8-23-2018 1:30 PM

Dynamic Calibration of EMG Signals for Control of a Wearable Elbow Brace

Taylor Stanbury
The University of Western Ontario

Supervisor
Trejos, Ana Luisa
The University of Western Ontario

Graduate Program in Biomedical Engineering
A thesis submitted in partial fulfillment of the requirements for the degree in Master of Engineering Science

Dynamic Calibration of EMG Signals for Control of a Wearable Elbow Brace

Taylor Stanbury

M.E.Sc Thesis, 2018

School of Biomedical Engineering

The University of Western Ontario

Abstract

Musculoskeletal injuries can severely inhibit performance of activities of daily living. In order to regain function, rehabilitation is often required. Assistive devices for use in rehabilitation are an avenue explored to increase arm mobility by guiding therapeutic exercises or assisting with motion. Electromyography (EMG), which are the muscle activity signals, may be able to provide an intuitive interface between the patient and the device if appropriate classification models allow smart systems to relate these signals to the desired device motion.

Unfortunately, there is a gap in the accuracy of pattern recognition models classifying motion in constrained laboratory environments, and large reductions in accuracy when used for detecting dynamic unconstrained movements. An understanding of combinations of motion factors (limb positions, forces, velocities) in dynamic movements affecting EMG, and ways to use information about these motion factors in control systems is lacking.

The objectives of this thesis were to quantify how various motion factors affect arm muscle activations during dynamic motion, and to use these motion factors and EMG signals for detecting interaction forces between the person and the environment during motion.

To address these objectives, software was developed and implemented to collect a unique dataset of EMG signals while healthy individuals performed unconstrained arm motions with combinations of arm positions, interaction forces with the environment, velocities, and types of motion. An analysis of the EMG signals and their use in training classification models to predict characteristics (arm positions, force levels, and velocities) of intended motion was completed.

The results quantify how EMG features change significantly with variations in arm positions, interaction forces, and motion velocities. The results also show that pattern recognition models, usually used to detect movements, were able to detect intended characteristics of motion based solely on EMG signals, even during complex activities of daily living. Arm position during elbow flexion–extension was predicted with 83.02 % accuracy by a support vector machine model using EMG signal inputs. Prediction of force, the motion characteristic that cannot be measured without impeding motion, was improved from 76.85 % correct to 79.17 % accurate during elbow flexion–extension by providing measurable arm position and velocity information as additional inputs to a linear discriminant analysis model. The accuracy of force prediction was improved by 5.2 % (increased from 59.38 % to 64.58 %) during an activity of daily living when motion speeds were included as an input to a linear discriminant analysis model in addition to EMG signals.

Future work should expand on using motion characteristics and EMG signals to identify interactions between a person and the environment, in order to guide high level tuning of control models working towards controlling wearable elbow braces during dynamic movements.

Keywords: motion classification, motion characteristics, dynamic movements, interaction forces, arm position, joint velocity, electromyography, EMG.

Acknowledgements

First and foremost, I would like to thank my supervisor, Dr. Ana Luisa Trejos. My thesis would not exist were it not for her constant support and encouragement. I feel extremely lucky to have had her as a mentor. Her guidance has allowed me to achieve many goals in my life. She continues to motivate me to pursue new challenges. I could not have asked for a better supervisor, I cannot thank her enough for all she has done for me.

The work of this thesis was funded by a Natural Sciences and Engineering Research Council (NSERC) by a Canada Graduate Scholarship and under grant RGPIN-2014-03815, and by the Ontario Ministry of Economic Development, Trade and Employment and the Ontario Ministry of Research and Innovation through the Early Researcher Award, and by infrastructure grants from the Canada Foundation for Innovation and the Ontario Research Fund.

A thank you goes to everyone who participated in trials, providing me with much needed data. I want to thank Yue Zhou for assisting with the mechanical design and 3D printing of parts, and Tyler Desplenter for guidance in using the Kinect. Thanks go to Emma Farago for providing help for the collection and analysis of EMG signals. I would like to thank everyone who currently is or has been part of the Wearable Biomechatronics Laboratory.

Lastly, I would like to thank my parents, Karen and Brent, and my sister Keelin, for their constant love and support.

Contents

List of Figures

List of Tables

Nomenclature and Acronyms

Acronyms

AD	Anterior Deltoid
ADL	Activity of Daily Living
AMF	Atomic Monitoring Function
ANN	Artificial Neural Network
AR	Autoregressive
BRA	Brachialis
BB_L	Biceps Brachii Long Head
BB_S	Biceps Brachii Short Head
BRD	Brachioradialis
CCOHS	Canadian Centre for Occupational Health and Safety
DOF	Degree of Freedom
ECR	Extensor Carpi Radialis
ECRB	Extensor Carpi Radialis Brevis
ECU	Extensor Carpi Ulnaris
EMG	Electromyography
ESM	Event-driven Safety Monitoring
FCR	Flexor Carpi Radialis

FCU	Flexor Carpi Ulnaris
FMI	Functional Independence Measure
GUI	Graphical User Interface
IMU	Inertial Measurement Unit
I/O	Inputs and Outputs
IP	Internet Protocol
ISPI	Infraspinatus
LD	Lateral Deltoid
LDA	Linear Discriminant Analysis
LOSO	Leave-One-Subject-Out
MAV	Mean Absolute Value
MAVS	Mean Absolute Value Slope
MDF	Median Frequency
MNF	Mean Frequency
MSK	Musculoskeletal
MSV	Mean Square Value
MVC	Maximum Voluntary Contraction
PD	Posterior Deltoid
PSM	Permanent Safety Monitoring
PT	Pronator Teres
RMS	Root Mean Square
sEMG	Surface Electromyography
SSC	Slope Sign Changes
SVM	Support Vector Machine
TRI_LAT	Triceps Brachii Lateral Head
TRI_LO	Triceps Brachii Long Head

TRI_M	Triceps Brachii Medial Head
WAMP	Wilson or Willison Amplitude
WL	Waveform Length
WMFT	Wolf Motor Function Test
WSIB	Workplace Safety and Insurance Board
ZC	Zero Crossings

Variables

a_k	Autoregressive coefficient
f	Frequency
F1	Force 1
F2	Force 2
F3	Force 3
i	EMG signal segment
j	Frequency bin
M	Length of the frequency bin
n	Order of the AR model
N	Length of the EMG signal
P	EMG power spectrum
P1	Position 1
P2	Position 2
P3	Posancialition 3
V1	Velocity 1
V2	Velocity 2
V3	Velocity 3
x	EMG signal

Units

cm	Centimetres
Hz	Hertz
kg	Kilograms
mm	Millimetres
ms	Milliseconds
N	Newton
Nm	Newton metre
rad	Radians
s	Seconds
°	Degrees
°C	Degrees Celsius

Chapter 1

Introduction

Musculoskeletal (MSK) disorders are among the leading causes of pain and discomfort in Canada. Eleven million Canadians are affected by MSK diseases each year [1]. The number of people affected is expected to increase with the aging population. MSK diseases cost the Canadian economy \$22.3 billion in 2000 [2]. The total cost consists of direct (health professional visits, rehabilitation) and indirect costs (loss of productivity or ability to perform activities, absence from work). Injuries contribute an additional \$15 billion each year [1]. Forty percent of lost time compensation claims in Ontario are due to MSK disorders, according to the Workplace Safety and Insurance Board (WSIB), demonstrating that MSK disorders lead to a loss of productivity [3, 4].

MSK disorders are usually chronic, causing long-term physical, psychological, and financial burdens [5]. Loss of function results in reduced ability to perform activities of daily living, including those required for self-care or in the workplace. Injuries to bones, joints, and muscles also result in reduced function and slow recovery [1]. Inactivity and injuries are risk factors for future health problems, and long rehabilitation strategies contribute to the financial burden to the individual and the health care system.

To improve the lives of Canadians burdened by MSK disorders and injuries, the Institute of Musculoskeletal Health and Arthritis' (IMHA) five year strategic plan (2014–2018) addresses disability, mobility and health as a main focus area for research in Canada [1]. A main theme is rehabilitation and restoring function to individuals with MSK disorders. Technological advancements can assist in rehabilitation, working towards the goal of improved mobility and well-being.

1.1 Motivation

The goal of rehabilitation is to help patients regain functional ability. During classical rehabilitation, therapists guide repetitive exercises to manage pain, regain range of motion, and build muscle strength. Physiotherapists may manually assist patients to perform movements or provide resistance during training [6]. Mobilization of joints is important to prevent stiffness after trauma, and orthotic braces are often used to progressively increase joint range of motion and to protect against further injury when patients are not in a rehabilitation therapy session [7]. However, poor adherence to rehabilitation programs, including not attending therapy sessions or not performing home-based exercises, is a barrier to health improvement [8].

Active-assisitive devices can be used to guide repetitive exercises [9], reducing the required amount of direct contact with a physiotherapist, and providing assistance outside of a therapy session. Electromechanical and robot-assisted devices, used as tools in rehabilitation, have helped improve patients' ability to perform activities of daily living, arm function, and muscle strength [10]. Such devices can interface with the patients by measuring muscle activity (electromyography), then detecting intended motions based on these signals in order to control the devices to assist movement and provide therapeutic training [11]. However, there are challenges in accurately detecting intended motions during unconstrained, dynamic movements.

1.2 General Problem Statement

The development of mechatronic devices to provide rehabilitation therapy and motion assistance after elbow surgery is of interest to the clinical community. Surface electromyography (EMG) signals are promising for monitoring muscle activity and to act as an interface between the patient and the device, by being used as inputs to the control systems. It has been noted that factors such as arm position, external forces, and movement speeds affect EMG signals, causing unfavourable control outcomes outside of a constrained laboratory environment.

This work aims to assess the influence of motion characteristics on EMG signals, quantifying their effect to inform the development of better motion classification models.

1.3 Research Objectives

The main goal of this thesis is to advance our understanding of the impact of motion characteristics during unconstrained dynamic arm movements on EMG signals used as inputs to control systems, working towards a smart wearable elbow brace. To achieve this objective, the work has focused on the following specific objectives:

- To develop software for calibration and collection of EMG, kinematic, and dynamic data from participants performing diverse movements while interacting with the environment.
- To collect EMG, kinematic, and dynamic data from healthy participants.
- To investigate trends in EMG feature values that vary in response to changes in motion characteristics during unconstrained movements.
- To investigate the usefulness of information about motion characteristics for motion classification.

1.4 Overview of the Thesis

The structure of the thesis is outlined below:

Chapter 2 Literature Review: A review of elbow rehabilitation techniques and assessments, EMG signals and features, motion classification for control of wearable devices, and a review of factors affecting EMG signals and motion classification accuracy.

Chapter 3 Design of Experiments: Includes the design of a repeated measures experimental protocol. Factors and levels are discussed.

Chapter 4 Equipment Set-Up: Outlines the measurement systems utilized and methods of data collection. This includes software development.

Chapter 5 Pre-Processing and Statistical Analysis: Describes the process of extracting relevant features from EMG signals. Features with statistical significance related to motion characteristics are discussed.

Chapter 6 Motion Characteristic Classification and Applications: Presents training of pattern recognition models to classify motion characteristics using EMG signal inputs. Iterations of classification models informed by statistical analysis results are described.

Chapter 7 Conclusions and Future Work: Highlights the contributions of this work. Recommendations for future work are also given.

Chapter 2

Literature Review

This chapter presents a review of literature in the areas of arm rehabilitation including assessment and assistive devices, arm motion including the tracking of motion, EMG signal features, the use of EMG features in motion classification for the control of wearable devices, and factors affecting EMG signals and control systems using these signals as inputs. A literature review was conducted using Google Scholar between September 2016 and July 2018. The keywords used for the searches included: myoelectric control, motion classification, dynamic movements, arm position, forces, motion velocity, EMG features, elbow rehabilitation, arm rehabilitation devices, prosthetic control, and a combination of some of those keywords. A total of 70 papers resulted from the search, a summary of which is presented in the following sections.

2.1 Elbow Rehabilitation

After surgery or injury to a joint of the body, such as the elbow, rehabilitation activities are commonly required. In rehabilitation, clinicians work with patients to regain functional ability [12]. The inability to move the elbow joint can inhibit many activities in daily life requiring use of the arm. It is important to mobilize the elbow early in rehabilitation to regain range of motion and prevent stiffness in this complex joint [7]. Four overlapping phases during a general rehabilitation guideline include: immediate motion, an intermediate phase, strengthening, and return to activity. In the first phase, pain is managed and motion is performed early to prevent more elbow stiffness. Then, exercises are continued to increase range of motion. More types of muscle contractions (isotonic contractions including concentric then eccentric contractions) can be introduced [6]. Progressively the muscles are strengthened, for example, via resistance training [13]. Returning to regular activity is usually completed gradually, especially for athletes, by increasing intensity of activities and joint use [6]. Various methods are used to evaluate and assess progress of rehabilitation, and a range of devices have been created for arm training, described further in the following sections.

2.1.1 Assessment

Clinical assessments used by professionals in rehabilitation assist in diagnosing problems and monitoring progress, commonly by evaluating performance of activities of daily living (ADLs) [12]. Activities of daily living are common movements performed repeatedly during daily life. They are goal oriented, performed with the purpose of completing a task. Measures used to assess movement function may include questionnaires, performance of tasks interacting with objects, with results consisting of scores on scales or various metrics [12]. A subset of these assessments include the Functional Independence Measure (FIM), the Barthel Index, Arm Motor Ability Test, Wolf Motor Function Test (WMFT), and the Fugl-Meyer Assessment [10]. These methods consist of activities of daily living and/or various range of motion and strength activities, some of which can be timed. Tests may be tailored to specific patient populations and injuries. For example, the WMFT has repeatedly been used to study chronic stroke patients [14].

Studies assessing arm motion (kinematics and dynamics) or measuring other outputs (sensor comparisons, device validation) use various motion measures and have participants perform a variety of relevent activities of daily living. These tasks may not include all aspects of clinical assessments but still produce valuable information. There is not one standard group of activities of daily living used. For example, in one study of upper extremity kinematics, participants performed four activities of daily living including touching their shoulder with their hand, simulating the motion of drinking, brushing hair, and moving their hand to their back pocket [15]. Another study looked at arm dynamics while performing ten activities of daily living, finding that motions of reaching the hand to the head or opposite side of the neck required the largest elbow rotations [16].

Another group looked at the kinematics of six activities of daily living: combing hair, perineal care, eating, reaching above the shoulder, washing axilla, and lifting a 4 kg weight [17].

2.1.2 Assistive Devices

Many research groups are working on developing wearable smart devices to provide therapy and assistance, commonly for people with impaired arm function after a stroke. A range of electromechanical and robotic devices reviewed perform functions throughout the rehabilitation process. These devices may provide movement while the user is passive, and they may assist or resist movements during training exercises [10]. It has been found that receiving therapy with an assistive device can improve arm function and performance of activities of daily living after a stroke [10]. Similar to developments in prosthetic devices, electromyography (EMG) signals, from both surface and intramuscular electrodes, monitor muscle activity and are sometimes used as an interface between the patient and the device. Regular elbow motion and an understanding of the arm muscle functions must be understood while developing such devices.

2.2 Motion

The elbow is a hinged joint performing mainly flexion–extension movements. Extension decreases the angle of the joint, while flexion increases the angle of the joint. Elbow flexion–extension movement and positions can be described by the degree of the angle between the forearm and the straight arm, with the fully extended straight arm being zero-degree flexion, as shown in Figure 2.1. Portions of the elbow anatomy are involved in pronation–supination of the forearm as well, as shown in Figure 2.2. However, the elbow is not involved in radial–ulnar deviation of the wrist, as shown in Figure 2.3. Major shoulder motions, adduction–abduction and flexion–extension, are shown in Figure 2.4.

Muscle activation is necessary for humans to perform motions or hold contractions. Arm muscles have generally been classified as extensors, which are active during elbow extension, flexors, which are activated during elbow flexion, and stabilizers. The biceps and brachioradialis, acting synergistically, are the main muscles that perform elbow flexion [18]. However, muscles do not

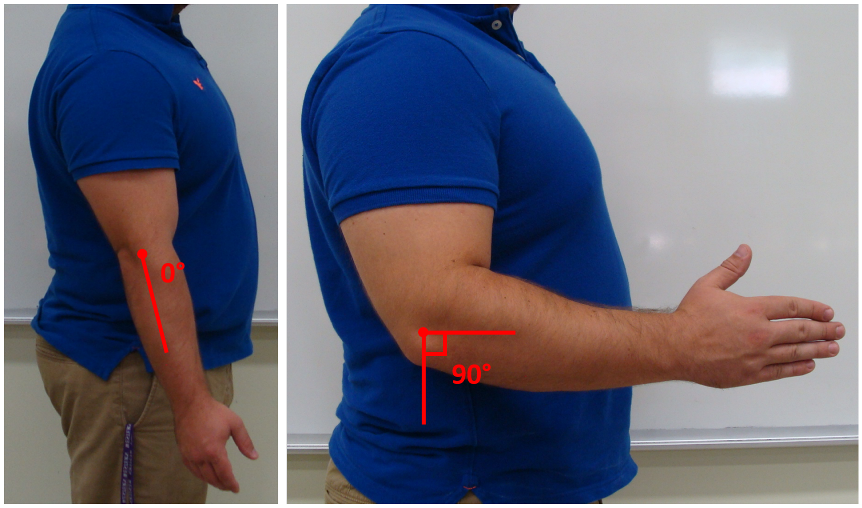

Figure 2.1: An elbow joint fully extended (*Left*), and flexed 90° (*Right*).

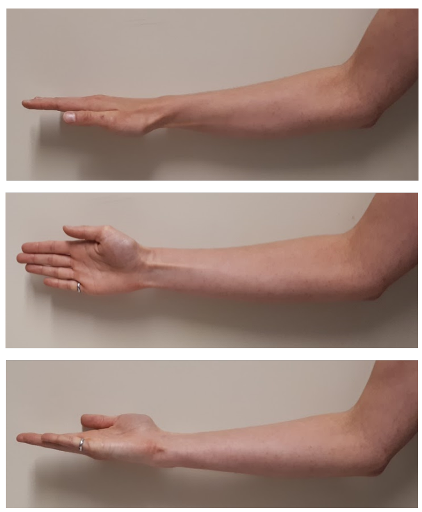

Figure 2.2: A forearm at 90° pronation (*Top*), neutral position (*Middle*), and at 90° supination (*Bottom*).

always fall into these strict categories as human motion is complex. Since ADLs are motions used to perform a task, they can include motions from multiple joints at the same time, with the individual movements adding together to perform a resultant motion.

Categories of motion can include isometric movements, and isotonic/dynamic movements with muscles in eccentric or concentric contraction. During isometric contractions, the joint angle and muscle length do not change. The joint angle and muscle length vary during isotonic movements. Muscle contractions can be classified as eccentric or concentric during isotonic movements. If

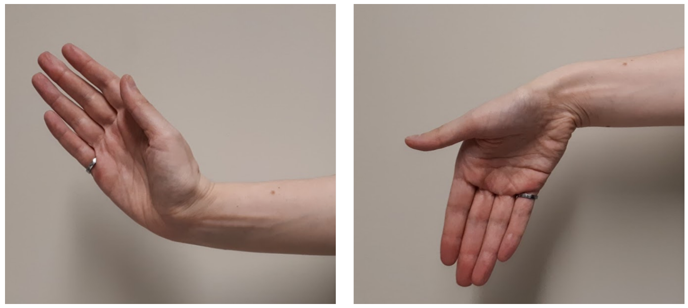

Figure 2.3: Wrist radial deviation (*Left*) and ulnar deviation (*Right*).

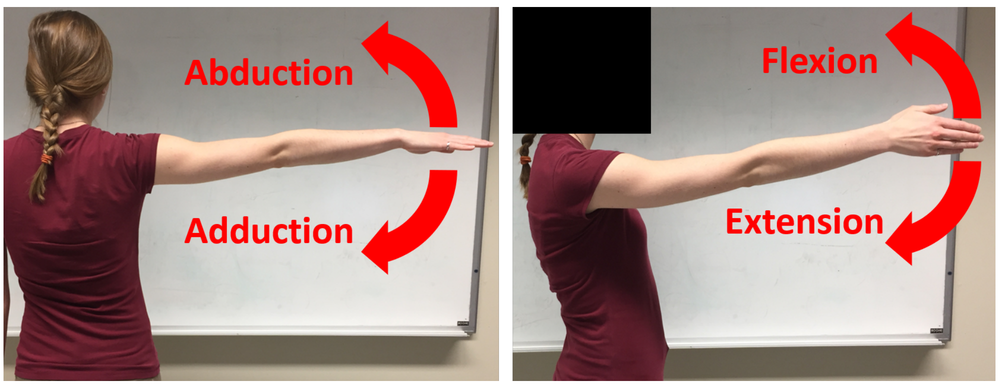

Figure 2.4: Shoulder adduction–abduction (*Left*) and flexion–extension (*Right*).

the muscle is activated and shortening, working to move the joint in the direction of motion, the muscle is performing concentric contractions. If the activated muscle is lengthening, resisting the direction of joint movement, the muscle is performing eccentric contractions. It is the coordination of individual muscle motor units and muscle groups that cause the resultant arm movement.

When performing or attempting to perform a motion or muscle contraction, electromyography (EMG) systems can detect levels of muscle activation. Superficial muscles can be measured with surface EMG (sEMG) electrodes attached to the skin while deeper muscles can only be measured with intramuscular electrodes. Many systems aimed at classifying wrist and hand motion for people with amputations in need of prosthetic devices, measure EMG signals using an arm band with electrodes spaced evenly around the arm. This does not give direct information for specific muscles. Instead, these signals are generally used with pattern recognition algorithms to classify motions. In other cases, EMG signals are gathered from specific muscles, with each channel giving information related to the muscle function. Muscles commonly measured using sEMG in order to detect intended arm motions and control devices are shown in Figure 2.5 and listed below [19, 19–28]:

- Biceps brachii short head
- Biceps brachii long head
- Brachialis
- Brachioradialis
- Pronator teres
- Infraspinatus
- Latissimus dorsi
- Upper trapezius
- Rhomboid major
- Pectorialis major
- Anterior deltoid
- Lateral deltoid
- Posterior deltoid
- Teres major
- Teres minor
- Triceps brachii long head
- Triceps brachii lateral head
- Triceps brachii medial head
- Extensor carpi ulnaris

- Flexor carpi ulnaris
- Extensor carpi radialis
- Flexor carpi radialis
- Palmaris longus
- Anconeous
- Extensor digitorum
- Flexor digitorum

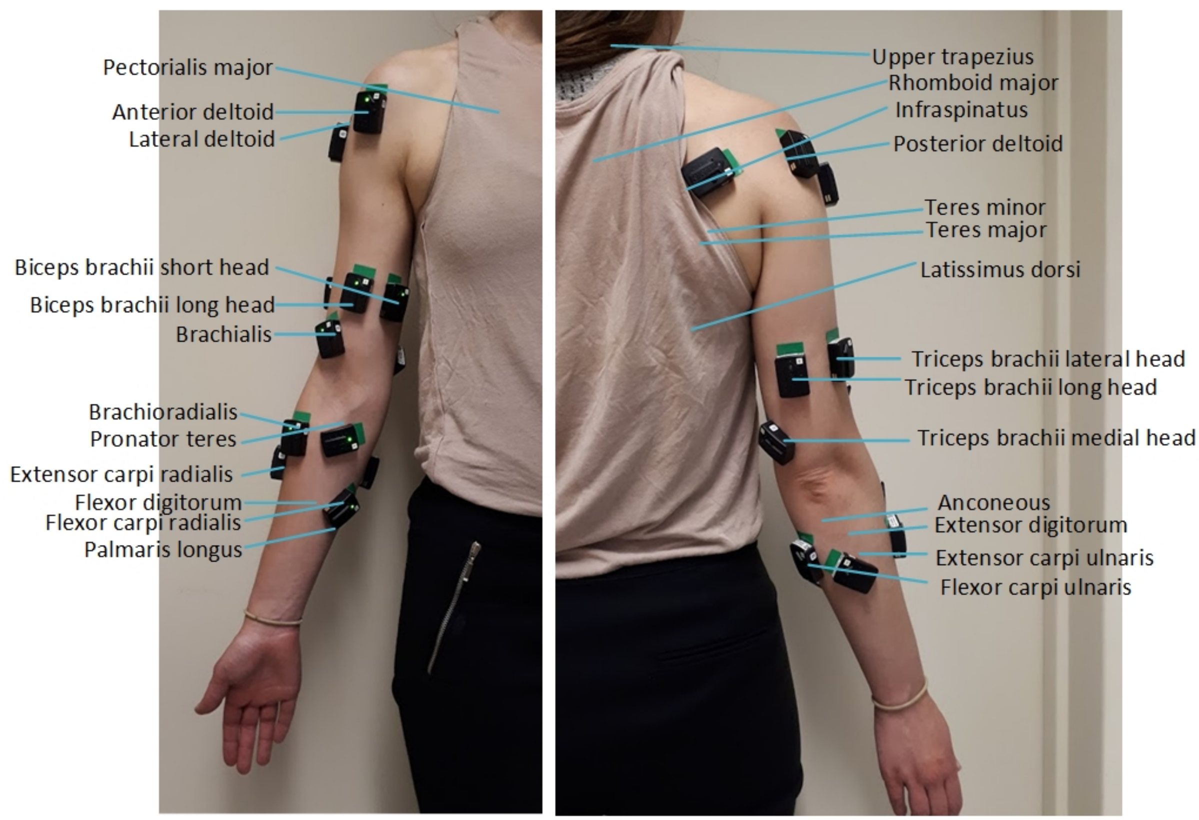

Figure 2.5: Upper extremity muscles commonly measured using sEMG, anterior view (*Left*) and posterior view (*Right*).

