG	HG	HG	HR	HGRP	HL	HLOUT/NR/HGIN
jar lid	NR/JLGIN	JLG	JLG	JLG	JLP	JLRP	JLS	JLOUT/NR/JLGIN
ball	NR/BGIN	BG	BG	BG	BSQ	BSQ	BSQOUT/NR	NR/BGIN
door knob	NR/DKGIN	DKG	DKG	DKG	DKTS	DKTR	DKTP	DKTOUT/NR/ DKGIN
key	NR/KGIN	KG	KG	KG	KGTS	KGTR	KGTP	KGOUT/NR/KGIN
scissors	NR/SCGIN	SCG	SCG	SCG	SCO	SCGO	SCC	SCOUT/NR/SCGIN
3-jaw chuck	NR/3JCGIN	3JCG	3JCG	3JCG	3JCR	3JCGR	3JCL	3JCOUT/NR/3JCGIN
tip pinch	NR/TPGIN	TPG	TPG	TPG	TPR	TPGR	TPL	TPLOUT/NR/TPGIN

chuck, with enough force so that it would not be dropped if the hand were lightly slapped. Apart for the above the subjects were allowed to choose the way in which they performed the actions.

Table 4.4 illustrates the protocol followed for all eight grip families. Each family (or group) was performed as one set of 12 repetitions over a two minute time span. A single repetition lasted 10 seconds and starts on second 5, 15, 25, ... 115, within each 120 second interval. For the hammer family, for example, the repetition started with the hammer grip transition in (HGIN) just prior to second 5 ($s_{x-1}4$), the hammer grip (HG) performed for three seconds (s_x5 - s_x7), followed by a hammer raise (HR) for one second (s_x8), hammer grip in raised position (HGRP) for one second (s_x9), the hammer lower (HL) for one (s_x10), and the transition out (HLOUT) followed by neutral/rest (s_x11 - s_x14) until the start of the next repetition.

An attempt was made during data collection to ensure the synchronization of the actions with the indicated times and durations. The EMGWorks$^{\text{TM}}$ timer was used to prompt the subject for the next action. However, this could only be done to a certain level of precision and so the timings, while within a few hundred milliseconds of the stated values, should be regarded as approximate.

Each subject's data collection resulted in eight separate files, one for each grip family. Figure 4.1 shows a block diagram of the eight files collected from each of the subjects.

File#	Grip Family	Repetition number												
		NA	1	2	3	4	5	6	7	8	9	10	11	12
1	hammer													
2	jar lid													
3	ball													
4	door knob													
5	key													
6	scissors													
7	3-jaw chuck													
8	tip pinch													
	elapsed secs.	0	5	15	25	35	45	55	65	75	85	95	105	115
	#seconds/rep.	5	10	10	10	10	10	10	10	10	10	10	10	6
	#100ms instances	100	200	200	200	200	200	200	200	200	200	200	200	120

Each of the 12 repetitions consists of these activity sequences for the 8 grip families (47 labelled classes). (Repetition #5 is shown.)

File#	activities within one family repetition
1	NR-HGIN-HG-HR-HGRP-HL-HLOUT-NR
2	NR-JLGIN-JLG-JLP-JLRP-JLS-JLOUT-NR
3	NR-BGIN-BG-BSQOUT-BSQOUT-NR
4	NR-DKGIN-DKG-DKTS-DKTR-DKTP-DKTOUT-NR
5	NR-KGIN-KG-KGTS-KGTR-KGTP-KGOUT-NR
6	NR-SCGIN-SCG-SCO-SCGO-SCC-SCOUT-NR
7	NR-3JCGIN-3JCG-3JCR-3JCGR-3JCL-3JCOUT-NR
8	NR-TPGIN-TPG-TPR-TPGR-TPL-TPLOUT-NR

Figure 4.1: Eight files were collected for each subject, one for each of the two minute data collection runs for the eight grip families. These files were separately processed and labelled before being combined for classification training and testing.

Each file contained the signal data (EMG and ACC) for one two minute run including 12 repetitions for the family. The DelSys EMGAnalysis^{TM} package was used to visualize and process the collected signal sequences. The Trigno sensors filtered the signals during collection with a 20-450 hz bandwidth using a flat Butterworth filter to preserve EMG signal amplitude and phase linearity. This eliminates noise while capturing most of the signal [37]. The mean absolute value (MAV) was computed for each sequence, specifying a collection window of a 100 millisecond signal segment with a 50 ms overlap. The window size was selected to allow for quick classification decisions needed for real time control applications. The trade-off of varying window sizes versus accuracy is discussed in [15] and [14].

To compute the MAV for the specified window size, let $f_j(iT_1), i = 1, 2, \ldots$ be the sampled data for channel $j = 1, \ldots, 8$; $T_1 = 1/2000$ second is the sampling period. $g_j(kT), k = 1, 2, \ldots$ is sampled filtered data computed using Mean Absolute Values (MAVs) for windows of width $2T$ at steps of size T using

$$g_j(kT) = \frac{1}{2N} \sum_{i=-N+1}^{N} |f_j(kT + iT_1)|, \tag{4.1}$$

where $T = 1/20$ second, $N = 100$, j is the sensor channel index, and k the sample index.

An 8-tuple $g_j(kT), j = 1, \ldots, 8$ computed using Eq. (4.1) is one training instance. The result was 20 training instances per second, or 2400 EMG instances per 120 second data collection run. The Trigno captures the ACC data at a different frequency and the MAV calculation results in 21.1 instances per second, or an additional 132 for the 120 second run. EMGAnalysis$^{\text{TM}}$ macros were used for this processing. Fig. 4.2 shows the MAV of eight sensor signals for the one entire 120 second stream for one hammer group data capture run. Fig. 4.3 shows a 20 second, two repetition sample for four of the action groups.

Figure 4.4 shows an overview of the process from subject data collection, through converting the signals to MAVs, and then to labelling the instances in the destination file. The file contains eight features, corresponding to the MAVs from the eight sensors, and a class label indicating the ground truth of the grip or movement. Figure 4.5 illustrates how the ultimate classification file for one subject was built from the data collection repetitions. Each of the eight families consisted of 12 repetitions, shown in the rows in the figures. The last repetition was slightly shorter than the others, but long enough to include all the necessary movements. The total number of instances in the file submitted to the classifiers is 18,720, shown in the lower right corner of the figure. One of these files was created for each subject and each was separately classified. There was no attempt to mix subject data within one classification run in this study. Figure 4.6 shows a breakdown of the number of collected instances for each subject for all 47 classes, and averages.

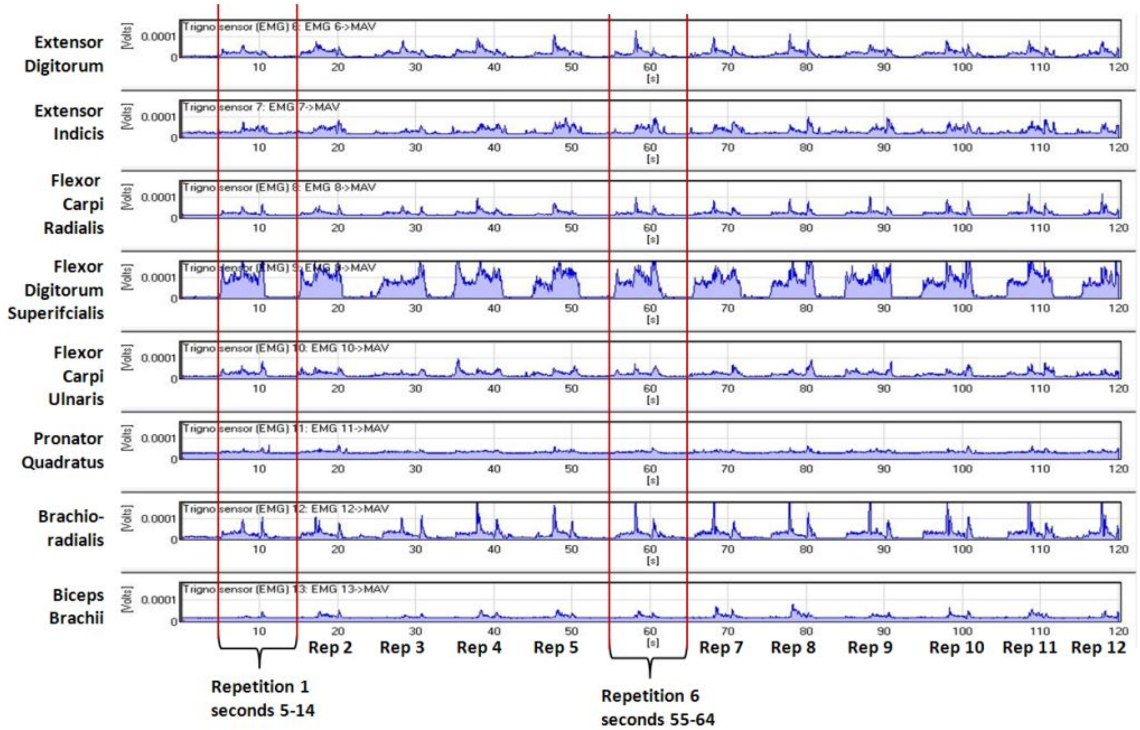


Figure 4.2: Graph of the mean absolute values (100 ms. window) for the eight sensor channels for an entire 120 second hammer group data capture run for one subject. A vertical slice from the repetition noted along the x-axis (the slice for the 1st and 6th are shown) up through the eight sensor graphs reveals the pattern for each repetition. Each repetition starts at seconds 5, 15, 25, ... with a grip transition in (HGIN) in the neighborhood of second 5, followed by the grip (HG), then a raise (HR), grip in raised position (HGRP), lower (HL), and a transition out (HLOUT) followed by a short rest (NR) before starting the starting next repetition.

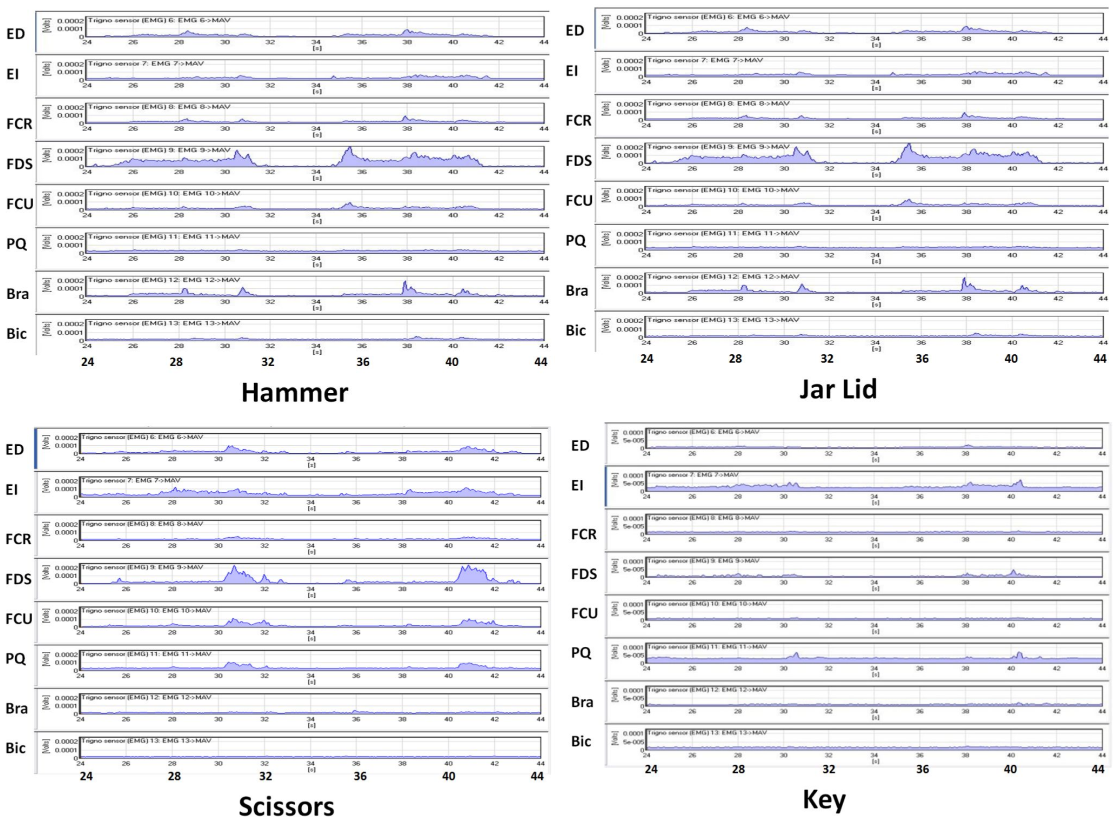
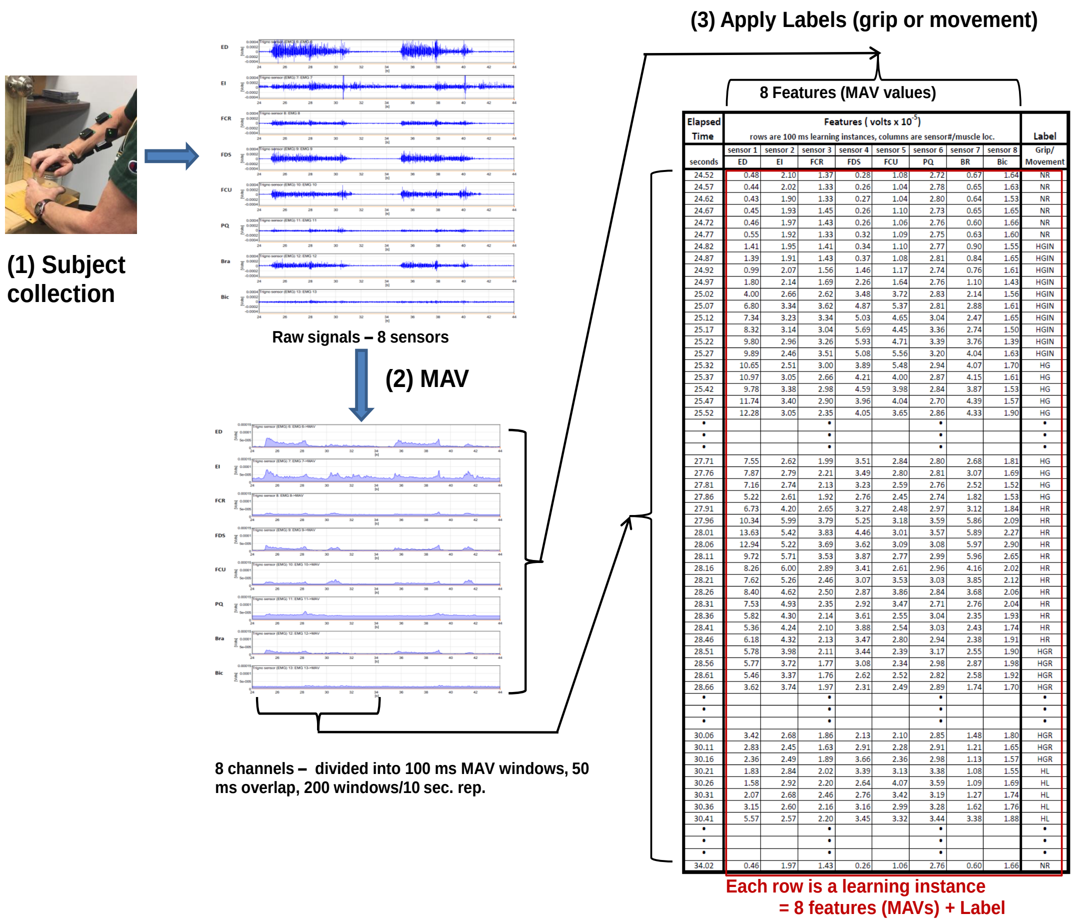


Figure 4.3: Graph of four of the eight action groups mean absolute values - 20 second interval (24-44) covering two complete repetitions (3rd and 4th) of each group's grip and movements. The four represent different types of grips: power (hammer), precision (jar lid), dynamic tripod (scissors), and precision handling - small objects (key). Pattern differences can be observed among the four. Note that the Key movement requires less force than the other three and shows a lower signal value despite its graph being scaled at a lower value than the others. Abbreviations for the muscle signal channels are shown on the left of each graph row.



Elapsed Time	Features (volts x 10^{-5}) rows are 100 ms learning instances, columns are sensor#/muscle loc.								Label
	sensor 1	sensor 2	sensor 3	sensor 4	sensor 5	sensor 6	sensor 7	sensor 8	Grip/
seconds	ED	EI	FCR	FDS	FCU	PQ	BR	Bic	Movement
24.52	0.48	2.10	1.37	0.28	1.08	2.72	0.67	1.64	NR
24.57	0.44	2.02	1.33	0.26	1.04	2.78	0.65	1.63	NR
24.62	0.43	1.90	1.33	0.27	1.04	2.80	0.64	1.53	NR
24.67	0.45	1.93	1.45	0.26	1.10	2.73	0.65	1.65	NR
24.72	0.46	1.97	1.43	0.26	1.06	2.76	0.60	1.66	NR
24.77	0.55	1.92	1.33	0.32	1.09	2.75	0.63	1.60	NR
24.82	1.41	1.95	1.41	0.34	1.10	2.77	0.90	1.55	HGIN
24.87	1.39	1.91	1.43	0.37	1.08	2.81	0.84	1.65	HGIN
24.92	0.99	2.07	1.56	1.46	1.17	2.74	0.76	1.61	HGIN
24.97	1.80	2.14	1.69	2.26	1.64	2.76	1.10	1.43	HGIN
25.02	4.00	2.66	2.62	3.48	3.72	2.83	2.14	1.56	HGIN
25.07	6.80	3.34	3.62	4.87	5.37	2.81	2.88	1.61	HGIN
25.12	7.34	3.23	3.34	5.03	4.65	3.04	2.47	1.65	HGIN
25.17	8.32	3.14	3.04	5.69	4.45	3.36	2.74	1.50	HGIN
25.22	9.80	2.96	3.26	5.93	4.71	3.39	3.76	1.39	HGIN
25.27	9.89	2.46	3.51	5.08	5.56	3.20	4.04	1.63	HGIN
25.32	10.65	2.51	3.00	3.89	5.48	2.94	4.07	1.70	HG
25.37	10.97	3.05	2.66	4.21	4.00	2.87	4.15	1.61	HG
25.42	9.78	3.38	2.98	4.59	3.98	2.84	3.87	1.53	HG
25.47	11.74	3.40	2.90	3.96	4.04	2.70	4.39	1.57	HG
25.52	12.28	3.05	2.35	4.05	3.65	2.86	4.33	1.90	HG
•			•			•			•
•			•			•			•
•			•			•			•
27.71	7.55	2.62	1.99	3.51	2.84	2.80	2.68	1.81	HG
27.76	7.87	2.79	2.21	3.49	2.80	2.81	3.07	1.69	HG
27.81	7.16	2.74	2.13	3.23	2.59	2.76	2.52	1.52	HG
27.86	5.22	2.61	1.92	2.76	2.45	2.74	1.82	1.53	HG
27.91	6.73	4.20	2.65	3.27	2.48	2.97	3.12	1.84	HR
27.96	10.34	5.99	3.79	5.25	3.18	3.59	5.86	2.09	HR
28.01	13.63	5.42	3.83	4.46	3.01	3.57	5.89	2.27	HR
28.06	12.94	5.22	3.69	3.62	3.09	3.08	5.97	2.90	HR
28.11	9.72	5.71	3.53	3.87	2.77	2.99	5.96	2.65	HR
28.16	8.26	6.00	2.89	3.41	2.61	2.96	4.16	2.02	HR
28.21	7.62	5.26	2.46	3.07	3.53	3.03	3.85	2.12	HR
28.26	8.40	4.62	2.50	2.87	3.86	2.84	3.68	2.06	HR
28.31	7.53	4.93	2.35	2.92	3.47	2.71	2.76	2.04	HR
28.36	5.82	4.30	2.14	3.61	2.55	3.04	2.35	1.93	HR
28.41	5.36	4.24	2.10	3.88	2.54	3.03	2.43	1.74	HR
28.46	6.18	4.32	2.13	3.47	2.80	2.94	2.38	1.91	HR
28.51	5.78	3.98	2.11	3.44	2.39	3.17	2.55	1.90	HGR
28.56	5.77	3.72	1.77	3.08	2.34	2.98	2.87	1.98	HGR
28.61	5.46	3.37	1.76	2.62	2.52	2.82	2.58	1.92	HGR
28.66	3.62	3.74	1.97	2.31	2.49	2.89	1.74	1.70	HGR
•			•			•			•
•			•			•			•
•			•			•			•
30.06	3.42	2.68	1.86	2.13	2.10	2.85	1.48	1.80	HGR
30.11	2.83	2.45	1.63	2.91	2.28	2.91	1.21	1.65	HGR
30.16	2.36	2.49	1.89	3.66	2.36	2.98	1.13	1.57	HGR
30.21	1.83	2.84	2.02	3.39	3.13	3.38	1.08	1.55	HL
30.26	1.58	2.92	2.20	2.64	4.07	3.59	1.09	1.69	HL
30.31	2.07	2.68	2.46	2.76	3.42	3.19	1.27	1.74	HL
30.36	3.15	2.60	2.16	3.16	2.99	3.28	1.62	1.76	HL
30.41	5.57	2.57	2.20	3.45	3.32	3.44	3.38	1.88	HL
•			•			•			•
•			•			•			•
•			•			•			•
34.02	0.46	1.97	1.43	0.26	1.06	2.76	0.60	1.66	NR

Figure 4.4: The figure shows the complete process from (1) data collection from the subject, (2) the processing of the signals to create the 100 ms MAV instances, and (3) to the labelling of the instances and creation of the classification file. The area boxed in red is the data presented to the classifier. Each row in the table is a single learning instance. Note that the eight sensor channel MAVs map to the eight features and that only data from one repetition is shown on the left side of the MAV block corresponding to the 10 second/200 instance data block in the file.