2.2.1 Motion Tracking

In order to study body kinematics and relate movements to muscle activity, motion can be tracked. A few main methods used to measure movement include electromagnetic tracking, inertial measurement units, and optical motion tracking.

Electromagnetic trackers can collect biomechanics movement data. However, metal in the environment can disturb the signals and markers being tracked must remain within a limited measurement range.

Inertial measurement units (IMUs) are a type of sensor that has been used for kinematic tracking. IMUs are usually small and not cumbersome when attached to landmarks on the body, and can be relatively inexpensive and portable compared to other systems such as optical trackers. Inertial sensors can contain a combination of gyroscopes, accelerometers, and a magnetometer. However, there are also many drawbacks. Drift is a common issue with use of gyroscopic information for position sensing, and the inertial data must be processed to relate it to a model of the body to extract meaningful joint angle and velocity information. There is not a set standard for using inertial sensing in motion analysis. Although gathering data from the sensors themselves can be relatively simple, the math required to calculate meaningful kinematic information with the data is more involved [29]. The entire algorithm calculating the kinematic information must be considered when determining accuracy, not just the individual sensors themselves. Inertial sensors have been shown to track human hip joint motion by fitting inertial sensor data in the sagittal plane to a sinusoidal curve [29]. As well, a combination of accelerometers and gyroscopes with Euler angle computation and Langrangian optimisation have been used to detect wrist, elbow, and shoulder positions in a controlled environment, with participants sitting while performing arm motions such as reaching, shrugging, forearm rotation, and tracing shapes on a desk with their hand [30]. This algorithm requires a kinematic model of the arm with lower and upper arm lengths [30].

Optical motion tracker systems may or may not require a marker. Without external markers, analysis of images to detect landmarks is needed. When markers are used, they are placed on the specific parts of the body to be tracked. The markers may be passive or active. Occlusion is a common problem with markers in optical tracking. If a specific marker is blocked by another part of the body or an object in the environment, the position of the body landmark related to that marker remains unknown.

The Microsoft Kinect sensor (Microsoft, USA), is a low cost alternative to large and expensive optical motion tracking systems, in some rehabilitation and human motion analysis applications. The Kinect uses depth information and its own skeletonization algorithm to output joint locations. Since there is no in depth calibration or choice of body model used, the Kinect estimates the body geometry with each frame [31]. Studies have been performed to assess accuracy with the camera focusing on a frontal view of a participant who is sitting still and breathing, and found that the length of the leg bones was varied by about 2 cm [31]. With the camera at a 45 angle to the frontal body plane, the variation in bone length was about 5 cm [31]. Another study found that for static poses, the Kinect was more accurate at identifying the joints of the upper extremities, with an accuracy of less than 100 mm for upper extremity joints, except for the hand, and lower accuracy for the lower extremity, except for the hip [32]. In general, the Kinect sensor has not identified joint positions of people in sitting postures as accurately as when people were in standing postures, as the sensor was built for standing game play [32]. Other drawbacks of the Kinect sensor are that occlusion of the body being tracked can inhibit tracking, and that clutter in the environment can cause the system to identify joints on other objects in the environment, such as a chair instead of the human body [31]. Other factors that can influence the accuracy of joint locations determined with the Kinect sensor are clothing (loose clothing may confuse the system) and body mass index [32]. However, if these precise measurements are not required, the Microsoft Kinect sensor can be a viable motion capture system of a lower price and faster calibration and setup than a motion capture system with markers.

The Kinect has also been used to perform and assess activities of daily living in virtual environments. An example is using the Kinect to track movements with an unscented Kalman filter-based system, and measuring speed-based performance metrics [33]. In this case, the accuracy and information available with the Kinect worked well enough to run a virtual reality system for which participants recovering from hemiparetic stroke sat in a chair performing upper extremity movements corresponding to virtual activities of daily living [33]. Metrics calculated from the mentioned virtual reality environment with data from the Kinect (duration to complete subtasks, normalized speed, "movement arrest period ratio") were compared to time metrics using the clinical Wolf Motor Function Test (WMFT), and it was found that the duration metric and the WFMT time were correlated with statistical significance [33].

Motion tracking provides true kinematic measurements of the arm during motion. When the arm movements are known, muscle activity can be related to these movements, and functions of muscles and trends of muscle activation during movements can be observed. Electromyography (EMG) measures this muscle activity.

2.3 EMG Signals

Electromyography (EMG) signals have been introduced in the development of assistive devices in rehabilitation and prosthetics. These signals are primarily used as an input to control systems to determine intended movements.

Electromyography (EMG) is a way to measure and record electrical activity of muscles. Electrodes can be placed on the surface of the skin over the underlaying muscle of interest, this is referred to as surface electromyography (sEMG). The electrodes measure voltages on a millivolt scale. These voltages are the combination of motor unit activations from multiple motor units firing in the muscles under the skin, underneath the electrode location. Electrodes can also penetrate the skin and muscle of interest with intramuscular EMG electrodes. By evaluating the signals recorded from the electrodes, information related to the muscle activation can be gathered.

sEMG sensors can vary in electrode shape and size, electrode material, inter-electrode distance, and construction. Electrode placement can vary with skin preparation, location and orientation of the electrodes, and fixation method. Hermans *et al.* tried to put together widely used guidelines for sEMG measurement, by looking at a variety of methods used and results, they determined recommendations for best practices [34].

To be useful, raw EMG signals must be processed. Usually, the first steps in processing raw EMG signals is gain amplification and filtering for the desired frequencies. Then, features categorized into the time domain, frequency domain, and time-frequency domain, can be extracted to gain meaningful information [35]. For most features, the EMG signals are first divided into windows, with or without overlaps, and then the features are extracted for each window.

Studies use varying window lengths and different overlap durations. One main factor that goes into choosing an appropriate window length is that for wearable devices, there is a desire to make the system work online in real time. If the system is to work in real time, there is a limit for the amount of delay that will be tolerated. In particular, delays of less than 300 ms have been found to be acceptable for electromyography controlled prosthetic wrist/hand devices to be usable in daily situations [36]. It has also been noted that window lengths between 150 and 250 ms is optimal [36]. However, window lengths as short as 40 ms and 50 ms have been used in other studies of myoelectric control and developing a neuromusculoskeletal model of the elbow with EMG inputs [24, 37]. Once the window length is chosen, there remains the option of overlapping windows by a number of samples, or not overlapping windows. Overlapping windows may improve accuracy at the expense of increased processing time [38]. While EMG signals provide a large amount of information about muscle activity, there are also limitations.

2.3.1 Limitations

An important limiting factor in studying EMG is that there can be crosstalk between signals gathered from muscles close to each other. Especially when the muscle activation is measured on the surface of the skin, the signals can have interference from surrounding muscles. However, if the EMG signals are being used to train algorithms to output motion information such as joint angle or force, and the crosstalk remains somewhat constant, the crosstalk could provide extra information to be used in the pattern recognition [27]. Related to crosstalk, electrode shift can influence EMG signals. sEMG electrodes attached to the surface of the skin can shift with respect to the muscle underneath, adding undesirable and difficult to remove signal variation [39]. Other factors affecting the quality of the EMG signals measured are sweat on the skin surface and electrode impedance changes [35]. Even with these limitations, EMG features from the time domain, frequency domain, and time-frequency domain are still being used in studying motion classification.

2.3.2 EMG Features

Although many time domain EMG features exist, some are defined very similarily and contain redundant information. A few features commonly used in the literature are listed below [35,40,41]:

- **Mean Absolute Value (MAV)** For MAV, the absolute value of an EMG signal is found, then this value is averaged for an EMG window, as follows:

$$MAV = \frac{1}{N}\sum_{i=1}^{N}|x_i| \tag{2.1}$$

  where N is the length of the EMG signal, and x_i is the EMG signal in segment i.

- **Slope Sign Changes (SSC)** Since slope of an EMG signal switches directions, the SSC refers to the number of times the slope changes from positive to negative and negative to positive, as follows:

$$\begin{aligned} SSC &= \sum_{i=2}^{N-1} f[(x_i - x_{i-1}) \times (x_i - x_{i+1})] \\ f(x) &= \begin{cases} 1, & \text{if } x \geq \text{threshold} \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{2.2}$$

- **Waveform Length (WL)** For a window, WL is the length of the EMG waveform, represented with the following equation:

$$WL = \sum_{i=1}^{N}|x_{i+1} - x_i| \tag{2.3}$$

- **Zero Crossings (ZC)** ZC for an EMG window refers to the number of times the amplitude of the signal crossed zero, as follows:

$$ZC = \sum_{i=2}^{N-1} [sgn(x_i \times x_{i+1}) \cap |x_i - x_{i+1}|]$$
$$sgn(x) = \begin{cases} 1, & \text{if } x \geq \text{threshold} \\ 0, & \text{otherwise} \end{cases} \tag{2.4}$$

- **Root Mean Square (RMS)** RMS is found by squaring the signal amplitude values, taking the mean of these squares over a window, and then calculating the square root, as follows:

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^{N} {x_i}^2} \tag{2.5}$$

- **Autoregressive Coefficients (AR coefficients)** AR coefficients are the coefficients of a linear combination model of previous EMG samples that could predict future EMG values, as follows:

$$x_i = \sum_{k=1}^{n} a_k x_{i-k} \tag{2.6}$$

where a_k is an autoregressive coefficient, and n is the order of the autoregressive model.

- **Wilson or Willison Amplitude (WAMP)** For WAMP, the difference in EMG amplitude between two segments is found. WAMP is the number of times this difference exceeds a threshold, as follows:

$$WAMP = \sum_{i=1}^{N-1} f(|x_i - x_{i-1}|)$$
$$f(x) = \begin{cases} 1, & \text{if } x \geq \text{threshold} \\ 0, & \text{otherwise} \end{cases} \tag{2.7}$$

Information is also contained in EMG signals in the frequency domain. Frequency domain features generally require more computational power and time to extract compared to time domain features. Two simple frequency domain features are described below [41]:

- **Mean Frequency (MNF)** The MNF is the average frequency of the EMG signal in the power spectrum, as follows:

$$MNF = \sum_{j=1}^{M} f_j P_i \Bigg/ \sum_{j=1}^{M} P_j \tag{2.8}$$

  where M is the length of the frequency bin, f_j is the frequency of the power spectrum at bin j, and P is the EMG power spectrum at frequency bin j.

- **Median Frequency (MDF)** The MDF is the median frequency of the EMG signal in the power spectrum, calculated as follows:

$$\sum_{j=1}^{MDF} P_j = \sum_{j=MDF}^{M} P_j = \frac{1}{2} \sum_{j=1}^{M} P_j \tag{2.9}$$

In addition, wavelets are an area being explored to describe information contained in EMG signals in the time-frequency domain, however, many groups continue to use only time domain features because of the ease of computation.

The information obtained with the different metrics can be redundant due to similarities in features, as shown in Phinyomark *et al.*'s comparison of 37 time and frequency domain features [41]. Frequency and frequency-time domain features can contain information that is lost in the time domain. However, extracting frequency domain features can be more complex and require more processing [42]. Studies have shown that there is not a significant improvement in motion classification when using time and frequency domain features compared to only using time domain features [41]. With systems intended to process information in real time with little delay, and with limited processing power (as devices must be wearable), working with only time domain features can generally give enough valuable information for the system to achieve its purpose (such as classifying motion within a limited timeframe).

A common time domain multi-feature set, known as the Hudgins feature set, is widely used for extracting information from EMG signals to be used as inputs to classifiers for motion classification. This set includes: mean absolute value, slope sign changes, waveform length, and zero crossings [27, 41, 43, 44]. In some instances, mean absolute value slope is also included in this list of key features, but not always [45, 46].

A group recommends using time domain features: mean absolute value (MAV), waveform length (WL), Wilson amplitude (WAMP), autoregressive coefficients (AR), mean absolute value slope (MAVS), and not using frequency-domain features due to the higher complexity in processing [41].

Often, studies will compare accuracies of pattern recognition algorithms using different sets of features as inputs. For example, in one study it was found that using multiple time domain features (MAV, SSC, WL, ZC) made a multilayer perceptron artificial neural network predict wrist forces based on sEMG signals more accurate than when using only the mean square value (MSV) feature [27]. In addition, two feature sets created by adding five wavelet marginals to the time domain features or adding the root mean square value and six autoregressive coefficients to the time domain features provided similar results (not significantly different) to the original time domain feature set, but still better than only using MSV [27]. Accordingly, it has been found that the features used to classify motions can affect the accuracy of motion classification more than the types of classifiers used [35].

2.4 EMG control of Wearable Devices

Many of these EMG features are being used as inputs for control systems of wearable devices to assist motion. Simple controllers not based on pattern recognition include: proportional control, finite state machines, and onset analysis [38]. Finite state machines involve states, transitions, and commands. The transitions are associated with the input signals and the states are motion commands [35]. These controllers can be intuitive to use and implement, comparing EMG signal levels to set thresholds, but are limited in the number of commands that can be implemented. Some complex sytems designed to predict joint trajectory utilize Hill-type models, relating EMG, force, joint angle, and contraction velocity [47]. Pattern recognition models are also used to classify intended motion. Once intended motions are identified, wearable devices can be commanded to assist with these motions. The use of models to classify motions is explored further in the next section.

2.4.1 Motion Classification

A variety of machine learning techniques are used to classify motion. With these techniques, systems can be trained to accept EMG features as inputs and connect them to motion classes, such as type of movement (i.e. wrist flexion–extension vs. rest). Use of these classifiers requires a training period to associate EMG patterns with the motion classes. The long training period, and training with limited data in constrained laboratory settings which do not necessarily translate well to clinical settings, are a couple of the limitations of pattern recognition models used in control systems [48]. It has been noted that there is a gap between research data collection and findings compared to usability results in daily living implementations, in regards to pattern recognition accuracy results for prosthetics [49].

Many types of classification systems exist, a few of the main classifiers are linear discriminant analysis (LDA) models, support vector machines (SVM), and artificial neural networks (ANN). Various combinations of features and classifiers are possible. In one study, SVMs were more accurate than neural networks for mapping sEMG signals to eight upper limb motions in real time [20]. However, accuracy is not the only factor to consider when choosing the optimal design, as SVMs consumed more time in this trial [20].

There is a gap between the accuracy of motion classification or pattern recognition systems in constrained laboratory settings, and usability in unconstrained daily activities. Frequently, in training the systems, the body, for example the arm, is held in a very specific position as the elbow is flexed and extended. However, changes in arm position and orientation, disturbances such as constant or variable external forces, analyzing signals gathered from different motion segments (static muscle activation vs. time-varying portions), and other noise factors can cause EMG signals to differ and reduce accuracy of motion classification systems [35, 48, 50].

2.5 Factors

Many factors affect EMG signals measured while motions are performed, and in turn, affect accuracy of motion classification algorithms. Some of the factors affecting the EMG signals are external to the muscle performance, not truly affecting muscle activations. Instead, these factors can cause noise and drift or change EMG electrode output signals when there are not real changes in the muscle activations [51]. Some changes in readings between systems could be caused by factors such as electrode size and type. Other factors may affect how motions are performed and could change muscle recruitment, even when the motion of interest is constant. These factors are not always understood and may not affect the signals coming from each muscle being measured at all times. Examples of these two types of factors are given below.

The following factors can affect EMG measurements:

1. Electrodes (material, style, surface, intramuscular, electrode spacing, sweating, skin cleanliness)
2. Placement (position of electrodes over muscle bodies, shifting of electrode location during use, crosstalk mixing signals from surrounding muscles)
3. System (amplification, filtering)

The following factors may affect muscle activation and the resulting EMG signals being measured:

1. Arm Position
2. Force
3. Velocity
4. Fatigue
5. Training Protocol

These variables can make the use of EMG signals as inputs to control systems challenging, as the signals can vary a lot, and it may not be known why they are changing. In laboratory settings, with variables controlled and motion very constrained, classification systems using EMG inputs generally have a higher accuracy than the same systems used in unconstrained daily activities [37]. The factors during daily living that can affect muscle activations are described further in the following sections.

2.5.1 Arm Position

In studies, EMG data are generally collected in very constrained laboratory settings with arms supported in specific resting positions, resulting in repeatable contractions. In the laboratory, shoulder movements can be avoided by fixing the upper arm to the body trunk. Other body movements have been limited by sitting participants in chairs and fixing their forearms to measurement devices that allow the participant to rotate the elbow in only one degree of freedom (DOF) [52]. Whereas in task-oriented situations or activities of daily living, limbs take on a variety of changing postures during contractions [49].

The actual muscle activations can change with limb posture and indirect joint angles, however, EMG readings can also change with limb position without being caused by changes in true muscle activation. When limbs move dynamically, the muscles contract or stretch, changing shape, and shifting beneath the skin. The movement of muscles under the electrodes may cause crosstalk effects to differ and alter the measurement conditions (such as distance from electrode to muscle), making electrode readings appear different even if the true muscle activation is not changing [39].

It is possible that different arm positions can cause activations in muscles not usually involved in the motion of interest. Muscles may need to activate to counteract gravity, and can play a larger or smaller role in some motions depending on joint angles. This is reflected in accuracies of pattern recognition of motion reducing with limb position variation [49].

For example, during trials of repetitive hand gripping while the arm was positioned with four shoulder flexion–extension angles and three elbow flexion–extension angles, EMG signal features (mean median frequency, RMS, slope of mean frequency, EMG work done) of the extensor carpi radialis brevis (ECRB) were not significantly different for different positions except for the EMG work done feature [53]. The ECRB muscle is located in the forearm, with the primary function of extending the wrist. Despite the ECRB not playing an active role in controlling elbow and shoulder joint angles, one of the EMG features of the ECRB tested was affected by those joint positions. Similar to how not all EMG features may be affected by position for one muscle, not all muscles may be affected by positions in the same way. In one study, the mean normalized sEMG envelope (indicating muscle activation) of the brachioradialis did not change with changes in elbow joint angles only [52]. Systems using EMG with the arm in varying positions can use these changes in signal as control inputs or can be designed to be robust and not affected by these variable signals.

In testing algorithms used to classify motion types or control devices, sets of data are used for training the system, and separate sets of data are used for testing the system. When data are collected from limbs in various positions, systems have been trained with data from one position or a combination of positions. If systems were trained with a data set from the arm in a single position, intra-position testing can describe when testing data are from the same position as training data, and inter-position testing can describe when testing data are from a position different from the position used in training data [54].

It has been found that errors in classifying forearm/wrist/hand motions using EMG signals from an electrode band around the forearm depended on limb postures (angles of joints not primarily moved by the muscles being measured) [49]. When linear discriminant analysis (LDA) classifiers were trained with four time domain surface EMG features to classify wrist/hand motions with the arm (shoulder and elbow angles) in different postures, the errors of classifiers trained in one position and tested classifying motions in the same arm position were lower than classification errors when a model trained with data from one arm position was tested with data from the other arm positions [49].

Another study observed effects of arm position, intra- and inter-position training, and subject type (people with or without amputations) on the accuracy of multi-layer perceptron artificial neural networks with EMG features as inputs and hand/wrist angles as outputs [54]. In this study, EMG and kinematic data were collected with the arm in three positions involving different elbow angles and shoulder adduction–abduction [54]. It was found that there was a significant difference between the artificial neural network performance measure for intra- and inter-position training/testing, with intra-position testing being more accurate [54]. This means that an artificial neural network classifying kinematics of wrist flexion–extension, wrist radial–ulnar deviation, and wrist pronation–supination did not work as well when using EMG data with the arm in a different position than the arm was in during training.

To counteract the decrease in accuracy with inter-position training and testing, it was found that if training data were pooled together from multiple positions instead of one position, the artificial neural network performance improved compared to inter-position training/testing performance [54]. The optimal number and types of arm positions that provide the best EMG data for training of motion classification algorithms requires further investigation. For a linear discriminant analysis (LDA) classifier, classification accuracy was better for classifiers trained with data from positions with the elbow at multiple angles compared to training data with the arm in multiple positions but only a flexed or extended elbow, not variations in elbow angle [49]. It was also shown that performance generally improved with an increase in the number of positions used in training, although there was also variation in performance with different combinations of positions used in training, and the amount of improvement with additional positions decreased as the number of positions included increased [49]. With the very large variation in dynamic movement of the upper limb, a very large training data set would be required to sample a range of positions of human movements. This would make training duration longer and require more repetitions of movements performed by the user of the system. Therefore, determining the best combination of and number of positions to use in training to balance the effort and time of training with the accuracy and usability improvements is desirable.

2.5.2 Resistance Force

In addition to arm position, interaction forces affect motion and EMG signals. In one study, classification error of an LDA model predicting hand actions based on EMG signals, increased by approximately 32 % when forces were introduced [35]. Activated muscles apply forces to joints causing movement or stabilization during isometric contractions. Increased recruitment of motor units, and increased firing rates of those motor units produces force [47]. The level of activation measured through sEMG can be related to force output, with higher sEMG signal amplitudes generally related to higher levels of force output [18]. However, this relationship is not always linear above force thresholds, or the force-sEMG relationship has a more parabolic shape in some muscles, for example muscles controlling finger movement [47]. In very controlled isometric contractions of the biceps, sEMG has been related non-linearly to force output at the wrist as well [18].

External forces acting on a joint during movement can cause a torque in the same direction of the joint rotation, assisting the main flexors or extensors causing the motion, or the external forces can oppose the joint motion, causing a torque acting in the opposite direction of the intended joint motion. An example of an opposition force would be lifting a load held in the hand by flexing the elbow, with the arm initially straight and upper arm held against the torso. In this case, the prime flexors, biceps and brachioradialis, are working against the added load. An example of an assisting force would be extending the arm from an initially bent (flexed) position with a load held in the hand and upper arm stationary against the torso. In this case, the added load is adding torque on the elbow joint in the same direction that the prime extensor, the triceps, is applying torque. As well, this may increase the amount of support or control the flexors may need to provide to support the added load in the extension movement.

During activities of daily living, varying levels of external forces can assist and oppose joint motion, as well as act on the limb in directions causing torques not aligned with the axis of rotation of the joint. Different loading on joints, as well as loading when the muscles are actively moving a joint versus when used for fine-tuning control, can cause changes in muscle activation patterns. The biceps assisting in controlling acceleration during elbow extension motions is an example of a fine-tuning role of a muscle (biceps), as the biceps is not a prime mover for elbow extension. For example, when performing elbow flexion against an external load the biceps and brachioradialis had similar muscle activation during sets of different joint angles, velocities, and loads, but during active elbow extension (fine-tuning roles) the biceps and brachioradialis had different muscle activations from each other in some sets [52]. In load bearing roles compared to fine tuning roles, the biceps and brachoiradialis were activated in different combinations during motions [52]. Changes in EMG signals with motion type is consistent with findings of muscle activations higher during concentric motions, lower during isometric contractions, and lowest during eccentric motions with constant force values [47]. Another example of the force-EMG relationship changing with motion is the force-EMG relationship was shown to change with changes in elbow joint angle [47].

Changes in muscle activation patterns measured by sEMG during upper limb movements producing different forces measured externally were distinct enough to calibrate parallel cascade identification modeling to estimate force [55]. Understanding the relationship between sEMG signals and generated force can be used to predict intended force based on EMG signals, and control devices.

For example, the efficacy of using bilateral mirrored training programs to predict intended output forces at the wrist during motion in two DOFs on a amputated limb has been studied [27]. In this case, forces measured at one wrist were used to train a multilayer perceptron artificial neural network to take sEMG signals from seven arm muscles on the ipsilateral (same) or contralateral arm as inputs and output intended force values [27]. The relationship between patterns in sEMG features and output forces could then be used to train with ipsilateral or contralateral measurements and control devices to determine and produce desired force levels.

A challenge when relating EMG to forces is accurately measuring the force outputs without interfering with motion. A pulley device has been used to apply constant loads to participants' hands in either direction, resisting elbow flexion or extension [52]. As well, a 1 DOF exoskeleton has been used to apply torques to the elbow joint and measure forces acting at the wrist joint with a 6 DOF force/torque sensor [55]. Though these devices could apply or measure force, the devices were limited to 1 DOF and more complex arm movements (shoulder rotations) were not permitted, the hand could only be moved in one plane. Another example of limited force measurements is the InMotion 2 (Interactive Motion Technologies, Watertown, MA) planar horizontal robot with a 6 DOF force sensor attached [28]. It was also found that by attaching a 6 DOF force sensor as the end effector on a KUKA robot arm (KUKA, Germany), a user could push against the robot during movements and receive feedback as forces are measured [56]. In contrast, this method with the KUKA robot arm was able to provide force measurements during movements with the hand following more complex paths.