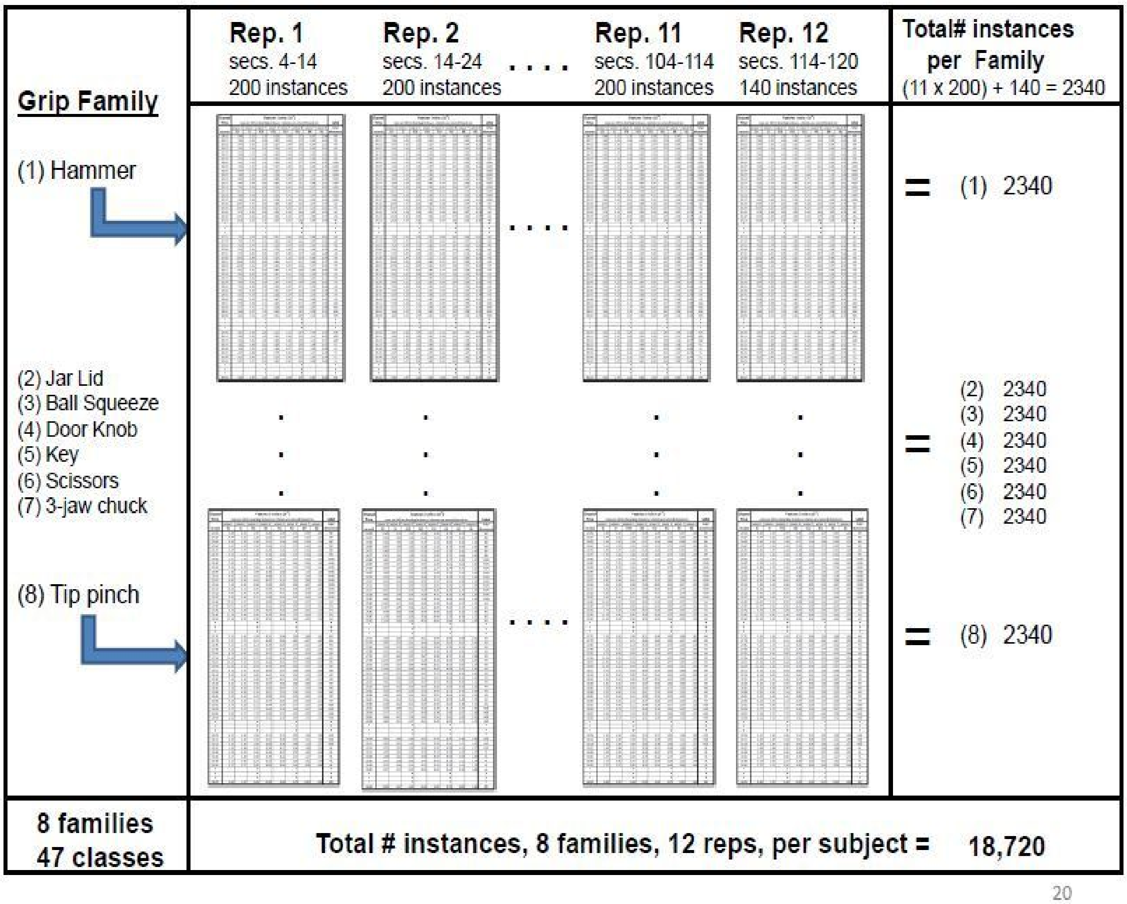


Figure 4.5: How a single subject's classification file was built from the 12 repetitions for each of the eight grip families. Each of the grip families is shown as a row of 2,340 learning instances, with a total of 18,720 per subject for all eight families.

activ.#	Code	subj. 1	subj. 2	subj. 3	subj. 4	subj. 5	avg.(5)	avg./rep.	% of total
1	NR	5442	5251	4644	6156	4456	5189.8	*54.1	27.76%
2	HGIN	196	179	184	112	191	172.4	14.4	0.92%
3	HG	612	519	634	604	530	579.8	48.3	3.10%
4	HR	139	157	210	146	203	171	14.3	0.91%
5	HGRP	289	403	259	367	297	323	26.9	1.73%
6	HL	136	149	131	144	221	156.2	13.0	0.84%
7	HLOUT	169	227	171	159	256	196.4	16.4	1.05%
8	JLGIN	158	91	255	84	189	155.4	13.0	0.83%
9	JLG	548	597	454	633	571	560.6	46.7	3.00%
10	JLP	206	206	234	163	190	199.8	16.7	1.07%
11	JLRP	290	337	338	327	310	320.4	26.7	1.71%
12	JLS	184	159	218	192	246	199.8	16.7	1.07%
13	JLOUT	172	287	307	131	181	215.6	18.0	1.15%
14	BGIN	203	267	180	169	289	221.6	18.5	1.19%
15	BG	624	574	583	518	401	540	45.0	2.89%
16	BSQ	494	499	655	596	550	558.8	46.6	2.99%
17	BSQOUT	185	291	117	242	370	241	20.1	1.29%
18	DKGIN	168	117	203	104	97	137.8	11.5	0.74%
19	DKG	696	614	526	582	517	587	48.9	3.14%
20	DKTS	274	313	268	212	197	252.8	21.1	1.35%
21	DKTR	261	321	264	324	465	327	27.3	1.75%
22	DKTP	291	276	284	283	215	269.8	22.5	1.44%
23	DKTOUT	187	159	244	167	203	192	16.0	1.03%
24	KGIN	213	105	235	96	262	182.2	15.2	0.97%
25	KG	605	613	616	578	493	581	48.4	3.11%
26	KGTS	135	149	156	148	190	155.6	13.0	0.83%
27	KGTR	361	367	339	341	275	336.6	28.1	1.80%
28	KGTP	147	158	244	160	177	177.2	14.8	0.95%
29	KGOUT	271	221	227	91	347	231.4	19.3	1.24%
30	SCGIN	235	221	237	128	248	213.8	17.8	1.14%
31	SCG	593	486	492	516	441	505.6	42.1	2.70%
32	SCO	130	166	166	147	169	155.6	13.0	0.83%
33	SCGO	350	360	429	419	350	381.6	31.8	2.04%
34	SCC	106	177	177	187	201	169.6	14.1	0.91%
35	SCOUT	383	337	351	367	432	374	31.2	2.00%
36	3JCGIN	195	160	211	152	232	190	15.8	1.02%
37	3JCG	579	579	610	529	485	556.4	46.4	2.98%
38	3JCR	159	166	247	168	251	198.2	16.5	1.06%
39	3JCGR	263	424	365	343	286	336.2	28.0	1.80%
40	3JCL	174	157	177	187	251	189.2	15.8	1.01%
41	3JCOUT	200	248	239	148	449	256.8	21.4	1.37%
42	TPGIN	215	167	278	145	291	219.2	18.3	1.17%
43	TPG	607	572	586	575	483	564.6	47.1	3.02%
44	TPR	137	152	216	163	274	188.4	15.7	1.01%
45	TPGR	279	430	316	405	291	344.2	28.7	1.84%
46	TPL	156	149	189	169	248	182.2	15.2	0.97%
47	TPLOUT	234	161	250	149	403	239.4	20.0	1.28%
	Total	**18651**	**18718**	**18716**	**18726**	**18674**	**18697**	**1558.1**	**100.00%**

* The average for 'NR' was 432.5. Since it was present in all 8 grip family repetitions, dividing by 8 yields 54.1.

Figure 4.6: Summary of the number of instances in each subject's classification file. The table shows a breakdown of all the data collected for the five subjects in this study, by class, and used for classification.

4.4 Labelling

The current labelling technique improved on earlier attempts. Both the earlier and current techniques are described in the next two sections.

4.4.1 Previous Labelling

The earlier phase of this research, reported in [1], labelled as follows. Data for all eight files for each of the subjects' collection runs were processed and labelled. Labels were assigned by reviewing the signal stream at the beginning of each of the action repetitions, seconds 5, 15, 25, ... 115. The stream consists of the elapsed captured time as well as the MAV of the signal values. The grips were assumed to start on the indicated times and were labelled accordingly. The start of subsequent movements within each repetition were indicated by a change in ACC signals for the two sensors on the hand. Since the ACC data were collected at a different frequency, the elapsed times do not exactly match the EMG times on a one-to-one basis. The ACC values were matched up with the closest EMG value based on the elapsed time of each, the difference never being more than 30 milliseconds. The summed difference of the ACC values between time t and $t+1$ were computed. The sum exceeding double the median value for the entire run was interpreted as being the start of an action and the EMG 8-tuple closest to the time instance was assigned the appropriate movement label.

Establishing the onset of the grips was a little more difficult since they are static activities and are not as clearly distinguished by a change in ACC readings. However, since the subjects were required to begin each grip with their hand off the object and move it to the object to establish the grip, there was a more modest ACC change as the grip was assumed. The end of a sequence of ACC changes during an elapsed time in which a grip was expected was used to label a data instance as the start of a grip and sustain it for up to three seconds. The result was that some label instances varied by a few processing windows before or after the expected time synchronization points indicated in Table 4.4. While this was never more than a 500 ms adjustment before or after the five second mark, the inexactness of the labelling process may have led to some misclassification at the onset of a static grip activity. While the process described above resulted in superior labelling compared with that used in [34], it remains a difficulty, has been reported elsewhere in the literature, for example [32], and as yet does not have a good solution.

Since the NR (neutral/rest) samples greatly outnumber the others, their numbers were reduced by including only the samples between seconds one and three of each run. This resulted in ≈ 60 (3×20) NR samples per run instead of ≈ 960, and an overall reduction of $\approx 7,680$ to 480 for all eight runs for a single subject. The NR instances outside the range of zero to five seconds occupy a gap in data collection between activities or activity repetitions during which the subject was only sometimes in the NR posture. During those gaps subjects occasionally performed some movement to relax or get ready for the next repetition. Because of this uncertainty, those NR instances could not be uniformly labelled correctly and are ignored. By contrast, subjects always started a run in the neutral/rest position and so sampling at seconds one to four ensures those instances are truly NR and do not include casual or unintended movements.

To ensure uniformity in evaluating the learning techniques, two of the approaches described in the earlier analysis truncated each labelled action to a maximum of 40 100 ms segments in length. This was done to facilitate the use of a technique called Dynamic Time Warping as a distance measure in one of the learning approaches, which will be explained later.

4.4.2 Current Labelling

The current labelling is an improvement over the earlier attempt. To reduce processing time, the data were resampled from a frequency of 2,000 to 500 hz. Classification quality was unaffected, probably because the data's MAV, not the raw signals, was used as the feature. An important additional benefit of resampling is that EMGWorks resamples both the EMG and ACC data and in the process up samples the ACC data from 148.1 hz to 500 hz. The result is a time series in which EMG and ACC data are matched. Prior to resampling the EMG and ACC data instances were slightly offset and a match of the timing of the two required time-consuming inspection and adjustment.

A goal of the current protocol is to improve the labelling of activities by systematically using the ACC data as an indicator of movement. This data played a relatively minor role in the earlier labelling. Here, however, plots of the EMG and ACC data streams for each activity group and each subject were time-aligned. Visual inspection of the plots indicated when movements started or stopped, and when direction of the hand changed. The changes approximately aligned with the activity timings shown in Table 4.4, but using the ACC breakpoints improved the accuracy of most labels.

All data streams (EMG and ACC) for each subject were plotted with MATLAB and visually inspected. Figure 4.7 shows one such plot for repetition #4 (seconds 34 - 43) of the hammer activity for one of the subjects. The ACC data was the key indicator used to set activity transition breakpoints. The transition from NR (rest) to the HGIN (transition-in) for a repetition can be clearly seen, as can the transition back to NR at the end of the repetition. Transition from the static grip posture, HG, to a raise, HR, is also obvious, as is the cessation of the raise and start of the grip/raised position (HGRP) posture.

These breakpoints were used to establish a change in activity label. This was done for all grip family repetitions for all subjects. The process was time-consuming, but resulted in more precise labelling.

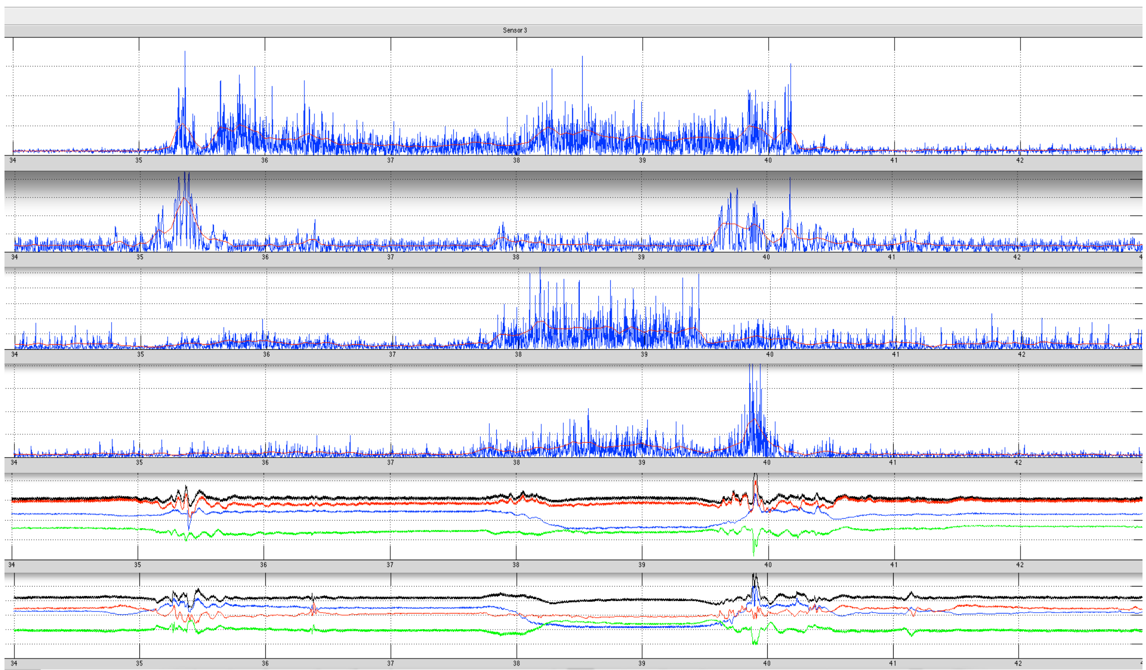


Figure 4.7: Plot of repetition 4 (seconds 34 to 43) for a hammer sequence showing the ACC signals from the hand sensors (bottom two signal rows) and the four most prominent sEMG signals (top four signal rows). Labels were established for the 100 ms segments based on changes in the ACC signals, indicating start or stop of a movement, and changes in the sEMG signals. The repetition starts with NR (second 34), then transitions to HGIN at 35.1, HG at 35.7, HR at 37.7, HGRP at 38.4, HL at 39.6, HLOUT at 40.5, and back to NR at 41.4. The timings are approximate, but the scale shown during actual labelling was more detailed and the recorded times were accurate within 100 ms.

Chapter 5: Learning approaches

Four approaches were used in learning the sEMG patterns leading to the recognition of selected grips and movements. The first uses several well-known classification techniques operating on 100 ms MAV windows as learning instances. The second two use a symbolic representation of the signal stream that divides it into discrete ranges. One of the two creates an affinity matrix to model the learning instances and employs nearest neighbor classification. The approach takes advantage of the time-series nature of the data stream by using a selected number of instances immediately before the one being classified to help make the classification decision. The other uses Dynamic Time Warping [38] as a distance measure in conjunction with nearest neighbor. Instead of treating each 100 ms instance in isolation, this approach considers all the instances that constitute an entire labelled activity to make classification decisions. The fourth approach, like the first, uses 100 ms MAV windows as its basic learning instance. Using a Hidden Markov Model (HMM) approach, it first performs a Random Forest classification of a signal instance, then uses the resulting class and state transition probabilities to perform a 'belief' calculation as a post-processing step.

Each approach is explained in more detail in the following sections.

5.1 Classification (Approach 1).

The key point in classification is the use of a class label for each training instance that must be manually assigned [11]. Here, a label is assigned to each 100 ms MAV window of each grip or movement, including the neutral/rest posture. A labelled 100 ms window is one training instance. Table 4.3 shows the code labels used to track the grips and movements. The label allows the classifier to build relationships between signals and the class (the grips and movements). The grips and movements form a 47 class problem in which a classifier is trained to recognize the classes from their sEMG signal patterns. The classifiers were trained and tested on their ability to recognize the class of each 100 ms instance in isolation, without considering any time sequence dependencies among them.

Classifiers are measured on the accuracy of their predictions. Accuracy is the percentage of test instances correctly identified from the total number evaluated. For multi-class problems, how well the classifier recognizes instances of each class - measured by the true positive rate (TPR) or Recall - is also of interest since it can vary.

After labelling, the data were normalized before applying the approaches described below. Each of the eight channels was individually normalized by subtracting the channel value mean and dividing by its standard deviation. The eight means and standard deviations of the training data channels were used to normalize held out test data.

Several classifiers were tried, including Decision Tree, Random Forest (RF), Support Vector Machine (SVM), and Nearest Neighbor (NN). The Weka toolset, v 3.6.11 [21] was used to perform all classification in this first approach. The Weka default values were used except as follows. The description of several classifiers and evaluation of parameter settings using 25 class data was reported in [1] and is summarized below. Those results showed the superiority of Random Forest, which is used in this approach.

The Decision Tree used the Weka J48 implementation of the C 4.5 decision tree algorithm. RF is an ensemble classifier that generates a stated number of trees using a subset of randomly chosen features for each generated tree. A voting process in which each of the generated tree's choice is tallied determines the winning grip or movement. The number of RF trees generated was varied from 15 to 100, with performance levelling off at 25. The RF results reported here were therefore generated using a 25 tree model. For K-nearest neighbor (K-NN), the normalized Euclidean distance measure was used, and the value of K, the number of neighbors used to determine the class, was varied from $1 \ldots 5$. 1-NN performed best and was reported.

While the aforementioned classification methods handle multi-class problems as part of their core algorithms, SVM is inherently binary. The Weka implementation employed here uses a 1-versus-1 approach and an implementation of the Hastie and Tribshirani pairwise coupling method [39]. For C classes, $C(C-1)/2$ classifiers are built. The pairwise class probability estimates are combined into a joint estimate for all classes and used to predict the class.

The SVM parameters were evaluated and set using a grid search. The Polynomial, RBF, and Pearson Universal Kernel (PUK) functions were tested. For the Polynomial kernel, the exponent parameter was tested at 2 and 3, with no improvement over the default of 1. The RBF kernel bandwidth parameter, γ, was tested for $0.01, 0.05$, and 1.0. The PUK parameters were tested for ω and σ values of $0.25, 0.5, 1$, and 5. The SVM regularization parameter, C, was tested for values $0.5, 1, 5, 10, 25, 50$, and 100, with accuracy leveling off at 5. The PUK kernel with $\omega = \sigma = 0.5$ and $C = 5$ were found to result in the highest accuracy when tested using data for the three subjects. These are the SVM settings used to produce the results reported in this paper.

Stratified ten-fold cross validation was performed using all training data to select parameters and evaluate classifier performance. Stratification ensures that a representative proportion of instances of each class is included in each of the ten folds.

5.2 Affinity Matrix (Approach 2)

One of the key ideas in this approach is the concept of building a class affinity matrix A from the training data and using it as the classification model. The matrix has one row for each of the 47 class actions listed in Table 4.3. The 100 ms MAV 8-tuples in the training dataset are converted to SAX words as described in the previous section. The words in the training dataset are used to create the matrix columns, each column consisting of a unique word encountered in the training dataset. The cells are the relative frequency, or affinity values, in the training data that each word is associated with the corresponding class rows. As a stream of words to be classified is processed, the column for that word is looked up in the matrix and, if found, recorded in a new matrix, P. If not found, this new word is written into the matrix and its matrix column and row in P populated with the column values for the closest word already in the matrix. Finally, as each new word is encountered and entered in P, the class activity decision is determined by summing the value of the affinities for the current occurrence and the affinity values for the previous w occurrences, where w is a parameter that was varied in several trials. The values are recorded in a separate matrix, $\bar{P}$, which holds the stream of summed affinity values. The class action decision is the maximum affinity value in the $\bar{P}$ row for the current occurrence. This is described in more detail below, and illustrated in Figure 5.1.

To build A, from the training dataset A_{ac} is computed as the number of times a word c occurs in action/class a. $\bar{A}$ is computed by normalizing A such that each row sums to 1, i.e.

$$\bar{A}_{ac} = A_{ac} / \sum_{c=1}^{N} A_{ac}, \forall a, c$$

From $\bar{A}$, $\hat{A}$ is computed by normalizing columns of $\bar{A}$, i.e. by making columns of $\hat{A}$ into unit vectors. N is the number of distinct words found in the training dataset. Note that N is usually less (much less for $n > 4$) than the total space of all possible words.

$\hat{A}$ is used in the recognition phase. Let the signal samples to be classified be $\mathbf{x}_i, i = 1, 2, \ldots m$, where m is the number of instances, converted to words, presented for classification.

A matrix P is created one row at a time, each row corresponding to a newly encountered word. The i-th row P_i of P is the c_ith column $\hat{A}_{c_i}$ of $\hat{A}$, where $c_i = \mathbf{x}_i$ or the nearest word to $\mathbf{x}_i$. The classification step simply estimates the class a_i as

$$a_i = \text{argmax}\{P_i\}, \tag{5.1}$$

i.e. finds the class corresponding to the index of the largest value in the row P_i.

These values can be quite noisy since they treat all word instances in isolation. An improvement involves taking into account the time series dependency of adjacent signal instances. Instead of using P for recognition, create $\bar{P}$ by summing the affinity values in the rows of P occurring immediately before the particular instance of P currently being classified. The number of values to be summed is determined by a window parameter w. For an instance x_i to be classified that is part of a sequence containing m instances, the predicted class for the ith instance in the sequence is computed using the w affinity values of the rows between $i - w$ and i as follows

$$\bar{P}_i = \begin{cases} \sum\limits_{j=1}^{i} P_j, & 0 < i \leq w \\ \sum\limits_{j=i-w}^{i} P_j, & w < i \leq m \end{cases} \tag{5.2}$$

Note that in Eq. (5.2), the first $(w-1)$ values in the sequence have less than w values and must be handled as a special subcase.

The predicted class a_i is estimated from $\bar{P}_i$ as in Eq. (5.1).

$\bar{A}$ and $\hat{A}$ were estimated by computing affinity values of unique words and activity classes. The columns of $\bar{A}$ and $\hat{A}$ correspond to unique words in the training set, and each column j corresponds to a unique word s_j. Call this set S. Given a signal 8-tuple $\mathbf{x}_i$ to be classified, if its corresponding word s_i exists in $\hat{A}$ it is added as a row to P_i and $\bar{P}_i$, and is used to recognize the class a_i using Eqs. (5.1-5.2). If s_i does not appear in $\bar{A}$, the set $S_i = \{s_{i_1}, s_{i_2}, \ldots\}$ of symbols is found in S which are closest to s_i using lexical distance $d_l(s_i, s) = \sum_{k=1}^{8} |s_i(k) - s(k)|$ for symbols s_i and s. Given S_i, columns $\bar{A}_{i_1}, \bar{A}_{i_2}, \ldots$ are added corresponding to symbols S_i to form a vector $\mathbf{p}_i$. $\mathbf{p}_i$ is normalized to 1 and transposed to form the row P_i which is then used for recognition using Eqs. (5.1-5.2).

Fig. 5.1 illustrates the structure of the Affinity and P matrices as well as their relationship. Matrix $\bar{P}$ has the same structure as P except that the values of $\bar{P}$ are created using the affinity summation scheme as described by Eq. (5.2).

The effect of the affinity summation scheme is to change the prediction of class “X” found in the middle of a long sequence of class “Y” by taking advantage of the temporal context

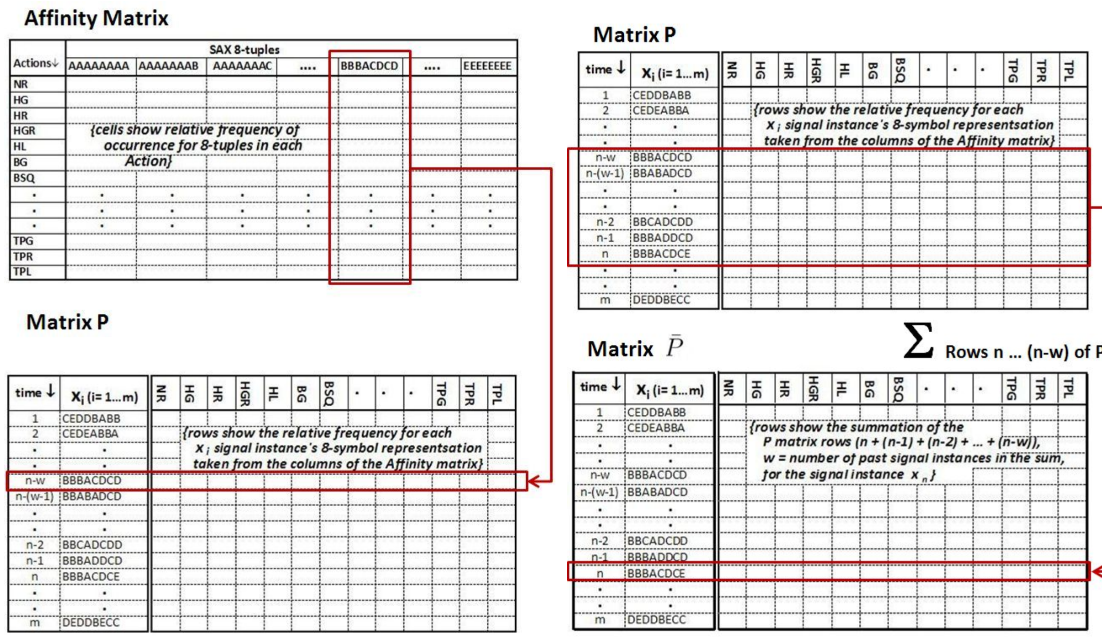


Figure 5.1: Left shows creation of the P matrix from the Affinity matrix. Right shows the creation of $\bar{P}$ from the P matrix with an affinity summation window of w (summation of the current classified instance plus the previous w rows) as described by Eq. 5.2.

information inherent in time sequences. In many instances, noise in the data can introduce clearly wrong predictions that this process corrects. Here the size of the summation window was varied in an attempt to find an optimum balance between an accurate prediction of a grip or movement sequence and limiting the size of the window. Larger values of w result in higher accuracies, but require that more information be known about the sequence. Since this technique uses information that occurs before the instance to be classified, it is suitable for use in real-time applications since no "future" information need to be seen prior making a classification decision.

5.3 Dynamic Time Warping (Approach 3)

The Affinity Matrix approach attempts to take advantage of the time series nature of the signal data by considering the context in which a 100 ms SAX 8-tuple instance (word) occurs.

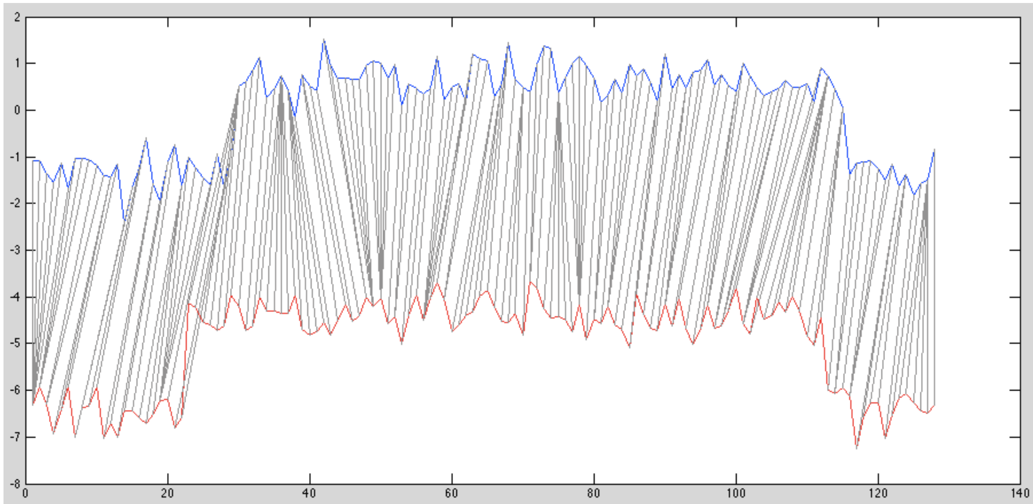


Figure 5.2: The Euclidean distance between two time series (top - blue line, bottom - red) will not be match along the time scale shown on the X-axis. The connecting lines show the Dynamic Time Warping (DTW) distance that attempts to match the two within a predetermined warping window. A parameter is set to govern the width of the path for comparison between the two series. Adapted from [41].

Specifically, it takes into account the prediction of the current instance under review, x_i, and the previous n instances. The approach described in this section also attempts to exploit the sequential nature of the data. Here, however, the shape of the entire activity (e.g., hammer raise, ball squeeze), which consists of a sequence of words, is matched against the patterns in the training data to find the corresponding action by using Dynamic Time Warping (DTW) [40] [38] [41] as a distance measure. A short description of DTW is given below, followed by the specifics of how it is used in this third learning approach.

Euclidean distance is a well-known distance measure in which sequences are aligned in a point-to-point fashion, i.e. the ith point in sequence Q is matched with the ith point in sequence C. Its simplicity and efficiency makes it a commonly used distance measure. While it often works well, it requires that both input sequences be of the same length and is sensitive to shifting along the time axis. For example, the top and bottom time series in Fig. 5.2 are not time-aligned and their values along the horizontal scale would not match.

Such a problem can generally be handled by more flexible distance measures such as

DTW. DTW uses dynamic programming to determine the best alignment to calculate the optimal distance. The warping window width parameter determines how much warping is allowed to find the best alignment [41]. A large window can increase processing time of the search and allow invalid matching between distant points. A small window, by contrast, could miss the best solution. Figure 5.2 demonstrates that with Euclidean distance, the dips and peaks in the two time series are misaligned and not matched at the same points on the horizontal scale, whereas DTW detects their alignment with nearby corresponding points as indicated by the lines connecting the top and bottom time series. The distance within which a match will be searched is governed by the warping window. While DTW is a more robust distance measure than Euclidean Distance, it is also more computationally intensive. [38] proposed an indexing scheme for DTW that allows faster retrieval. Nevertheless, DTW is still at least several orders slower.

In this approach, signal instances in the test data are presented for an entire activity for classification. An activity is one of the 47 actions (grips or movements) listed in Table 4.3 and consists of a sequence of consecutive words from one single repetition. For example, a group of the action 'hammer raise', HR, consists of a sequence $hr_1, hr_2, \ldots hr_m, m \leq 40$, collected in that order during one of the 12 hammer raise repetitions. Activities in the dataset were truncated to a maximum of 40 words since that is the maximum number of words in the movements and the significant parts of the grips occur in the first 40. The test action is then compared with all actions in the training data starting with only the first word in the test sequence (hr_1), then the first and second (hr_1, hr_2), then the first through third (hr_1, hr_2, hr_3), and so on until all words in that test action are compared ($hr_1 \ldots hr_m$). The size of the comparison is limited to either m, the number of words in the test sequence, or the number in the activity sequence from the training dataset if shorter.

For each comparison, the DTW is measured using a modified version of [42] and a K-nearest-neighbor classification (K-nn) scheme used to determine the class. Here $k = 1$ was computed and recorded for each comparison window ranging from $1 \ldots m$. A warping window of ± 5 was specified, for a total width of 11. The distance measure is a modified lexical distance between the test and training actions. The individual test and training words are compared one letter at a time and their differences are summed as follows:

$$D(a,b) = \sum_{i=1}^{8} d(a_i, b_i), \quad d(a_i, b_i) = \begin{cases} 0, a_i = b_i \\ |a_i - b_i| - 1, a_i \neq b_i \end{cases} \tag{5.3}$$

For example,

$D(`AAAAAAAA', `AACCBBEE') = 0 + 0 + 1 + 1 + 0 + 0 + 3 + 3 = 8.$

Computing the modified lexical distance in this way avoids the problem of assigning a distance of one to two adjacent values where both are close to the boundary between them and whose value would be much closer to zero than one [41].

This approach matches signal sequences whose shapes are similar but slightly out of line. The window parameter controls the flexibility of the match and the trade-off of large-versus-small was previously discussed. The approach also measures how soon a test sequence can be correctly recognized since the comparison is done in increasing numbers of words in a particular action. The results and conclusion sections discuss this.

5.4 Hidden Markov Model/Belief Calculation (Approach 4)

In the fourth approach a Hidden Markov Model (HMM) was developed, tested, and analyzed to operate on individual 100 ms MAV windows as learning instances. There is no conversion to SAX symbols. Similar to Approach 1, a Random Forest (RF) classifier was used to create a score for each instance that represents a probability that the instance belongs to one of the 47 classes. As each classification decision is made, a calculation was applied that uses context information to modify the decision and hopefully improve it. The goal is to identify a 'belief' state that reflects the true class of the instance being classified.

The RF voting process returns a score vector, one element for each of the 47 classes, that represents the percentage of generated trees that identified the specific class as the true one. The score vector is the observation part of the model since it is a decision based on direct input from the sEMG sensors. In Approach 1, the class with the highest score for a given 100 ms instance is judged to be the true class. No further processing occurred.

In Approach 4, additional processing is performed on the vector that uses contextual information to improve the accuracy of the classification decision. First, the RF scores (class probabilities) used to compute the previous instance's predicted class are considered, since the previous instance's class affects the current one. For example, a transition from a hammer grip to a hammer raise in adjacent 100 ms time intervals is possible, while a transition from a hammer raise to a key turn is not. Moving the hand from a power grip used to hold a hammer to the finer grip needed to grasp and begin turning a key requires more time and is essentially impossible to execute in 100 ms. While some RF scores show a high result for a particular class, indicating a strong decision, others show smaller scores spread across many classes. To account for this variation in scoring strength, the previous instance's scoring vector is considered in deciding the current instance's class.

The approach uses a single HMM, similar to [26] and the notation from [43] to describe it. Initially a simple transition matrix T was used in which the transitions follow the order in Table 4.3. T is a 47-by-47 matrix showing a probability of moving from a given class (row) to another class (column) in the next 100 ms time segment. The probabilities in the table were created by assigning relative weights using a best estimate for reasonable transitions. For example, a transition from neutral/rest to any transition-in state is possible and is assigned an appropriate positive value. On the other hand, a transition from hammer raise to a key turn in one 100 ms time slice is impossible and the associated table entry is given a low value. From neutral rest, there is an equal probability of staying in it or moving to a transition to any grip. Additional transition probabilities are defined in T using the weights in Table 5.1 and normalizing along the row to convert the weights into probabilities. The weights were determined empirically and are not necessarily optimum. Extra weight is given to 'no change' in state and moving from one grip or movement to the next sequential one (e.g., from hammer grip to hammer raise).

As in Approaches 2 and 3, the eight activities are divided into 12 repetitions each. The

Table 5.1: Weights assigned to the transitions in matrix T. The rows in T were normalized to convert them into transition probabilities.

transition from...	**transition to...**	**weight**
neutral/rest	neutral/rest	35
neutral/rest	transition-in - any grip	10
any state	the same state (NC)	50
transition-out - any grip	neutral/rest	10
any grip	transition-out - same family	10
any movement	grip - same family	10
transition-out - any grip	neutral/rest	10
any grip/movement	next grip/movement - same family	25
any grip/movement	previous grip/movement - same family	10
impossible transitions		.001

following is repeated twelve times. One repetition is held out for testing and to learn the parameters of the observation model from the remaining eleven repetitions, which form a training set.

In the learning phase the observation model is learned and has two components. The first uses the Random Forest (RF) classifier to classify 100 ms windows in the training set. The second generates a matrix G, a row-normalized 47×47 matrix where g_{ij} correspond to likelihood that a sample from a state s_i will be recognized as a state s_j^*. G is obtained as follows. Given the learning data $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, with labels $L = \{l_1, \ldots, l_N\}$ an RF classifier f_{RF} is generated, which for every 8-dimensional data sample returns a 47-dimensional vector corresponding to likelihood of each state/class.

The learned classifier is applied to the training data to generate scores $f_{RF}(\mathbf{x}_i)$, $i = 1, \ldots, N$, which correspond to the probability distribution over all states/classes. A matrix G^* is created where a row i of G^* is given as a sum of all $f_{RF}^T(\mathbf{x}_j)$ for which $l_j = s_i$, i.e. the sum of all scores for which the label was the state s_i. G is obtained by a row normalization of G^*. Given any test sample $\mathbf{y}$, $O = \text{diag}\{G f_{RF}(\mathbf{y})\}$ is used as observation, i.e. a column vector is turned into a diagonal matrix.

In the testing phase 200 data points $Y = \{\mathbf{y}_1, \ldots, \mathbf{y}_{200}\}$ are used which correspond to one repetition for one of the eight activities as input. The forward filtering algorithm is used to compute state beliefs. Each sequence **f** is initialized with belief 1 for neutral rest and 0 for all other states. The algorithm is then

$$\mathbf{f}_{1:t+1} = \alpha O_{t+1} T \mathbf{f}_{1:t}, \quad t = 1, \ldots, 200$$

where α is the normalization constant. The classification result is obtained as

$$s_t = \mathrm{argmax}\{\mathbf{f}_{1:t}\},$$

i.e. the class for which the belief is maximal at t.

Chapter 6: Results

This chapter presents the results for the the five subjects and four approaches, with a separate section for each approach. For each approach data collected from the subjects were analyzed using the 47 grips and movements listed in Table 4.3. The results are discussed and compared in the following chapter. Detailed results in the form of confusion matrices for the same two subjects are shown for each approach to ensure a consistent comparison. Those are subject #4, who showed relatively high accuracy, and subject #2, with relatively low accuracy.

6.1 Approach 1

Results reported in [1] showed Random Forest with 25 trees (RF25) having the highest overall accuracy compared with other classification methods. RF25 was therefore used in Approach 1 and its results reported here. The Weka workbench was used to build the classification model and evaluate using stratified ten fold cross-validation. The average accuracy for the five subjects was 71.28%. Table 6.1 shows the overall accuracy for all subjects for Approach 1.

Figures 6.1 and 6.2 are confusion matrices showing results for subject 4 and 2, respectively. Subject 4's accuracy was relatively high: 76.49%. Subject 2's was relatively low: 66.52%.

6.2 Approach 2

For the second Learning Approach, Affinity Matrix, a custom MATLAB® solution was developed. Unlike Approach 1, which only classified individual words, this one sums the

subj4	NR	HGI	HG	HR	HGR	HL	HLO	JLGI	JLG	JLP	JLRP	JLS	JLO	BGI	BG	BSQ	BSQ	DKG	DKG	DKT	DKT	DKT	DKT
NR	6366	2	0	0	4	0	13	2	6	2	30	6	8	6	5	0	26	2	7	0	19	2	9
HGIN	11	28	27	2	7	10	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HG	0	6	547	8	13	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HR	0	2	9	111	17	1	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HGR	0	0	10	11	310	6	0	0	4	0	0	2	0	0	0	0	0	0	0	0	0	0	0
HL	1	4	2	8	10	101	0	0	0	1	0	3	0	0	0	0	0	0	0	1	0	1	0
HLOUT	43	1	3	0	1	5	59	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
JLGIN	7	0	0	0	3	0	0	28	21	4	1	1	0	0	0	0	0	0	2	0	0	1	0
JLG	1	0	0	0	1	0	0	13	556	8	0	0	0	0	0	0	0	1	23	0	0	0	0
JLP	2	0	0	0	0	0	0	0	15	96	17	0	0	0	0	0	0	0	2	1	0	13	1
JLRP	108	0	0	0	0	0	1	0	1	7	176	5	2	0	0	0	0	0	3	0	4	0	3
JLS	4	0	0	0	1	0	0	0	5	0	1	138	1	0	0	0	0	1	3	2	1	5	0
JLOUT	60	0	0	0	0	0	2	0	0	0	17	12	8	0	0	0	0	0	0	1	6	0	1
BGIN	13	0	0	0	0	0	0	0	0	0	0	0	0	108	48	0	0	0	0	0	0	0	0
BG	3	0	0	0	0	0	0	0	0	0	0	0	0	20	490	5	0	0	0	0	0	0	0
BSQ	0	0	0	0	0	0	0	0	0	0	0	0	0	1	4	583	8	0	0	0	0	0	0
BSQOUT	105	0	0	0	0	0	0	0	0	0	0	0	0	4	0	3	130	0	0	0	0	0	0
DKGIN	12	0	0	0	1	1	0	0	10	0	2	1	0	0	0	0	0	13	47	0	0	8	0
DKG	20	0	0	0	0	0	0	0	25	0	1	2	1	0	0	0	0	4	478	7	1	1	0
DKTS	1	0	0	1	0	1	3	0	1	0	0	1	0	0	0	0	0	0	1	189	6	0	0
DKTR	160	1	0	0	1	0	0	0	2	0	4	1	2	0	0	0	0	2	7	17	87	3	8
DKTP	11	1	0	0	0	2	0	2	8	6	0	9	0	0	0	0	0	1	11	0	1	212	3
DKTOUT	70	0	0	0	0	0	0	0	0	5	13	0	0	0	0	0	0	0	0	0	8	11	37
KGIN	27	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
KG	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
KGTS	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
KGTR	96	0	0	0	0	0	0	0	0	0	3	0	1	0	0	0	0	0	1	0	1	1	0
KGTP	27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0
KGOUT	64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCGIN	30	2	4	0	11	0	2	0	5	0	0	1	1	0	0	0	0	1	2	0	0	0	0
SCG	57	0	9	1	31	0	3	0	24	0	0	1	0	0	0	0	0	0	1	0	0	0	0
SCO	4	0	3	0	0	0	1	0	1	0	1	13	0	0	0	0	0	0	0	0	0	0	0
SCGO	53	0	0	0	1	0	11	0	3	0	0	1	0	0	0	0	0	0	1	0	1	0	0
SCC	2	8	10	3	10	3	1	1	0	0	0	3	0	0	0	0	0	0	1	0	0	1	0
SCOUT	24	1	11	0	12	3	12	2	6	1	1	10	0	0	0	0	0	1	1	2	2	2	0
3JCGIN	39	0	0	0	1	0	1	0	2	0	1	0	0	0	0	0	0	0	12	0	0	0	0
3JCG	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	11	0	0	0	0
3JCR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3JCGR	2	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	1	0	0	0	0
3JCL	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3JCOUT	100	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0
TPGIN	39	0	0	0	0	0	0	0	0	1	3	0	0	0	0	0	0	0	5	0	0	3	4
TPG	0	0	0	0	0	0	0	0	0	2	0	2	0	0	0	0	0	1	6	0	0	0	1
TPR	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	4	0	0	0
TPGR	6	1	0	0	0	0	0	0	3	0	1	0	0	0	0	0	0	0	2	0	0	0	1
TPL	24	0	0	0	0	0	1	0	0	1	5	0	0	0	0	0	0	0	2	0	0	0	5
TPLOUT	57	0	0	0	0	0	0	0	0	0	9	0	1	0	0	0	0	1	2	0	4	0	3

subj4	KGI	KG	KGT	KGT	KGT	KGO	SCG	SCG	SCO	SCG	SCC	SCO	3JC	3JC	3JCR	3JC	3JCL	3JC	TPG	TPG	TPR	TPG	TPL	TPLOUT
NR	1	6	0	27	1	4	5	18	10	33	2	19	5	2	0	10	1	17	2	1	2	10	4	13
HGIN	0	1	0	0	0	0	1	3	0	0	7	1	0	5	0	5	2	1	0	0	0	0	0	0
HG	0	0	0	0	0	0	1	13	0	0	4	2	0	0	0	9	0	0	0	0	0	0	0	0
HR	0	0	0	0	0	0	0	0	0	0	3	1	0	0	0	0	0	0	0	0	0	0	0	0
HGR	0	0	0	0	0	0	1	14	0	0	4	5	0	0	0	0	0	0	0	0	0	0	0	0
HL	0	0	0	0	0	0	0	1	0	0	6	5	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	0	0	0	1	0	1	1	13	1	13	1	8	2	0	0	2	0	1	0	0	0	0	1	0
JLGIN	0	0	0	0	0	0	0	0	0	4	0	8	0	0	0	0	0	1	0	0	0	2	1	0
JLG	0	0	0	0	0	0	0	14	1	1	0	2	1	0	0	4	0	0	0	3	0	4	0	0
JLP	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	1	2	0
JLRP	0	0	0	2	0	0	0	0	0	2	0	0	0	0	0	1	0	2	0	1	0	1	6	2
JLS	0	0	0	0	0	0	2	4	2	3	2	12	0	0	0	0	0	1	0	3	0	0	0	1
JLOUT	0	1	1	6	0	0	3	0	0	5	0	0	0	1	0	0	0	1	1	0	0	2	1	2
BGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKGIN	0	1	0	0	1	0	0	0	0	0	0	0	0	2	0	0	0	1	0	2	0	0	1	1
DKG	0	3	0	2	0	0	0	3	0	4	0	2	3	10	0	0	2	0	0	9	0	2	2	0
DKTS	0	0	0	0	0	0	0	0	4	1	0	1	0	1	0	0	0	0	0	0	1	0	0	0
DKTR	0	0	2	4	2	0	0	0	0	2	0	3	1	3	0	0	2	0	2	1	2	3	0	2
DKTP	0	1	1	0	0	0	0	0	0	0	1	2	1	4	0	2	2	1	0	0	0	1	0	0
DKTOUT	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	2	1	0	3	9	0	0	4	2
KGIN	14	41	2	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	3	3	0	0	2	0
KG	7	527	7	7	3	0	0	0	0	0	0	0	0	2	0	0	0	0	2	7	0	3	5	3
KGTS	0	6	100	19	0	0	0	0	0	0	0	0	0	1	0	2	1	0	0	1	4	11	0	0
KGTR	0	3	10	211	2	2	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	5	2	0
KGTP	1	9	0	14	94	6	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	5
KGOUT	2	0	0	3	8	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
SCGIN	0	0	1	2	0	0	9	19	2	2	3	16	0	1	0	2	1	0	0	0	0	11	0	0
SCG	0	0	0	0	0	0	1	326	3	19	1	27	0	4	0	2	0	0	0	0	0	4	2	0
SCO	0	0	0	0	0	0	1	7	71	32	0	8	0	0	0	0	0	0	0	4	0	1	0	0
SCGO	0	0	0	0	0	0	0	40	23	250	5	17	0	4	0	0	0	0	0	2	0	7	0	0
SCC	0	0	0	0	0	0	2	4	7	1	109	21	0	0	0	0	0	0	0	0	0	0	0	0
SCOUT	0	0	0	0	0	0	3	48	21	24	29	147	0	0	0	0	0	0	0	0	2	2	0	0
3JCGIN	0	1	0	5	0	0	0	2	0	1	0	4	37	38	0	2	3	1	0	0	0	2	0	0
3JCG	0	2	3	0	0	0	0	2	0	1	0	0	16	444	10	12	18	0	0	0	3	4	1	0
3JCR	0	0	0	0	0	0	0	1	0	0	0	0	0	2	143	15	1	0	0	0	6	0	0	0
3JCGR	0	0	0	1	0	0	0	4	0	0	0	0	1	13	15	288	15	0	0	0	0	0	0	0
3JCL	0	2	0	3	1	0	0	0	0	0	0	0	2	21	3	28	115	3	1	0	0	0	1	0
3JCOUT	0	0	0	2	1	0	0	0	0	0	0	0	3	2	0	0	4	30	1	0	0	0	1	2
TPGIN	1	12	0	0	1	0	0	0	2	0	0	0	0	0	0	0	0	0	24	42	0	0	4	4
TPG	0	4	2	0	0	0	0	0	3	5	0	0	0	0	0	0	0	0	5	535	4	1	4	0
TPR	0	0	3	0	0	0	0	0	0	0	0	1	0	0	2	0	0	0	0	7	129	16	0	0
TPGR	0	0	2	2	0	0	2	0	0	2	0	0	0	2	0	1	0	0	0	2	16	350	11	1
TPL	1	23	0	5	3	0	0	0	0	1	0	0	0	2	0	0	0	0	1	3	0	20	69	3
TPLOUT	0	8	0	1	12	1	0	0	0	2	0	0	0	0	0	0	0	1	4	0	0	3	3	37