2.5.3 Velocity

In addition to arm position and force, varying motion activation patterns can be related to varying joint rotation velocities. Muscle activation of the biceps and brachioradialis has been observed to increase with increasing velocities during elbow flexion [52]. However, during fine-tuning tasks (extension of the elbow), muscle activation of the biceps decreased with increasing velocities, while the brachioradialis mean normalized sEMG envelope increased with increasing angular velocity [52]. This highlights that the effects of velocity on muscle activation can depend on the muscle and type of motion. Other studies note the possible impact of velocity on muscle behaviour, but then hold velocity constant while studying EMG signals during elbow flexion–extension movements [50]. Root mean square error of a "parallel cascade identification model" estimating forces at the wrist based on EMG inputs, increased from 8.3 %, when forces and velocities were not varied, to 33.3 %, when variation in forces and velocities were introduced [55].

2.5.4 Fatigue

Furthermore, the effects of fatigue on muscle activation are not completely understood. However, it has been observed that when performing isometric contractions and maintaining a specified force, the amplitude of an sEMG signal of a muscle can increase and signal power shifts to the lower end of the spectrum [18, 47]. In studying EMG, rest periods are commonly given during trials between contractions, and motions are performed in randomized orders to minimize effects of fatigue. For example, in one study, 60 s rest periods were given between 45 s duration contraction measurement periods to reduce the effects of fatigue [52], but the reasoning behind why these durations were chosen is unclear. In situations where a set of contractions are performed to train systems for control of devices, rest times for fatigue avoidance can greatly increase training periods. In training an artificial neural network for prosthesis control, 5 minutes of rest allotted between 25 second contractions to avoid fatigue was presented as a limitation [27].

2.5.5 Training Protocol

In many experiments focused on the design and testing of myoelectric controlled devices, the EMG pattern recognition classifiers are being trained for long periods of time in a very controlled lab setting. In daily activities, the body is not constrained in the same way, with factors affecting the EMG signals and intended movements not matching the movement profiles used in the training period.

Development of improved training protocols is being studied to make pattern recognition control systems more generalizable to arm movements outside of laboratory settings [37]. It has been found that involving data from dynamic portions of muscle contractions (instead of only static portions) in the training protocol of classifiers, improved the accuracy of LDA and SVM classifiers used [48].

Another challenge in EMG control is exposure to external forces, involuntary muscle activations, and the after effects of changes in EMG signals in response to removal of dynamic external forces [57]. To account for external forces and varying levels of muscle contractions, it was found that SVM models trained with data from dynamic arm positions and dynamic levels of muscle contractions performed better at classifying finger motions under a variety of conditions (static versus dynamic arm positions and contraction levels, and external disturbance forces) than classifiers trained with data only from static postures and contraction levels [37].

Another suggestion is to incorporate other data, such as signals from accelerometers, with the surface EMG signals in the training and use of classifiers [51]. Using a combination of kinematic and EMG signals in prosthesis control was demonstrated in a simulated virtual reality environment [58].

In this chapter, the motivation for investigating effects of motion characteristics on EMG signals and motion classification were reviewed. A literature review of methods of rehabilitation and functional assessment, types of arm motion and effects on EMG signals, and the uses of these EMG signals in motion classification have been presented.

Chapter 3

Design of Experiments

It was found that a variety of motion factors can influence EMG signals. Also, variations in EMG signals introduce difficulties in using these signals as inputs to classification models and control systems identifying intended motions and controlling wearable devices. An experiment was designed to investigate the effects of motion characteristics on EMG signals to improve use of EMG signals. The methods of the experiment designed along with key measurements collected are described as follows.

3.1 Methods

The three main movement factors being observed were: arm position, resistance force, and velocity. The experiments were organized in a factorial design with the goal of collecting EMG and kinematic data of arm movements. Arm motions were divided into three categories: isometric, single elbow flexion–extension motions, and more complex activities of daily living (ADLs), as explained in Section 3.1.1. All three types of motion were included to permit investigation of differences in EMG levels and factor interaction across the motion types. In one session, participants completed all three sets of tests: isometric contractions, single motion, and activities of daily living.

Each motion factor was varied between two or three levels as arm movements were performed. The arm position factor consisted of three levels: position 1 (P1), position 2 (P2), and position 3 (P3), as described in Section 3.1.2.1. Arm position was not specified during ADLs, as the shoulder orientation could not be constrained as the motions simulated performing tasks. Force values and directions changed between three levels for isometric contractions and elbow flexion–extension motions, and two levels during ADLs. The force levels are presented in Section 3.1.2.2. Velocity was split into three levels: stationary, slow, and fast. Velocity details are explored in Section 3.1.2.3. During isometric contractions, the joint angle does not change, therefore, the arm was held stationary with no variation in velocity. The corresponding elbow flexion–extensions varied between slow and fast speeds. During ADLs, velocity could only be varied between two levels, slow or fast. Stationary isometric contractions would not have permitted the completion of ADL tasks. Table 3.1 displays the factor variation for the three motion types.

Table 3.1: Factor variation for motion tests.

	Motion sets		
Factors	Isometric	Single Motion	Activities of Daily Living
Position	X	X	—
Force	X	X	X
Speed	—	X	X

Overall, the combined isometric contractions and flexion–extension motions resulted in a 3 by 3 repeated measures design, with three factors (position, force, velocity) varying between three levels each. As well, the ADL motions were performed as part of a 2 by 2 repeated measures design with two factors (force and velocity) varying between two levels each. The motion sets and factor levels are described further in the following section.

3.1.1 Motion Sets

As described in Chapter 2 Section 2.2, motion can be divided into isometric contractions and dynamic movements with isotonic muscle contractions. Therefore, in this study, EMG signals were measured during isometric contractions and dynamic movements. The dynamic movements were divided into simple elbow flexion–extension, and more complex activities of daily living (ADLs).

3.1.1.1 Isometric Contractions

During isometric contractions, the participants were expected to hold their arm still. The elbow angle did not change during the contraction, however, separate isometric contractions were held with the elbow fully extended, or the elbow flexed 90°. These contractions were held with the arm in three different positions (shoulder orientations), and three different forces were applied to the hand.

3.1.1.2 Single Motions

The first type of dynamic motions was simple elbow flexion–extension. In these motion trials, the arm was held in the starting position with the elbow fully extended, the elbow joint was rotated to 90° flexion, then extended again. One repetition consisted of the full movement from extended elbow, to flexed elbow, and return to extended elbow. These flexion–extension movements expanded on the isometric contractions, by being performed with the arm held in the three corresponding arm positions (shoulder orientations), three force levels applied to the hand, and at two velocities (slow, fast).

3.1.1.3 Activities of Daily Living

Upper extremities consist of multiple joints (shoulder, elbows, wrist) with various degrees of freedom. Arms are involved in many different activities throughout the day, moving through a wide range of motion. To consider more scenarios, activities of daily living were tested.

As presented in Chapter 2 Section 2.1.1, various sets of activities of daily living, or other movements are performed for the assessment of upper extremity kinematics, dynamics, and functionality. The specific activities of daily living included can vary. In particular, due to the elbow being the joint of interest in this study, motions that produced large variations in elbow flexion–extension were of interest.

The following two activities of daily living were selected as a sample of arm movements to measure:

1. Lowering and raising arm above horizontal (reaching above shoulder level in front of body)

2. Hand to mouth (simulating eating and drinking)

During performance of these motions, resistance force was varied between two levels and the velocity at which the motion was performed at varied between two levels. The levels of the motion factors are described further in the following section.

3.1.2 Factors

The three main movement factors being observed were arm position, resistance force, and velocity. These factors were varied through multiple levels in multiple combinations during the described motion trial movements, while muscle activation was measured, and kinematic information was collected.

3.1.2.1 Arm Position

For isometric measurements, and moving the arm through single flexion–extension motions, the orientation of the upper arm was held in three different positions. The shoulder and torso were not physically constrained which allowed for some movement of the upper arm to occur naturally, regardless of instructing participants to remain stationary. This was reflective of how motions are comfortably performed during daily activities. The arm positions are displayed in Table 3.2 and Figure 3.1.

Table 3.2: Arm orientations

Position	**Description**	**Shoulder Angles**
P1	Arm down along torso	0° abduction, 0° flexion
P2	Arm horizontal, stretched forwards	90° flexion
P3	Arm horizontal, stretched to side	90° abduction

During isometric contractions in P1, the elbow was fully extended and also held still at 90° flexion, as if the flexion–extension movement was paused. These joint angles are demonstrated in Figure 3.2. These data were gathered to determine baseline measurements.

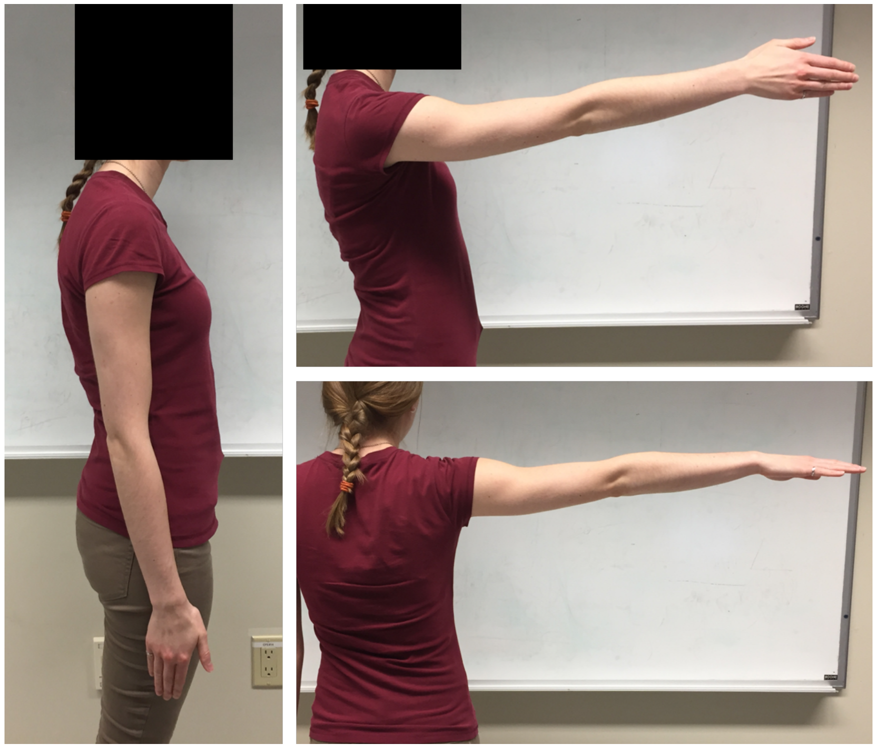

Figure 3.1: Arm positions: P1 (*Left*), P2 (*Top right*), and P3 (*Bottom right*).

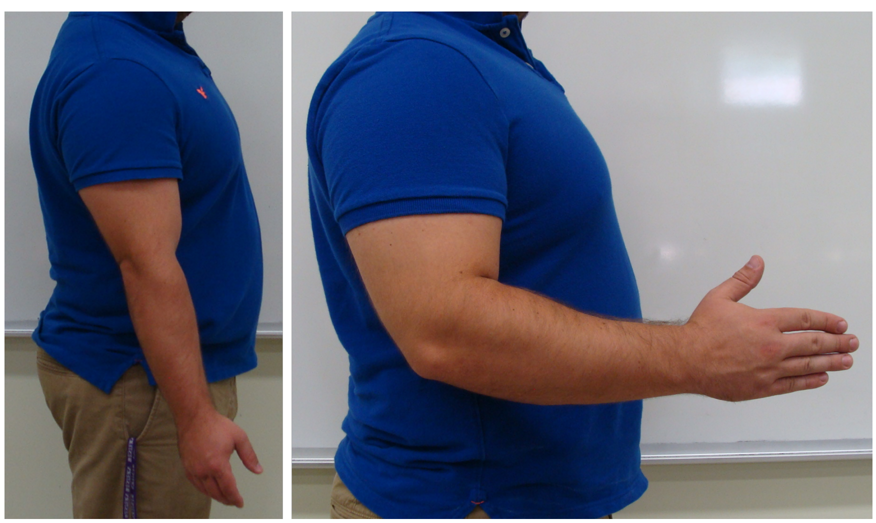

Figure 3.2: Arm with elbow extended (*Left*), and elbow flexed by 90° (*Right*).

3.1.2.2 Resistance Force

The participants grasped a handle end effector of the collaborative robot while performing motions, as described in Chapter 4. Forces were applied to the participants' hands through the handle, and the contact forces transmitted from the participant's hand to the handle were measured. The three force levels during isometric contractions and elbow flexion–extension were: 0 N, 22 N in the direction resisting elbow flexion, and 22 N resisting elbow extension. 22 N was chosen to represent the weight felt to lift objects such as a bag of potatoes or textbooks. During activities of daily living, two force levels (11 N and 22 N) were applied to the participant's hand. The 11 N or 22 N forces were applied directly downwards to simulate the force of gravity acting on objects a person may carry.

3.1.2.3 Velocity

Velocity is a factor affecting muscle activation. Movements in this experiment were performed at three different velocities: 0°/sec during isometric contractions, a slow quasi-static speed (approximately 11°/sec), and a faster speed (approximately 23°/sec). Participants were guided to perform stationary isometric contractions, then elbow flexion–extension and two activity of daily living motions at the two different speeds. In order to perform the motions at two different speeds, participants were instructed to perform the slow trials in about 8 seconds (duration from full extension to 90° elbow flexion), and complete the motion segment in about 4 seconds for the faster speed. For the faster speeds participants did not move as fast as possible because motions were to be executed in a controlled manner.

3.1.3 Constant Experiment Elements

While the motion characteristics of interest were varied, other aspects of the experiment were held constant to reduce the introduction of variables. The constant experimental elements were the protocol for measuring maximum voluntary contractions, hand positions, and breaks between motions to mitigate effects of fatigue. However, since movements were unconstrained, some variation in arm positions and speeds did exist and were not eliminated. Small variations were seen as acceptable because these data are being used to work towards control of a device during unconstrained movements during daily living with interactions with the environment, such as lifting objects. The ways in which maximum voluntary contraction tests and hand positions were held constant, and fatigue effects mitigated are described in the following sections.

3.1.3.1 Maximum Voluntary Contraction

Studies use various strategies to conduct the measurement of EMG signals during the maximum voluntary contraction (MVC) of muscles. The purpose of measuring EMG signals during maximum contraction is the EMG signals during other motions can then be normalized, allowing for comparisons of EMG patterns between subjects, not only within subjects. The duration that contractions were held for during measurement, the style of contraction such as slowly ramping up to maximum muscle contractions before holding versus only holding maximum contractions, duration of rest periods in between contractions to avoid fatigue, and the number of repetitions varied in the literature. MVC was not measured in every study. Usually, MVC was measured if EMG data from various movements or contractions were going to be normalized to the MVC EMG signals in order to make fair comparisons between study subject and muscle group EMG values.

In this work, MVC was measured by holding the upper arm against the torso with the elbow flexed 90° while the hand gripped the handle of a stationary robot. The participants maximally contracted the arm for one trial, attempting to flex the elbow (raise the hand), and a second trial, attempting to extend the elbow (lower the hand), each for a 5 second duration. The robot was stiff, resisting movement. Surface EMG and force measurements were recorded as the MVC values. The measurement of MVCs was completed at the start of the measurement session for each subject, first with isometric elbow flexion and second with isometric elbow extension.

3.1.3.2 Hand Position

During movements, forearm position or rotation was held constant in a neutral position. However, a wrist brace was not worn by participants in order to constrict movements. The participants were merely instructed to hold their forearm and wrist in a constant neutral position. The forearm and wrist were held in a neutral orientation to limit changes in mechanics with angle and line of activation for muscles involved in elbow flexion–extension. As well, muscles involved in elbow flexion–extension can also be involved in forearm pronation–supination. With the forearm in a constant neutral position, changes in activation levels of these muscles should have stayed representative or related mainly to elbow flexion–extension.

Also, during all of the motions, the participants were holding onto a handle. The same handle was held during the 38 motion trials and MVC trials. During motions outside of a laboratory while interacting with the environment, people would not always be holding items of the same shape or orientation. However, for this study, the grip force and orientation was not a motion factor of interest, therefore, the handle interface was held constant throughout the trials.

3.1.3.3 Fatigue

Three repetitions of each trial were performed. To prevent extreme muscle fatigue and discomfort due to overworked muscles, rest periods were given between each repetition, and between each set. Ten seconds of rest were given between each repetition, and approximately 1 minute of rest was given between motion sets. Participants mentioned some tiring of muscles during motion sets with the arm at 90° shoulder abduction, however they were able to complete the tasks with adequate recovery during rest times.

3.2 Measurements

In order to collect the required force, kinematic, and EMG data, various systems were required. To transmit controlled forces to participants' hands, a handle interface was designed. As EMG signals were the main measurements of interest, particular arm and shoulder muscles of interest were selected. The handle interface through which forces were applied to participants' hands and the key muscles of interest are described in the following sections.

3.2.1 Force

In order to track hand motion, provide desired and measurable stiffness, and measure forces as participants performed motions interacting with a physical environment, participants were instructed to hold onto a handle attached to a robot while performing motions. The robotic equipment selection and set-up is described in detail in Chapter 4, while the handle design is presented in the next section.

3.2.1.1 Participant Interface

A handle end effector was designed as an interface between the user and equipment applying forces. The requirements of the handle interface were as follows:

1. Must attach easily to the equipment applying forces (robot flange).
2. Must have a comfortable/ergonomic grip diameter.
3. Must not have moving parts. No moving parts while performing motions was important, as movable parts would change the characteristics of the end effector, which would make the robot-calculated force measurements inaccurate.

A straight handle extending from the robot flange was selected. The handle was to be gripped with the long axis perpendicular to the forearm with the wrist and forearm in neutral positions, during stationary and single motion trials. During the activities of daily living, the handle remained vertical, perpendicular to the ground, simulating lifting a cup without it being tipped. This decision was justified by previous work, in which it was found that people with dexterity disabilities, such as arthritis, performed better at lifting small weights on a device with a vertical handle, as opposed to a horizontal handle [59].

With the handle orientation decided, the grip diameter was determined. Hand size and the size of items grasped affect hand grip strength, so a diameter within the range in which people can perform high grip strength activities was chosen [60]. For a study of people without dexterity disabilities, the mean grasping diameter of the hand (maximum bending diameter of the hand with the thumb and middle finger just touching while grasping an object) was 40.42 mm, with a range of 26.93 mm for the 5th percentile to 46.31 mm for the 95th percentile [59]. These grasping diameters reflected the maximum diameter of objects people were able to grasp with their thumb and middle finger touching, not necessarily ergonomic object sizes. For stability in a power grip, the Canadian Centre for Occupational Health and Safety (CCOHS) recommended a handle diameter range of 30–50 mm, specifically 40 mm [61].

For the first prototype shown in Figure 3.3, a handle diameter of 40 mm was chosen to fit within the recommended ranges. To prevent slippage, the diameter of the ends was widened. For comfort and grip, the stiff plastic material of the handle was coated with a softer, more rubbery material. A full set of pilot trials was completed with two participants with this first prototype. Subject one (S1) had no complaints about the handle and completed all of the tests. The second subject (S2) commented on minor thumb soreness where the thumb was in contact and moving against the handle. S2 remarked that the grip diameter felt large. In response to this feedback, a second prototype was designed, shown in Figure 3.3. The second prototype had a grip diameter of 30 mm, which was at the lower end of the recommended range of handle diameters for power grips [61]. 30 mm was also the measured diameter of the grip portion of a dumbbell, an exercise weight that people with varied hand sizes grip and lift. Since participants were performing motions similar to dumbbell exercises, a handle similar to the size of a dumbbell handle was reasonable to use.

Figure 3.3: First prototype (*Left*), and final design (*Right*) of a handle interface.

The handle length was selected to accommodate the common hand breadth sizes noted from available anthropometric data. In one report, mean hand breadth sizes of the right and left hand were reported as 90.5 mm and 89.9 mm, respectively, with a maximum breadth of 115.9 mm for the right hand and 115.5 mm for the left hand [59]. Work presented in [62] recommended a handle length between 100 mm and 150 mm to accommodate hand breadth. A minimum handle length of 100 mm was recommended by the CCOHS to prevent compression in the palm due to a handle not spanning the breadth of the hand [61]. A handle length of 140 mm was selected to ensure enough room for the hand.

The handle was modeled in Solidworks (Dassault Systèmes, USA), and then 3D printed (Polyjet, Stratasys, USA) with a plastic material. The grasping surface of the handle was coated with a softer rubber material during printing to provide more grip. The handle was securely attached to the robot flange with four M6 screws. With the handle designed and fabricated, motions trials could then be performed with forces being applied to participants' hands via the handle.

3.2.2 Muscles Measured

During trials for all motion types, sEMG measurements were recorded using the Trigno wireless EMG sensors (Delsys, USA). This system had 16 channels with wireless electrodes that adhered to the skin surface above the belly of the muscle with sticky tape. Each of these electrodes measured the muscle activity and had a three DOF accelerometer.

The prime elbow flexion–extension muscles were selected for measurement, as well as other muscles in the arm and shoulder area. In particular, shoulder muscles involved with shoulder abduction and flexion (raising the arm) were included because the effect of arm posture was one of the main factors being studied. The forearm was held in a neutral position during most tasks. Effects of pronation and supination of the forearm were not investigated in detail, as the arm was not actively pronating or supinating. However, muscles involved in forearm rotation can stabilize the forearm and are sometimes involved in elbow flexion–extension as well, such as the brachioradialis. Therefore, sEMG signals were collected from selected forearm muscles. The forearm muscles measured included wrist flexors and extensors. Table 3.3 lists the selected muscles for which EMG measurements were collected.

3.3 Conclusion

This chapter outlined the experimental design of this study. It described how motion factors (arm position, interaction forces, and velocities) were being varied during isometric contractions, elbow flexion–extension, and ADLs. Types of data collected were also introduced. The specific collection systems selected are described further in the following chapter, along with how the equipment was

Table 3.3: Upper limb muscles measured.

Channel	**Muscle**	**Acronym**	**Function**
1	biceps brachii short head	BB_S	flexor of elbow, forearm supinator, involved in flexing shoulder
2	biceps brachii long head	BB_L	flexor of elbow, supinator
3	brachialis	BRA	flexor of elbow
4	brachioradialis	BRD	flexor of elbow, pronator
5	triceps brachii long head	TRI_LO	elbow extension
6	triceps brachii lateral head	TRI_LAT	elbow extension
7	triceps medial head	TRI_M	elbow extension
8	pronator teres	PT	elbow extension, forearm pronation
9	infraspinatus	ISPI	shoulder rotation, stabilizer in rotator cuff
10	anterior deltoid	AD	shoulder vertical and horizontal flexion, shoulder rotation
11	lateral deltoid	LD	shoulder abduction
12	posterior deltoid	PD	shoulder vertical and horizontal extension, shoulder rotation
13	extensor carpi ulnaris	ECU	wrist extension
14	extensor carpi radialis	ECR	wrist extension
15	flexor carpi ulnaris	FCU	wrist flexion
16	flexor carpi radialis	FCR	wrist flexion

calibrated to run the designed experimental protocol.

Chapter 4

Equipment Set-Up

Chapter 3 discussed the experimental design. Three main types of data were collected for these experiments including body kinematics, EMG, and force. The measurements of these motion characteristics were divided between three systems: a Microsoft Kinect motion sensor, Trigno Wireless EMG Sensors, and a KUKA robot. This chapter describes the equipment features and how the systems were used to conduct the experimental protocol.

4.1 Kinect

In addition to the dynamic and EMG data collected, availability of kinematic data was also desired to potentially relate the muscle activity to the actions being performed better. The KUKA robot was able to provide position data of the robot itself, and by extension, the position of the participants' hands as they held the robot handle, as described in Section 4.3. The position information along with timestamps could be used to calculate speeds as well. However, using the robot, only the position of the users' hand was known, losing any other information about the users' body. Criteria for a suitable motion tracking system are listed below.

1. The physical set-up, such as markers, must not interfere with EMG collection.

2. Metal components must not interfere with the motion tracker.

3. The sensor must accommodate the movement envelope of participants (elbow flexion–extension

while standing with the arm in various orientations, and selected ADLs).

4. Provide elbow position and angle measurements.

5. Capable of saving kinematic data to a .txt or .csv file format suitable for further processing offline.

6. Provide timestamps.

Magnetic tracking was not used because of metal interference with robotic equipment being used and the large range of motion capture required. Optical trackers without markers were systems of interest to reduce calibration and equipment set-up time with each subject. A Microsoft Kinect motion tracking system (Microsoft, USA) was chosen to provide additional kinematic data as it fulfilled the requirements identified. The Kinect sensor is shown in Figure 4.1, sitting above a computer monitor. The Kinect tracked motion activity by visually detecting the joints of the person in the view range. An application written in C# was used to acquire the joint position data of each of the joints available in the Kinect body tracking, then record these position points in a text file, to be imported into MATLAB at a later time.