Figure 6.1: Confusion matrix for subject 4, Approach 1 (Random Forest with 25 trees). Classes listed in the rows are the ground truth for one word instance. The total number of predicted classes for all subjects are shown in the cells corresponding to the class in the columns. Higher numbers (200+ or $\approx 1\% - 5\%$ of total) are shown in black, moderate numbers (100-199, $\approx 0.5 - 1\%$) in dark gray shade, smaller (30-99) in light gray, small in light blue (5-29, $< 0.12\%$), and fewer than 5 in white. Strong dark colors along the diagonal indicate correct predictions. Many incorrect predictions are clustered near their action group (e.g., hammer, tip pinch, 3-jaw-chuck) inside the grip family's box along the diagonal. Note the confusion between the tip pinch group (TPx) and neutral/rest (NR).

Table 6.1: Overall accuracies for all subjects for the Approach 1 RF25 classifier. Ranges are 63.70 for subject 5 (less than fully-abled) to 82.69 for subject 1, with an average of 71.28% for all subjects.

subject#	**Accuracy(%)**
1	82.69
2	66.52
3	67.00
4	76.49
5	63.70
average	71.28

subj2	NR	HGIN	HG	HR	HGR	HL	HLOU	JLGIN	JLG	JLP	JLRP	JLS	JLOUT	BGIN	BG	BSQ	BSQO	DKGI	DKG	DKTS	DKTR	DKTP	DKTO
NR	5046	5	0	1	2	0	9	0	2	1	6	1	8	17	75	0	12	4	14	4	24	0	6
HGIN	34	43	45	0	0	0	8	1	5	2	4	3	0	0	2	0	0	0	9	3	2	0	0
HG	0	5	490	7	6	0	2	0	3	0	4	0	0	0	0	0	0	0	0	0	0	0	0
HR	0	0	5	110	25	11	1	0	0	2	0	0	0	0	0	1	0	0	0	0	0	0	0
HGR	0	0	10	15	342	8	0	1	5	1	4	0	0	0	0	0	0	0	4	5	0	1	0
HL	1	1	0	7	13	108	5	0	1	0	0	0	0	0	0	4	0	0	0	6	0	0	0
HLOUT	70	5	13	0	4	9	57	0	0	0	4	2	1	5	0	0	6	0	1	5	13	0	0
JLGIN	16	0	0	0	0	0	0	7	56	0	3	2	0	0	0	0	0	0	3	0	0	0	1
JLG	2	0	0	0	1	0	1	2	526	11	11	2	2	0	0	0	0	0	27	9	1	0	1
JLP	2	0	2	0	0	0	0	0	2	163	29	4	0	0	0	0	0	0	0	1	0	0	0
JLRP	4	4	0	0	0	0	0	0	18	25	263	7	0	0	0	0	0	0	5	2	0	0	0
JLS	1	1	0	0	0	0	0	2	3	17	8	67	44	0	0	0	1	0	10	0	0	0	0
JLOUT	71	1	0	0	0	0	1	0	1	1	2	23	149	3	0	0	0	2	5	2	6	1	6
BGIN	87	0	0	0	0	0	3	0	0	0	0	0	1	78	30	0	9	1	0	0	19	0	2
BG	188	1	0	1	2	2	1	0	0	0	1	0	0	14	310	3	3	0	1	0	4	0	0
BSQ	0	0	0	2	1	6	1	0	0	1	0	0	0	1	0	481	0	0	0	1	0	0	0
BSQOUT	107	1	2	0	3	0	5	0	2	1	5	0	2	11	5	3	78	0	3	5	8	4	0
DKGIN	32	0	0	0	0	1	1	0	1	1	0	4	3	0	0	0	0	34	21	0	2	9	1
DKG	25	1	1	0	1	0	1	0	28	0	3	5	4	0	0	0	0	5	528	8	0	2	0
DKTS	2	0	0	0	8	0	1	1	2	2	9	1	2	0	0	0	1	2	24	250	1	1	0
DKTR	132	0	0	0	0	0	0	0	1	0	1	0	3	12	1	0	7	3	13	12	110	2	2
DKTP	6	0	0	0	1	0	0	0	3	1	1	1	3	5	0	0	3	3	6	7	6	220	4
DKTOUT	54	0	0	0	0	0	0	0	0	0	2	0	4	5	0	0	2	1	0	1	6	16	57
KGIN	42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0
KG	209	0	0	0	0	0	1	0	0	0	0	0	0	2	0	0	0	0	2	0	0	0	0
KGTS	7	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	2	1	0	1
KGTR	55	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0
KGTP	11	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	3	0	0	0	3	0	0
KGOUT	113	0	0	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0
SCGIN	33	5	0	0	2	1	8	0	0	1	6	0	0	2	2	0	5	0	0	2	4	0	0
SCG	48	1	1	0	4	0	2	0	0	1	5	1	0	3	5	0	1	0	0	3	0	0	0
SCO	18	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0	0	0	0	1	3	0	0
SCGO	5	1	0	0	1	0	1	0	0	2	2	5	1	0	0	0	3	0	0	0	0	0	0
SCC	8	0	0	0	0	0	2	0	0	13	3	0	0	0	0	0	0	0	0	2	0	2	0
SCOUT	27	2	0	0	1	3	6	0	1	7	8	1	0	7	2	0	4	0	0	4	2	1	0
3JCGIN	71	1	3	0	0	0	1	1	1	0	4	1	1	2	1	0	0	0	2	2	5	0	0
3JCG	137	1	0	0	0	0	0	0	0	0	2	0	0	0	7	0	2	0	0	0	0	0	0
3JCR	7	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
3JCGR	34	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
3JCL	18	1	1	0	0	0	1	0	0	0	0	0	0	1	3	0	0	0	0	0	0	0	0
3JCOUT	47	0	0	0	0	0	1	0	0	0	1	0	1	5	1	0	7	0	0	0	2	0	0
TPGIN	94	0	0	0	0	0	0	0	0	0	0	0	0	4	1	0	0	0	2	0	2	0	1
TPG	352	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	1	0	0
TPR	31	0	0	0	0	0	0	0	0	0	0	0	1	2	1	0	0	0	0	0	2	0	0
TPGR	182	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	1	0	1
TPL	84	0	0	0	0	0	0	0	0	0	0	0	0	4	5	0	0	0	1	0	0	0	0
TPLOUT	84	0	0	0	0	0	0	0	0	0	0	0	0	2	6	0	0	0	1	1	3	0	1

subj2	KGIN	KG	KGTS	KGTR	KGTP	KGOU	SCGIN	SCG	SCO	SCGO	SCC	SCOU	3JCGI	3JCG	3JCR	3JCGR	3JCL	3JCOU	TPGIN	TPG	TPR	TPGR	TPL	TPLOUT
NR	1	93	0	34	0	22	12	50	2	1	6	27	2	72	11	18	7	26	4	106	11	48	3	8
HGIN	0	0	0	0	0	0	3	1	0	1	0	0	0	0	1	12	0	0	0	0	0	0	0	0
HG	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HR	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0
HGR	0	0	0	0	0	0	2	3	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0
HL	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	1	2	0	2	1	0	7	7	0	0	0	2	0	2	0	2	2	1	0	0	1	1	0	1
JLGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	2
JLG	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
JLP	0	0	0	0	0	0	0	0	0	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0
JLRP	0	0	0	0	0	0	1	2	0	3	1	2	0	0	0	0	0	0	0	0	0	0	0	0
JLS	0	0	0	0	0	0	0	2	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0
JLOUT	0	0	0	0	0	0	0	1	0	3	0	0	0	0	1	0	1	2	0	0	2	3	0	0
BGIN	0	4	3	0	0	1	0	5	1	1	0	8	0	1	0	0	2	8	0	0	1	0	1	1
BG	0	2	1	8	0	2	0	2	0	3	0	0	0	15	0	0	1	2	0	5	1	0	1	0
BSQ	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0
BSQOUT	0	6	2	0	2	0	1	8	2	5	0	10	0	1	0	1	0	5	0	2	0	2	0	0
DKGIN	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1
DKG	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
DKTS	0	0	0	0	0	0	0	2	0	2	0	2	0	0	0	0	0	0	0	0	0	0	0	0
DKTR	0	5	3	2	2	0	0	1	1	0	0	0	3	1	0	0	0	1	1	0	0	2	0	0
DKTP	0	0	0	0	0	0	0	0	1	1	2	1	0	0	0	0	0	0	0	0	0	0	0	1
DKTOUT	0	0	3	0	0	0	0	0	0	0	0	0	0	2	1	0	0	1	0	0	2	2	0	0
KGIN	0	44	0	1	1	1	0	0	0	0	0	0	1	2	0	0	0	0	2	3	0	6	0	0
KG	2	330	9	6	3	11	0	0	0	0	0	0	0	3	0	0	0	0	1	15	0	17	0	2
KGTS	0	5	86	39	3	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0	0	0
KGTR	0	7	17	271	8	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2	0
KGTP	0	9	7	15	89	11	0	3	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0
KGOUT	0	28	1	2	4	59	0	0	0	0	0	0	0	2	0	0	0	0	0	3	0	6	0	0
SCGIN	0	1	1	2	0	0	51	64	1	3	2	18	0	0	0	1	4	1	0	0	1	0	0	0
SCG	0	0	1	1	0	0	21	275	12	45	0	17	1	26	1	3	6	1	1	0	0	0	0	0
SCO	0	0	0	0	0	0	0	3	89	31	10	3	0	0	0	2	0	0	0	0	3	0	0	0
SCGO	0	0	0	0	0	0	2	19	19	264	9	25	0	0	0	0	0	0	0	0	1	0	0	0
SCC	0	0	0	0	0	0	3	6	5	3	102	27	0	0	0	0	0	1	0	0	0	0	0	0
SCOUT	0	0	0	0	0	0	15	23	7	16	32	160	0	2	1	0	2	3	0	0	0	0	0	0
3JCGIN	1	8	0	0	0	0	1	11	0	0	0	0	6	23	1	0	2	6	2	0	0	2	0	1
3JCG	0	8	0	2	0	0	0	8	0	0	0	0	2	377	7	2	4	20	0	0	0	0	0	0
3JCR	0	0	0	0	0	0	0	0	0	0	0	0	0	10	113	28	3	0	0	0	2	0	0	0
3JCGR	0	1	0	0	0	0	3	1	0	0	0	0	0	2	13	358	6	0	0	0	1	1	0	0
3JCL	0	0	0	0	0	0	0	7	0	0	0	5	0	8	4	20	58	29	0	0	0	1	0	0
3JCOUT	0	0	1	1	0	0	0	2	0	2	0	2	1	27	0	0	6	141	0	0	0	0	0	0
TPGIN	0	14	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	13	0	13	1	3
TPG	1	40	0	2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	146	5	15	1	2
TPR	0	0	2	1	0	0	0	0	0	0	0	0	0	0	3	1	0	0	0	3	96	9	0	0
TPGR	3	21	0	3	0	6	0	1	0	0	0	0	0	0	3	0	0	0	1	9	8	187	0	0
TPL	0	16	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	11	0	3	10	5
TPLOUT	0	14	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	2	8	0	9	6	21

Figure 6.2: Confusion matrix for subject 2, Approach 1 (Random Forest with 25 trees). Despite showing lower overall accuracy, subject 2's results show the similar patterns as those for subject 4.

computed affinity value for the instance being classified with the values of the previous w values from the input stream.

To evaluate the approach, the eight action group datasets are divided into 12 segments, one for each action repetition. Segment one, for example, includes the words in the first repetition of the hammer grip family (HGIN, HG, HR, HGRP, HL, and HLOUT), followed by the jar lid family (JLGIN, JLG, JLP, JLRP, JLS, and JLOUT), the ball family (BGIN, BG, BSQ, BSQOUT), etc. Likewise, the second segment includes the second repetition of all action groups. Neutral/rest (NR) words from seconds 1-3 for each of the action group runs were added to each of the 12 runs. This was done to inject a representative sample of that posture ($\approx 6\%$ of total) since including all NRs would result in their being $\approx 35\%$ of the total and unbalancing the dataset. Twelve separate classification training and testing runs were performed. Each segment was withheld as a test set for a run one time, with the remaining 11 used to build the Affinity Matrix. Accuracies from classifying the test sets in the twelve runs were averaged.

This approach requires two parameters: the number of SAX symbols used to discretize the signals, and w, the number of words immediately before the test instance used to create the affinity sum. The number of symbols was varied from 5 to 15, and the number of words from 3 to 30. Table 6.2 shows the various parameter combinations. The table contains the high-low range for the three subjects for each combination. The graph in Figure 6.3 shows the average accuracy for the three subjects for word counts up to 30. The lines for symbols 7, 11, and 15 are tightly clustered and superior to 5 for all word counts. Improvement flattens out between 20 and 30 words, indicating parameter settings that will produce maximum accuracy.

Figures 6.4 and 6.5 are confusion matrices showing results for subject 4 and 2, respectively, for 11 symbols and 30 words. Both subjects shoed their highest accuracies at those parameter levels. Subject 4's accuracy was relatively high: 79.15%. Subject 2's was 78.07% - lower, but closer to Subject 4 compared with results for Approach 1.

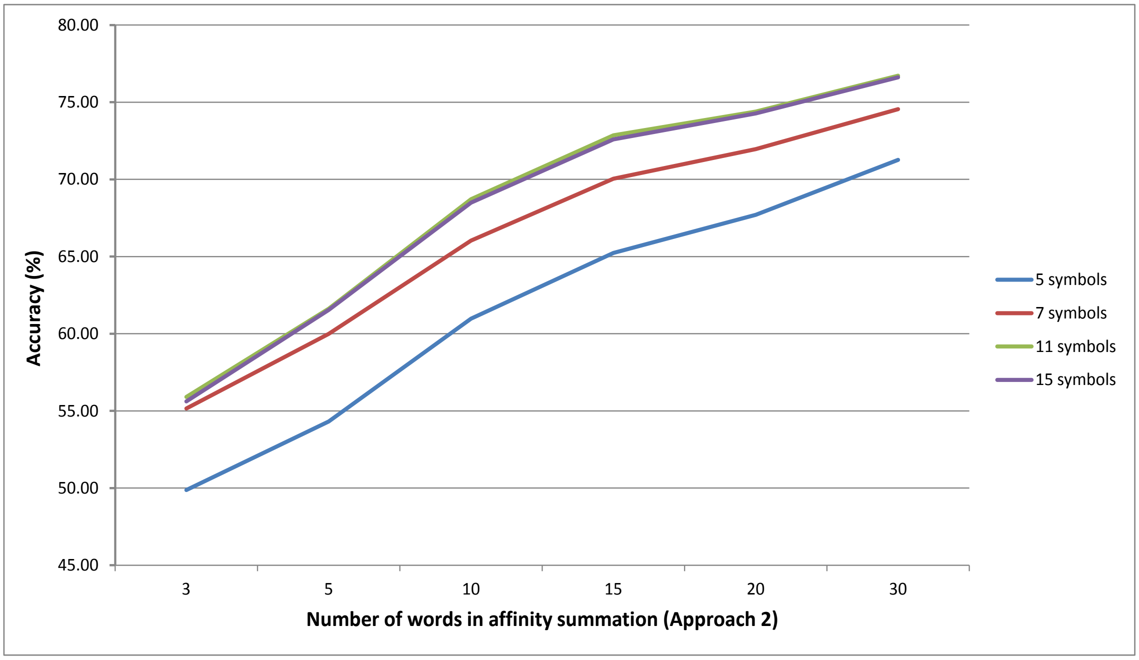


Figure 6.3: Graph of the Affinity Matrix approach for the average accuracy of all subjects for various numbers of symbols and words used in the affinity summation. The accuracy for 7, 11, and 15 symbols is tightly clustered and therefore similar, but superior to trials using only 5. Improvement is steep until the vicinity of 20 to 30 words for all numbers of symbols.

Table 6.2: Affinity result ranges among the 5 subjects for selected numbers of SAX symbols and number of words w used in affinity summation.

# symbols	**3**	**5**	**10**	**15**	**20**	**30**
5	[39.61, 64.76]	[43.50, 69.69]	[49.95, 75.98]	[54.58, 79.41]	[57.60, 80.64]	[61.41, 82.81]
7	[44.39, 68.88]	[48.74, 73.64]	[55.17, 78.73]	[60.27, 81.37]	[62.79, 82.22]	[65.50, 83.42]
11	[44.30, 70.06]	[49.83, 75.16]	[57.31, 81.05]	[62.72, 83.58]	[65.58, 84.38]	[69.32, 85.03]
15	[44.46, 69.54]	[50.25, 74.63]	[58.44, 79.88]	[62.99, 82.31]	[65.31, 83.15]	[68.69, 83.98]

subj4	NR	HGIN	HG	HR	HGR	HL	HLOU	JLGIN	JLG	JLP	JLRP	JLS	JLOUT	BGIN	BG	BSQ	BSQO	DKGI	DKG	DKTS	DKTR	DKTP	DKTO
NR	261	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HGIN	0	3	6	3	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
HG	0	0	348	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HR	0	0	0	38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HGR	0	0	14	10	233	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HL	0	0	0	0	2	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	1	0	4	0	0	0	33	0	0	0	0	0	5	2	0	0	0	0	0	0	0	0	0
JLGIN	0	0	0	0	0	0	0	9	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
JLG	0	0	0	0	0	0	0	7	344	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLP	0	0	0	0	0	0	0	0	0	45	1	0	0	0	0	7	0	0	0	0	0	2	0
JLRP	0	0	0	0	0	0	0	0	0	0	195	0	9	0	0	0	0	0	0	0	0	0	9
JLS	0	0	0	0	0	0	0	0	0	0	1	65	0	0	0	0	0	0	0	0	0	1	0
JLOUT	0	0	0	0	0	0	0	0	0	1	3	4	13	0	0	0	6	0	0	0	2	0	0
BGIN	0	0	0	0	0	0	3	0	0	0	0	0	0	51	6	0	0	0	0	0	0	0	0
BG	0	0	0	0	0	0	0	0	0	0	0	0	0	5	359	0	0	0	0	0	0	0	0
BSQ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	372	0	0	0	0	0	0	0
BSQOUT	0	0	0	0	0	0	12	0	0	0	0	1	1	0	0	0	104	0	0	0	0	0	0
DKGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8	5	0	0	0	0
DKG	11	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0	0	27	296	0	0	0	0
DKTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	104	0	0	0
DKTR	7	0	0	0	0	0	14	0	0	0	3	0	27	0	0	0	9	0	0	16	115	0	20
DKTP	14	0	0	0	0	4	0	3	0	0	0	2	0	0	0	0	0	0	3	0	0	145	0
DKTOUT	1	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	16	7	38
KGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGTR	0	0	0	0	0	0	0	0	0	0	12	0	14	0	0	0	0	0	0	0	0	0	0
KGTP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGOUT	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCGIN	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	7	0	0	0	0
SCG	0	0	4	0	20	0	0	0	9	0	3	0	0	0	10	0	0	0	0	0	0	0	0
SCO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCGO	2	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	25	0	0	0	0	0	0
SCC	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCOUT	0	0	13	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3JCGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3JCG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	0	0
3JCR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3JCGR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3JCL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3JCOUT	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TPGIN	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TPG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8
TPR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0
TPGR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TPL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TPLOUT	91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

subj4	KGIN	KG	KGTS	KGTR	KGTP	KGOU	SCGIN	SCG	SCO	SCGO	SCC	SCOU	3JCGI	3JCG	3JCR	3JCGR	3JCL	3JCOU	TPGIN	TPG	TPR	TPGR	TPL	TPLOUT
NR	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	86
HGIN	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
HG	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	15	0	0	0	0	0	0	0	0
HR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HGR	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	0	0	0	0	0	0	3	11	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLGIN	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
JLG	0	0	0	0	0	0	0	13	0	4	0	0	0	0	0	4	0	0	0	0	0	0	0	0
JLP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLRP	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLS	0	0	0	0	0	0	0	0	7	0	5	5	0	0	0	0	0	0	0	0	0	0	0	0
JLOUT	0	0	0	6	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	0	0
BGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	3	0	0	0	0	0	0	0	0
BG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQOUT	0	0	0	0	0	0	0	0	0	22	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKGIN	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
DKG	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	1	0
DKTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKTR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKTP	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0
DKTOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0
KGIN	8	3	0	0	1	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
KG	15	325	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	29	0
KGTS	0	0	41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
KGTR	0	0	0	207	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGTP	0	4	0	7	40	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGOUT	0	0	0	0	1	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCGIN	0	0	0	5	0	0	6	2	0	0	1	2	0	0	0	1	0	0	0	0	4	2	0	0
SCG	0	0	0	0	0	0	11	223	0	33	0	4	0	2	0	3	0	9	0	0	0	0	7	0
SCO	0	0	0	0	0	0	0	3	34	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCGO	0	0	0	0	0	0	0	38	14	224	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCC	0	0	0	0	0	0	0	0	3	0	74	0	0	0	0	0	0	0	0	0	0	0	0	0
SCOUT	0	0	0	0	0	0	0	52	10	14	55	111	0	0	0	0	0	0	0	0	0	0	0	0
3JCGIN	0	0	0	0	0	0	0	0	0	0	0	0	33	6	0	0	0	5	0	0	0	0	0	0
3JCG	0	0	0	0	0	0	0	0	0	0	0	0	31	325	0	0	0	0	0	0	0	1	0	0
3JCR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0
3JCGR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	231	0	0	0	0	0	0	0	0
3JCL	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	76	0	0	0	0	0	0	0
3JCOUT	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0
TPGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	19	0	0	0	8
TPG	0	0	0	0	0	0	0	0	14	7	0	0	0	0	0	0	0	0	3	270	0	0	0	62
TPR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	11
TPGR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	236	0	48
TPL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	39	10
TPLOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	32	82	23	39	50	50

Figure 6.4: Confusion matrix for subject 4, Approach 2 (Affinity). Classes listed in the rows are the ground truth for one word instance. As with Approach 1, note strong dark colors along the diagonal, indicating correct predictions, and the tendency for incorrect predictions to be clustered within the same action group box.

subj2	NR	HGIN	HG	HR	HGR	HL	HLOU	JLGIN	JLG	JLP	JLRP	JLS	JLOUT	BGIN	BG	BSQ	BSQO	DKGI	DKG	DKTS	DKTR	DKTP	DKTO	KGIN	KG	KGTS	KGTR	KGTP	KGOU	SCGIN	SCG	SCO	SCGO	SCC	SCOU	3JCGI	3JCG	3JCR	3JCGF	3JCL	3JCOU	TPGIN	TPG	TPR	TPGR	TPL	TPLOUT
NR	274	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	0	0	62
HGIN	0	42	12	0	0	0	6	0	0	0	3	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0
HG	0	6	346	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HR	0	0	0	54	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HGR	0	0	0	25	258	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HL	0	0	0	1	0	41	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	1	16	4	0	11	2	62	0	0	0	0	0	0	5	0	0	0	0	0	11	1	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	4	1	0	0	0	0	0	0	0
JLGIN	0	0	0	0	0	0	0	1	12	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLG	0	0	0	0	0	0	0	2	330	0	31	0	0	0	0	0	0	0	6	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLP	0	0	0	0	0	0	0	0	0	107	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLRP	0	0	0	0	0	0	0	0	0	74	155	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLS	0	0	0	0	0	0	0	0	0	21	0	34	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLOUT	1	0	0	0	0	0	0	0	0	0	0	53	126	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0
BGIN	0	0	0	0	0	0	6	0	0	0	0	0	0	80	7	0	10	0	0	0	27	0	0	0	0	0	0	0	13	0	1	0	0	0	0	0	0	0	0	5	10	0	0	0	0	0	0
BG	0	0	0	0	0	0	0	0	0	0	0	0	0	13	351	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0
BSQ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	352	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQOUT	1	0	0	0	0	0	0	0	0	0	0	0	0	2	22	0	142	0	0	0	0	1	0	0	1	0	0	8	0	0	10	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0
DKGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	4	0	2	1	8	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKG	0	0	0	0	0	0	0	0	0	0	0	10	21	0	0	0	0	0	321	4	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	194	0	0	0	0	0	0	0	0	0	0	0	0	0	5	6	0	0	0	0	0	0	0	0	0	0	0	0
DKTR	0	0	0	0	0	0	0	0	0	0	1	0	17	3	0	0	1	0	0	4	169	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	9	3	0	0	0
DKTP	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	2	0	0	0	5	148	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKTOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	3	13	42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	7	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	6
KG	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	29	229	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	2	2	69
KGTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	27	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13
KGTR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	221	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	38
KGTP	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	52	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6
KGOUT	25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	6	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28
SCGIN	0	0	0	0	0	0	9	0	0	0	0	0	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	50	17	5	0	0	16	0	0	0	0	1	0	0	0	0	0	0	0
SCG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	201	0	19	0	22	0	47	0	14	3	0	0	0	0	0	0	0
SCO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	28	9	5	0	0	0	0	0	0	0	0	0	14	0	0	0
SCGO	0	3	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	10	217	11	4	0	0	0	0	0	0	0	0	0	0	0	0
SCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	2	0	51	17	0	0	0	0	0	0	0	0	0	0	0	0
SCOUT	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	13	11	54	146	0	0	0	0	0	3	0	0	0	0	0	0
3JCGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	9	0	0	3	0	0	1	0	0	0	0	0	12	0	3	0	14	0	4	0	4	0	0
3JCG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	4	334	0	0	0	2	0	0	0	0	0	0
3JCR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	54	2	0	0	0	0	0	0	0	0
3JCGR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	307	0	0	0	0	0	0	0	0
3JCL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	7	42	1	0	0	0	0	0	0
3JCOUT	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	0	0	0	133	0	0	0	0	0	0
TPGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	42	6	0	9	0	0
TPG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	7	32	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	276	0	23	13	0
TPR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	43	0	0	0
TPGR	26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	12	0	1	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	18	16	231	1	0
TPL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	0	1	22	0
TPLOUT	165	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	64	13	49	13	44	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 6.5: Confusion matrix for subject 2, Approach 2 (Affinity). Although lower in overall accuracy, subject 2's results show the similar patterns as those for subject 4. A notable exception is the confusion of the TPLOUT class with the key grip (KG) family, indicated by dark colors along the bottom row and last columns corresponding with the intersection of those families.

Table 6.3: Dynamic Time Warping result ranges among the 3 subjects for selected numbers of SAX symbols and number of words used in the comparison.

# symbols	3	5	10	15	20	30
5	[27.48, 45.92]	[42.73, 64.18]	[53.01, 72.70]	[56.91, 81.03]	[60.46, 82.27]	[66.13, 85.82]
7	[23.58, 45.57]	[41.31, 66.67]	[54.61, 73.76]	[59.57, 81.21]	[64.01, 83.87]	[67.91, 85.46]
11	[25.89, 50.00]	[40.07, 66.31]	[55.67, 74.47]	[58.69, 81.38]	[64.18, 84.74]	[68.26, 86.70]
15	[25.18, 47.52]	[40.96, 68.49]	[53.35, 73.76]	[58.69, 81.56]	[65.07, 84.04]	[67.91, 86.88]

6.3 Approach 3

For the third Learning Approach, Dynamic Time Warping (DTW), the data was segmented as described for the second approach. Unlike that approach, which classified individual words based on summed affinity values of immediately preceding words, this approach only classifies activity segments, as described in the DTW approach section. This approach only requires one parameter: number of SAX symbols. However, since the comparison is made for an increasing number of words, the success rate at each word count is noted and reported. Results are therefore shown for various combinations of SAX symbols and word counts.

Table 6.3 shows results for the various parameter and word count combinations. The table contains the high-low range for the three subjects for each combination. The graph in Figure 6.6 shows the average values for the five subjects for word counts up to 30. The lines for all symbol values are tightly clustered and improve rapidly until reaching between 20 and 30 words.

Figures 6.7 and 6.8 are confusion matrices showing results for subject 4 and 2, respectively, for seven symbols and 30 words. Both subjects showed their highest accuracies at those parameter levels. Subject 4's accuracy was relatively high: 75.53%. Subject 2's was lower: 67.91%.

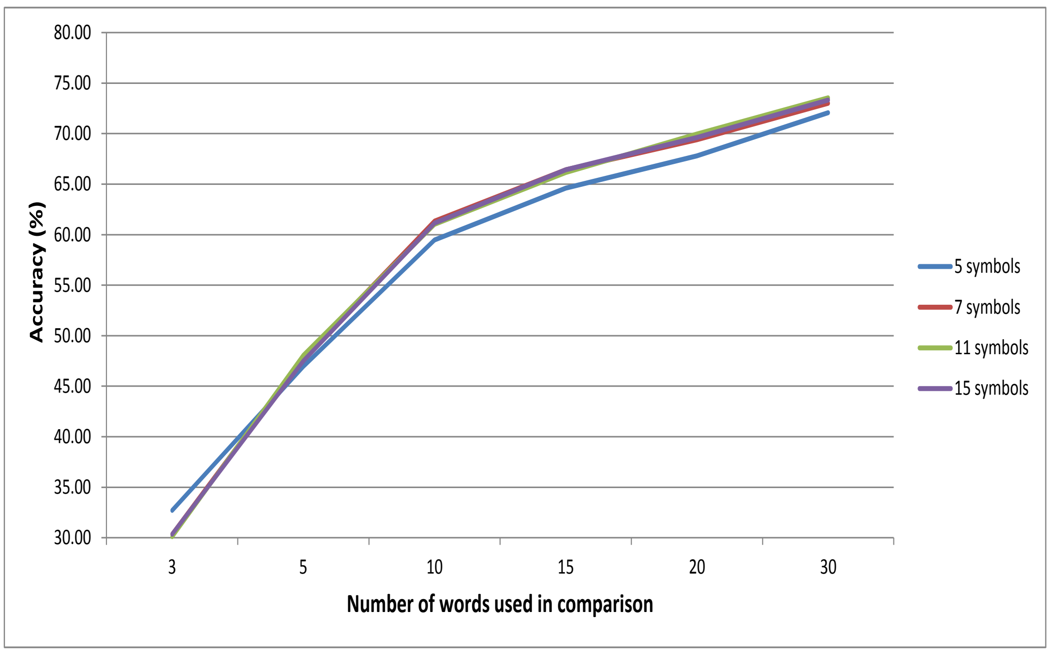


Figure 6.6: Graph of the Dynamic Time Warping approach for the average accuracy of all subjects for various numbers of symbols and words participating in the comparison. The accuracy flattens out between 20 and 30 words for all numbers of symbols. The number of SAX symbols used does not have much affect on the accuracy.

subj4	NR	HGIN	HG	HR	HGR	HL	HLOU	JLGIN	JLG	JLP	JLRP	JLS	JLOUT	BGIN	BG	BSQ	BSQO	DKGII	DKG	DKTS	DKTR	DKTP	DKTO	KGIN	KG	KGTS	KGTR	KGTP	KGOU	SCGIN	SCG	SCO	SCGO	SCC	SCOU	3JCGI	3JCG	3JCR	3JCGR	3JCL	3JCOU	TPGIN	TPG	TPR	TPGR	TPL	TPLOUT
NR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
HGIN	0	11	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HG	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0
HR	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HGR	0	0	1	3	7	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HL	0	0	0	2	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	0	0	0	0	0	0	9	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
JLGIN	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLG	0	0	0	0	0	0	0	1	7	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLP	0	0	0	0	0	0	0	0	0	10	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLRP	0	0	0	0	0	0	0	0	0	0	7	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
JLS	0	0	0	0	0	0	0	1	0	0	0	7	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLOUT	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQOUT	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKGIN	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
DKG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
DKTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKTR	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	1	0	0	0	4	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
DKTP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	7	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKTOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGTR	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGTP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCGIN	0	2	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCG	0	2	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
SCO	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
SCGO	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	5	3	0	2	0	0	0	0	0	0	0	0	0	0	0	0
SCC	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0
SCOUT	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	3	1	2	2	0	0	0	0	0	0	0	0	0	0	0	0
3JCGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	10	0	1	0	0	0	0	0	0	0	0	0
3JCG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	6	0	2	1	0	0	0	0	0	1	0
3JCR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0
3JCGR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	9	1	0	0	0	0	0	0	0
3JCL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0
3JCOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0
TPGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	1	8	0	0	0	0	0
TPG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	2	9	0	0	0	0
TPR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
TPGR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	8	0	0
TPL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	2
TPLOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	1	0	0	0	0	0	0	2	2	0	2	2

Figure 6.7: Confusion matrix for subject 4, Approach 3 (DTW). Classes listed in the rows are the ground truth for one word instance. As with Approaches 1 and 2, note strong dark colors along the diagonal, indicating correct predictions, and the tendency for incorrect predictions to be clustered within the same action group box.

subj2	NR	HGIN	HG	HR	HGR	HL	HLOU	JLGIN	JLG	JLP	JLRP	JLS	JLOUT	BGIN	BG	BSQ	BSQO	DKGII	DKG	DKTS	DKTR	DKTP	DKTO	KGIN	KG	KGTS	KGTR	KGTP	KGOU	SCGIN	SCG	SCO	SCGO	SCC	SCOU	3JCGI	3JCG	3JCR	3JCGR	3JCL	3JCOU	TPGIN	TPG	TPR	TPGR	TPL	TPLOUT
NR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	3	1
HGIN	0	10	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HG	0	0	11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HR	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HGR	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HL	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	0	0	0	0	0	1	7	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
JLGIN	0	0	0	0	0	0	0	11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLG	0	1	0	0	0	0	0	5	5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLP	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
JLRP	0	0	0	0	0	0	0	0	0	3	7	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
JLS	0	0	0	0	0	0	0	1	0	1	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLOUT	0	0	0	0	0	0	0	0	0	0	0	5	4	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0
BG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
BSQ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQOUT	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
DKG	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	10	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKTS	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0
DKTR	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	7	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0
DKTP	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
DKTOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	9	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
KGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
KGTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGTR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
KGTP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
KGOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
SCGIN	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	7	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	2	0	0	1	0	0	1	0	0	5	0	0	0	0	0	0	0
SCO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0
SCGO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	6	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0
SCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0
SCOUT	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	5	1	0	0	0	0	1	0	0	0	0	0	0	0
3JCGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	1	0	0	0	0	0	0	0
3JCG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	9	0	0	0	1	0	0	0	0	0	0
3JCR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0
3JCGR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	1	0	0	0	0	0	0	0
3JCL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0
3JCOUT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0
TPGIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	2	1
TPG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	0	0	1	1
TPR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
TPGR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	9	0	0
TPL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0
TPLOUT	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	0	1	3

Figure 6.8: Confusion matrix for subject 2, Approach 3 (DTW). Although lower in overall accuracy, subject 2's results show the similar patterns as those for subject 4.

6.4 Approach 4

Since the Random Forest classifier showed the best results reported in [1] and were used in Approach 1, it was used here as well to produce the class probabilities used as the basis for the observation model in the HMM.

For Learning Approach 4, signal instances for each grip family in the test data were presented for an entire 10 second repetition for classification. For example, each of the the repetitions for the hammer activity consisting of {NR,HGIN,HG,HR,HGRP,HL,HLOUT,NR} was presented for classification and belief calculation in the order in which it was performed. The 12 test segments consisted of one of these sequences for each family as listed in Table 4.4. Each family's 10 second segment was classified separately. Maintaining this order allowed the belief logic to use the time sequence nature of the data since it was the order in which the data was collected from the subject.

The 10 second repetitions each included 200 100 ms instances for repetitions $1-11$ and approximately 140 for the last repetition since the last one was truncated to seven seconds by the collection protocol. The first four seconds (80 100 ms instances) of each family's repetitions were not included in training or testing. Those instances were not closely tracked during data collection and included many neutral/rest, which were already over-represented in each subject's data. Approximately 18,700 100 ms instances were therefore collected for each subject. Figure 6.9 shows how data was divided for training and testing.

This approach was implemented in Matlab. The Matlab *Treebagger* class was used for training and the *predict* method used for prediction and to generate the scores. This simple forward filtering method improved the overall results 10-15% for all subjects. What the overall score does not show is the nature of improvements. Most of the errors occur within a family (hammer, key, etc.), at transitions, and between transitions and neutral rest.

Figure 6.10 compares the confusion matrices for subject 4 for Approach 4. Matrices for both the Random Forest classification and subsequent HMM belief calculation are included to show the effect of the belief calculation. Subject 4 had relatively high accuracy. The top is the result of the Random Forest classification which had an overall classification accuracy

Repetition#→	NA	1	2	3	4	5	6	7	8	9	10	11	12
hammer													
jar lid													
ball													
door knob													
key													
scissors													
3-jaw chuck													
tip pinch													
elapsed secs.	0	4	14	24	34	44	54	64	74	84	94	104	114
#seconds/rep.	4	10	10	10	10	10	10	10	10	10	10	10	7
#100ms instances	80	200	200	200	200	200	200	200	200	200	200	200	140

Repetition #5
held out for testing

Figure 6.9: Segmentation of repetition data for Approach 4. The above shows repetition # 5 as held out for testing while the other repetitions were used for training. The instances in the held-out testing repetition for each grip family were presented to the trained classifier as a unified whole with the instances in the order in which the subjects performed the repetition for that family. Each test therefore consisted of presenting 8 different 10 second, 200 instance segments (7 second, 140 instance for repetition # 12) for classification and belief calculation. The 10 second segments were presented for classification in the top-to-bottom order shown in the above table hammer, jar lid, ... tip pinch. Maintaining the order for each family repetition was necessary to consider the time sequence nature of the signal stream. The process was repeated 12 times so that each repetition participated in training 11 times and was held out once for testing.

Table 6.4: Overall accuracies for for Approach 4. The results for the Random Forest classification based on a single instance is shown, followed by the improved accuracy when the belief calculation was made.

Method	**subj1(%)**	**subj2(%)**	**subj3(%)**	**subj4**	**subj5**	**avg.(%)**
RF25	76.40	60.96	61.32	71.82	57.19	65.54
Belief Calc.	84.30	72.50	70.20	80.29	68.14	75.09

of 71.82%. The bottom shows the improved CM after applying the belief calculation and showed an improved accuracy of 80.29%.

Figure 6.11 compares the confusion matrices for subject 2 for Approach 4. Subject 2 had relatively low accuracy. The top is the result of the Random Forest classification of 60.96%. Bottom shows the improved CM after applying the belief calculation resulting in an accuracy of 72.5%.

Figure 6.12 is a plot of the accuracies for all five subjects of the Random Forest classification prior to applying the belief calculation. Results for classification using various numbers of trees from 3 to 100 is shown. Accuracy improves noticeably to around 15 trees then begins to tail and shows no improvement beyond 50.

Figure 6.13 lists the accuracies for all five subjects of the Random Forest classification prior after applying the belief calculation. As before, accuracy improves noticeably to around 15 trees then levels off at 25, and shows no improvement beyond 50.

Figure 6.14 compares the improvement of the belief calculation over RF classification for subjects 1 and 2. Subject 1 had higher accuracies compared with 2 and belief improved accuracy by approximately 8%. Subject 2, by contrast, had lower accuracies for all trees, but belief improved them by approximately 11%.