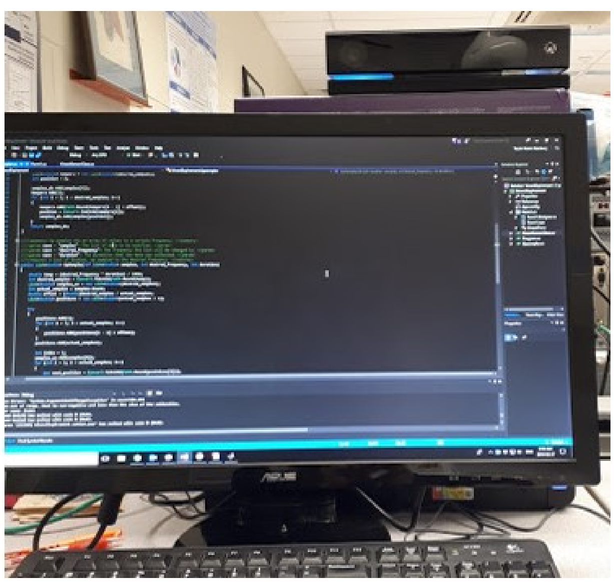

Figure 4.1: Experimental set-up with Kinect sensor located above a computer monitor used during data collection.

4.2 Trigno

Collection of EMG signal measurements was required for this experiment, in addition to kinematic and dynamic data of motions. Sensors were needed to measure EMG signals from the surface of the skin. To ensure the sensors would be usable for the experiment, key requirements of the EMG system were identified. The key criteria for the EMG sensors are listed below.

1. The EMG sensors must not interfere with movement, they must not constrict natural arm movement.
2. Accommodate the movement envelope of a person standing and performing elbow flexion–extension motions and ADLs of interest.
3. Sensors must have a range of at least approximately 3 m, to suit the testing configuration with the other equipment.
4. 16 EMG channels are needed to be collected simultaneously.
5. The sensors must attach to the surface of the skin, they must not be invasive.
6. Capable of saving EMG data to a .csv file format suitable for further processing offline.
7. Provide timestamps.

Based on this criteria, the Trigno wireless surface EMG sensors (Delsys, USA) were chosen to collect the EMG data. This system met the outlined requirements and had additional beneficial features. These EMG sensors could be charged, and then used continuously for multiple hours. The wireless sensors were charged before each use, and had a working range of 20 m [63]. The sensors adhered to the surface of the skin above the main bulky area of the muscles of interest with sticky tape, as shown in Figure 4.2. Each sensor provided 1 of 16 EMG channels, and 3 degree-of-freedom accelerometer measurements. The proprietary software provided by Delsys, EMGworks Acquisition, was utilized to collect and save the data. Afterwards, using the analysis software, EMGworks Analysis, the raw files were converted into .csv files to be easily accessed using MATLAB.

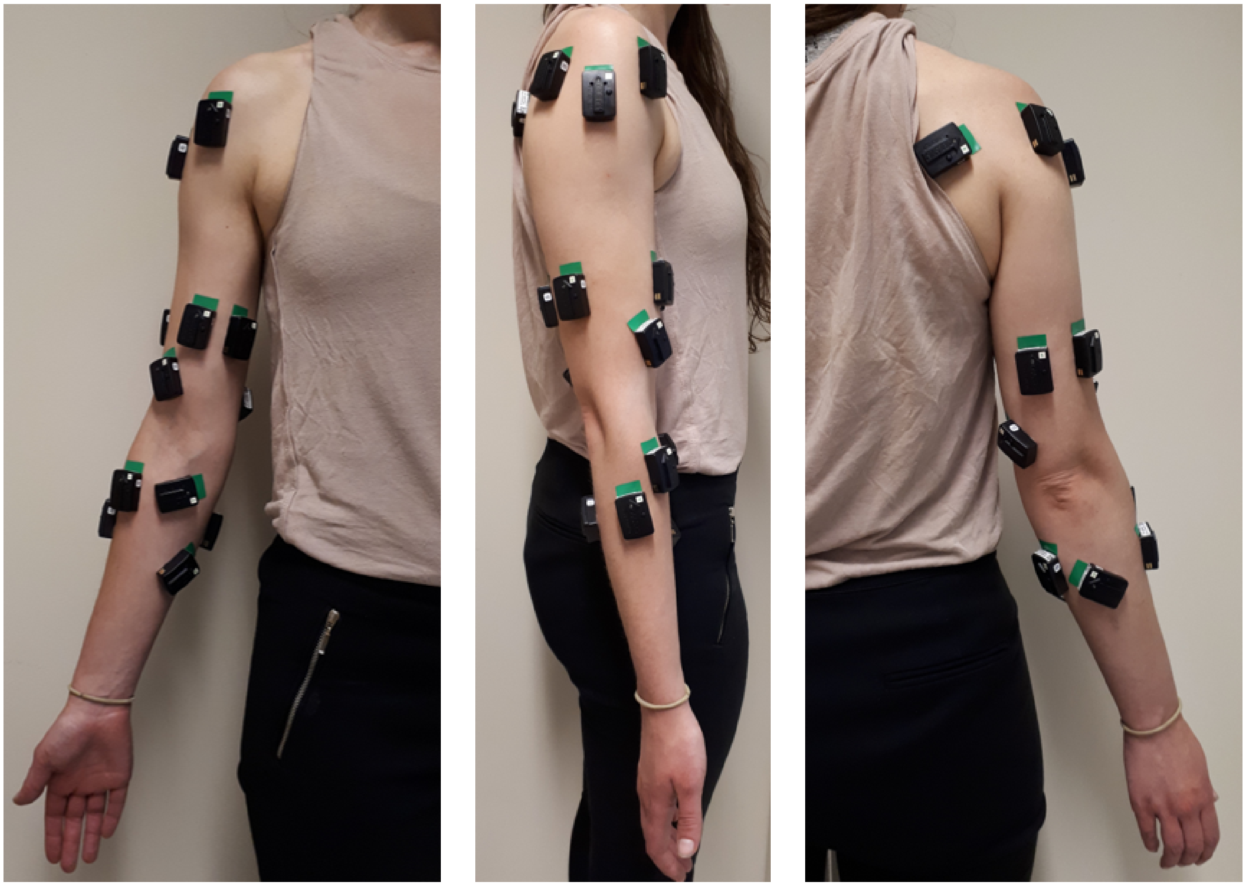

Figure 4.2: Trigno EMG sensors attached to the dominant arm over 16 muscles of interest.

4.3 KUKA lbr iiwa

A main part of these experiments was observation of human movement while interacting with the environment with various force levels. The force between the participant's hand and the environment needed to be manipulated and controlled for each participant. The option to manipulate the direction of the forces in a way that was more complex than simply instructing participants to carry a weight, was required. This was necessary to simulate more diverse interactions between a person and the environment during movements. In order to apply the force levels specified in the experimental design to participants' hands, and measure the contact forces, a list of requirements was drafted for the equipment. The key criteria for the equipment used to apply and measure forces is presented below.

1. Able to manipulate force value and direction, and capable of measuring both.

2. Interaction could be passive, i.e., moved by user, with specified resistance levels.
3. Capable of safely collaborating with humans.
4. Able to accommodate elbow flexion–extension movements and simulate interaction with the environment during activities of daily living.
5. Capable of saving force (value and direction) and position data to a .csv file format suitable for further processing offline.
6. Provide timestamps.

Based on these requirements a KUKA lbr iiwa collaborative robot (KUKA, Germany), shown in Figure 4.3, was chosen to implement and measure force levels during human movements. This robot was capable of safely interacting with humans. The robot has 6 joints with an extra turning flange, providing redundancy as it moves with 6 degrees of freedom (x, y, z translation, a, b, c rotation). Torque sensors in each joint provide torque and force feedback. This force feedback capability allowed for the robot to be passive while moved by a user, or to apply forces during movements with variable stiffness. The stiffness of the robot was controlled in an impedance control mode. In effect, the robot acted like springs with a programmed stiffness attached to the end effector, pulling it back to a set position. Forces in certain directions were also overlaid as motions were performed.

The following sections outline how the robot was set up and controlled during the experimental trials.

4.3.1 KUKA Projects

Robot applications were developed using the KUKA Sunrise.Workbench program installed on a desktop computer (Intel® Core™ i7-6700 CPU 3.4 GHz Desktop running Windows 10). This software acted as a development environment for aspects such as writing programs, setting safety features, configuring robot tools, workpieces and frames, and setting up robot inputs and outputs (I/O).

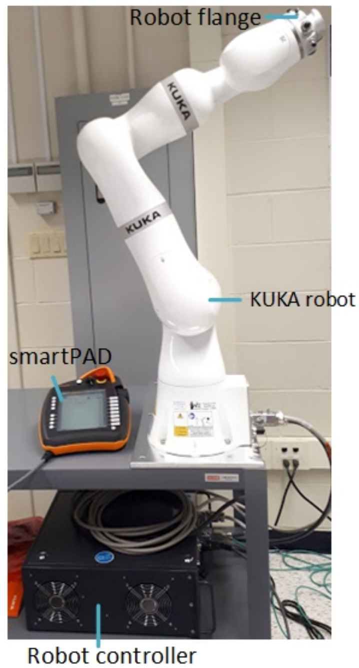

Figure 4.3: KUKA lbr iiwa collaborative robot.

The programs were written in a Java-based language within the KUKA Sunrise.Workbench program taking advantage of many built in KUKA robotics libraries. The project was loaded onto the robot controller for running of the programs during the experimental trials. As robot programs ran, the only way to interface with the robot was through the smartPAD teach pendant human-machine interface, input buttons, and force feedback.

4.3.1.1 Programs

In order to use the robot during the experiments, to apply the desired forces to the participants and log the measurements of interest, two programs were written in the KUKA Sunrise.Workbench program. The two programs and basic functions are listed below.

1. `SetStartPositions`: This program was run by the experimental coordinator, with a trial participant assisting, to set up the starting positions for MVC measurements and the 38 motion trials, but the motions were not yet performed by the participant.

2. `MeasuringForDynamicCalibration`: This program was run by the experimental coordinator

while a participant was interacting with the robot. The robot guided MVC measurements, interacted with the participant during the 38 motion trials by applying desired forces to the participant's hand or acting passively with low stiffness and gravity compensation. During the trials, the robot controller logged position, force, and timestamp data of interest. Position and force measurements were sampled at a rate of 1000 Hz.

The second program, `MeasuringForDynamicCalibration`, required more collaboration between the robot and participant than the first program, `SetStartPositions`. However, in the default robot mode, the robot moves rigidly and is not compliant. A challenge was to have the robot apply a constant force in a chosen direction, while still allowing the participants to move the robot freely and smoothly. The Cartesian Impedance Control Mode and the Cartesian Sine Impedance Control Mode, native to the Sunrise.Workbench programming environment for the KUKA lbr iiwa, were utilized to control stiffnesses of the robot (how the robot resisted forces or was passive) and forces overlaid over movements. Through iterations, parameters of the impedance control modes were calibrated to control how the robot simulated activities and interactions with the environment. A more detailed explanation of the flow of each program, and the parameters of the robot impedance control modes implemented is presented in the following sections.

4.3.1.1.1 Setting Motion Trial Start Positions: `SetStartPositions` Before motion trials could be completed by the participants, the starting positions of the motions were established. No other calibration was required. The robot frames held all of the robot position information used during robot motion commands.

For running the program to set up starting positions (save robot frames), the desired program was selected from the **Applications** menu via the smartPAD. Figure 4.4 shows the pendant screen at the beginning of the application. This is where messages stating the progress of the application and dialog boxes appeared. Other menus could also be accessed from this window.

Figure 4.5 outlines the basic program flow. The darker coloured boxes indicate that input was required from the user, i.e., the experimental coordinator, or coordinator from this point onwards. Input could be transmitted to the robot via the the smartPAD, or buttons on the robot flange. To run the program the coordinator pressed and held both the enabling switch and play button

Figure 4.4: The smartPAD screen view at the beginning of `SetStartPositions` application.

on the smartPAD.

1. As with all programs, initialization was the first step. In this case, variables were defined, the handle tool was attached to the robot, I/O were specified, I/O conditions were defined, and instances of Cartesian Impedance Control Modes were configured. Table 4.1 displays the parameter settings for the control mode used when the robot was commanded to hold a position in this program with a high resistance to movement.

Table 4.1: Cartesian Impedance Control Mode parameters for robot holding positions with resistance to movement.

Cartesian Impedance Control Mode Parameters	**Value**
Translational Stiffness [N/m]	5000 (max)
Rotational Stiffness [Nm/rad]	300 (max)
Additional Control Force [N]	0

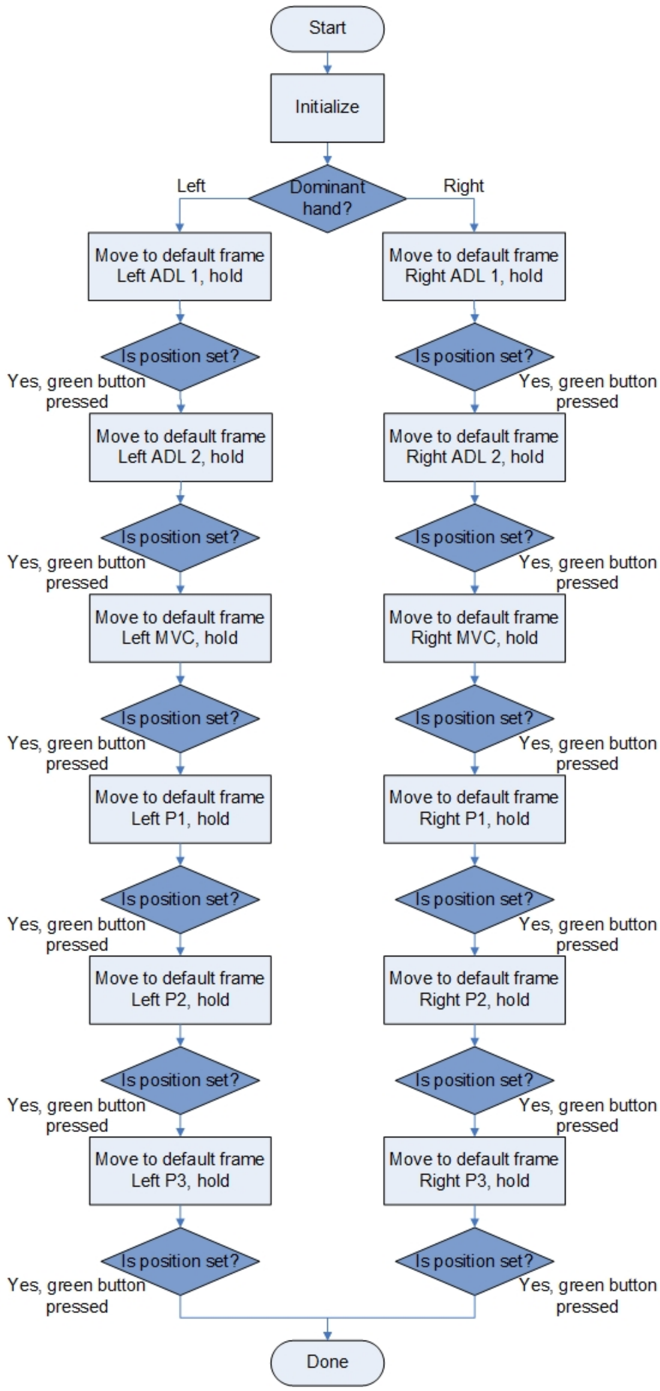

Figure 4.5: Flow of `SetStartPositions` program run on KUKA robot.

2. After initialization, a dialog box appeared on the smartPAD (human-machine interface pendant) allowing for selection of the dominant hand (right or left) of the participant. The coordinator selected the appropriate choice (left hand button or right hand button) via the

touchscreen. This choice determined whether the robot then moved through a series of default start frames for motion trials completed with the participant's left hand or right hand.

3. The robot then moved to the default start frame for ADL 1. When the robot reached this position, the program paused, and the coordinator did not need to continue holding the enabling switch and play button. The participant was instructed to stand with their arm in the desired position. The coordinator could press and hold an enabling switch on the robot flange to put the robot in a hand guiding mode. This mode enabled gravity compensation and put the robot in a passive state where the coordinator could move it with little resistance. The coordinator adjusted the robot to ensure that the handle interface was placed at a proper height and position for the participant, Figure 4.6.

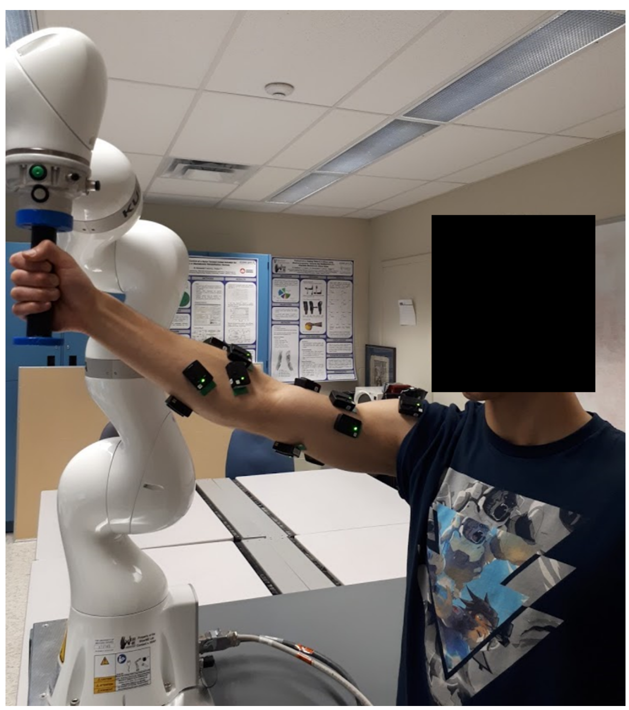

Figure 4.6: The ADL 1 start position for a right-handed participant holding the robot handle interface.

When the robot was adjusted to the desired position, the coordinator updated the corresponding start position frame to the current robot position via the smartPAD. From the home menu of the smartPAD, the coordinator selected the **Frames** menu to access the list of frames, Figure 4.7. The corresponding frame was updated to equal the current robot position. When updating the positions, the reference frame must be correctly set with the "Handle" tool and "Handle centre" as the frame reference.

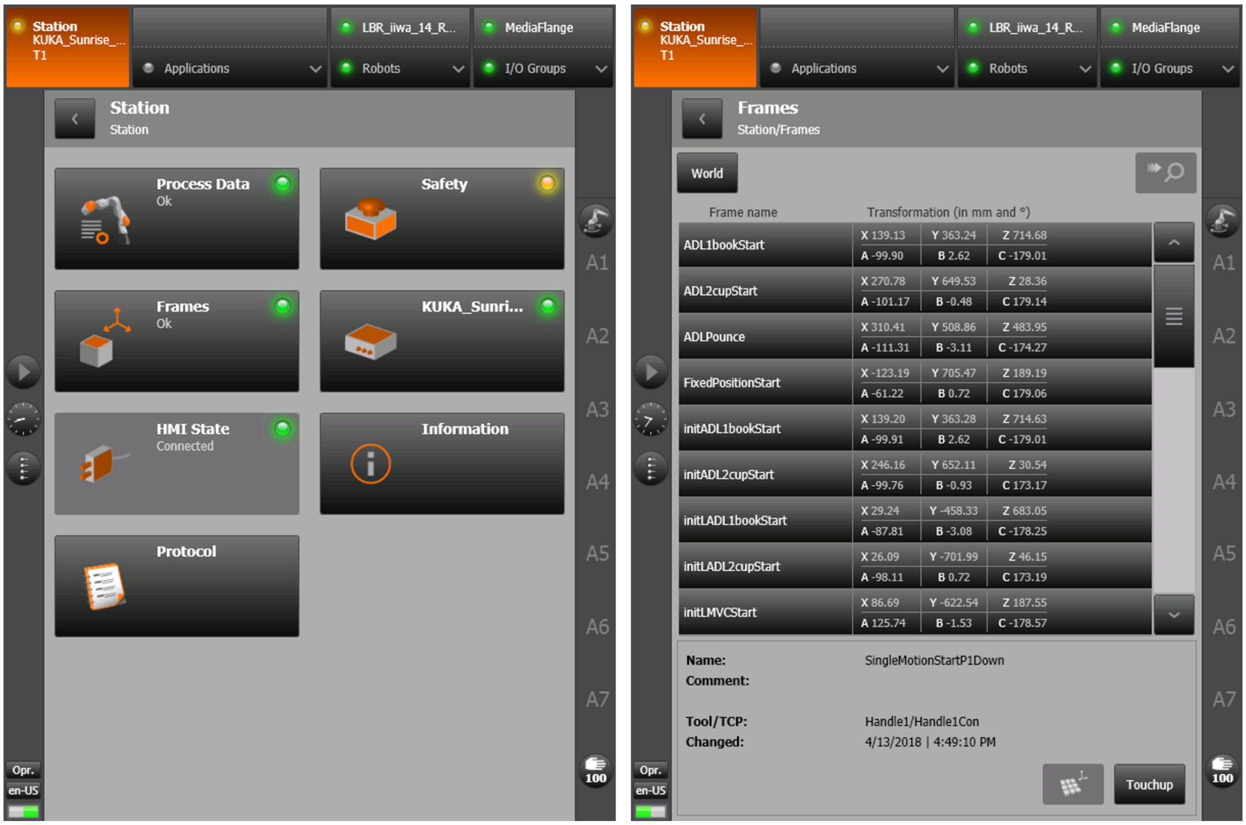

Figure 4.7: The smartPAD home screen (*Left*), and list of frames (*Right*).

4. The robot required the green button on the robot flange to be pressed in order to continue with the program. The green button activation was the condition required to break free from holding its position and to move to the next position. After robot position adjustment and frame updates, the program was started again from the pause position by holding the enabling and play switches, and pressing the green button on the robot flange.

5. The robot then moved to the next default frame (ADL 2, MVC, P1, P2, P3). Again, the robot was adjusted to be in the proper position for the current participant, as shown in Figures 4.8 and 4.9. The default start frames for all motion trial starting positions were configured and saved as part of the robot application prior to participant trials and did not

need to be adjusted while running experiments with participants.

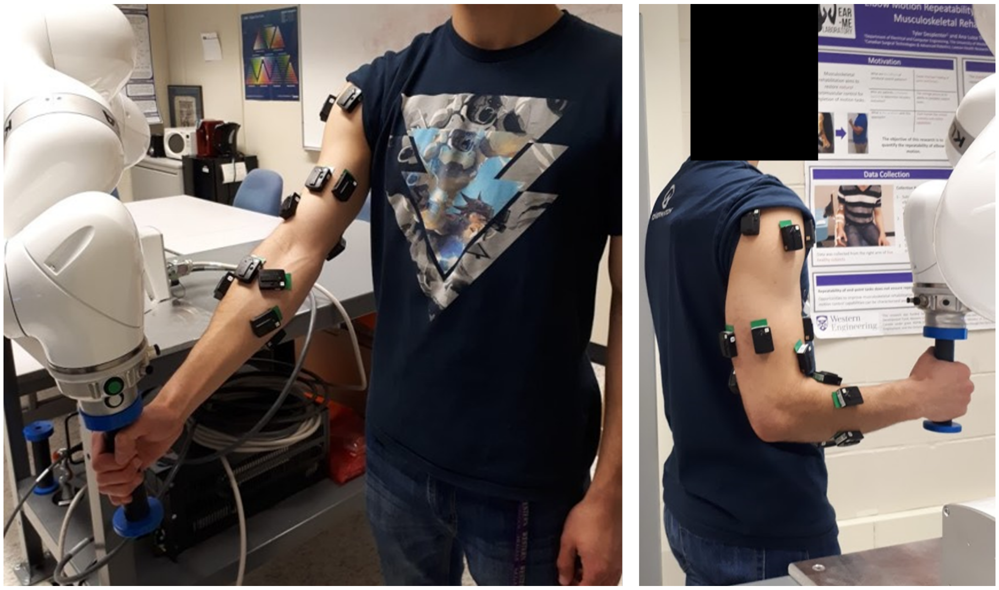

Figure 4.8: The ADL 2 (*Left*) and MVC (*Right*) start positions for a right-handed participant holding the robot handle interface.

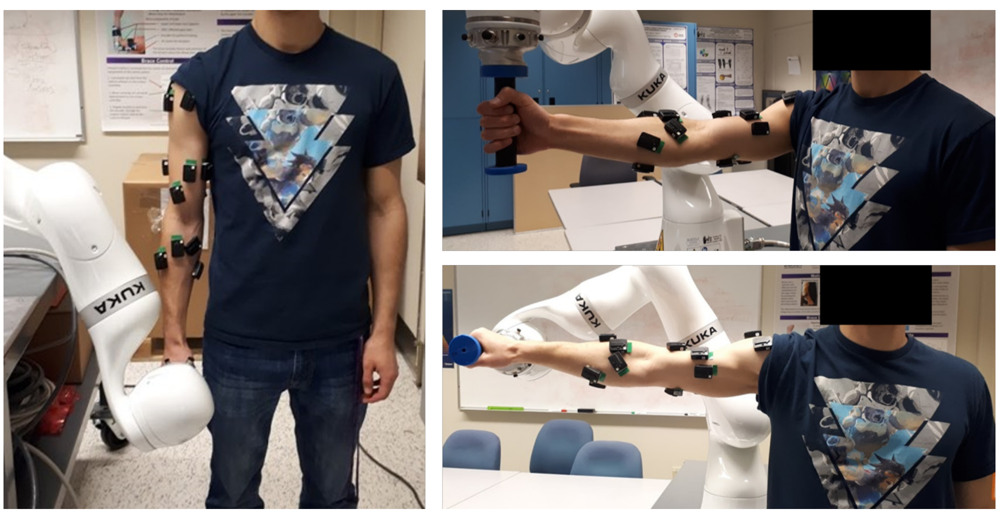

Figure 4.9: A right-handed participant holding the robot handle interface in start positions P1 (*Left*), P2 (*Top Right*), and P3 (*Bottom Right*).

6. Once all of the frames were updated, the application ended. At this point, the frames used for running motion trials were tailored to the current participant.

4.3.1.1.2 Guided Motion Trials, Logging Measurements: `MeasuringForDynamicCalibration`

During this program, the robot applied desired forces through the handle during motions specified by the coordinator via the smartPAD, recorded data, and saved the log files to the robot controller.

Figure 4.10 provides an overview of the robot program used to guide motion trials and log measurements.

This program consisted of the functions (methods) listed below:

1. `main()`

   The robot application was run through this main program. The application was selectable from the **Applications** menu on the smartPAD. Figure 4.11 shows the smartPAD view at the start of the program running. After initialization, this main program prompted input from the coordinator to specify specific tests to be run (MVC and motion trials), updated variables in response to the user input, and ran methods (`doMVC`, `doStatReps`, `doFlexExtReps`, `doADLReps`) to conduct the specified tests. The details of the individual steps are described below.

   (i) Initialization

   Initialization was the first step. At this stage, variables were defined, the handle was attached, and instances of Cartesian Impedance Control Modes and Cartesian Sine Impedance Control Modes were configured.

   To simulate lifting weights, or pushing and pulling against items in the environment while performing activities of daily living, without constraining motions, three Cartesian Sine Impedance Control Modes were configured, as seen in Table 4.2.

   For all the motion test categories (stationary, flexion–extension, activities of daily living), the Null Space Damping was set to 0.7 Nm*s/rad and Null Space Stiffness was set to 50 Nm/rad. This Null Space Stiffness determined how far the robot moved when an external force was applied to the robot. The Null Space Damping parameter determined the oscillation as the robot was deflected from the planned path. Setting a low Null Space Stiffness made the robot compliant in its redundant degree of freedom [64]. In this way, the robot could respond to obstacles in its path during motion. If the robot

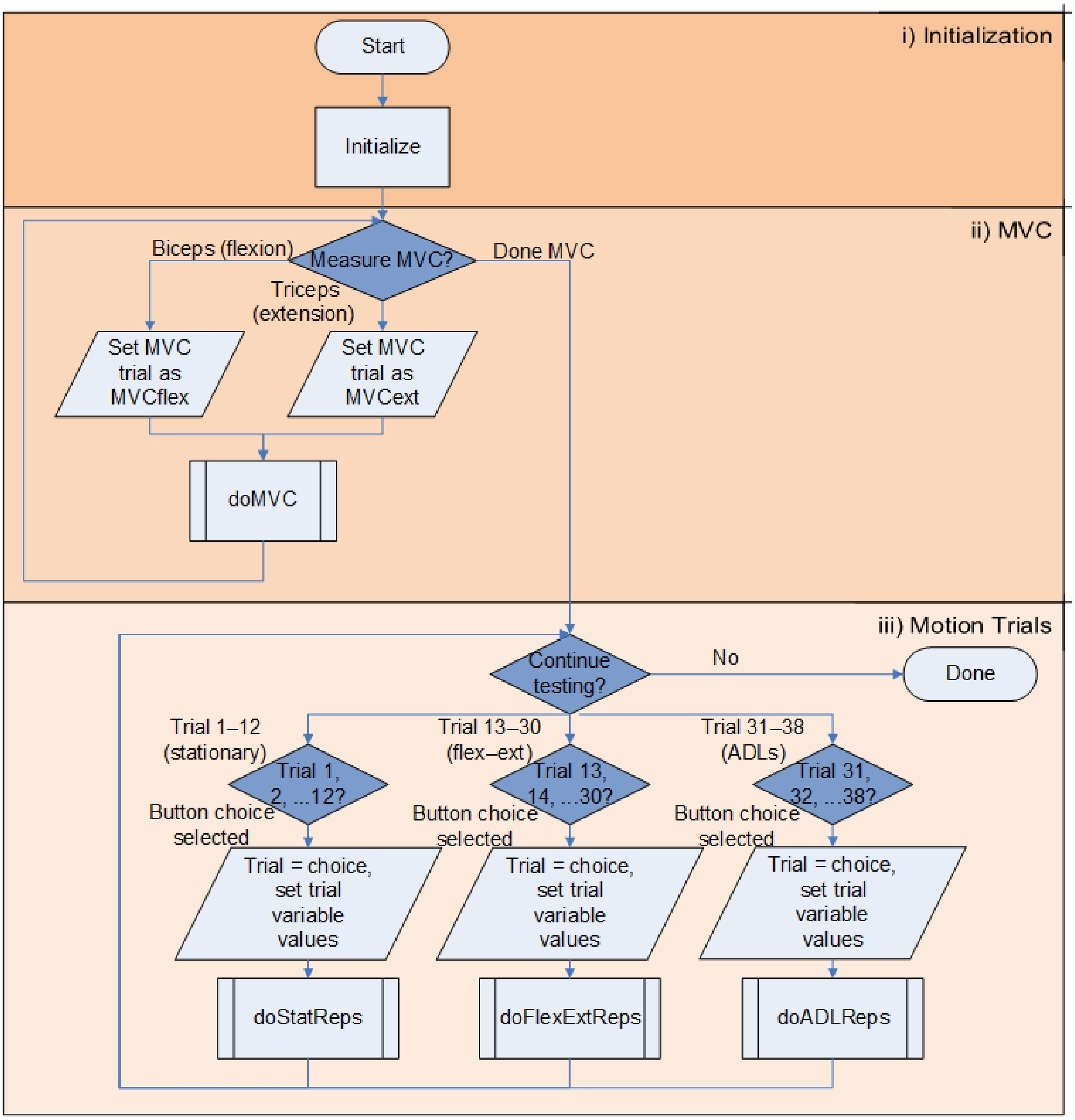

Figure 4.10: Overall flow of the `MeasuringForDynamicCalibration` program run on the KUKA robot.

was pushed or it collided with something it would respond with a low stiffness instead of colliding rigidly.

In addition to Null Space Damping, Translational and Rotational Damping were also set to 0.7, a damping parameter setting recommended by the KUKA manual for pre-

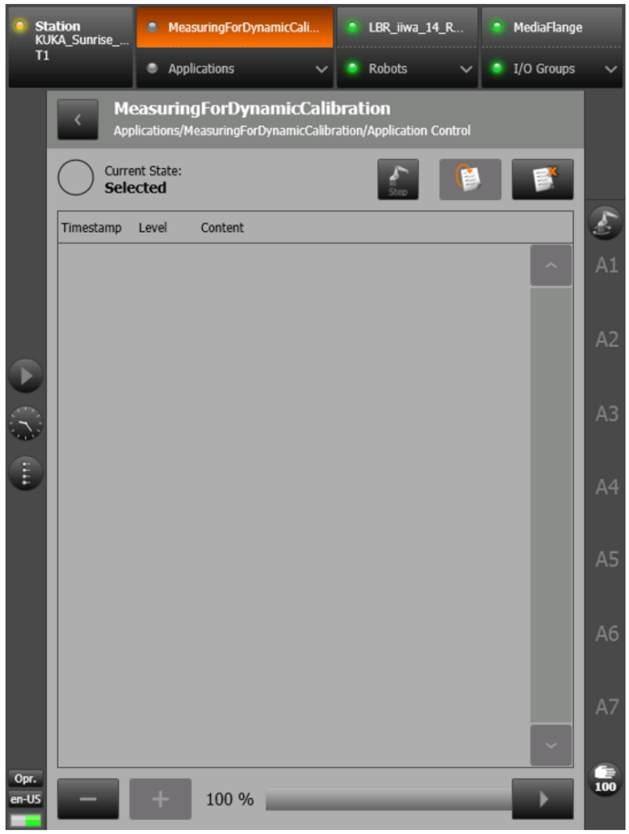

Figure 4.11: The smartPAD screen view at the beginning of `MeasuringForDynamicCalibration` application.

dictable robot motion [64]. The Translational and Rotational Stiffnesses, Force Bias and Force Limit parameter settings are described in detail in the method descriptions corresponding to the motion types. Stationary motion parameters are described in the method 3 `doStatReps` explanation, Flexion–Extension parameters are described further in the method 4 `doFlexExtReps` explanation, and the ADL parameters are described in the method 5 `doADLReps` explanation.

The three Cartesian Sine Impedance Control Modes were implemented in addition to a Cartesian Impedance Control Mode shown in Table 4.3, deployed at each time the robot was paused holding its position with maximum resistance. This mode was updated from the mode used in `SetStartPositions` (Table 4.1) to reset maximum control forces and torques, and null space damping and stiffness after these parameters were used at other times during the motion trials. Translational Stiffness was set to 5000 N/m to make

Table 4.2: The configuration of Cartesian Sine Impedance Control Mode parameters used to simulate environment interaction.

Cartesian Sine Impedance Control Mode Parameters	**Motion Type**		
	Stationary	**Flexion–Extension**	**ADLs**
Translational Stiffness [N/m]	0	0	0
Rotational Stiffness A [Nm/m]	0	0	0
Rotational Stiffness B [Nm/m]	0	0	300
Rotational Stiffness C [Nm/m]	0	0	300
Null Space Stiffness [Nm/rad]	50	50	50
Translational Damping	0.7	0.7	0.7
Rotational Damping	0.7	0.7	0.7
Null Space Damping [Nm*s/rad]	0.7	0.7	0.7
z Force Bias [N]	(0, +22, -22)*	(0, +22, -22)*	(11, 22)*
z Force Limit [N]	(absolute value z Force Bias)*	(absolute value z Force Bias)*	(absolute value z Force Bias)*
Rise Time [s]	2	2	2
Hold Time [s]	5	indefinite	indefinite
Fall Time [s]	2	0	0

the robot have very high resistance to movement in the x, y, and z axes. Setting Maximum Translational Control Force to 5000 N allowed for high translational forces to occur without the robot decreasing resistance to movement. Rotational Stiffness was set to 300 Nm/rad to make the robot have very high resistance to a, b, or c torques. Setting Maximum Rotational Control Torque to 300 Nm allowed for this high resistance to torques to occur. Null Space Damping was set to 0.7 Nm*s/rad, as recommended in the KUKA programming manual to ensure that the robot moves predictably and smoothly [64]. Null space stiffness was set at 200 Nm/rad so that the robot was not compliant in its redundant degree of freedom, ensuring the robot did not move as forces were applied to it.

(ii) MVC

After initialization was complete, a dialog box appeared on the smartPAD display asking the coordinator if the MVC was to be measured. The coordinator responded with the touchscreen by selecting one of three buttons: 'Biceps (flexion)', 'Triceps (extension)', or 'Done MVC'. If biceps or triceps was selected, force value and direction variables within the program were set accordingly. Then the `doMVC` method was called and completed. The `doMVC` method permitted the measurement of MVC as the participant attempted to flex or extend the elbow against resistance, as described in method 2 `doMVC`. Afterwards,

Table 4.3: The configuration of Cartesian Impedance Control Mode parameters for robot holding positions with maximum stiffness.

Cartesian Impedance Control Mode Parameters	**Value**
Translational Stiffness [N/m]	5000 (max)
Rotational Stiffness [Nm/rad]	300 (max)
Additional Control Force [N]	0
Maximum Translational Control Force [N]	5000 (max)
Maximum Rotational Control Torque [Nm]	300 (max)
Null Space Damping [Nm*s/rad]	0.7
Null Space Stiffness [Nm/rad]	200

the application looped back to inquiring the user if MVC was to be measured.

Once the 'Done MVC' button was selected instead of 'Biceps (flexion)' or 'Triceps (extension)', then the application moved on to the next step: running motion trials.

(iii) Motion Trials

To start the motion trials, a dialog box appeared on the smartPAD display asking if testing was to continue. The coordinator responded via the touchscreen by selecting a button indicating the type of trial ('Trial 1–12 (stationary)', 'Trial 13–30 (flexion–extension)', 'Trial 31–38 (ADLs)') or 'No' to end the trials.

If a trial type was selected, another dialog box appeared with selectable buttons for each motion trial in that range. The user was prompted to select a button indicating the desired trial. Once a trial was selected, trial variables used in the remainder of the application were updated. Force levels and direction, log filenames, start positions, velocity, and motion type were set automatically within the program according to the chosen trial, hence, the coordinator was not required to input these manually. Based on the type of motion trial selected, a method, `doStatReps` or `doFlexExtReps` or `doADLReps`, was called to perform the three repetitions of the specified stationary, or flexion–extension, or ADL motion trial. Once the repetitions were completed, the application looped back to asking the coordinator if testing was to continue.

2. `doMVC`

For maximum voluntary contraction in the direction of elbow flexion and extension, the robot moved to its MVC starting frame (updated in `setStartPositions`), and held its position steady for 5 seconds. At this point, the participant held the handle interface. If more time was required for the participant to get in the proper position, the coordinator could let go of the play or enabling switches to pause the program.

The blue LED ring on the robot flange lit up while the robot held its position with a high stiffness Cartesian Impedance Control Mode, Table 4.3. This mode was implemented because both translational and rotational stiffnesses were at the robot maximum, meaning that the robot resisted movement if the handle was pulled or pushed. As shown in Table 4.3, two Cartesian Impedance Control Mode parameters, Maximum Translational Control Force and Maximum Rotational Control Torque, were set high at 5000 and 300, respectively. These maximum control force and torque parameters ensured the robot stayed in position. If these maximum control forces and torques were low, the robot would move like a spring in response to the participant pulling or twisting the handle with a force or torque above the specified value.

At this step, the participant attempted to flex or extend their elbow while holding the robot handle, and the robot resisted movement. The robot measured and recorded position and forces for five seconds. Then the blue LED ring turned off to indicate to the participant that they are allowed to stop flexing or extending, the robot stopped recording to the log file saved locally on the robot controller, and the robot paused its movement.

3. `doStatReps`

   The stationary isometric contraction (zero velocity) tests, began with the robot moving to the starting position for the selected trial. The Cartesian Sine Impedance Control Mode, Table 4.2, was updated automatically with the Force Bias and Force Limit values corresponding to the current trial. The robot held its position for ten seconds allowing the participant to rest and get in position, holding the handle.

   Then, the blue LED ring turned on, and the robot held its position with the updated Cartesian Sine Impedance Control Mode. The low translational and rotational stiffnesses meant that the robot would move with little resistance in response to being pushed or twisted, to ensure that the participant's motion was not constrained. The Force Bias meant that the robot pushed or pulled with the specified force with a line of action along the long axis of the handle (z axis of the robot flange), as seen in Figure 4.12. Setting the direction of the Force Bias in the positive or negative z direction corresponded to resisting or assisting elbow flexion as the participant held the handle, depending on how the handle was oriented.

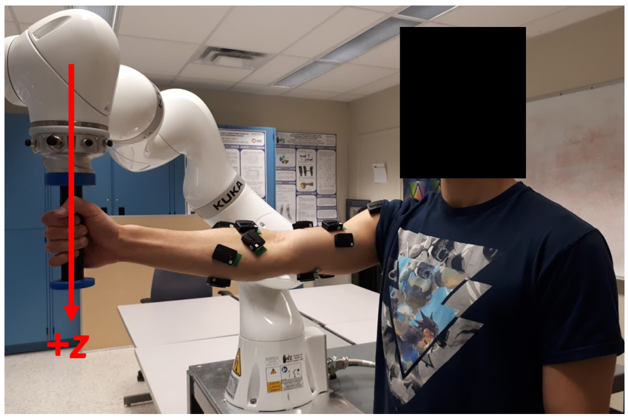

Figure 4.12: The z axis of the robot flange and line of action of force biases applied to the participant hand via a handle interface.

For the stationary motion trials, the participant tried to maintain a constant position while holding the handle, as the robot pulled with the specified force value in the specified direction. The Force Limit parameter equalling the Force Bias value meant that if the participant pushed or pulled the handle in the z direction with a force greater than the absolute value of the specified force, the robot would respond by moving to ensure a constant force was applied.

As shown in Table 4.2, the 2 second Rise Time meant that the force increased to the desired amount over 2 seconds, instead of suddenly applying the total force. A rise time of 2 seconds was found to be sufficient to ease the application of the force so that it was not too sudden. The 5 second Hold Time controlled the robot to maintain the specified force level for 5 seconds, then the force decreased back to 0 N over two seconds, as specified by the Fall Time parameter.

At the end of the repetition, the LED lights turned off and the robot held its position with its original holding stiffness, supporting itself and not applying greater forces to the environment, for 10 seconds until the next repetition. This process was repeated for a total of 3 repetitions. Log files containing timestamped robot position and external force measurements for each repetition were saved to the robot controller.

4. `doFlexExtReps`

The flexion–extension tests began with the robot moving to the start position for the selected trial and pausing for 10 seconds. The Cartesian Sine Impedance Control Mode for flexion–extension movements was updated automatically with the Force Bias and Force Limit values accordingly, as shown in the third column of Table 4.2.

At the beginning of a repetition, the blue LED ring on the robot flange lit up, the robot was in the impedance mode with force overlay corresponding to the specified trial, and the robot system was recording and saving timestamped position and force measurements to a log file. During the repetition, the participants held the handle interface, and performed an elbow flexion–extension movement. The robot could be pushed or twisted with little resistance, due to low translational and rotational stiffness parameters. Depending on the specific trial, the robot was passive with gravity compensation, allowing the participant to move it with little resistance, or allowed the participant to move the robot while the robot applied a force, equalling the Force Bias parameter, along the long axis of the handle, robot flange z axis. Depending on the direction of this force overlay (Force Bias), with or against the direction of motion, the participant pulled against it or was assisted. The Force Limit equalling the absolute value of the Force Bias, allowed for the robot to provide the stiffness required to overlay the corresponding Force Bias.

At the end of each repetition, the coordinator pressed the green user button on the robot flange to signal the end of the repetition and stop the recording of that repetition. The Hold Time of the impedance control mode was set to be indefinite, and the ending condition of the motion was activation of the green button. This Hold Time parameter setting and end condition meant that the participant was able to perform their motion without a time limit enforced by the robot. The end of the repetition was signalled to the robot by the coordinator pressing the green button on the robot flange. Due to the repetitions ended by a button press in this method, the Fall Time parameter was not used, therefore it was set to the default 0 s. When a repetition ended, the blue LED ring turned off and the robot resumed its original high stiffness mode holding its position stationary for 10 seconds until the next repetition. Three repetitions occurred for each of these trials.

5. `doADLReps`

   To begin the activity of daily living tests, the robot moved to its starting position for the selected trial and paused for five seconds. The Cartesian Sine Impedance Control Mode for ADL movements was updated with the Force Bias and Force Limit values for the specified trial, as shown in the fourth column of Table 4.2.

   At the beginning of a repetition, the blue LED ring on the robot flange lit up, the robot switched to its updated impedance mode with force overlay, as well as recorded and saved timestamped position and force measurements to a log file. The impedance control mode used to simulate the activities of daily living was very similar to the mode used for flexion–extension movements, but with a higher stiffness for two rotational directions. For the simulated activities of raising the arm above horizontal lifting an object off a shelf (ADL 1), and lifting a cup to the mouth (ADL 2), rotation about the handle axis (z axis of the robot flange) was not restricted (rotational stiffness a set to 0 Nm/rad). As the handle was to remain vertical, maximum stiffnesses of 300 Nm/rad were set for b and c rotations to limit and prevent tipping of the handle off of the vertical axis. With the specified Force Bias being applied downwards towards the ground (simulating gravity acting on an object), the participant was free to perform the motion without tipping the handle. The Force Limit value set to equal the absolute value of the Force Bias value meant that the robot was capable of applying the corresponding force through the handle to the participant, but not a force higher than the Force Limit.

   To end the repetition, stopping the force application and recordings, the coordinator pressed the green user button on the robot flange. The blue LED ring turned off, signaling no force overlay and the robot returned to its high stiffness mode, not moving for 10 seconds until the next repetition. A total of three repetitions were completed for the current trial.

4.3.1.2 Transferring Projects

Once projects were setup in Sunrise.Workbench, they were transferred to the robot controller via an ethernet cable, by clicking the 'Synchronize Project' button, as shown in Figure 4.13. During development of programs, projects were loaded from the controller to the computer to collect and save robot frame and tool information. All safety configurations transferred as part of the project. For projects to be transferred, computer IP (internet protocol) settings were updated in order to communicate with the robot, as shown in Figure 4.14.

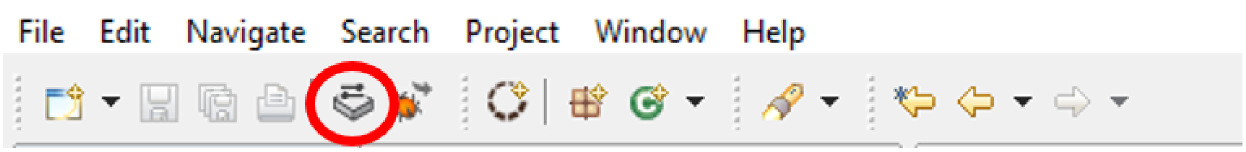

Figure 4.13: Sunrise.Workbench toolbar with 'Synchronize Project' button.

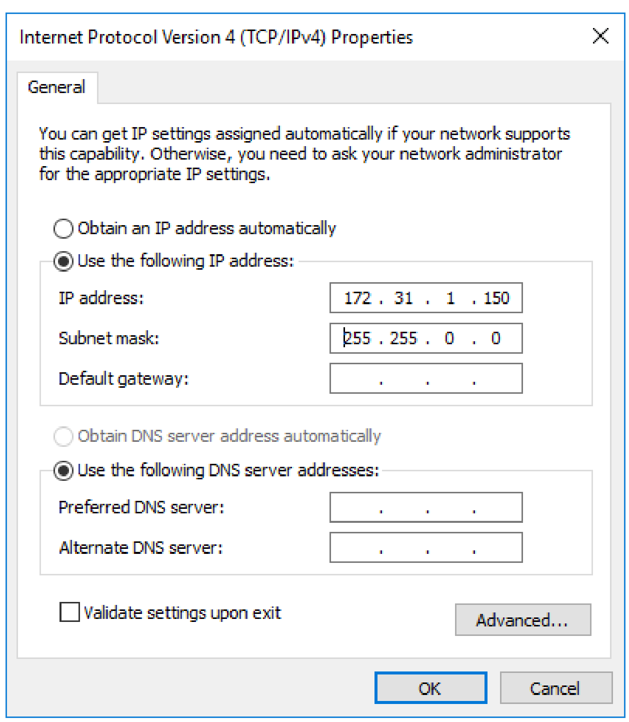

Figure 4.14: IP settings required to communicate with the robot.

4.3.1.3 Tool Set-Up

In order for robot movements and measurements to adjust to external forces properly, all components attached to the robot needed to be configured. In this study, a handle end effector was attached to the robot flange to interface between the participants and the robot.

After mounting the handle to the robot, "tool load data" was determined using the built in robot controller functionality. A handle tool was created in the project object templates in Sunrise.Workbench, Figure 4.15, and then transferred to the robot. In the Robots view, **Load data** was selected, **Determining the load data** was pressed, **Redetermine mass** was selected, and then the program was allowed to run, as shown in Figure 4.16.

When the automatic load data determination was complete, the load information was applied to the tool, and the entire project was synchronized between the controller and Sunrise.Workbench program. Synchronization saved the configuration to Sunrise.Workbench so that the tool could be used in programs.

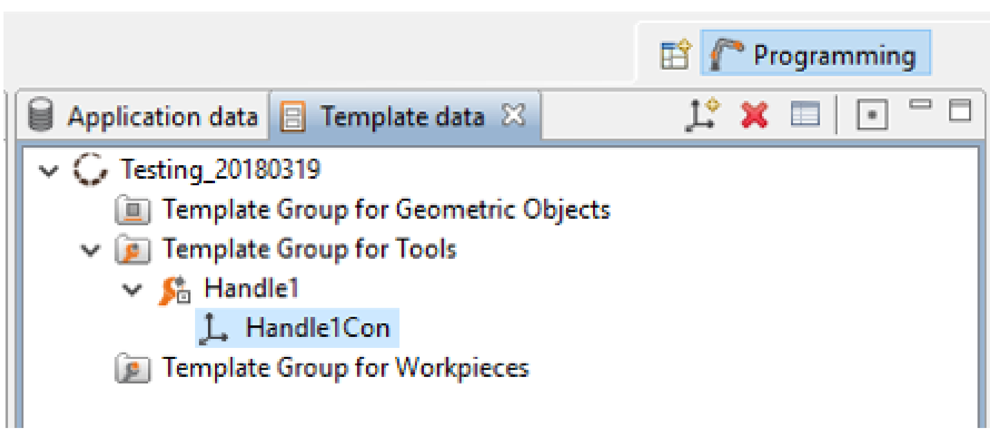

Figure 4.15: Sunrise.Workbench template data for tools and workpieces.