Table 6.4 summarizes results for all subjects. The RF25 results are shown along side the belief calculation that used the RF25 probability scores as input. Belief improved the overall Random Forest Approach 4 accuracies for each subject by $\approx 8\%$ to 11%. The cross-subject average was improved by $\approx 9\%$ over the Matlab Treebagger classifier.

subj4	NR	HGIN	HG	HR	HGRP	HL	HLOU	JLGIN	JLG	JLP	JLRP	JLS	JLOUT	BGIN	BG	BSQ	BSQO	DKGI	DKG	DKTS	DKTR	DKTP	DKTO	KGIN	KG	KGTS	KGTR	KGTP	KGOU	SCGIN	SCG	SCO	SCGO	SCC	SCOU	3JCGI	3JCG	3JCR	3JCGF	3JCL	3JCOU	TPGIN	TPG	TPR	TPGR	TPL	TPLOUT
NR	5793	2	0	0	4	0	10	1	5	0	48	5	6	6	14	0	15	1	13	0	17	2	9	3	10	0	17	7	2	1	30	15	34	1	19	2	0	1	5	2	15	1	1	0	17	5	17
HGIN	6	28	23	5	5	6	0	0	1	0	0	0	0	1	5	2	0	0	0	0	0	0	0	0	1	0	0	0	0	1	3	0	1	9	2	0	6	0	4	2	0	1	0	0	0	0	0
HG	0	10	524	9	14	2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	21	3	0	2	8	0	0	0	9	0	0	0	0	0	0	0	0
HR	0	4	14	98	20	0	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0
HGRP	0	1	24	16	261	7	0	0	5	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	2	27	0	0	9	12	0	0	0	0	0	0	0	0	0	0	0	0
HL	1	6	4	6	6	91	0	0	0	1	0	3	0	0	0	3	0	0	0	0	0	3	0	0	0	0	0	0	0	1	3	0	0	9	7	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	45	2	3	1	1	2	38	0	0	0	0	1	0	5	0	0	6	0	0	1	2	0	0	0	0	0	0	0	0	2	14	1	14	1	12	0	1	0	3	0	2	0	0	0	1	1	0
JLGIN	6	0	0	0	2	0	1	28	20	3	1	1	0	0	0	2	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	2	1	8	1	1	0	0	0	1	0	0	0	2	1	0
JLG	5	0	0	0	1	0	0	18	480	14	0	1	3	2	3	0	0	0	32	0	0	0	0	0	0	0	0	0	0	0	51	1	6	0	2	0	0	0	8	0	0	0	4	0	2	0	0
JLP	3	0	0	0	1	0	0	0	8	82	20	0	3	1	0	8	0	1	2	2	0	16	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	1	0
JLRP	118	0	0	0	0	0	0	0	0	12	137	6	6	3	2	0	0	2	5	0	3	0	4	0	0	1	1	1	0	0	1	0	3	0	0	0	1	0	0	0	1	2	1	0	3	10	4
JLS	5	0	0	0	0	0	1	2	4	3	2	125	5	0	0	0	3	1	3	1	0	7	0	0	0	0	0	0	0	0	3	2	6	3	11	0	0	0	0	0	1	0	2	0	1	0	1
JLOUT	60	0	0	0	0	0	1	1	0	6	21	9	3	0	0	0	0	0	3	1	3	0	1	0	1	1	7	0	0	0	0	0	4	0	1	0	1	0	0	0	0	1	0	0	4	2	0
BGIN	26	0	0	0	1	1	3	0	1	2	1	0	0	69	47	0	0	0	1	0	0	1	0	0	0	1	1	0	0	0	4	0	0	0	2	0	1	1	4	0	1	0	0	0	1	0	0
BG	13	1	0	0	1	0	0	1	6	0	0	0	0	25	462	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	4	0	0	0	0	0	2	0	0
BSQ	1	0	1	5	1	9	0	4	1	2	0	3	0	0	3	563	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQOUT	156	0	0	0	0	2	11	0	1	1	2	7	0	0	0	1	24	0	0	6	4	0	1	0	0	0	0	0	0	0	0	2	17	0	4	0	0	0	0	0	0	0	0	1	1	1	0
DKGIN	13	0	0	0	1	1	0	1	10	3	1	1	0	2	0	1	0	12	41	0	0	5	0	0	2	0	0	0	0	0	0	0	1	0	1	1	1	0	0	1	1	0	3	0	0	1	0
DKG	30	0	0	0	0	0	0	0	34	0	1	3	0	0	0	0	1	16	411	5	2	1	0	1	4	1	1	0	0	0	6	1	2	0	3	5	26	0	0	4	1	0	14	0	1	7	1
DKTS	1	0	0	1	0	1	4	0	0	0	0	0	0	0	0	0	3	0	2	177	8	0	0	0	0	0	0	0	0	0	1	1	2	1	0	0	1	0	0	0	0	0	3	6	0	0	0
DKTR	164	0	0	0	1	0	0	1	2	0	4	1	5	1	3	0	2	2	8	16	71	4	10	0	0	1	3	2	0	0	0	0	1	0	5	0	4	0	0	1	3	2	0	2	2	0	3
DKTP	11	1	0	0	1	0	1	1	6	5	0	10	0	0	1	1	1	3	15	0	1	200	5	0	1	0	0	0	0	0	1	0	0	3	3	4	2	0	1	2	0	1	1	0	1	0	0
DKTOUT	69	0	0	0	0	0	0	0	0	2	14	0	0	0	3	0	0	0	1	0	9	15	27	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	10	0	0	7	1
KGIN	24	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	18	36	1	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	2	4	0	0	4	1
KG	5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	4	0	0	0	0	14	488	7	5	7	0	0	0	0	0	0	0	0	5	0	0	0	0	2	10	0	2	16	12
KGTS	2	0	0	0	0	0	0	0	0	0	2	0	1	2	1	0	0	0	0	0	0	0	0	0	6	94	17	1	0	0	0	0	0	0	0	0	2	1	1	0	0	1	1	4	11	1	0
KGTR	124	0	0	0	0	0	0	0	0	0	6	0	3	1	2	0	0	0	0	0	1	0	0	0	8	7	168	5	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	9	5	0
KGTP	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	13	0	10	94	5	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	2	6
KGOUT	66	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	2	10	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5
SCGIN	26	1	2	2	12	0	1	1	6	0	2	1	1	0	2	0	0	0	3	0	0	0	0	0	0	1	2	0	0	4	18	0	5	3	17	2	1	0	2	1	0	0	0	1	10	1	0
SCG	79	2	15	0	30	2	3	0	35	0	1	1	1	3	1	0	0	0	3	0	0	0	0	0	0	0	0	0	0	6	245	2	31	3	39	1	2	0	5	1	0	0	0	0	5	0	0
SCO	4	0	3	0	0	0	2	0	2	0	0	14	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	2	7	55	35	1	14	0	0	0	0	0	0	1	5	0	0	0	0
SCGO	57	0	0	0	3	0	10	0	4	0	0	3	0	0	0	0	5	0	4	1	0	0	0	0	0	0	0	0	0	2	56	33	202	3	19	0	6	0	0	0	0	0	2	0	9	0	0
SCC	2	11	14	2	8	6	0	1	0	0	0	4	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	0	1	8	8	3	90	25	0	0	0	0	0	0	0	0	0	0	0	0
SCOUT	24	1	15	0	10	2	9	1	8	0	2	11	0	3	1	2	3	1	0	2	1	0	0	0	0	0	0	0	0	6	46	20	37	32	125	2	2	0	0	0	0	0	0	0	1	0	0
3JCGIN	37	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	13	1	1	1	0	0	1	0	2	0	0	0	4	0	2	0	3	37	31	1	3	3	5	0	1	0	3	1	0
3JCG	0	2	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	25	0	1	1	0	0	7	5	1	0	0	0	9	0	1	0	0	28	390	9	14	27	0	0	0	2	2	3	0
3JCR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	3	142	14	1	0	0	0	5	1	0	0
3JCGR	11	0	0	0	0	0	0	0	5	0	0	0	0	1	3	0	0	0	2	0	0	0	0	0	0	2	0	0	0	0	2	0	0	0	0	1	16	15	260	24	0	0	0	0	1	0	0
3JCL	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	1	2	1	0	0	0	0	0	0	0	2	24	3	26	116	4	1	0	0	0	0	0
3JCOUT	93	0	0	0	0	0	1	1	0	0	1	1	1	0	0	0	0	1	0	0	2	0	0	0	0	0	6	1	0	0	1	0	0	0	0	4	2	0	0	4	26	1	0	0	0	0	2
TPGIN	36	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	3	0	0	3	2	1	13	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	18	50	0	0	9	5
TPG	0	0	0	0	0	0	0	0	0	6	3	7	0	0	0	0	0	2	13	0	0	0	3	0	7	1	0	0	0	0	0	8	13	0	0	0	0	0	0	0	0	17	483	3	3	6	0
TPR	1	0	0	0	0	0	0	0	2	3	0	0	0	1	0	0	0	0	1	5	1	0	0	0	0	5	0	0	0	0	1	0	1	0	0	0	1	1	0	0	0	0	5	119	16	0	0
TPGR	9	0	0	0	0	0	0	0	1	0	1	0	3	1	0	0	0	0	2	0	0	0	0	0	1	4	4	0	0	6	2	0	5	0	0	0	2	0	1	0	0	0	2	13	337	10	1
TPL	20	0	0	0	0	0	1	0	0	2	6	0	0	0	1	0	0	0	3	0	1	0	5	1	21	1	10	5	0	0	0	0	1	0	0	0	1	0	0	1	0	1	2	0	20	61	5
TPLOUT	64	0	0	0	0	0	0	0	0	0	4	0	2	0	0	0	0	0	2	0	9	0	2	0	10	0	1	10	3	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	2	7	28

subj4	NR	HGIN	HG	HR	HGRP	HL	HLOU	JLGIN	JLG	JLP	JLRP	JLS	JLOUT	BGIN	BG	BSQ	BSQO	DKGI	DKG	DKTS	DKTR	DKTP	DKTO	KGIN	KG	KGTS	KGTR	KGTP	KGOU	SCGIN	SCG	SCO	SCGO	SCC	SCOU	3JCGI	3JCG	3JCR	3JCGF	3JCL	3JCOU	TPGIN	TPG	TPR	TPGR	TPL	TPLOUT
NR	5825	1	2	0	0	0	14	1	0	0	29	2	10	8	4	0	3	2	6	0	0	0	20	0	6	0	6	0	14	1	24	17	48	0	31	1	2	0	0	0	69	2	0	0	1	0	7
HGIN	8	31	33	3	0	0	0	2	0	0	0	0	0	3	7	2	1	1	0	0	0	0	0	0	1	0	0	0	0	2	6	0	0	0	0	4	6	1	0	0	1	0	0	0	0	0	0
HG	0	0	554	12	13	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	5	0	0	0	7	1	1	0	0	0	0	0	0
HR	0	0	9	109	23	0	0	0	0	0	0	0	0	0	0	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HGRP	0	0	27	18	310	11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HL	1	0	4	5	11	119	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	40	0	6	2	2	6	103	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLGIN	14	0	0	0	0	0	0	21	26	5	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	6	4	0	0	0	4	1	0	0	0	0	1	0	0	0	0	0	0
JLG	0	0	0	0	0	0	0	5	552	17	1	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0
JLP	0	0	0	0	0	0	0	0	10	124	25	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
JLRP	80	0	0	0	0	0	0	0	0	10	222	8	1	1	1	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
JLS	8	0	0	0	0	0	0	0	6	2	2	126	11	3	0	0	0	0	2	0	0	0	0	2	0	0	0	0	0	1	2	9	2	2	13	0	0	0	0	0	0	0	1	0	0	0	0
JLOUT	47	0	0	0	0	0	0	0	1	5	31	13	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	1	0	0	0	0	0	0	0	0	0	4	0	0
BGIN	30	0	0	0	0	0	0	0	0	0	0	0	0	80	58	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BG	2	0	0	0	0	0	0	0	0	0	0	0	0	0	507	7	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	587	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQOUT	143	0	0	0	0	0	0	0	1	0	0	0	0	0	0	4	82	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKGIN	17	0	0	0	0	0	0	0	2	0	0	0	0	4	0	0	0	21	50	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	2	4	0	0	0	1	0	1	0	0	0	0
DKG	26	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	510	7	0	0	4	0	2	0	0	0	0	0	0	0	0	0	0	1	27	0	0	0	3	0	1	0	0	0	0
DKTS	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	3	199	5	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0
DKTR	70	1	0	0	0	0	0	0	0	0	0	0	0	3	2	0	1	2	8	21	189	6	11	1	0	0	0	0	0	1	0	0	0	0	0	5	3	0	0	0	0	0	0	0	0	0	0
DKTP	5	1	0	0	0	0	0	0	1	6	0	0	0	0	0	0	0	3	26	1	12	207	11	0	1	0	0	0	0	0	0	0	0	0	4	1	4	0	0	0	0	0	0	0	0	0	0
DKTOUT	70	0	0	0	0	0	0	0	0	0	0	0	0	1	3	0	2	0	0	0	0	19	72	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGIN	33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	17	40	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	3	0	0	0	0
KG	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	537	16	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	11
KGTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	121	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0
KGTR	91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	16	224	6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGTP	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	113	19	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0
KGOUT	41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	45	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCGIN	37	2	0	0	3	0	0	1	2	0	0	0	0	1	3	0	2	2	0	0	0	0	0	0	0	0	0	0	0	12	24	3	0	0	31	1	0	0	0	0	2	0	0	0	1	1	0
SCG	89	0	4	0	12	2	3	0	15	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	288	3	15	0	81	0	0	0	0	0	0	0	0	0	0	0	0
SCO	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	10	55	42	0	35	0	0	0	0	0	0	0	0	0	0	0	0
SCGO	42	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	58	28	270	4	14	0	0	0	0	0	0	0	0	0	0	0	0
SCC	1	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	8	0	87	85	0	0	0	0	0	0	0	0	0	0	0	0
SCOUT	22	0	7	3	2	0	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	36	3	23	18	249	0	0	0	0	0	0	0	0	0	0	0	0
3JCGIN	45	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	37	53	2	0	0	12	0	0	0	0	0	0
3JCG	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	9	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	25	465	13	3	10	1	0	0	0	0	0	0
3JCR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	150	13	0	0	0	0	2	0	0	0
3JCGR	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	15	292	25	3	0	0	0	0	0	0
3JCL	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	3	25	127	12	0	0	0	0	0	0
3JCOUT	50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	2	92	0	0	0	0	0	0
TPGIN	36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	18	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	27	59	0	0	0	1
TPG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	1	2	0	0	0	0	0	0	0	0	14	17	0	0	0	0	0	0	0	0	0	533	3	1	0	0
TPR	0	0	0	0	0	0	0	0	0	1	0	2	2	0	0	0	0	0	0	7	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	128	14	0	0
TPGR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	372	13	0
TPL	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	25	105	24
TPLOUT	57	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	1	6	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	4	76

Figure 6.10: Comparison of the confusion matrices for subject 4's results in Approach 4. Top is the Random Forest CM, bottom shows improvement after belief logic is applied.

subj2	NR	HGIN	HG	HR	HGRP	HL	HLOUT	JLGIN	JLG	JLP	JLRP	JLS	JLOUT	BGIN	BG	BSQ	BSQOUT	DKGIN	DKG	DKTS	DKTR	DKTP	DKTOUT	KGIN	KG	KGTS	KGTR	KGTP	KGOUT	SCGIN	SCG	SCO	SCGO	SCC	SCOUT	3JCGIN	3JCG	3JCR	3JCGR	3JCL	3JCOUT	TPGIN	TPG	TPR	TPGR	TPL	TPLOUT
NR	4339	8	2	0	1	0	8	3	1	1	7	2	14	28	87	0	16	3	18	4	26	0	20	3	96	2	38	3	22	8	73	1	2	9	31	10	77	7	16	10	35	7	117	14	57	8	17
HGIN	37	30	40	2	1	1	5	0	7	2	5	3	0	2	1	0	0	1	12	5	2	2	0	0	0	0	0	0	0	4	0	0	0	0	1	2	2	0	12	0	0	0	0	0	0	0	0
HG	2	15	454	8	12	2	2	0	5	4	4	0	0	0	0	0	2	0	1	0	0	0	0	0	0	0	0	0	0	6	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
HR	0	0	4	101	35	11	1	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0
HGRP	1	2	17	24	317	8	0	1	7	1	8	0	0	0	0	0	0	0	5	8	0	0	0	0	0	0	0	0	0	0	2	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
HL	0	1	0	6	12	104	6	1	0	0	0	0	0	0	0	5	0	0	3	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	73	9	13	0	6	7	45	0	0	0	3	0	0	7	0	0	0	0	2	6	17	0	0	2	4	0	0	0	0	5	13	0	0	0	6	2	2	0	1	3	1	0	0	0	0	0	0
JLGIN	15	2	0	0	0	0	0	2	54	0	5	2	2	0	0	0	0	1	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	2
JLG	2	0	1	0	0	0	0	20	462	16	23	8	3	1	0	0	2	0	42	10	1	0	0	0	0	0	0	0	0	1	2	0	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0
JLP	0	0	1	1	0	0	0	0	5	133	44	7	1	0	3	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	8	1	0	0	0	0	0	0	0	0	0	0	0	0
JLRP	7	4	2	0	1	0	1	0	21	45	219	3	3	0	1	0	0	0	6	6	0	0	0	0	0	0	0	0	0	5	3	0	4	2	4	0	0	0	0	0	0	0	0	0	0	0	0
JLS	2	2	0	0	0	0	0	3	7	19	3	45	49	0	0	0	1	1	15	3	0	0	0	0	0	1	0	0	0	0	2	0	4	1	0	1	0	0	0	0	0	0	0	0	0	0	0
JLOUT	68	0	0	0	0	0	2	0	0	3	1	38	128	3	1	0	1	1	13	1	6	0	11	0	0	0	0	0	0	0	2	0	1	0	0	0	0	0	1	2	2	0	0	1	1	0	0
BGIN	95	0	0	0	0	0	3	0	0	0	0	0	2	55	33	1	8	0	0	0	22	0	3	0	1	3	2	1	3	1	7	1	1	0	6	1	2	0	0	2	10	0	0	1	0	1	2
BG	240	0	1	2	5	1	0	0	0	2	0	0	0	15	238	2	4	0	1	0	12	0	0	0	3	1	7	2	1	0	5	0	1	0	1	0	14	2	1	3	3	0	5	0	0	1	1
BSQ	0	0	0	2	2	7	0	0	0	1	0	0	0	0	0	478	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	5	0	0	0	0	0	0	0	0	0	0	0	0
BSQOUT	117	0	2	1	1	1	2	0	0	1	6	1	3	6	8	2	55	0	4	5	12	7	0	0	4	2	1	6	1	1	10	3	11	0	10	1	2	0	1	0	1	0	1	0	2	0	0
DKGIN	29	1	0	0	0	0	1	0	0	1	1	5	2	0	0	0	0	29	23	0	4	7	4	1	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	2	0	0	3	0	1
DKG	25	2	1	0	0	0	0	2	35	0	4	18	25	0	0	0	2	6	448	33	1	6	1	0	1	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0
DKTS	1	1	0	1	7	1	0	1	12	2	11	4	3	0	0	0	1	2	18	232	2	0	0	1	0	0	0	0	0	0	4	0	0	1	7	0	0	0	0	0	0	1	0	0	0	0	0
DKTR	125	0	0	0	0	0	6	0	4	0	0	0	4	17	3	0	11	5	10	13	91	2	4	0	6	1	1	1	0	0	1	2	0	0	1	4	0	0	0	0	1	2	0	0	1	2	3
DKTP	4	2	0	0	0	0	0	2	2	0	1	2	3	4	0	0	4	4	12	6	6	204	13	0	0	0	0	0	0	0	1	2	0	3	0	0	0	0	0	0	0	0	0	0	0	0	1
DKTOUT	52	0	0	0	0	0	1	0	0	0	3	0	4	6	0	1	2	3	2	0	6	23	47	0	0	1	0	1	0	0	0	1	0	0	0	0	2	3	0	0	0	0	0	0	1	0	0
KGIN	40	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	42	1	1	1	2	0	0	0	0	0	0	2	1	0	0	0	0	0	4	0	7	0	1
KG	215	0	0	0	0	0	1	0	0	0	0	0	0	2	0	0	1	0	0	1	1	0	0	7	280	11	7	6	14	0	0	0	0	0	0	0	6	0	0	0	0	2	33	1	22	2	1
KGTS	10	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	2	0	0	1	0	0	1	0	5	90	32	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	3	1	0	0
KGTR	60	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	5	25	255	7	1	1	4	0	0	0	0	0	0	0	0	0	0	0	0	2	1	3	0
KGTP	12	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	2	0	0	0	2	1	1	0	6	8	16	85	17	0	4	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
KGOUT	113	0	0	0	0	0	0	0	0	0	0	0	0	2	6	0	0	0	0	0	0	0	0	0	25	0	0	12	49	0	0	0	0	0	0	0	1	0	0	0	0	0	4	0	9	0	0
SCGIN	48	2	4	0	3	0	3	0	0	1	8	0	0	1	2	0	4	0	0	1	5	0	0	0	0	2	0	0	0	42	52	1	3	1	20	0	1	2	1	6	4	0	0	2	2	0	0
SCG	53	0	1	0	3	0	4	0	0	0	8	1	1	3	11	0	1	0	1	3	1	0	0	0	2	1	3	0	0	29	216	8	58	1	24	2	43	0	2	2	3	0	0	1	0	0	0
SCO	19	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1	0	0	1	6	0	0	0	0	0	0	0	0	3	4	72	30	13	12	0	1	0	1	0	0	0	0	1	0	0	0
SCGO	4	1	0	0	0	0	0	0	0	1	2	6	5	0	0	0	5	0	0	2	0	0	0	0	0	1	0	0	0	1	29	29	228	10	35	0	0	0	0	0	0	0	0	1	0	0	0
SCC	5	0	0	0	0	0	1	0	0	12	7	0	0	0	0	0	0	0	0	2	0	4	1	0	0	0	0	0	0	1	5	10	4	89	35	0	0	0	0	0	1	0	0	0	0	0	0
SCOUT	32	0	0	0	1	2	0	0	1	7	8	0	1	3	0	0	10	0	0	7	2	3	0	0	0	1	0	0	0	11	28	10	28	30	140	0	3	2	0	3	4	0	0	0	0	0	0
3JCGIN	76	1	5	0	0	0	1	1	1	1	3	1	1	1	0	0	0	0	1	1	6	0	0	1	7	0	1	0	0	0	11	0	0	0	0	7	20	0	0	2	5	3	0	0	2	0	1
3JCG	123	0	0	0	0	0	0	1	0	0	2	0	0	2	16	0	0	0	1	0	0	0	0	0	10	0	2	0	0	0	16	0	0	0	0	3	364	8	1	4	25	0	1	0	0	0	0
3JCR	10	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	8	113	26	4	1	0	0	2	0	0	0
3JCGR	58	3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	2	0	0	0	0	3	0	0	0	0	0	0	1	18	323	12	0	0	0	0	2	0	0
3JCL	19	2	0	0	0	0	0	0	0	0	0	0	2	3	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	7	0	0	0	5	1	12	5	20	44	28	0	0	0	1	0	0
3JCOUT	44	0	0	0	0	0	0	0	0	0	2	0	0	6	5	0	8	0	0	0	3	0	0	0	0	0	1	0	0	1	7	0	1	1	3	1	35	0	0	7	123	0	0	0	0	0	0
TPGIN	95	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1	2	2	0	2	0	1	0	9	1	6	1	1	0	1	0	0	0	0	1	0	0	0	0	0	12	14	0	14	0	2
TPG	362	0	0	0	0	0	0	0	0	0	0	0	0	1	10	0	0	0	0	0	0	0	0	2	48	0	3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	2	114	3	24	0	2
TPR	31	0	0	0	0	0	0	0	0	0	0	0	2	3	1	0	0	0	0	0	2	0	3	0	0	4	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	97	6	0	0
TPGR	188	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	0	0	0	2	1	33	1	3	1	5	1	2	0	0	0	0	0	0	2	0	0	0	0	17	6	163	0	1
TPL	92	0	0	0	0	0	0	0	0	0	0	0	0	6	4	0	1	0	0	0	2	0	0	0	14	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	1	4	3
TPLOUT	101	1	0	0	0	0	0	1	0	0	0	0	0	2	1	0	1	0	1	1	5	0	0	1	15	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	3	1	9	0	9	2	5