4.3.1.4 Safety

While working in close proximity with the robot, safety measures were taken, to ensure the participants and coordinators running the experiment remained safe. People could be in direct contact while collaborating with the robot as it was running because of the safety configurations in place. The KUKA lbr iiwa has torque sensors in each of its joints, which were used to measure torques

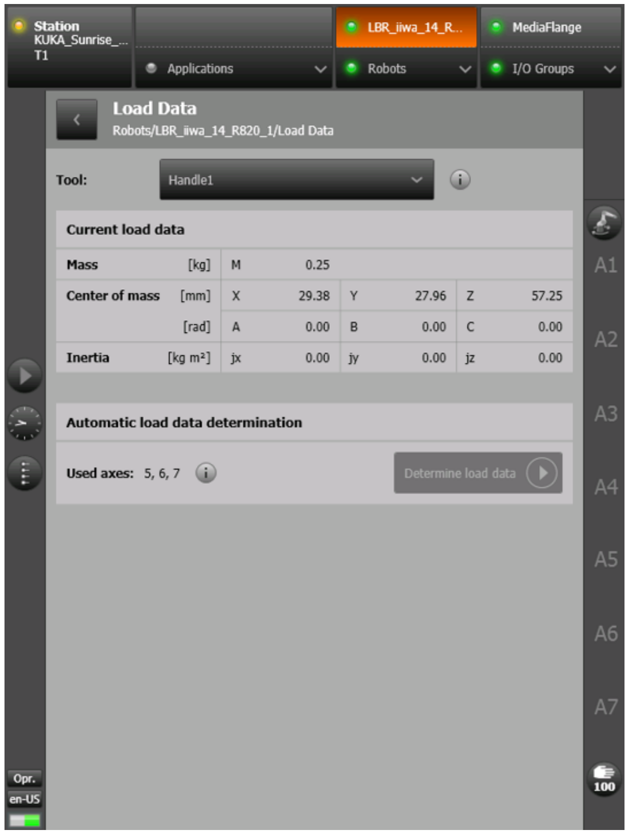

Figure 4.16: The smartPAD screen view for determining and saving tool loads.

and calculate forces, which allowed for force limits to be set.

The robot programs were run in a testing mode instead of in an automatic mode, as one safety precaution. KUKA provided operating modes, including: T1 (manual reduced velocity), T2 (manual high velocity), and AUT (automatic). The differences in functions are described in Table 4.4. In the test modes (T1 and T2), an enabling button and play button needed to be pressed and held for the program to run, compared to the automatic mode which allows the program to run automatically once started. The test modes facilitated pausing and restarting of the program in between trials. As well, the coordinator was able to immediately stop the robot at any time by letting go of one or both of the enabling and play buttons. `SetStartPositions` was run in T1 mode, while `MeasuringForDynamicCalibration` was run in T2 mode to facilitate higher velocities.

Before the robot could move, a safety configuration within Sunrise.Workbench was required to be set, loaded to the robot controller, and given permission to take effect. The safety configuration, shown in Figure 4.17, consisted of KUKA Permanent Safety Monitoring (KUKA PSM), Customer

Table 4.4: Operating modes of the KUKA robot.

Operating Mode	Program Conditions	Jog Mode
T1	Reduced velocity, maximum 250 mm/s Manual	Maximum 250 mm/s
T2	Programmed velocity Manual	Not possible
AUT	Programmed velocity	Not possible

Permanent Safety Monitoring (PSM), and Event-driven Safety Monitoring (ESM).

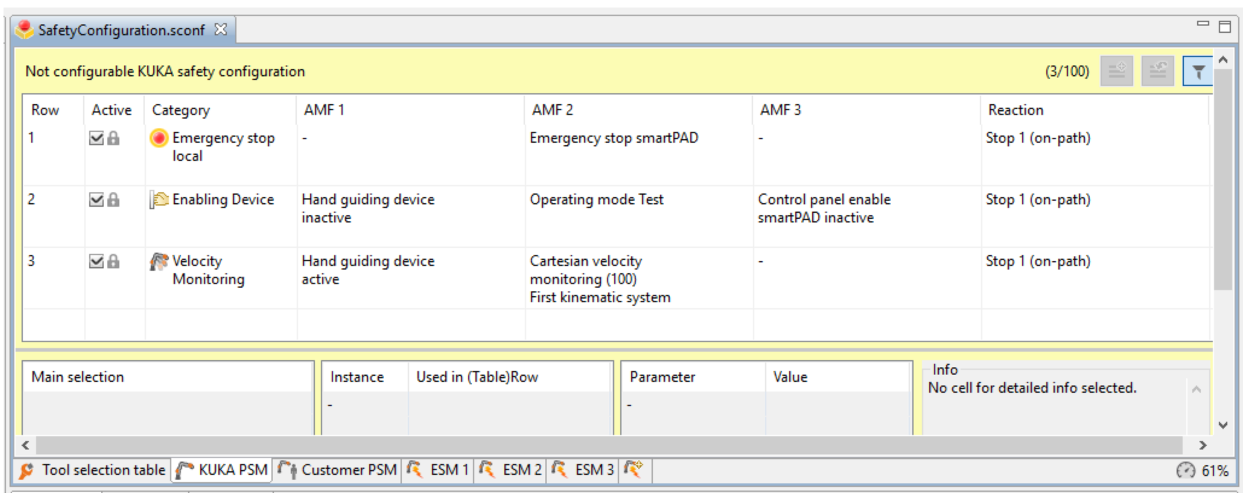

Figure 4.17: Sunrise.Workbench safety configuration, KUKA PSM.

KUKA PSM settings consisted of non configurable safety functions, which could not be changed. These included the use of the emergency stop button on the smartPAD, the requirement of holding enabling buttons when the robot was running in test modes T1 and T2, and velocity limits (250 mm/s) when the robot was in a hand guiding (robot passive) mode.

Customer PSM settings were configurable safety checks that were monitored while the robot was running. Velocity monitoring was one PSM setting that was added, as shown in Figure 4.18. With the cartesian velocity monitoring, the maximum robot and tool velocity was set at 500 mm/s. If this velocity was exceeded (KUKA Atomic Monitoring Function (AMF) is violated), a safety stop 1 (on-path) was triggered. This speed limit was introduced because the motions necessary during the trials fell under this speed and higher velocities were not required. Therefore, if the participant were to let go of the handle while the robot was applying a force overlay, or if the participant was not resisting the forces applied by the robot, or if the participant was not moving the robot smoothly in a controlled manner, then the robot stopped before any damage was caused.

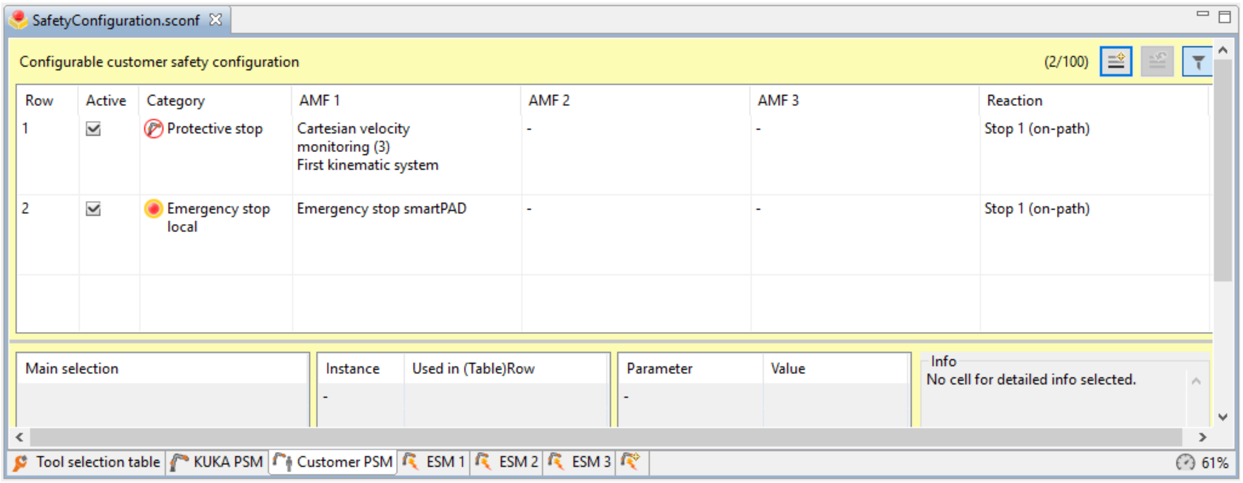

Figure 4.18: Sunrise.Workbench safety configuration, Customer PSM.

These two sets of PSM (Permanent Safety Monitoring) functions were running and being checked while the robot was running any program or application contained within the project. When the project (safety configuration, individual programs, tool and workpiece setups) were loaded on the robot controller, these safety checks were being monitored.

Three event-driven safety monitoring (ESM) AMFs (Atomic Monitoring Functions) were also added, as shown in Table 4.5. The monitoring of these safety functions took place at specific points during the programs when they were activated, and were not monitored when deactivated. A maximum of one ESM could be activated and monitored, while the remainder needed to remain inactive. If the AMF safety conditions were ever violated, the robot stopped on path and paused the program. The program could be played after being enabled again with the enabling buttons.

ESM 2 was only utilized during the programming process to move the robot to various positions and continue running the program, while ESM 1 and 3 were used during the main programs used for testing participants. Notable differences between the ESM settings were that ESM 1 had a collision detection external torque limit, however, ESM 3 did not have this added torque sensing

Table 4.5: Event-driven safety monitoring.

ESM	**AMF**	**Reaction**
ESM 1	Emergency stop smartPAD	Stop 1 (on path)
	Collision detection (maximum external torque 30 Nm)	Stop 1 (on path)
	Cartesian velocity monitoring (maximum velocity 500 mm/s)	Stop 1 (on path)
ESM 2	Hand guiding device inactive	Stop 1 (on path)
ESM 3	Emergency stop smartPAD	Stop 1 (on path)
	Cartesian velocity monitoring (maximum velocity 500 mm/s)	Stop 1 (on path)

in the safety controls.

ESM state 1 was enabled while: the coordinator was providing input with the smartPAD touchscreen, the robot was moving between positions, or the robot was paused with low forces (lower than 22 N). Directly before an interaction with forces of 22 N, or higher forces during MVC completion were expected, the ESM state was switched to ESM 3 to allow the high forces to occur without the robot conducting a safety stop. After the high force interaction, the ESM state was switched back to ESM 1. This ensured that during the majority of the testing time, the extra end effector collision detection was activated.

4.3.1.5 Obtaining Log Files

The data collected during runtime of the program, were written to log files on the robot controller. In order to access the files and transfer them to a desktop computer, a USB was inserted into the robot controller and a diagnostic package was written to the USB, after each participant completed all of the trials. The option to write the diagnostic package to the USB was selected with the smartPAD from the Robots view. The log files of interest were then found in the file path \KRCDiag_0_2018-05-15T17_36_18\Files\KRC\Roboter\Log\DataRecorder.

4.4 Protocol

Each of the individual systems described were used in conjunction. Using this equipment, the protocol followed to conduct the experimental trials for data collection from the participants is listed below:

1. Systems were turned on and tested to ensure programs were running without problems. These systems included: the KUKA robot, Kinect sensor, and Trigno EMG collection unit.

2. The participant read through and signed the letter of information and consent form, shown in Appendix B.2.

3. The coordinator filled out the Trial Form, shown in Appendix B.3, with the participant's information (subject number, age, dominant hand, gender, weight [kg], height [cm], waist circumference [cm], wrist circumference [cm], hip circumference [cm], forearm circumference [cm], forearm length [cm], upper arm length [cm], room temperature [°C], time of day, level of activity [number of times exercising/week]).

4. The robot program `SetStartPositions` was run. As described in Section 4.3.1.1.1, the coordinator guided the participants through positioning their arm in the start positions for the various motions, while the coordinator adjusted the position of the robot and updated the robot frames via the smartPAD.

5. The surface of the skin where EMG sensors were to be attached was cleaned with an alcohol swab. These areas were above the 16 muscles of interest.

6. EMG sensors were attached with double sided sticky tape to the surface of the skin over the muscles of interest. The EMG sensors were located over the muscle belly, following SENIAM guidelines and anatomy diagrams.

7. The trial data were collected. One repetition each of elbow flexion MVC and elbow extension MVC were completed. Then the three repetitions of each of the 38 motion trials were performed in randomized order. Prior to data collection, the order of trials was randomized using the `randi` function in MATLAB, after the random number generator was seeded based on the current time using `rng('shuffle')` to ensure a new sequence of numbers would be produced.

8. The EMG sensors were removed from the participant's arm once all repetitions of all trials were completed. The participant's involvement in the experiment was finished at this point.

9. Log files were obtained from the KUKA robot, EMG files from the Trigno system were converted to .csv file format, and any extra notes were added to the notes section of the Trial Form by the coordinator. The Trial Form is shown in Appendix B.3.

4.4.1 Timestamps

The EMG, kinematic, and force data were collected using three separate systems, Trigno, Kinect, and KUKA robot, respectively. Data points from the separate systems were required to be synchronized offline after collection since the systems were not connected at the time of the trials. Timestamps recorded by each system were used to match data points obtained from the three different systems.

In the files holding joint measurements collected with the Kinect system, the second line held a timestamp of the first data point identifying the initial real world time, and the last line held the amount of time in seconds that had passed from the first data point to the last data point. With the timestamp information and the frequency of measurements, the real world time of any of these position measurements could be determined.

In the saved files from the Trigno system, the time label of the first measurement was '0', with the time labels of subsequent measurements counting up in time since the first measurement. When viewing the files in the Trigno software, a real world time timestamp for the beginning of the measurement file was available and recorded, in order to match the Trigno measurement times with other systems referencing real world time. To relate each of the KUKA robot data points to data collected with the Kinect and Trigno systems, each data point in the log files from the KUKA robot were labeled with an epoch timestamp.

When first observing the EMG data (from the Trigno system) and the force data (from the KUKA robot), it was discovered that there was an offset between the real world timestamps. The KUKA robot system time appeared to be offset from true world time by about 30 seconds. This offset also drifted when the robot was turned on and off since the KUKA robot was not connected to the network with the same time server as the other computers used during data collection. To determine the offset, the real world time, seen watching a live web update on the computer system with time matching the Trigno and Kinect systems, of 10 unique activity timepoints that corresponded to the start and end of log files saved by the KUKA robot were recorded. The time difference between the observed time and robot recorded timestamps were then calculated and averaged to determine the actual robot offset. Human error or delay in observing the time was minimal enough for this study. Since during troubleshooting the robot time offsets were seen to fluctuate slightly when the robot was turned off then restarted after a period of a couple days, this time recording process was completed for each participant trial, without turning the robot system off in between.

With this information, the data from the Trigno, KUKA robot, and Kinect system could be synchronized, using the timestamps to relate the data points to a common reference time.

4.5 Conclusion

This chapter described how the equipment was used to conduct motion trials to collect the kinematic, dynamic, and EMG measurements of interest. The software developed for data collection and calibration of the KUKA robot was also outlined. The majority of the set-up was required for the KUKA robot to smoothly guide the motion trials. In order to gain insight into the raw EMG data collected, processing and analysis of the EMG data were the next steps. This analysis is described in the following chapter.

Chapter 5

Pre-Processing and Statistical Analysis

The previous two chapters presented the experimental design and execution of data collection. Surface EMG data were measured from 16 muscles as 24 healthy participants performed motions. The first purpose of conducting this experiment was to gain insight into how muscle activations changed with motion characteristics: arm position, force, and velocity. In order to do that, this chapter presents the processing of the collected data and statistical analysis of the processed EMG signals. The results of this chapter can be used to inform the use of EMG signals in detecting characteristics of intended motion in the next chapter. To conduct a statistical analysis, features needed to be extracted from the raw EMG signals. The processing of the raw EMG signals is presented in the following section.

5.1 Pre-Processing EMG Signals

The data during trials were all collected, and then processed after data collection was completed. Processing was completed offline, not in real time, using MATLAB R2016b (MathWorks, USA). See MATLAB scripts written for processing data in Appendix A.2. It was found that data from one of the Trigno sensors (Sensor 8) were inconsistent as the sensor disconnected from the system and did not measure EMG signals during many of the trials and repetitions. This occurred for

21 out of 24 subjects (87.5 %). Due to the sensor inconsistently recording, all recordings from Trigno Sensor 8 (attached over the pronator teres) were omitted during further statistical and classification analysis. As well, during measurement, Trigno Sensor 7 disconnected momentarily for Participant 9, and Trigno Sensor 9 disconnected for Participant 7. Since these were isolated incidents, not all of Sensor 7 and Sensor 9 data were excluded for every subject. Sensor 7 was only excluded from further analysis for Participant 9, and Sensor 9 data were only excluded for Participant 7, since including the partial measurements could skew the results. Table 5.1 lists the 15 muscles used in further processing.

Table 5.1: Muscle channels

Muscle Channel	**Muscle**	**Acronym**
1	biceps brachii short head	BB_S
2	biceps brachii long head	BB_L
3	brachialis	BRA
4	brachioradialis	BRD
5	triceps brachii long head	TRI_LO
6	triceps brachii lateral head	TRI_LAT
7	triceps medial head	TRI_M
8	infraspinatus	ISPI
9	anterior deltoid	AD
10	lateral deltoid	LD
11	posterior deltoid	PD
12	extensor carpi ulnaris	ECU
13	extensor carpi radialis	ECR
14	flexor carpi ulnaris	FCU
15	flexor carpi radialis	FCR

5.1.1 Segmenting EMG Repetitions

EMG signals for multiple repetitions were collected in the same file. Timestamps recorded with the hand position and force data from the KUKA robot, indicated the starting and end times of the various movements and repetitions. These timestamps of the beginning and end of each repetition were identified and then synchronized to the EMG files. The EMG signals were segmented into separate repetitions.

5.1.2 Filtering

EMG signals were sampled at a rate of 2000 Hz. Offline, these signals were bandpass filtered with a 4th-order butterworth band-pass filter with a lower boundary of 20 Hz and an upper boundary of 450 Hz. These bandpass window limits were chosen because it has been noted that EMG signals of interest using surface EMG electrodes are between the frequencies of 20 and 450 Hz [65]. A notch filter eliminated the 60 Hz power noise.

5.1.3 Normalizing EMG Signals

The EMG signals gathered indicate the activation of the muscle below the sensor, the combination of the individual motor units firing. In order to compare the signals between participants and muscles, the filtered signals were normalized relative to the absolute maximum of the EMG signals gathered from the corresponding muscle during the maximum voluntary contraction exercises.

5.1.4 EMG Feature Extraction

Features of EMG signals were extracted to observe how EMG signals changed with the influence of the arm position, force, and velocity levels during dynamic movements. The features extracted fall under the time domain or frequency domain categories, and were extracted using existing MATLAB functions [66]. Table 5.2 lists the extracted features.

To extract the features, the window size and overlap of windows were held constant. The window length was set at 500 samples (approximately 250 ms) and the overlap of the windows was 250 samples (approximately 125 ms) long. The parameters were set at these values because these levels have been commonly used and have been shown to be effective [43]. Segments were overlapped because segments over 200 ms need to be overlapped to potentially have time to process signals and control devices in real time with less than a 300 ms delay between muscle contraction and device movement [67]. A variety of window sizes and overlaps have been shown to work [35,68]. However, the window length resulting in higher classification accuracy can vary depending on the features and classifiers used, with longer window lengths closer to 400 ms performing better than very short (50 ms) window lengths [35]. Another study including varying window lengths found

Table 5.2: Features extracted

Feature Number	Feature	Acronym
1	mean absolute value	MAV
2	slope sign changes	SSC
3	waveform length	WL
4	zero crossings	ZC
5	root mean square	RMS
6	first autoregressive coefficient	AR1
7	second autoregressive coefficient	AR2
8	third autoregressive coefficient	AR3
9	fourth autoregressive coefficient	AR4
10	mean frequency	MNF
11	median frequency	MDF

150–250 ms to be the optimal range in window length for acceptable classification error in real-time control [69]. As well, it has been shown that increasing window lengths from 125 ms to 500 ms increased classification accuracy, but the improvements above 250 ms were not significant [43]. Therefore, the effects of window size and overlap were not of interest in this study and thereby held constant. Also, it has been shown that the efficacy of pattern recognition models depended more on, or were influenced more by, the features used (type, number, specific combination) and type of training data used than by the classifier [48].

Four time domain features belonging to the Hudgins set were extracted. This set of time domain features included: mean absolute value (MAV), slope sign changes (SSC), waveform length (WL), and zero crossings (ZC). Mean absolute value slope, included as part of the Hudgins set in some cases, was not observed in the initial study. A second group of time domain features seen in the literature consists of root mean square (RMS) and autoregressive coefficients (AR) [67, 70]. RMS and fourth order autoregressive coefficients (AR1, AR2, AR3, AR4) were also extracted.

Additionally, two frequency domain features extracted were the mean frequency (MNF) and median frequency (MDF). These were included to provide more frequency information than is represented in the SSC and ZC time domain features.

5.2 Statistical Analysis

With features extracted from the filtered and normalized EMG signals, a statistical analysis was completed to observe relationships between motion factors and EMG signal features.

Correlations between factor levels and EMG feature values for the muscles of interest were observed. To evaluate these connections or lack of connections between motion characteristics and EMG signals, a repeated measures analysis was completed using the Statistical Package for the Social Sciences v. 24 (SPSS). A statistical significance of 0.05 was used.

For each muscle and each of the extracted features, the feature value was averaged over the entire repetition of a movement, then the mean of the repetition averages was collected to give one value for the feature per movement. This process was completed for each subject. Each muscle and feature was analyzed separately. There was some variation in EMG feature values throughout a movement trial, however, the feature values were averaged over an entire motion to investigate the motion as a whole. In future studies, motions could be segmented further, potentially segmented into elbow flexion versus extension portions of a movement. EMG feature measurements were observed to be repeatable between the three repetitions, with intraclass correlation. Specifically for the two participant pilot study data, there was some variation in EMG measurements between repetitions, however trends of features increasing or decreasing with movement repetitions were not observed for the BB_S for S1, or the TRI_LO for S2.

For the factorial experimental design, EMG data were analyzed with repeated measures tests using SPSS. For the basic flexion–extension tests, processed EMG signals from flexion–extension motions and isometric contractions with the elbow fully extended were grouped together. Activity of daily living motion trials were not included in this group for analysis. The three factors observed were arm position, interaction force between the users' hand and environment (robot), and goal velocity. Each factor had three levels. The three nominal arm positions (P1, P2, P3) were the arm starting down by the side of the torso (0° shoulder abduction, 0° shoulder flexion), the arm horizontal with hand stretched forwards (90° shoulder flexion), and arm horizontal with hand stretched to the side (90° shoulder abduction). The three force levels (F1, F2, F3) in this group of trials were 0 N, 22 N resisting elbow flexion, and 22 N resisting elbow extension. The three velocities (V1, V2, V3) were stationary (0°/s), slow (approximately 11°/s, 8 seconds to perform motion), and fast (approximately 23°/s, 4 seconds to perform motion). The velocities in this case were the speeds at which the participants were instructed to move during the tests, the goal velocities.

The next group for analysis was the motion trials for ADL 1, which consisted of a simulation of picking an object off of a shelf just above shoulder height, lowering the hand, and raising the hand back to the starting position. The two factors varied during this activity were force and velocity. Unlike the single flexion–extension motions and isometric contractions, the arm position was not specified for these activities as specific daily tasks were being mimicked. The two force levels (F1, F2) for this set were 11 N and 22 N. The direction of this force remained constant with the robot pushing vertically downwards on the participants' hands. The two velocity levels (V1, V2) were slow and fast, again. The 0°/s stationary level was not included in this case as performance of the task required movement not isometric contractions.

Similarly, the analysis group for ADL 2, which included a simulation of lifting a cup off of a table to drink, was analyzed with the two factors of force and velocity. The two force levels (F1, F2) were 11 N and 22 N, the two velocities (V1, V2) were slow and fast. The results of all statistical analyses described are presented in the next section.

5.2.1 Statistical Results

An initial statistical analysis was performed in order to observe the effect of changing position, force, and velocity levels on feature values of 15 arm and shoulder muscles. The results for basic flexion–extension, and two ADL motions are presented in the follow sections.

5.2.1.1 Flexion–Extension Statistical Results

For flexion–extension, there were many significant differences between all position levels, between all force levels, and between all velocity levels. The features with significant differences for position, force, or velocity levels, varied with each muscle assessed. Significant changes in feature values related to changes in the levels of the motion characteristics are displayed in Tables 5.3, 5.4, and 5.6. The rows correspond to the 15 muscles of interest (Table 5.1). The columns designate Features

1–11 (Table 5.2). In the body of the table, a coloured cell represents a significant difference in the feature value with changing factor level for that corresponding muscle, whereas white cells represent no significant difference for the intersecting feature and muscle combination.

Table 5.3: Significant differences found in the assessment of position during flexion–extension motion for the various features and muscles assessed.

Muscles/Features	**MAV**	**SSC**	**WL**	**ZC**	**RMS**	**AR1**	**AR2**	**AR3**	**AR4**	**MNF**	**MDF**
BB_S											
BB_L											
BRA											
BRD											
TRI_LO											
TRI_LAT											
TRI_M											
ISPI											
AD											
LD											
PD											
ECU											
ECR											
FCU											
FCR											

Table 5.3 displays a summary of the statistical results of EMG features related to arm position. This table shows that at least two feature values had significant differences with changing arm positions for all 15 muscles of interest. The differences in arm position between P1, P2, and P3 were shoulder flexion angles and shoulder abduction angles. It was expected that feature values for the prime movers responsible for shoulder flexion, abduction, and rotation would be significant. Based on anatomy, the main function of the LD is shoulder abduction. In P3, the participant's shoulder was abducted 90°, compared to P1 and P2 with 0° shoulder abduction. Consistent with the LD performing shoulder abduction, multiple features were significantly different for the LD between P1 and P3 (nine features), as well as P2 and P3 (7 features). However, multiple features of the LD muscle activation also changed significantly between P1 and P2 (9 features), where participants were maintaining a constant shoulder abduction angle. For example, mean SSC values for the LD were highest when the arm was held in P1, mid-range when the arm was held in

P2, and lowest in P3 (60.617 vs. 51.0955 vs. 47.703, $p < 0.001$), as shown in Figure 5.1 together with the ZC metric (which was also significant between P1 and P2). A full comparison of mean EMG feature values for each muscle, and significant differences in these values corresponding to varying levels of arm position, force, and velocity during elbow flexion–extension, is provided in Appendix C.1.

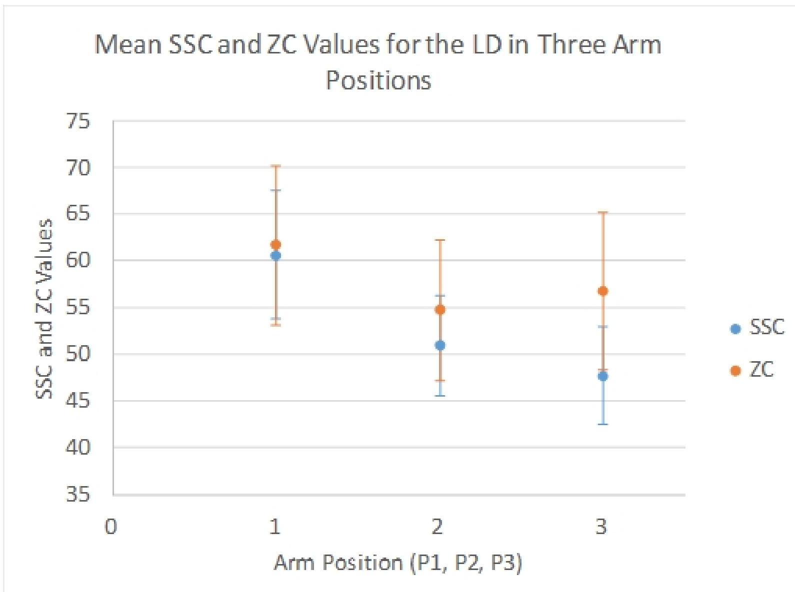

Figure 5.1: Mean SSC and ZC values for the LD in three arm positions. Error bars indicate standard deviations.

Anatomically, the AD contributes to shoulder flexion, and the PD contributes to shoulder extension. In P2, the participant's arm was oriented with 90° flexion, compared to P1 and P3 with 0° shoulder flexion. Multiple EMG features varied significantly based on arm position for both the AD and PD. For the AD, more features changed significantly between P1 and P2 (nine features), P1 and P3 (11 features) than P2 and P3 (five features). For example, mean WL values for the AD were lowest in P1 and higher in P2 and P3 (2.083 vs. 11.486 and 10.035, $p < 0.001$), as displayed in Figure 5.2. While for the PD, more features changed between P1 and P3 (nine features), P2 and P3 (nine features) than between P1 and P2 (three features). For example, mean WL feature values were lowest for the PD in P1 and P2 compared to P3 (3.342 and 3.456 vs. 15.846, $p < 0.001$).

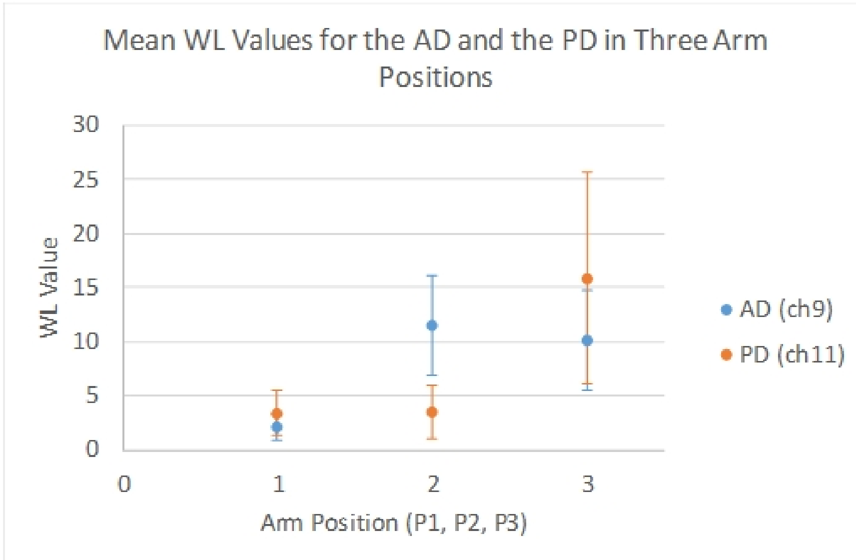

Figure 5.2: Mean WL values for the AD and the PD in three arm positions. Error bars indicate standard deviations.

Also, other arm muscles, such as the prime elbow flexors and extensors, BB (_S and _L) and TRI (_LO, _LAT, and _M), as well as muscles in the forearm (ECU, ECR, FCU, and FCR) had significant changes in values for subsets of features when the arm was in different positions. For example, mean ZC values for the BB_L were significantly different when the arm was in P1, P2, and P3 (59.791 vs. 52.945 vs. 56.325, $p < 0.001$). As well, mean WL values for the ECU varied significantly when the arm was in P1, P2, and P3 (12.215 vs. 15.925 vs. 18.866, $p < 0.001$). These findings suggest that muscle activation was influenced by joint positioning tasks for joints for which those muscles were the not main activators, and from which they were separated by another intermediate joint (the elbow separated the forearm muscles from the shoulder). Alternatively, changes in arm positions changed the way in which the arm muscles coordinated to perform a consistent task. These differences in muscle activations with changes in arm position were also observed with the arm performing flexion–extension motions with varying external forces (F1, F2, F3) applied to the hand, and at varying velocities (V1, V2, V3).

As shown in Table 5.4, force had a large impact on muscle activation. The patterns of muscle activation in response to environmental interaction occurred across different arm positions and elbow rotation velocities. This indicates that the impact of force on the EMG signals was strong.

Table 5.4: Significant differences found in the assessment of force during flexion–extension motion for the various features and muscles assessed.

Muscles/Features	MAV	SSC	WL	ZC	RMS	AR1	AR2	AR3	AR4	MNF	MDF
BB_S											
BB_L											
BRA											
BRD											
TRI_LO											
TRI_LAT											
TRI_M											
ISPI											
AD											
LD											
PD											
ECU											
ECR											
FCU											
FCR											

Five EMG features (MAV, SSC, WL, ZC, RMS) for all 15 muscles had significant differences in values during different environmental interactions. The remaining 6 features (four autoregressive coefficients, MNF, and MDF) changed significantly with force for the majority of the muscles measured, but not for all. The environmental forces were applied through the robot handle to the hand of the participant. The range of features impacted over each of the muscles suggested that the muscles throughout the entire arm and shoulder were working together synergistically in response to external forces.

Table 5.5: WL means and standard deviations at three force levels for elbow flexors and extensors.

	WL Mean			Standard Deviation		
Muscle	**F1**	**F2**	**F3**	**F1**	**F2**	**F3**
BB_S (ch1)	1.494	5.140	1.864	1.166	3.929	2.131
BB_L (ch2)	2.016	6.056	1.757	1.063	3.532	1.220
BRA (ch3)	2.228	5.823	2.561	1.352	2.969	1.960
BRD (ch4)	3.017	4.909	3.234	4.243	3.968	4.522
TRI_LO (ch5)	1.957	1.593	5.525	1.186	0.994	2.998
TRI_LAT (ch6)	2.777	2.303	8.231	1.808	1.573	5.188
TRI_M (ch7)	3.086	1.963	7.928	2.911	1.300	5.554

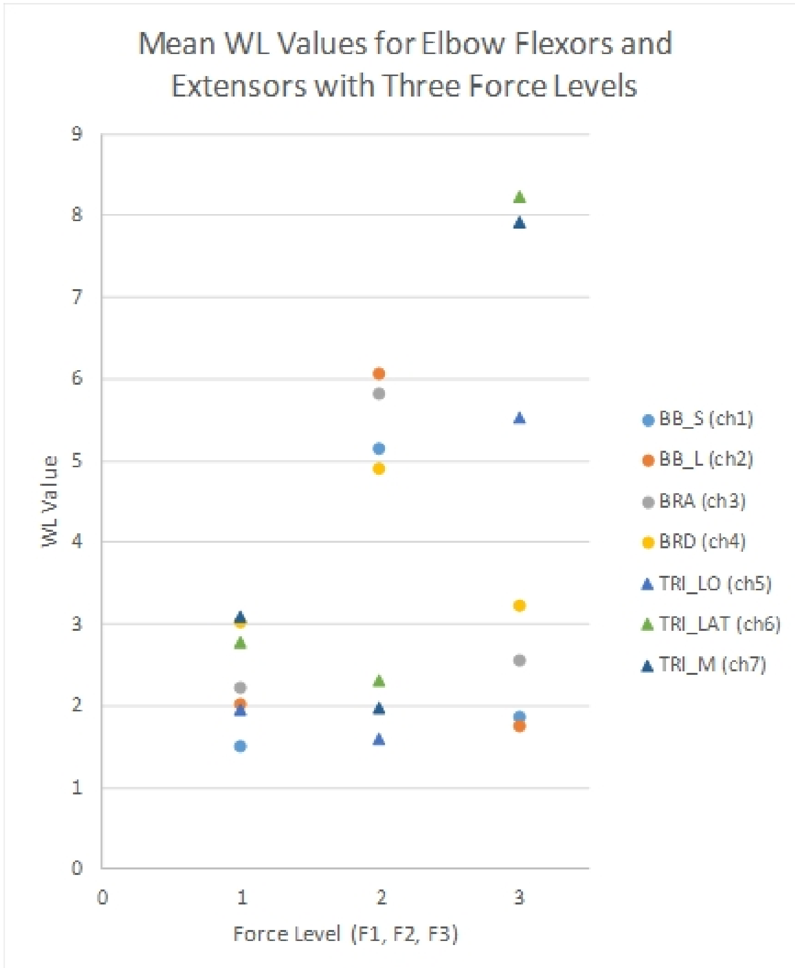

Figure 5.3: Mean WL values for elbow flexors and extensors at three force levels. Standard deviation error bars not shown in plot for clarity. Standard deviations are shown in Table 5.5.

The variety of muscles were activated differently in response to the changing forces to stabilize the wrist, elbow, and shoulder joints. To demonstrate, mean WL length values for a subset of muscles, the prime elbow movers, are presented in Figure 5.3. The mean WL feature values for BB_S were higher for F2 than F1 and F3 (5.140 vs. 1.494 and 1.864, $p < 0.001$). Whereas the mean WL of the TRI_M was highest for F3, lower for F1, and lowest for F2 (7.928 vs. 3.086 vs. 1.963, $p < 0.001$). These differences appeared even with changing arm positions and velocities.

Fewer muscle/feature combinations were found to have significant differences with changing velocities, as seen in Table 5.6. 108 muscle/feature combinations changed significantly with velocity level changes, compared to 124 and 152 muscle/feature combinations that changed significantly with different positions or force levels, respectively. Similar to the force motion characteristics, a trend of MAV, SSC, WL, ZC, and RMS changing significantly for more muscles than the remaining six features (except the fourth AR coefficient) was observed. Two examples of feature values changing with varying motion velocities are described. For the BB_S, the mean SSC feature was highest when the arm was stationary (V1), and also varied significantly between V2 (slow) and V3 (fast) motions (54.815 vs. 52.173 vs. 53.483, $p < 0.001$). Similarly, for the TRI_M, mean SCC values were higher during isometric contractions (V1) than slow (V2) and fast (V3) movements (59.427 vs. 56.277 and 56.471, $p < 0.001$).

Table 5.6: Significant differences found in the assessment of velocity during flexion–extension motion for the various features and muscles assessed.

Muscles/Features	**MAV**	**SSC**	**WL**	**ZC**	**RMS**	**AR1**	**AR2**	**AR3**	**AR4**	**MNF**	**MDF**
BB_S											
BB_L											
BRA											
BRD											
TRI_LO											
TRI_LAT											
TRI_M											
ISPI											
AD											
LD											
PD											
ECU											
ECR											
FCU											
FCR											

This section explored the results of the statistical analysis of EMG signals during flexion–extension motions. All three motion characteristics of interest (position, force, velocity), had some impact on EMG signal feature values. This impact was observed even though the motions were performed with limited constraints and motion characteristics changed simultaneously, in a repeated measures experimental design. Velocity had fewer muscle/feature combinations with statistical significance, while MAV, SSC, WL, ZC, and RMS were consistently significantly different for different force and velocity levels for most muscles (all 15 muscles for force). The results of the statistical analysis for ADL motions are described in the following section.

5.2.1.2 ADL Statistical Results

For activities of daily living (ADL) 1 and 2, there were fewer feature and muscle combinations that had significant differences for the force and velocity levels, when compared to the flexion–extension results. For ADL 1, 100 feature/muscle combinations varied significantly with force level, and 46 varied significantly with velocity. For ADL 2, there were 98 feature/muscle combinations with significant differences for force, and 36 combinations with significant differences for velocity. These results for ADL 1 are shown in Table 5.7 and Table 5.8. These results for ADL 2 are shown in Tables 5.9 and 5.10.

Table 5.7: Significant differences found in the assessment of force during ADL 1 motion for the various features and muscles assessed.

Muscles/Features	MAV	SSC	WL	ZC	RMS	AR1	AR2	AR3	AR4	MNF	MDF
BB_S	■	■	■		■			■	■		
BB_L	■	■	■	■	■		■	■	■	■	
BRA	■	■	■	■	■	■	■	■	■	■	
BRD	■	■	■	■	■			■	■		
TRI_LO		■		■				■	■	■	■
TRI_LAT	■	■	■	■	■	■		■	■	■	■
TRI_M		■		■			■	■	■	■	
ISPI	■	■	■	■	■		■	■	■	■	■
AD	■	■	■		■			■	■		
LD	■	■	■	■	■			■	■	■	
PD	■	■	■	■	■		■	■	■	■	■
ECU			■								
ECR		■	■								
FCU	■	■	■		■				■		
FCR	■	■	■		■						

Overall, force levels influenced EMG signals in a more consistent manner compared to velocity. As well, force and velocity impacted EMG signals less consistently for ADL motions compared to the basic elbow flexion–extension motion. This was expected as the ADL motions combined more rotations of the shoulder joint moving the upper arm making the movements more complex. As well, for the force characteristic, the nominal difference in force value between levels was only

Table 5.8: Significant differences found in the assessment of velocity during ADL 1 motion for the various features and muscles assessed.

Muscles/Features	MAV	SSC	WL	ZC	RMS	AR1	AR2	AR3	AR4	MNF	MDF
BB_S											
BB_L											
BRA											
BRD											
TRI_LO											
TRI_LAT											
TRI_M											
ISPI											
AD											
LD											
PD											
ECU											
ECR											
FCU											
FCR											

11 N during ADLs, whereas the nominal difference in force level investigated during basic elbow flexion–extension was plus and minus 22 N.

For both ADL 1 and ADL 2, the first five features (MAV, SSC, WL, ZC, and RMS) had significant differences in values during different velocities for more muscles than the remaining six features. This trend was observed during flexion–extension motions as well. This suggests these features may be more robust if used to indicate intended velocity, but less stable if signals unchanging in response to velocity changes were desired.

A few examples of the impact of force and velocity changes on EMG signals are presented. During ADL 1, mean WL values for the BB_S increased with an increase in force from F1 (11 N) to F2 (22 N) (3.231 vs. 4.803, $p < 0.001$). Similarly, mean WL values were higher for F2 (22 N) than F1 (11 N) for the TRI_LAT during ADL 1 (2.553 vs. 2.196, $p = 0.013$). During ADL 2, mean ZC values increased with an increase in velocity from slow (V1) to fast (V2) movements for the TRI_M (58.626 vs. 61.239, $p = 0.006$). The BB_S followed a similar trend with ZC values increasing with an increase in speed from V1 to V2 (50.593 vs. 52.211, $p = 0.029$). A full comparison of mean feature values and significant differences during ADL 1 and ADL 2 motions is provided in Appendix C.2 and C.3. The relationship between EMG signals and motion characteristics is explored further

Table 5.9: Significant differences found in the assessment of force during ADL 2 motion for the various features and muscles assessed.

Muscles/Features	MAV	SSC	WL	ZC	RMS	AR1	AR2	AR3	AR4	MNF	MDF
BB_S											
BB_L											
BRA											
BRD											
TRI_LO											
TRI_LAT											
TRI_M											
ISPI											
AD											
LD											
PD											
ECU											
ECR											
FCU											
FCR											

in the next chapter with patterns in EMG signals used to predict motion characteristics.

5.3 Conclusion

Chapter 5 described the processing of the collected EMG signals, extraction of time domain and frequency domain features, and the relationship between changing motion characteristics and muscle activation, as represented by the feature values. Arm position, force, and velocity all had an impact on EMG features throughout the arm. The relationship is explored further and used to determine intended arm motion in Chapter 6.

Table 5.10: Significant differences found in the assessment of velocity during ADL 2 motion for the various features and muscles assessed.

Muscles/Features	MAV	SSC	WL	ZC	RMS	AR1	AR2	AR3	AR4	MNF	MDF
BB_S											
BB_L											
BRA											
BRD											
TRI_LO											
TRI_LAT											
TRI_M											
ISPI											
AD											
LD											
PD											
ECU											
ECR											
FCU											
FCR											

Chapter 6

Motion Characteristic Classification and Applications

In the previous chapter, it was shown that motion characteristics such as arm position, interaction forces between a person's hand and the environment, and movement velocity all significantly impacted the activation of a variety of arm and shoulder muscles, as seen as variations in up to 11 EMG feature values. This insight into the behaviour of arm muscle activation during various motions was then used to classify levels of motion characteristics based on EMG signals. However, classification of motion characteristics was a basic example of a control scheme for a wearable device. A smoother device would need to be controlled using more complex model-based strategies. The importance of basic motion characteristics affecting EMG signals remains. Results of motion characteristic classification and further applications of using measurable motion characteristics to improve classification of interaction forces are described in this chapter.

6.1 Motion Characteristic Classification

In the previous statistical analysis, it was shown that changing levels of motion characteristics changed muscle activation as represented by EMG feature values. Depending on the motion factor (position, force, velocity) and motion type (simple elbow flexion–extension or ADLs), various feature values changed for differing muscles. In this section, the EMG features of various arm muscles were used in MATLAB to train two types of classifiers, linear discriminant analysis (LDA) and support vector machine (SVM), to detect classes of arm position, forces, and motion velocities. Refer to Appendix A.2.1 for MATLAB scripts developed to implement classification of motion characteristics. Determining motion characteristics such as force from EMG signals could then lead to better control of active-assistive devices.

6.1.1 Training Sets

EMG data from groups of motions were used to train classification models. Classifiers have tended to categorize hand movements less accurately when force levels or limb positions during testing changed or differed compared to the constrained conditions the training data came from [35]. Since including training data from various force levels or various arm positions has reduced motion classification error when various force levels and limb positions were introduced [35], diverse training sets were used in this work. The classifiers were trained and the accuracy of the trained classifiers was tested with the same data collected and used in the statistical analysis, after feature extraction. This training data consisted of EMG features from motions with varying arm positions, forces, and velocities. Separate models were trained with data collected during four different motion types listed below.

1. Flexion–Extension movements

   Observations from isometric contractions and flexion–extension movements were combined together in one training set. The EMG observations for ADL 1 and ADL 2 were excluded. These data were more representative of a single arm motion (flexion and extension of the elbow) without additional movements of the shoulder and wrist. This group included observations from diverse unconstrained motions with three motion factors (position, force, velocity) varying between three levels each.

2. ADL 1

   A second set of training data consisted of EMG data collected during performance of ADL 1 only for each participant. Two motion factors (force and velocity) varied between two levels each, during ADL 1.

3. ADL 2

   A third set of training data included EMG features collected during ADL 2 movements for every participant. For ADL 2, two motion factors (force and velocity) varied between two levels.

4. ADL 1 and ADL 2

   The fourth set of observations contained EMG data collected during ADL 1 and ADL 2 movements performed by all 24 participants. Data from both ADLs were combined to generate a more diverse training set.

Data were separated by type of motion to train independent models because force levels differed between flexion–extension motions and ADLs. As well, unlike flexion–extension motions, arm position was not a controlled variable during ADLs.

The training input was divided into predictors, which consisted of EMG feature values from the various arm muscles measured, and labels pertaining to motion characteristic classes. Each observed set of predictors was assigned a label. The observations consisted of data from a subset of trials corresponding to the four types of motion (flexion–extension, ADL 1, ADL 2, ADL 1 and ADL 2), for each participant. The predictors and class labels are described in more detail in the following sections.

6.1.1.1 Predictors

For this initial collection of tests, all feature values (11) for all available muscles (15) were given as predictors to the classifier. Along with the set of predictors (11 feature values for all 15 muscles available), a label was assigned to each motion.

6.1.1.2 Labels

The labels for each movement corresponded to classes of the position, force, and velocity motion characteristics. Independent classification models were trained to classify arm position, force, or velocity. For the flexion–extension trial group, three position classes (P1, P2, P3) were determined. As well, three force classes (0 N, +22 N, -22 N) were classified. For the velocity motion characteristic, motions were classified into three classes (stationary (0°/s), slow, fast) or two classes (stationary, moving).

For the ADL trial groups, these motions were separated into two force classes (11 N, 22 N), or two velocity classes (slow, fast). Predictors with known labels were used to train classifiers to receive predictors as inputs and output the expected class. Native MATLAB functions, `fitcdiscr` and `fitcecoc`, received sets of training predictors and labels to generate LDA and SVM models, respectively, for scripts see Appendix A.2.1. The method for checking classifier accuracy is outlined in the following section.

6.1.2 Model Accuracy

To test the accuracy of the trained classification models, a leave-one-subject-out (LOSO) cross-validation technique was used to exclude testing sets from training sets [37, 71, 72]. For the 24 subjects, the observations from Subjects 2–24 were used to train the classification model, then the model was tested by classifying the motion of Subject 1. Motions were classified using the native MATLAB function `predict(model, test_predictors)`. Inputs to this function included a trained model (`model`) and the new set of predictors (`test_predictors`), and the output was a vector of class labels that the model assigned to the predictors. These new labels, predicted by the trained models, were compared to the true labels corresponding to the test predictors, to determine how many labels the model assigned correctly to the motion predictors. Then Subject 2 was excluded from the training observation set (consisting of all remaining subjects, 1 and 3–24), and used as a test set for checking the classification accuracy. This was repeated, leaving each subject out of the training data then testing the model with that subject, to give a final averaged accuracy. This LOSO technique was used to give a better idea of model generalization and true accuracy without overfitting. The accuracies of the LDA and SVM models to assign the described predictors to the proper position, force, and velocity classes are demonstrated in the next section.

6.1.3 Classification Results

The results of training classifiers to detect motion characteristics are displayed in Table 6.1. Independent classification models were trained and tested using the LOSO cross-validation technique for four groups of motion trials: flexion–extension, ADL 1, ADL 2, or ADL1 and ADL 2 motion data.

Table 6.1: LDA and SVM motion characteristic classification accuracies using 11 EMG feature inputs from 15 arm and shoulder muscles.