subj2	NR	HGIN	HG	HR	HGRP	HL	HLOUT	JLGIN	JLG	JLP	JLRP	JLS	JLOUT	BGIN	BG	BSQ	BSQOUT	DKGIN	DKG	DKTS	DKTR	DKTP	DKTOUT	KGIN	KG	KGTS	KGTR	KGTP	KGOUT	SCGIN	SCG	SCO	SCGO	SCC	SCOUT	3JCGIN	3JCG	3JCR	3JCGR	3JCL	3JCOUT	TPGIN	TPG	TPR	TPGR	TPL	TPLOUT
NR	4289	7	0	0	0	0	3	0	0	0	0	0	3	57	127	0	3	6	17	2	0	0	8	1	48	1	8	0	54	2	75	1	0	10	42	4	56	14	1	0	48	16	278	19	23	7	21
HGIN	34	56	54	3	0	0	1	1	1	1	0	0	0	4	3	0	1	0	9	6	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
HG	0	1	489	8	1	0	7	0	3	0	2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	3	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HR	0	0	4	112	31	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HGRP	0	0	2	17	367	13	2	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HL	0	0	0	5	14	123	5	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HLOUT	81	0	6	0	1	8	90	0	0	0	0	0	0	7	4	0	0	0	3	2	0	0	3	2	5	0	0	0	0	6	7	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0
JLGIN	19	0	0	0	0	0	0	13	52	0	0	0	1	0	0	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	2
JLG	1	0	0	0	0	0	0	1	525	19	18	4	5	0	0	0	0	0	23	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLP	0	0	0	0	0	0	0	0	8	155	43	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLRP	0	0	0	0	0	0	0	0	19	48	262	3	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLS	0	0	0	0	0	0	0	0	11	20	4	17	102	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
JLOUT	70	0	0	0	0	0	0	0	0	2	0	1	206	0	0	0	0	0	6	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BGIN	37	0	0	0	0	0	0	0	0	0	0	0	0	118	87	2	8	0	0	0	0	0	0	0	1	0	0	0	0	0	6	0	0	0	1	0	2	0	0	0	5	0	0	0	0	0	0
BG	116	0	0	0	0	0	0	0	0	0	0	0	0	4	412	2	19	0	0	0	0	0	0	0	0	1	2	0	0	0	1	0	0	0	0	0	7	3	0	0	0	0	7	0	0	0	0
BSQ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	486	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BSQOUT	119	1	4	0	0	0	0	3	3	0	0	0	0	0	4	6	136	0	2	1	2	0	0	0	0	0	0	0	0	0	3	0	1	0	2	0	3	0	0	0	0	0	1	0	0	0	0
DKGIN	28	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	38	28	0	0	2	0	2	4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	5	1	5	1	0	0
DKG	13	3	0	0	0	0	0	0	0	0	0	0	22	0	0	0	0	4	527	40	0	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DKTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30	270	4	3	1	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0
DKTR	90	0	0	0	0	0	0	0	0	0	0	0	0	18	0	0	0	2	14	14	170	3	5	1	1	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	1	0	0	0	0	0
DKTP	2	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	4	1	18	7	9	200	28	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0
DKTOUT	47	0	0	0	0	0	0	0	0	0	0	0	0	8	1	2	0	0	2	0	0	28	68	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0
KGIN	31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	53	0	0	0	3	0	0	0	0	0	0	0	1	0	0	0	0	0	6	0	4	2	4
KG	73	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	428	15	1	5	67	0	0	0	0	0	0	0	0	0	0	0	0	2	18	0	4	0	0
KGTS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	100	45	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGTR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	20	331	9	4	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGTP	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	14	108	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KGOUT	70	0	0	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0	0	11	2	1	1	133	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SCGIN	33	1	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	48	90	1	0	0	39	1	0	0	0	0	0	0	0	6	0	0	0
SCG	36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	298	6	59	1	41	0	32	3	0	1	4	0	0	0	0	0	0
SCO	15	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	73	36	6	20	0	0	0	0	0	0	0	0	8	1	0	0
SCGO	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	4	13	296	10	27	0	0	0	0	0	0	2	0	2	0	0	0
SCC	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	5	8	85	71	0	0	0	0	0	0	0	0	0	0	0	0
SCOUT	36	3	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	13	4	20	23	223	0	1	0	0	0	0	0	0	0	0	0	0
3JCGIN	79	2	3	0	0	0	0	1	2	2	2	0	0	4	0	0	1	0	1	0	0	0	0	0	6	0	0	0	0	2	6	0	0	0	0	5	25	1	0	0	15	2	0	0	0	0	1
3JCG	48	0	2	0	0	0	2	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	11	0	0	0	2	0	3	0	0	0	0	2	467	21	0	0	17	0	0	0	0	0	0
3JCR	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	114	32	1	0	0	0	5	0	0	0
3JCGR	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	11	375	16	0	0	0	0	0	0	0
3JCL	16	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	6	29	53	36	0	0	0	0	0	0
3JCOUT	43	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	18	2	0	0	179	0	0	0	0	0	0
TPGIN	49	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	1	3	0	0	0	0	0	8	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	34	41	0	12	4	8
TPG	132	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	37	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	356	6	18	5	10
TPR	8	0	0	0	0	0	0	0	0	0	0	0	0	1	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	123	13	0	0
TPGR	68	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	41	26	264	21	5
TPL	83	0	0	0	0	0	0	0	0	0	0	0	0	4	3	0	0	0	0	0	0	0	0	0	6	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	2	4	4	15
TPLOUT	93	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	1	1	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	2	18	0	7	0	17

Figure 6.11: Comparison of the confusion matrices for subject 2's results in Approach 4. Top is the Random Forest CM, bottom shows improvement after belief logic is applied.

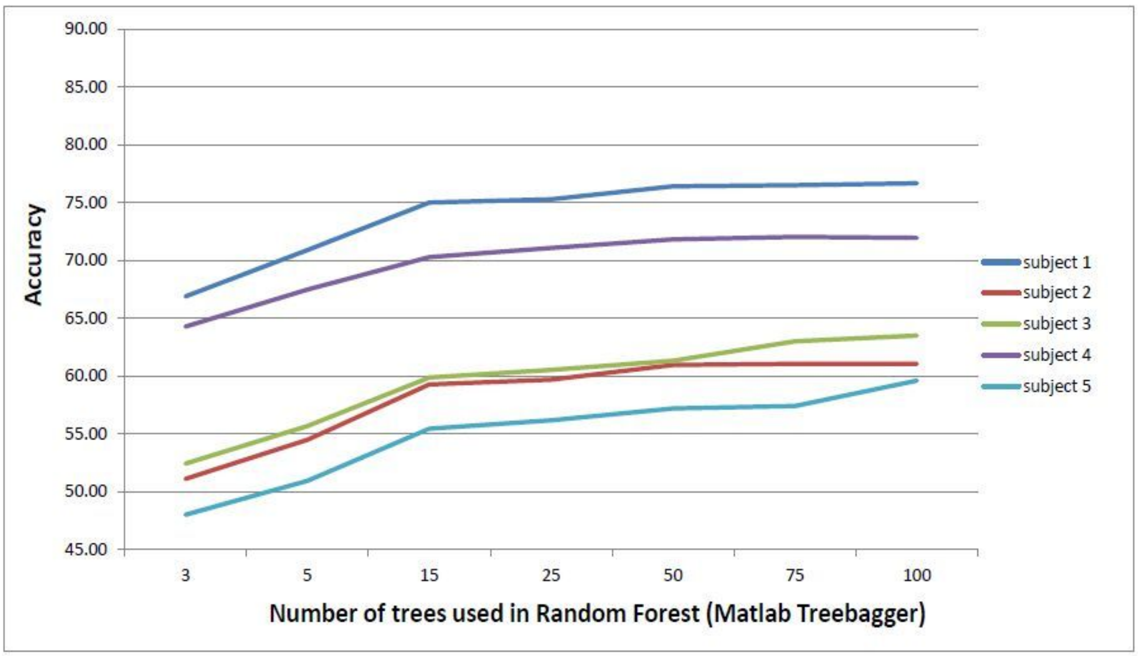


Figure 6.12: Comparison of the results of Random Forest classification for all five subjects using the Matlab *Treebagger* class and *predict* method for various numbers of randomly generated trees. Accuracy improves noticeably up to 15 trees then levels off around 25 and shows no improvement beyond 50. A separate evaluation was done on the number of trees in the Approach 4 Random Forest since the method is different from the Approach 1 Weka implementation. In both cases, however, 25 trees was found to be optimum.

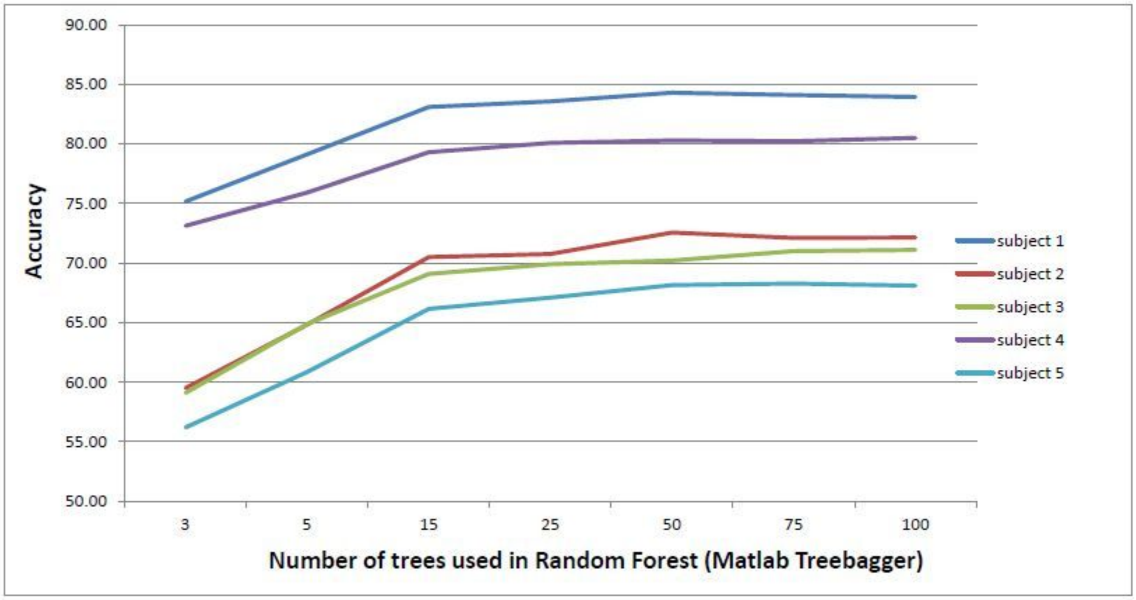


Figure 6.13: Comparison of the results of Random Forest classification after applying the belief calculation for all five subjects using the Matlab Treebagger class for various numbers of randomly generated trees. Accuracy improved noticeably up to 15 trees then begins to tail and shows no improvement beyond 50.

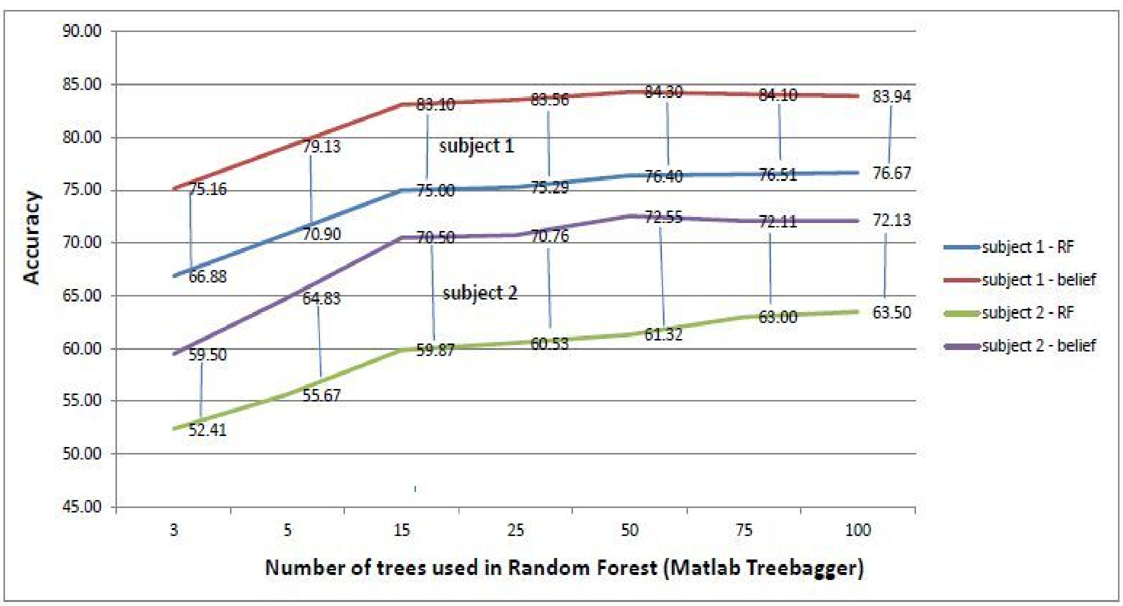


Figure 6.14: Comparison of the results of the improvement the belief calculation brought to classification accuracy for subjects 1 (relatively high accuracy) and 2 (relatively low). Improvement was approximately 8% for subject 1 and 10% for 2.

Chapter 7: Discussion and Conclusions

Representing the grips and movements in Table 4.3 - prehensile patterns - effectively, required accepting the challenge of having to select both electrical signal data from multiple extrinsic muscles of the hand as well as position data obtainable from the accelerometer. Not all relevant signals contributing to prehensile activity were captured from a few surface electrodes - deeper muscles may not be adequately represented. An appropriate set of functional prehensile patterns was selected that was likely to be useful in informing clinicians about which muscles and hand/wrist movements are associated with these activities. With further improvements they can form the basis for applications involving myoelectric control of hand grips and movements.

7.1 Summary of Results for the Four Approaches

The choice of which methods to use to reduce the captured data and recognize the patterns was empirical and approached from the perspective of trying to identify the best fit. An MAV window of 100 ms was used. In the first approach, standard classifiers (Decision Tree, Nearest Neighbor, Support Vector Machine, and Random Forest) were trained to recognize individual 100 ms instances without considering any other information. The best performer, Random Forest (25 trees), yielded an average accuracy of 71.28% for 47 classes. While this is comparable to results obtained elsewhere for similar problems, it is less than what would be needed to be useful in real life scenarios. Besides having a too-low accuracy, a difficulty arises when an activity consisting of multiple 100 ms signal instances results in an inconsistent stream of predicted movements. How would a stream of {HR, HR, HL, HR, JLS, HR, JLS} be interpreted? Smoothing the result stream, as reported in [34] may help, but does not erase all ambiguity.

The second approach, Affinity, attempted to improve classification accuracy in two ways. The first used the SAX concept to discretize the real-valued signals into a set of symbols. This had the effect of normalizing the data while reducing the total signal space to a finite number of symbol combinations. It also allows for an easier visual interpretation of signal values. For example, it is plain to see that for a five symbol alphabet 'EEEEEEEE' represents a set of high signal values compared with 'AAAAAAAA'.

The second improvement injected time-context into the process of classifying a signal by considering immediately preceding values in the stream. Affinity values for a single 100 ms instance can reflect a strong probability for a specific class, in some cases with a 90%+ probability for one class and low or zero values for the rest. In other cases, the affinity can be spread across ten or more classes with a strong preference for none and the selected class's affinity being below 20%. In those instances, the case for selecting the class with the highest affinity is weak. The class decision was improved by calculating the sum of the class values for the current with some number of previous instances to better reflect the "sense of the neighorhood" in terms of identifying the true class. Experimenting with various combinations of SAX symbols and numbers of words in the affinity summation led to an improved average accuracy of 76.72% for 11 symbols and 30 words. To achieve this accuracy, 30 words or 1.5 seconds of signal data needed to be read and processed before a class decision was made, thus delaying the decision.

While the Affinity approach improves prediction accuracy using information from adjacent 100 ms instances, the third approach considered the entire movement. This approach also relied on converting the signal values to SAX symbols prior to classifying. The approach segmented the data stream into the 47 activities listed in Table 4.3. It also employed the Dynamic Time Warping concept to account for small time shifts in the signals and allow for a relaxed and more realistic comparison. In this approach the number of SAX symbols was varied and the resultant accuracy noted for various numbers of words used in the classification. Surprisingly, the number of SAX symbols had little impact on accuracy, but increasing the number of words improved accuracy before levelling off between 20 and 30.

At that level the average accuracy was 73.01%. As with Affinity, achieving this accuracy imposed a processing delay - in this case of 30 words, or 1.5 seconds.

Approach 4 is similar to Approach 1 in that it processed the 100 ms MAV instances of the signal stream and did not convert them to SAX symbols. In this approach the instances were classified using an HMM employing a Random Forest classifier to compute a class probability for each instance based on the number of trees that identified the class as the true one. This approach then performed a post-processing step by calculating a 'belief' value for the possible classes based on the previous instance's probabilities, class-to-class transition probabilities, and the probabilities of the current instance's prediction conditioned by a confidence factor. Like Approach 1, 4 processed the 100 ms instances as they were encountered and did not require segmentation of labelled activities for classification, as in Approaches 2 and 3.

Approach 4 classifier training followed a similar process as Approach 1, using individual 100 ms windows as training instances. In post-processing, however, continuous movement 10-second, 200-instance segments were presented for classification. This post-processing belief calculation took advantage of the time-ordering within the segments by considering adjacent instances in making class decisions. Accuracy of the belief calculation tests ranged from 68.14% to 84.3%, an improvement of $\approx$ 8% to 11% for the various subjects. The average for the five subjects was 75.09%.

In summary, Approach 2 (Affinity) improved on the classifiers used in Approach 1, and Approach 3 (DTW) improved on 1 but not 2. Including context information from the signal stream helps, as does considering entire movement sequences. This is not surprising since the goal is to recognize complete movements, and not small slivers of movements. Incorporating a wider swath of data improves this recognition. Approach 4 accuracies were superior to those from Approaches 1 and 3, but are slightly lower than the results for the best parameter combinations in Approach 2. However, Approach 4 involved whole activity segments, not just segments as was the case with Approaches 2 and 3.

Figure 7.1 lists the comparative accuracy and standard deviation for all subjects and

Appr#	Method (parms)	subj1		subj2		subj3		subj4		subj5		average	
		Accuracy	Stdev	Accuracy	Stdev	Accuracy	Stdev	Accuracy	Stdev	Accuracy	Stdev	Accuracy	Stdev
1	RF (25)	82.69	3.60	66.52	2.90	67.00	4.70	76.49	5.59	63.70	4.90	71.28	4.34
2	Aff (11/10)	81.05	9.90	68.28	8.81	63.42	7.75	73.55	13.59	57.31	10.10	68.72	10.03
2	Aff (11/20)	84.38	8.93	75.52	7.86	68.82	7.59	77.69	12.88	65.58	9.63	74.40	9.38
2	Aff (11/30)	85.03	8.00	78.07	7.53	72.02	7.18	79.15	12.52	69.32	8.35	76.72	8.72
3	DTW (11/10)	73.76	6.53	54.61	10.16	58.69	8.17	60.99	7.72	58.69	6.99	61.35	7.91
3	DTW (11/20)	83.87	7.31	64.54	5.62	64.01	7.60	70.57	5.29	64.01	6.75	69.40	6.51
3	DTW (11/30)	85.46	6.56	68.26	6.61	67.91	8.71	75.53	7.32	67.91	8.29	73.01	7.50
4	HMM	84.30	4.20	72.50	3.10	70.20	4.10	80.29	5.49	68.14	5.10	75.09	4.40

Figure 7.1: Overall accuracy and standard deviation for the four approaches, all five subjects, and average of the five. Separate results are shown for three parameter combinations for Approaches 2 and 3. In those cases the numbers in parentheses are the number of SAX symbols generated from the signal stream and the number of words used in the approach calculations.

approaches. The accuracy is the average of all 12 cross-fold tests for each subject and approach, and the standard deviation was calculated from the accuracy of those 12 tests for each subject-approach combination. For Approaches 2 and 3, multiple results are shown for the various combinations of SAX symbols and signal words used. The Affinity Approach (2) is nearly equivalent to Approach 1 for each subject and the average when 11 symbols and 15 words are employed, and superior at 20+ words. The DTW Approach (3) is nearly equivalent to Approach 1 at 20 and 30 words, but inferior below that level. It requires 30 words to reach equivalence for the Affinity Approach at the 20 word level. The results for the HMM approach (4) are nearly equivalent to the best in Approach 2, and superior to all others, while only requiring information from the current and immediately preceding 100 ms signal instance. Approach 2 required 20+ instances, or one second of signals, to achieve similar results. Approach 3 required 30 words, or 1.5 seconds of data, yet achieves slightly inferior results. Note that the Random Forest and HMM approaches show lower standard deviations compared with the Affinity and DTW approaches.

7.2 Comparison with Results from Similar Studies

Table 7.1 compares the accuracy in this study for Approach 4 (HMMs) with results achieved by other researchers. Other results reported in the literature almost exclusively classify small window segments of 500 ms or less. Since Approach 4 produced the best results for the window based approaches in this study, its results are compared. The top two rows are results reported in this dissertation. The first involved identifying 47 classes of continuous movement with an average accuracy of 75.09%. The other shows an accuracy of 85.68% when the transition movements are not counted. Transitions had low recognition accuracy - a problem reported in other studies. A common solution is to ignore them in reporting results. The higher results are a more consistent comparison with other studies, while the lower results are included to show applicability to continuous movement.

Atzori is from the NINAPRO project and is the closest comparator to the research reported here in terms of types and number of hand and finger movements involved. That study attempted to recognize 50 classes of mixed hand, wrist, and individual finger movements. The movements involved executing and holding a posture - for example, grasping a bottle or flexing an index finger. The study was limited to identifying isolated single movements and did not attempt to identify a continuous set that would be used in an ADL. The best results used a relatively large feature set of 336 and achieved accuracy equivalent to that reported here (75.27%). Various other combinations of features were also reported. The RMS feature results are included in the table since the number, twelve, is the closest configuration to the eight MAV features used in this study. Accuracy for this reduced set was somewhat lower (71%).

The other three results in the table show high accuracy, but involved many fewer movement classes and are typical of studies involving control applications. As with the Atzori results, the focus in all was in executing and holding a single posture for a short period of time rather than attempting to identify a continuous stream.

Table 7.1: Comparison with results from studies involving grips and movements of the wrist, hands, and fingers. The top two rows are results reported in this dissertation - the first involving continuous movement and 47 classes, the other ignores the transition movements that showed relatively low accuracy. Atzori is the closest comparator to this study and involved 50 classes with two sets of features used giving different accuracies. The other three listed show high accuracy, but involved many fewer movement classes.

Researcher	**Grips & Movements**	**# classes**	**# chan**	**# features**	**Acc.(%)**
Shuman (HMM)	Hand-continuous	47	8	8	75.09
Shuman (HMM)	Hand-ignore transitions	31	8	8	85.68
Atzori [33]	Hand& finger-single movements	50	12	336 (All)	75.27
Atzori [33]	Hand& finger-single movements	50	12	12 (RMS)	71.00
Tenore [31]	Finger movements	12	32	128	84.9-99.7
Castellini [19]	Hand grips	6	10	10	89.00
Khokhar [36]	Wrist movements	19	4	24	88.00

7.3 Performance Comparison of the Approaches

The performance of the approaches developed here varied and are listed in Table 7.2. The timings were recorded while running implementations of the approaches on an Apple iMAC 3.06 GHz Intel Core 2 Duo Processor with 4 GB of 667 MHz memory. The implementations were written in Matlab. The DTW distance measure was a C language version from [42].

The timings were collected by inserting Matlab 'tic' and 'toc' timings in the implementation code. Since these record wall clock times, no other processes were active on the iMAC while the timings were collected to ensure that only time required by the training and testing was counted. A complete set of training and test runs were conducted as described earlier. Twelve runs were executed with 11 repetitions for training and one for testing, repeated twelve times so that each repetition participated in testing once. Training and test timings were collected for all runs and averaged.

The Affinity and DTW approaches both have very low training times since there is relatively little model building in either. Both times listed include converting the signal values to SAX symbols. By contrast, the Random Forest and HMM approaches both require substantial training time since they rely on generating 25 decision trees.

Classification times show the opposite trend. Affinity requires .024 seconds (24 ms) to classify a single 100 ms instance, while DTW requires 0.115 (115 ms) to classify an entire

Table 7.2: Comparison of the performance of the four approaches developed and evaluated in this study.

Approach	**Classification unit**	**Acc.(%)**	**Training time (s)**	**Classif. time (s)**
Affinity (SAX)	100 ms MAV window	76.72	1.9	.024
DTW/NN (SAX)	entire movement segment	73.01	1.62	.115
RF(25 trees)	100 ms MAV window	65.54	78.0	.0008
HMM	100 ms MAV window	75.09	78.0	.00083

segment based on 30 SAX words (each word representing a 100 ms signal instance). Random Forest requires 0.0008 (.8 ms) to classify one 100 ms instance after model building. Likewise, HMM only requires 0.00083: 0.0008 to classify with the RF model and an additional 0.00003 to compute the belief calculation of a single instance.

While the Random Forest and HMM approaches are very fast to classify an instance, Affinity only requires three times the amount, and even DTW, a method known to be slow, only requires .115 seconds. However, Affinity and DTW both require that 30 words (30 100 ms instances) of a segment be reviewed before a classification decision can be made. Random Forest and HMM only require a single 100 ms instance to render a decision. Requiring 30 instances before a decision means a delay of 1.5 seconds and is far greater than the actual classification time itself. The implications of this for the use of Affinity and DTW in control applications is discussed later.

7.4 Source of Errors

While the accuracy range of the approaches reported here is promising considering the large number of classes involved, the best method only achieved an accuracy of 76.72%, or a 23.28% error rate. Classifier error can be decomposed into three components: variance, bias, and noise [11,44]. Variance refers to the effect of variability of the training data on the classifier's decision boundaries. Classifiers whose decision boundaries change a lot when the training data changes are said to have high variance. Bias is the impact of the classifier's design on its decision boundary. Those with complex boundaries that separate the training data classes with high accuracy are said to have low bias. Noise is the intrinsic error in the target data.

The ideal goal is to construct a classifier with low variance and low bias, which is difficult to achieve because there is a trade-off between the two. Classifiers with low variance have simpler decision boundaries and are less sensitive to variation in the training data. However, the simple boundary can result in many misclassified training instances, resulting in high bias. On the other hand, classifiers with complex boundaries have low bias because they have fewer misclassified training instances. Changes in the training data, however, can result in significant changes in the decision boundary, resulting in high variance. Further, the complex boundary may not generalize well by correctly classifying new, previously unseen instances.

Approaches 1 and 4 attempt to balance bias and variance by employing Random Forest, an ensemble of decision tree classifiers. Decision Trees are sensitive to changes in the training data and have high variance. By combining them into an ensemble of trees Random Forest improves the generalization error and lowers the variance of the underlying decision trees [44].

The following two subsections discuss three areas of errors specific to this particular problem.

7.4.1 Recognizing Activities versus Sub-activities

The previous chapter includes confusion matrices showing classification results for the various approaches and subjects (Figures 6.1, 6.2, 6.4, 6.5, 6.7, 6.8, 6.10, and 6.11). All the confusion matrices show strong results along the diagonal (darker colors and higher numbers, where displayed), indicating correct classification, and errors occurring within the grip families shown as boxes along the diagonals and as neutral/rest in column 1.

The percentage of correct classifications is good considering the large number of classes to be recognized. The many errors occurring within the grip family boxes indicate that the accuracy for recognizing the base grip is very high, even if the specific sub-activity is wrong. A further look at the errors reveals that many misclassified instances occur immediately before or after the true class on the diagonal. This could be caused by errors in labelling, or by a failure to accurately recognize transitions from a given class to its successor. Likewise, identifying a neutral/rest instance in a sequence of otherwise active grips or movements could also be due to labelling issues, but might also reflect the reality that the subject was actually in a rest state for a very short period (100-200 ms) before completing the movement. Relaxing the classification success criteria to allow for correct identification as being within the entire grip family or neutral/rest results in higher classification accuracy.

7.4.2 Ambiguous classes and overlapping patterns

More generally than the problem of confusing instances within the same grip family is the idea that not all class patterns are separable by any boundary. They do not all fall in their own unique space because their patterns overlap. This is a particularly difficult problem that might resist any solution. A certain level may be unavoidable because of this intrinsic noise in the data.

7.4.3 Issues Concerning Labelling

An important task in using supervised learning techniques such as classification is labelling the signal instances with the grip or movement associated with the signal to establish its ground truth. Each subject's 16 minutes of collected data resulted in $\approx 19,000$ instances that required labelling. This was time-consuming, prone to error, and is a well-known problem with supervised learning.

There are techniques for improving the labelling process and making it less onerous and more accurate. For the problem described in this dissertation, using other data modalities can help, as was the case with using accelerometer data. However, this still required manual data review and, while it made the resulting labels more accurate, did not completely eliminate mislabelled instances and increased the amount of time needed for the process.

The labelling problem is a major reason the activities performed by subjects followed a timed script. That simplified the mapping of activities to signals since it provided a chronological starting point for recognizing transitions and assigning appropriate labels. Using accelerometer data provided additional data that allowed for more precision in the labels. Ultimately, it will be desirable to track activities unconstrained by scripts - activities that a typical person might perform throughout the day. These will span longer time periods than a minute or two and will consist of grips and movements in many different combinations. Assigning ground truth to such signal streams will require an automated labelling process that uses modalities such as accelerometer, positional, or pressure inputs. With such a process a much larger data set can be collected and transition tendencies, important in Approach 4, can be learned.

7.5 Belief Calculation Versus Random Forest

The Approach 4 HMM used the two step process of (1) calculate Random Forest (RF) probabilities, and (2) apply the 'belief' calculation to identify the class. As was shown in the Results chapter, 'belief' improved the RF accuracy by 8% to 11% for the various subjects, and the overall accuracies ranged from 68.14% to 84.3%. Improvements are possible. For example, the transition matrix was created using a 'best judgement' method on how likely transitions from activity to activity were to occur. After initial set-up, no attempts were made to systematically tune the values. Would different values improve overall accuracy? Would they improve the accuracy (true positive rates) of individual grips or movements? Would tuning the values risk over-fitting them to specific subject data and not generalize well? This was not explored here, but is an interesting question for future research.

7.6 Practicality of the Approaches

How practical would the approaches be in real-life situations? For the use-case involving after-the-fact review and batch processing of the signal stream, all the approaches are practical. For those cases, Approach 2 or 3 provide high accuracy and the requirement for a one or two second data segment to achieve satisfactory accuracy is not a problem.

For real-time control systems, however, rapid turn-around of data input and activity identification is essential. A 300 ms processing window is generally used as a maximum value [14] in such applications. Approach 1 only requires collection and analysis of a 100 ms signal slice before rendering a classification decision. The 300 ms timing requirement would be met, at the cost of slightly lower overall accuracy (ranging from 63.7% to 82.69%) and some inconsistency in the predicted signal stream classes.

Approach 2 only requires consideration of the current signal instance and some that were already seen, and yields high accuracy. The range for 11 SAX symbols and 30 comparison words as listed in Table 6.2 was from 69.32% to 85.03%. Achieving this accuracy required 30 processing windows, or 1.5 seconds, of data before an activity is recognized to the stated accuracy and is therefore too slow for the real time requirement. Reducing the number of windows to five (.25 seconds of data) could bring the processing time to under 300 ms, but lowers the accuracy range to 49.83% to 75.16%.

Approach 3 requires that an entire activity segment be considered. The results show that 30 words, or 1.5 seconds of signal data, is required to reach the maximum accuracy reported here. The range for a 30 word segment is, from Table 6.3, 67.91% to 85.46% when using seven SAX symbols. A 1.5 second delay is too long in most real-time applications and would have to be shortened to consider this approach in those settings. Reducing the number of windows to five to meet the timing requirement results in an accuracy range of 40.07% to 68.49% (for all symbol and word combinations), which is unacceptably low. For those without a real-time requirement, the accuracy when considering a 30-word, 1.5 second data segment is acceptable and makes the approach usable.

Approach 4's accuracy range was 68.14% to 84.3%. The approach is the most attractive one for real time settings. Its accuracy range is somewhat better than for Approach 1 and it has similar timing characteristics. Unlike 1, it considers the relative order and timing of activities. Potential improvements can therefore be made by tuning the transition matrix and considering more previous states.

The findings reported here report on four different approaches that can be successfully applied to different use-cases. They support the view that prehensile patterns can be distinguished by combining electrical and mechanical properties of the task. This is both clinically useful and opens the way for an approach to help simulate hand functional activities. With improvements it may also prove useful in real life settings, including real time control applications.

Chapter 8: Future work

Future work should address some of the shortcomings of the approaches reported here. More prehensile patterns should be investigated, leading to the goal of recognizing continuous movement, not just discrete action segments. A step toward achieving that goal is to create a more holistic model that combines the electric signal, the mechanical components, and the dynamic components to picture the activity in its entirety.

Exploring the recognition of individual tasks and their differences in accuracy would be useful in breaking down the total prehensile space into those that can be easily recognized and those that cannot. In this research the ball squeeze (BSQ) and hammer grip (HG) were well-recognized with high true positive rates for all methods. Tip pinch grip (TPG) and key grip (KG), by contrast proved difficult to recognize. For the difficult cases, additional analytical tools can be considered such as recognizing family of movements organized around their base grip. For those, a hierarchical strategy could be used to recognize the family, for example a hammer grip, and then operate only on those instances belonging to that family in a secondary step to identify the specific movement involved such as hammer raise and lower.

An important goal in exploring prehensile patterns is to model continuous, real time movement that people typically perform during the course of a day. This would address the real-time control use case. The research reported here relied on training and testing scripted activities. Approach 4 began considering continuous segments of 10 seconds, but within the constraints of the scripts.

The following items briefly present and discuss potential research paths for exploring the recognition of continuous movement.

Expanding the number of grips and movements. While the 47 activities used in this research is large compared with most other reported results, it is far less than what would be needed in real life control of a robotic or prosthetic wrist and hand. Increasing the number would help move toward a real-world solution, but might stretch classification techniques beyond their ability to identify them with enough accuracy to meet requirements.

Using more data modalities. EMG was central to this research. However, additional modalities could be introduced to improve recognition. For example, accelerometer data was used here in a limited way to help label transitions, but it could also be used to build a feature set. Other modalities would be necessary to bring additional capabilities to recognition. Examples include pressure devices to determine the amount of force used in a grip, and positional sensors to identify the location and speed of a movement.

More subjects. This is perhaps the most obvious expansion of the research, but an important one needed to give confidence in the universality of the results. An ideal subject pool would include a diverse set of people representing different ages, sexes, ethnicity, and abilities. Data collection is limited by the practicality that it is time consuming. Still, expanding the pool beyond five subjects would be an excellent way to continue.

Collecting more data. Successfully modelling continuous movement requires the collection and processing of considerably more data than was done in the research reported here. Data reflecting the performance of routine, unscripted activities would require many minutes and probably hours of signal input. The large variety of movements, need for many examples of each, the many transitions, and the incomplete movements that are started but not finished all need to be collected and modelled.

Improving labelling. The large amount of data required to model continuous movement cannot be manually labelled, as was done in this research. The large volume renders manual methods infeasible. Collecting additional modalities, including accelerometer, pressure, and positional, and synchronizing them with the EMG data stream gives the possibility of using the other modalities to create an automated process to quickly and accurately label the large datasets that would be needed.

Experimental parameters. This research did not exhaustively explore the various parameter spaces. The signal stream windowing can be varied to a smaller or larger amount, and the overlap changed or dropped. Transition probabilities used in Approach 4 can be changed to give different weights to transitions. Additional features beyond the MAV used here can be tried - the Background section described some used elsewhere.

Hierarchical models. Multiple models, including classification, could be used to recognize a grip family in one step and movements performed while the hand is assuming that grip in a second step. In such a hierarchical approach, different families could be trained with different techniques suited to their particular needs. This has the potential to expand the number of recognized activities while keeping the demands on classifiers to a reasonable number of classes.

Modelling activity atoms. Techniques including clustering could be used to break down activities into smaller units, or atoms. These atoms could be assigned as symbols and an alphabet developed that would be used as descriptors of the larger grips and movements. Instead of trying to identify whole activities, the atoms would be recognized and an eventual movement identified from its known atomic parts.

Bibliography

[1] G. Shuman, Z. Duric, D. Barbara, J. Lin, and L. H. Gerber, "Improving the recognition of grips and movements of the hand using myoelectric signals," *BMC Medical Informatics and Decision Making*, vol. 16(2), p. 78, 2016.

[2] J. R. Cram and E. Criswell, *Cram's Introduction to Surface Electromyography.* Sudbury, MA: Jones & Bartlett Learning, 2011.

[3] L. J. Hargrove, K. Englehart, and B. Hudgins, "A comparison of surface and intramuscular myoelectric signal classification," *Biomedical Engineering, IEEE Transactions on*, vol. 54, no. 5, pp. 847–853, 2007.

[4] C. J. De Luca, "The use of surface electromyography in biomechanics," *Journal of applied biomechanics*, vol. 13, pp. 135–163, 1997.

[5] M. Nordin and V. H. Frankel, *Basic biomechanics of the musculoskeletal system.* Philadelphia, PA: Lippincott Williams & Wilkins, 2001.

[6] maricopa. [Online]. Available: http://www.maricopa.edu

[7] J. Napier, "The prehensile movements of the human hand," *J Bone Joint Surg*, vol. 38B, pp. 902, 913, 1956.

[8] T. R. Farrell and R. Weir, "A comparison of the effects of electrode implantation and targeting on pattern classification accuracy for prosthesis control," *Biomedical Engineering, IEEE Transactions on*, vol. 55, no. 9, pp. 2198–2211, 2008.

[9] Delsys. [Online]. Available: http://www.delsys.com.

[10] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2^{nd} ed. Wiley-Interscience, 2000.

[11] C. M. Bishop, *Pattern recognition and machine learning.* New York: Springer, 2006.

[12] A. Jain, R. Duin, and J. Mao, "Statistical pattern recognition: A review," *Pattern Recognition and Machine Intelligence, IEEE Trans. on*, vol. 22, no. 1, pp. 4–37, 2000.

[13] G. Shuman, "Using forearm electromyograms to classify hand gestures," *Proceedings, IEEE International Conf. on Bioinformatics and Biomedicine*, 2009.

[14] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *Biomedical Engineering, IEEE Transactions on*, vol. 50, no. 7, pp. 848–854, 2003.

[15] M. A. Oskoei and H. Hu, "Myoelectric control systemsa survey," *Biomedical Signal Processing and Control*, vol. 2, no. 4, pp. 275–294, 2007.

[16] K. Englehart, B. Hudgins, and P. Parker, "A wavelet-based continuous classification scheme for multifunction myoelectric control," *Biomedical Engineering, IEEE Trans. on*, vol. 48, no. 3, pp. 431–438, 2001.

[17] B. Hudgins, P. Parker, and R. N. Scott, "A new strategy for multifunction myoelectric control," *Biomedical Engineering, IEEE Transactions on*, vol. 40, no. 1, pp. 82–94, 1993.

[18] K. Englehart, B. Hudgins, P. A. Parker, and M. Stevenson, "Classification of the myoelectric signal using time-frequency based representations," *Medical engineering & physics*, vol. 21, no. 6, pp. 431–438, 1999.

[19] C. Castellini and P. van der Smagt, "Surface emg in advanced hand prosthetics," *Biological Cybernetics*, vol. 100, pp. 35–47, 2009, 10.1007/s00422-008-0278-1. [Online]. Available: http://dx.doi.org/10.1007/s00422-008-0278-1

[20] J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing sax: a novel symbolic representation of time series," *Data Mining and knowledge discovery*, vol. 15, no. 2, pp. 107–144, 2007.

[21] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools and techniques*. Burlington, MA: Morgan Kaufmann, 2005.

[22] mathworks. [Online]. Available: http://www.mathworks.com.

[23] H. Suzuki, R. Conwit, D. Stashuk, L. Santarsiero, and E. Metter, "Relationships between surface-detected emg signals and motor unit activation," *MedSci Sports Exerc.*, vol. 34, no. 9, pp. 1509–1517, 2002.

[24] Y. Huang, K. Englehart, B. Hudgins, and A. Chan, "A gaussian mixture model based classification scheme for myoelectric control of powered upper limb prostheses," *Biomedical Engineering, IEEE Transactions on*, vol. 52, no. 11, pp. 1801–1811, 2005. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=arnumber=1519588

[25] A. Chan and K. Englehart, "Continuous classification of myoelectric signals for powered prostheses using gaussian mixture models," in *Engineering in Medicine and Biology Society, 2003. Proceedings of the 25th Annual International Conference of the IEEE*, vol. 3, 2003, pp. 2841–2844 Vol.3. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=arnumber=1280510

[26] A. D. Chan and K. B. Englehart, "Continuous myoelectric control for powered prostheses using hidden markov models," *Biomedical Engineering, IEEE Transactions on*, vol. 52, no. 1, pp. 121–124, 2005. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=arnumber=1369595

[27] A. T. Au and R. F. Kirsch, "EMG-based prediction of shoulder and elbow kinematics in able-bodied and spinal cord injured individuals," *Rehabilitation Engineering, IEEE Transactions on*, vol. 8, no. 4, pp. 471–480, 2000.

[28] S. Bitzer and P. van der Smagt, “Learning emg control of a robotic hand: towards active prostheses,” in *Proceedings of ICRA, the International Conference on Rootics and Automation*, 2006, pp. 2819–2823.

[29] G. Tsenov, A. Zeghbib, F. Palis, N. Shoylev, and V. Mladenov, “Neural networks for online classification of hand and finger movements using surface emg signals,” in *Proceedings of the 8th Seminar on Neural Network Applications in Electrical Engineering*, 2006, pp. 167–171.

[30] T. S. Saponas, D. S. Tan, D. Morris, and R. Balakrishnan, “Demonstrating the feasibility of using forearm electromyography for muscle-computer interfaces,” in *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, ser. CHI ’08. New York, NY, USA: ACM, 2008, pp. 515–524. [Online]. Available: http://doi.acm.org/10.1145/1357054.1357138

[31] F. Tenore, A. Ramos, A. Fahmy, S. Acharya, R. Etienne-Cummings, and N. Thakor, “Multi-subject/daily life activity emg-based control of mechanical hands,” *IEEE Transactions on Biomedical Engineering*, vol. 56, pp. 1427,1434, 2009.

[32] I. Kuzborskij, A. Gijsberts, and B. Caputo, “On the challenge of classifying 52 hand movements from surface electromyography,” in *Proceedings of the International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2012, pp. 4931–4937.

[33] M. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A.-G. M. Hager, S. Elsig, G. Giatsidis, F. Bassetto, and H. Muller, “Electromyography data for non-invasive naturally-controlled robotic hand prostheses,” *Scientific Data*, 2014. [Online]. Available: http://dx.doi.org/10.1038/sdata.2014.53

[34] G. Shuman, Z. Duric, D. Barbara, J. Lin, and L. H. Gerber, “Using myoelectric signals to recognize grips and movements of the hand,” *Proceedings, IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 388–394, 2015.

[35] C. Castellini, A. E. Fiorilla, and G. Sandini, “Multi-subject/daily-life activity emg-based control of mechanical hands,” *Journal of neuroengineering and rehabilitation*, vol. 6, no. 1, p. 41, 2009.

[36] Z. O. Khokhar, Z. G. Xiao, C. Menon *et al.*, “Surface emg pattern recognition for real-time control of a wrist exoskeleton,” *Biomedical engineering online*, vol. 9, no. 1, p. 41, 2010.

[37] D. A. Winter, *Biomechanics and motor control of human movement.* Hoboken, NJ: John Wiley & Sons, 2009.

[38] E. Keogh and A. Ratanamahatana, “Exact indexing of dynamic time warping,” *Knowledge and Information Systems (2005)*, vol. 7, pp. 358–386, 2004.

[39] T. Hastie and R. Tribshirani, “Classification by pairwise coupling,” *Annals of Statistics*, vol. 26, no. 2, pp. 451–471, 1998.

[40] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *Acoustics, Speech, and Signal Processing, IEEE Transactions on*, vol. 26, no. 1, pp. 43–49, 1978.

[41] J. Lin, R. Khade, and Y. Li, "Rotation-invariant similarity in time series using bag-of-patterns representation," *J Intell Inf Syst*, vol. 39, pp. 287–315, 2012.

[42] Q. Wang, "http://www.mathworks.com/matlabcentral/fileexchange/43156-dynamic-time-warping–dtw-," Signal Analysis and Machine Perception Laboratory, Department of Computer and Electrical Engineering, Rensselaer Polytechnic Institute, 2013, accessed 20 March 2016.

[43] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3^{rd} ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2009.

[44] P. N. Tan, M. Steinbach, and V. Kumar, *Introduction to DATA MINING.* Boston, MA 02116, USA: Pearson Education, Inc., 2006.

Curriculum Vitae

Prior to completing this Doctor of Philosophy degree Gene Shuman worked at the U.S. Department of State, spanning the tenures of Secretaries Kissinger and Kerry. During that time he held a variety of Information Technology related positions, including Software Developer, Systems Programmer, Associate Director of the Paris Regional Financial Services Center, Branch Chief for Core Financial Systems, and Director for Consular Systems Development and Support. He holds a Master of Science degree in Computer Science from the University of Maryland, College Park, and a Bachelor of Science degree in Mathematics and Computer Science from Penn State University, University Park. He is a member of ACM and the Project Management Institute and is a certified Project Management Professional. His research interests include Pattern Recognition, Data Mining, Biomechanics applications, Electromyography, Software Engineering, and Project Management.