Motion Type	**Characteristic**	**Classes**	**LDA Accuracy** [%]	**SVM Accuracy** [%]
flexion–extension	position	3 (P1, P2, P3)	78.70	83.02
flexion–extension	force	3 (0 N, +22 N, -22 N)	73.77	74.54
flexion–extension	velocity	3 (stationary, slow, fast)	43.98	47.22
flexion–extension	velocity	2 (stationary, moving)	67.90	71.91
ADL 1	force	2 (11 N, 22 N)	56.25	63.54
ADL 1	velocity	2 (slow, fast)	56.25	53.13
ADL 2	force	2 (11 N, 22 N)	53.13	59.38
ADL 2	velocity	2 (slow, fast)	53.13	57.29
ADL 1, ADL 2	force	2 (11 N, 22 N)	54.69	58.33
ADL 1, ADL 2	velocity	2 (slow, fast)	48.96	54.17

An LDA model was first trained with a flexion–extension observation set to determine the position (P1, P2, P3) of the arm during movements. This LDA model performed with a classification accuracy of 78.7 %. This accuracy improved to 83.02 % with an SVM model. Even with very diverse training data with varied force levels and velocity levels, position could be determined with less than 25 % error.

The accuracy of force classification was lower at 73.77 % and 74.54 % accuracies, for LDA and SVM models, respectively. In this case, not as much improvement was observed with the SVM classifier compared to the LDA classifier. However, interaction with the environment could be determined with just over 25 % error when arm position and velocity varied, influencing the EMG signals.

Both classifier types were very poor at determining three velocity classes (stationary, slow, fast), but accuracy improved when only two velocity levels (stationary, moving) were classified. Poor discrimination was expected between the slow and fast velocity classes as these were the goal velocities at which the participants were instructed to move. The actual joint rotation and hand speeds varied between participants, and within participant trials they could vary between repetition, even though participants were instructed to perform motions over consistent durations. With the statistical analysis, significant differences were found for EMG features pertaining to motions in these goal velocity classes, not only the position and force motion characteristics. Therefore, significant differences in signals levels do not necessarily indicate positive pattern recognition. However, both LDA and SVM type classifiers could better recognize patterns in EMG signals pertaining to isometric contractions compared to elbow flexion and extension. LDA models classified stationary contractions versus movement with 67.9 % accuracy, while SVM models were 71.91 % accurate.

Force and velocity classification was worse for ADL motions than the flexion–extension movements, as presented in Table 6.1. These classification levels were expected to be lower because the motions were more complex, and the force was applied constantly downwards not perpendicular to the forearm, meaning the torque experienced at the elbow could differ throughout the motion even though the force value felt at the hand was constant. As well, there was a smaller difference between the force levels (11 N and 22 N) for ADLs compared to the force levels (0 N, +22 N, -22 N) experienced during the basic flexion–extension motion trials. Also, there were fewer training motions for ADLs (4 ADL 1 motion trials per subject) relative to the 27 flexion–extension motion trials per subject, meaning that there were fewer training prediction/label observations for the classifiers.

There were minimal constraints for flexion–extension motions in this study, and the classifiers were trained with data covering various arm positions, force levels, and velocities. Motion constraints were limited and diverse training sets were used in an effort to make the classifier training sets of EMG signals more representative of the variety of elbow movements during day to day living. However, the much poorer classification of motion characteristics for ADLs compared to flexion–extension motions shows that the control problem of using EMG signals to determine intended elbow motion characteristics for ADLs is more complex than elbow flexion–extension with limited upper arm movement.

Usually, motion classifiers using arm muscle EMG input have been used to determine intended motion types such as wrist flexion–extension, grabbing objects, and pointing. In contrast, these results demonstrated the potential to determine more information about intended motion related to arm position and force. These initial classifiers had error rates above 10 %, the maximum error rate for a system classifying motions to be considered usable [35]. However, these initial classifiers determining force would most likely not be implemented in force control of a wearable device. The intention of these classifiers determining levels of motion characteristics and not purely motions, was to demonstrate the tangible impact of the motion characteristics on the EMG feature values from a variety of arm and shoulder muscles, and to use this information to better inform other control systems. For instance, knowing if a user is interacting with the environment or not (force classification) could be useful in guiding or tuning control models using EMG signal inputs. Further iterations of classifiers determining force levels, representing interactions between a person and their environment, and using motion characteristic information such as arm position and hand velocity as additional inputs to improve this classification is continued in the next section.

6.2 Classification Iterations

Purely classifying basic motion characteristics based on EMG signals as a control input would only provide basic control of a wearable device designed for rehabilitation. In these devices, the development of more complex control systems may require parameters that need to be tuned. Parameter values may be optimized using data from constrained motions performed at a specific velocity, and no environmental interaction. Or they may have been calibrated with the user's arm in one position (or shoulder orientation) instead of in a variety of arm positions that occur during activities of daily living. However, this study showed that arm position, force and interaction with the environment, and joint velocity do impact EMG signals significantly for many features and many arm muscles.

For such control schemes, it could be useful to have arm position, interaction forces, and velocities as known inputs if measurable. These motion characteristics could then direct the control path. For instance, if control model parameters are tuned for the model to perform optimally at a particular speed, the actual speed of motion could be used to tune these parameters further for other velocities as the motions are being performed. As well, knowing whether or not the device user is interacting with the environment, and with which force level, could be used to tune the control model or change the level of assistance or stiffness or compliance. For example, an assistive device may be required to provide more support if the user is holding a heavy object instead of just their own hand. However, higher muscle activation required to hold a heavy object stationary should not be mistaken as an intent to flex the elbow.

Arm position and joint velocity or hand speed are simpler to measure with a wearable device than interaction forces between a person's hand and the objects in the environment. Therefore, classifying interaction forces is the most important goal as it would not be possible to directly measure it. To improve the ability of models to correctly classify force during motion, the data included in training sets for pattern recognition models were modified. Since it was shown that arm position and joint speed influenced muscle activations in addition to force levels, arm position and hand velocities (related to joint rotation speeds) were included as inputs to classifiers in addition to EMG signal features. As well, the sizes of the EMG feature vectors were reduced. The specifics of the the predictor groups implemented in training are as follows.

6.2.1 Force Predictors

To compare to the previous classifications, all 11 EMG features for all 15 muscles measured were included as predictors in the first training sets. A modified training set that slightly reduced the number of muscles/features included was also generated. Only the specific EMG feature and muscle combinations found to have significant differences with changing force levels during flexion–extension motions (Chapter 5 Section 5.2) were included. However, the size of the feature vector was not reduced by a large amount. Thereby, the muscle/feature combinations included were reduced further to include only the popular and robust Hudgins set EMG features (MAV, SSC, WL, ZC) for the biceps brachii and triceps brachii muscles (BB_S, BB_L, TRI_LO, TRI_LAT, TRI_M). The muscles were limited to BB and TRI as devices have been controlled with only surface EMG signals from BB and TRI as inputs [35].

Since the influence of arm position and movement velocity on EMG signals has been shown, combinations of position labels (P1, P2, P3) and motion speed values and labels were included as predictors in addition to the muscle/feature EMG sets. Actual average hand speed (representative of joint rotation speed), or goal movement speed labels (stationary vs. moving, or stationary vs. slow vs. fast) were used as the speed predictors. The actual average hand speed was calculated from the KUKA robot handle x, y, z position measurements in the world frame collected during the performance of the motions. During elbow flexion–extension, the hand speed was related to the joint rotation speed. The speed was averaged over the entire motion duration. The labels corresponding to the predictor sets are outlined in the next section.

6.2.2 Classifier Training

The observation force labels remained the same as the initial classifications in Section 6.1.1.2, 0 N, +22 N, and -22 N for flexion–extension motions, and 11 N and 22 N for ADL motions. Again, the leave-one-subject-out technique was used to check the accuracy of the trained model with data observations for subjects that remained independent from the training dataset. An LDA classifier was implemented as it is commonly recommended and used to classify Hudgins set EMG signal features as the LDA model is robust, does not have extra parameters and can produce accuracies similar to other more complex models [35, 73]. The results of training the LDA classification models to identify force levels using the various predictor sets for flexion–extension motions and ADL movements are described in the following section.

6.2.3 Iterated Classification Results

The results of the LDA models trained with the new predictor sets, reducing the EMG muscle-/feature vectors and adding position and movement speed information, were compared to the original force classification systems. The goal was to improve the force classification accuracy. Flexion–extension and ADL motions were analyzed separately.

6.2.3.1 Improved Flexion–Extension Force Classification Results

Table 6.2 displays the results of LDA model force classification accuracies for flexion–extension motions using different predictor sets. Classification accuracy was given as percent correct, the Muscles/Features column outlines which EMG values were included as predictors, and the green cells indicate if position labels, actual average hand speed, or speed labels were included as predictors as well.

A few general trends in classification accuracy were observed. First of all, the overall classification accuracy using all feature 11 values for all 15 muscles (the full feature vector) resulted in the largest errors classifying 0 N, +22 N, and -22 N. Adding position labels or motion speed

Table 6.2: Accuracy of LDA model classifying three force levels (0 N, +22 N, -22 N) during elbow flexion–extension motions with varying predictor sets. Coloured boxes mean that the predictors named in the corresponding column title were included as inputs to the LDA model of the intersecting row.

Accuracy [%]	**Muscles/Features**	**Position Label**	**Actual Average Hand Speed**	**Speed Label (Stationary, Moving)**	**Speed Label (Stationary, Slow, Fast)**
73.77	all				
73.61	all	■			
73.92	all		■		
74.23	all			■	
74.38	all				■
73.30	all	■	■		
73.61	all	■		■	
72.84	all	■			■
75.00	force significance				
74.38	force significance	■			
75.31	force significance		■		
75.00	force significance	■	■		
76.85	BB, TRI, Hudgins set				
78.40	BB, TRI, Hudgins set	■			
77.62	BB, TRI, Hudgins set		■		
77.16	BB, TRI, Hudgins set			■	
77.16	BB, TRI, Hudgins set				■
78.40	BB, TRI, Hudgins set	■	■		
79.17	BB, TRI, Hudgins set	■		■	
78.55	BB, TRI, Hudgins set	■			■

information did not produce a large change in accuracy. The accuracies ranged from 72.84 % to 74.38 % correct. A possible explanation for the lack of change in accuracy is that the predictor sets were already so large with the full muscle/feature vector, that adding a single position and/or speed predictor did not make much difference. As well, the predictors included many muscle EMG feature values already influenced by position and speed so adding position and speed predictors was redundant.

Next, the classification accuracies of LDA models trained with a slightly reduced muscle/feature set (only the EMG muscle/feature combinations with significant differences with changing force levels) were slightly higher. These accuracies fell in the 74.38–75.31 % range. Adding the arm position or average hand speeds did not cause much difference in classification errors.

Most notably, the further reduced muscle/feature vector (Hudgins set features for BB and TRI) as predictors resulted in the highest classification accuracies ranging from 76.85 % to 79.17 %, as presented in Figure 6.1. This was promising, as collecting EMG signals from many muscles and the processing of the signals and extraction of features took time and computational power. Reducing the amount of EMG processing required in controlling a device in real time is advantageous. The lowest classification accuracy (76.85 %) using this reduced muscle/feature predictor vector occurred when only the EMG features were included as inputs to the LDA classifier without additional position and/or speed predictors. Although the increase in accuracies was small, augmenting the EMG predictor vector with arm position labels and speed labels (stationary, moving) resulted in the lowest error (classification accuracy of 79.17 %). Adding position labels reduced the error slightly more than adding only speed information. This demonstrated that position and speed information had some positive influence on classification of force with a reduced EMG feature predictor set. Further investigation to determine if these trends occurred for more complex ADL motions and smaller differences in force levels continues in the next section.

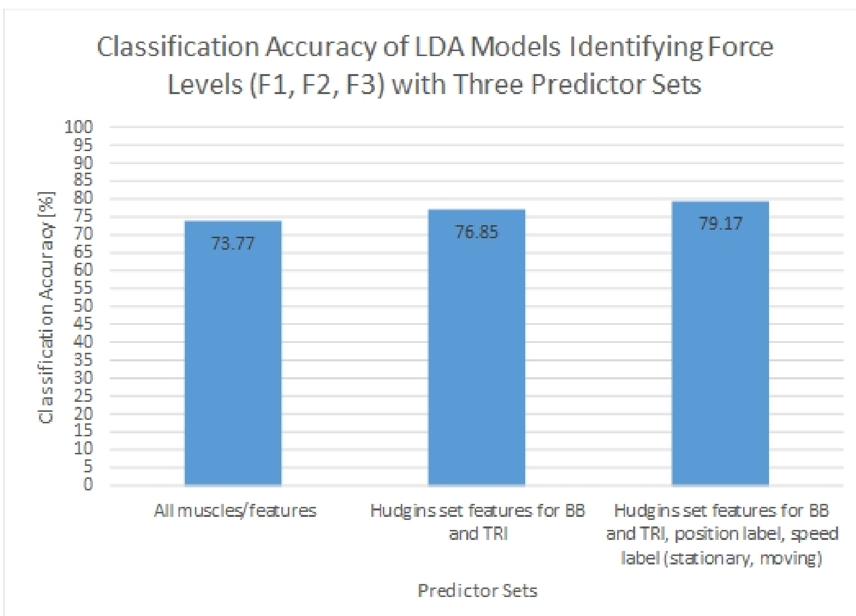

Figure 6.1: Classification accuracy of LDA models identifying force levels (F1, F2, F3) with three predictor sets during elbow flexion–extension.

6.2.3.2 Iterated ADL Force Classification Results

The LDA model force classification accuracy results for ADL motions with various predictor sets are displayed in Table 6.3.

Table 6.3: Accuracy of LDA model classifying two force levels (11 N, 22 N) during activities of daily living motions with varying predictor sets. Coloured boxes mean that the predictors named in the corresponding column title were included as inputs to the LDA model of the intersecting row.

Accuracy [%]	Motion Type	Muscles/Features	Actual Average Hand Speed	Speed Label (Slow, Fast)
56.25	ADL 1	all		
54.17	ADL 1	all		
53.13	ADL 2	all		
54.17	ADL 2	all		
54.69	ADL 1, ADL 2	all		
53.13	ADL 1, ADL 2	all		
59.38	ADL 1	BB, TRI, Hudgins set		
64.58	ADL 1	BB, TRI, Hudgins set		
63.54	ADL 1	BB, TRI, Hudgins set		
61.46	ADL 2	BB, TRI, Hudgins set		
60.42	ADL 2	BB, TRI, Hudgins set		
61.46	ADL 2	BB, TRI, Hudgins set		
58.85	ADL 1, ADL 2	BB, TRI, Hudgins set		
59.90	ADL 1, ADL 2	BB, TRI, Hudgins set		
58.33	ADL 1, ADL 2	BB, TRI, Hudgins set		

This table and Figure 6.2 show that a reduced muscle/feature vector of Hudgins set features for the BB and TRI (BB_S, BB_L, TRI_LO, TRI_LAT, TRI_M), instead of 11 features for each of 15 muscles, increased LDA model force classification accuracy for ADLs. The overall accuracies were much lower for the ADLs than basic flexion–extension, yet this decrease in error by using the reduced muscle/feature EMG set reflected the same trend observed with basic flexion–extension motions.

In contrast to the flexion–extension motions, arm position was not included as a predictor in this analysis of ADLs. The participants could not perform the ADL motions by remaining stationary, therefore, the only speed labels used as predictors were slow or fast, not stationary.

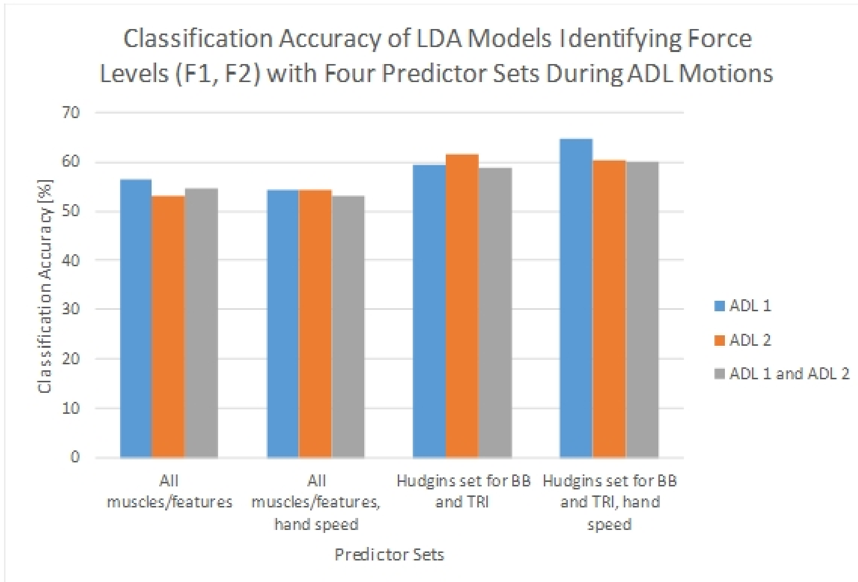

Figure 6.2: Classification accuracy of LDA models identifying force levels (F2, F3) with four predictor sets during ADLs.

Unlike flexion–extension movements, adding in speed information did not correspond with a consistent change in accuracy for identifying force levels during ADL 2 or the combination of ADL 1 and ADL 2 movements. However, providing speed information (actual average hand speed or goal speed labels (slow, fast)) as predictors in addition to the Hudgins set feature values for BB and TRI did increase the accuracy of force level (11 N, 22 N) classification during ADL 1. The addition of speed labels as classifier inputs increased the accuracy from 59.38 % correct, with only EMG signals as predictors, to 63.54 % correct. The highest classification accuracy was 64.58 % using actual average hand speed as an input in addition to the reduced EMG set, as shown in Table 6.3 and Figure 6.2.

These results were limited by amount of data available for training and testing the classifiers, as well, the motions were complex with little constraint and multiple factors (force, velocity) affecting the motion. The limited constraints and varying conditions is reflective of ADL motions outside of a laboratory environment. Even with these limitations, a 5 % increase in force classification accuracy was observed for ADL 1 when speed information was provided as an input to the classifier in addition to EMG features. Additionally, the lowest force classification error during flexion–extension motions resulted when position and speed information, along with the reduced EMG feature vector, was fed into the LDA model. This demonstrated that knowing arm positions and motion speeds influenced the results of classifying interaction forces. The amount of improvement that position and speed motion characteristics provide in force classification decisions based on EMG signals can be investigated further. The identification of interaction force levels means classifiers could be used to detect when a person is in contact with the environment.

6.3 Conclusion

This chapter presented the results of using the data collected during the performance of motions by healthy people to classify motion characteristics. EMG features were fed into LDA and SVM classification models, which then detected arm position, force levels, or movement. Informed by the previous EMG feature statistical analysis, iterations of the pattern recognition model training consisted of reducing the size of the EMG feature vector and adding position and motion speed predictors to the model inputs. More consistent improvements were seen in force prediction for flexion–extension motions compared to ADL movements. Motion characteristic information that influences EMG signals may be used to guide tuning of more complex control systems. Further applications of these findings are discussed in the following chapter, along with the main conclusions and contributions from this work.

Chapter 7

Conclusions and Future Work

The work presented in this thesis was aimed at analyzing the effects of motion characteristics on EMG signals, and using the trends to inform the investigation of classifying levels of motion characteristics, particularly force (interaction between a person and the environment). A literature review was performed to show the gaps in informed motion classification and control with various motion factors simultaneously influencing EMG signals. EMG signals were readily used as inputs to control systems to detect intended movements, however, they perform more poorly outside of a constrained laboratory environment. The effect of motion factors during unconstrained motion on EMG feature metrics for the various upper limb muscles would inform the use of EMG signals as control system inputs.

A repeated measures EMG study was designed to quantify the way in which EMG signal features changed with varying arm positions, interaction forces between the person and the environment, and motion velocities during unconstrained elbow flexion–extension and activities of daily living. In order to conduct the experiment, software was developed to calibrate and control a collaborative KUKA robot in the application of precise forces between the robot end effector and the participant to simulate interaction with the environment, and to collect kinematic and dynamic data.

EMG signals measured from 15 arm and shoulder muscles during motion trials were processed and analyzed. Significant differences were observed between various EMG feature values as arm position, or interaction force, or goal velocity changed. Fewer feature/muscle combinations had significant differences in values with changing velocities, than changing forces or arm positions. The knowledge of the impact of motion characteristics on EMG features across a range of upper extremity muscles is important to inform the use of EMG signals as control inputs for a smart wearable brace for motion assistance during rehabilitation.

Additionally, pattern recognition (LDA and SVM) models were fed EMG features and trained to detect levels of motion characteristics. Three arm positions and three classes of interaction with the environment were identified by two LDA models, one classifying arm position and one classifying force, with accuracies of 78.70 % and 73.77 %, respectively, during elbow flexion–extension. For comparison, in the literature, an LDA model was trained and tested with data with varying force levels, to identify ten hand position classes with an error of 19 % [35]. Three velocities of motion (stationary, slow, fast) could not be distinguished well with LDA or SVM models. However, isometric contractions versus active motion were determined during elbow flexion–extension, although the classification error was 32.10 %. Compared to basic elbow flexion–extension movements, models classifying forces and velocities had higher errors during ADLs. This is consistent with the usability of pattern recognition control systems only weakly correlated to offline accuracy in the literature [37, 74].

The results of the initial analysis of motion characteristics were used to increase the accuracy of force classification. Interaction forces were the motion characteristic of interest for classification as it is more complex to determine than arm position and joint velocity, with a basic wearable device. Including arm position and hand speed as inputs to an LDA model, in addition to a reduced EMG feature set (four features for biceps and triceps only), influenced the ability of the model to determine three force levels during elbow flexion–extension. The classification accuracy of LDA models predicting force levels increased from 76.85 %, without position or velocity inputs (only the reduced EMG set), to a range of 77.16–79.17 %, with position labels and/or velocity as additional inputs to the model. However, the same amount of improvement in classifying force levels was not observed during ADL 2. Yet, during ADL 1 movements, force classification error was reduced by 5.20 % to improve the classification accuracy to 64.58 % when hand speed was included as an input to the classifier, in addition to the reduced EMG set. This is comparable to the literature where a PCI model estimated force with 33.3 % error during arm contractions with varying force and velocity [55].

The results of these analyses showed that motion characteristics had significant effects on EMG signal features across a range of arm and shoulder muscles during unconstrained elbow flexion–extension motions, and less straightforward effects during more complex activities of daily living. Improvements in classifying forces from different types of interactions with the environment using position and speed information as well as EMG signals suggest that motion characteristic inputs can influence the use of EMG signals in control.

7.1 Contributions

The contributions of the work presented in this thesis are as follows:

- Software was designed to calibrate a KUKA robot and have the robot perform a repeated measures motion trial study simulating activities with specified forces, and collecting dynamic data of the motions.

- This work collected and processed a unique dataset of EMG signals and corresponding arm kinematic and dynamic data during dynamic motions. This can be used for further study and calibration of control systems that use EMG signals during complex unconstrained motions with changing arm positions, person–environment interaction forces, and motion velocities.

- Insight into EMG signals during unconstrained motion was gained. EMG signals were observed with combinations of multiple factors affecting the motion. Previous studies have looked at effects of motion velocity or force or position. This work confirmed the effects of arm position, forces from environmental interaction, and motion velocity on EMG signals. This study highlighted and quantified the effect of changing levels of each of these factors simultaneously, during various styles of movement (isometric contractions, basic flexion–extension, and activities of daily living). Statistical differences were observed in EMG features for a variety of arm muscles corresponding to changes in motion characteristics.

- EMG signal features were used to determine corresponding motion characteristics. Classification of intended arm position and interaction force was more accurate than classification of goal velocity.

- The significance of motion factors on EMG feature values highlighted with the statistical analyses led to the inclusion of motion characteristic information during classification of interaction forces. Detection of changes in interaction forces during dynamic and unconstrained motions using only EMG signals, arm position, and movement speed information was completed. Improvement of force classification was observed with a reduction in feature vector to four Hudgins set EMG features (MAV, SSC, WL, ZC) for BB and TRI muscles, and addition of arm position and hand speed information for the first time. It justifies the consideration of motion characteristics in further exploration of using EMG signals as inputs to a control system detecting intended elbow motion.

7.2 Recommendations for Future Work

Recognizing that the insights gained from this study can be applied further, and expansion of the experiment to include patient cohorts could identify more trends, research avenues to explore in future work are presented below:

- To provide smoother data collection, increase ease of modifying measurement parameters during trials, reduce the time of experimental trials, and reduce processing time and error introduced in synchronizing the data from various systems, EMG and KUKA robot data collection can be integrated at the time of measurement, and a graphical user interface (GUI) could be created for visualization and control.
- Expand the variety of motions measured, and ways in which motions can be characterized.
    - Test more unconstrained motions mimicking a wider range of activities of daily living. Ensure data are collected for clinically relevant movements that patients would be performing or attempting to perform during rehabilitation. Detection of motion characteristics was much less accurate for ADLs compared to basic elbow flexion–extension, but these complex unconstrained motions are the types of movements that wearable

devices would be required to assist users perform. Therefore, data for these motions is required to develop improved control systems.

– Label motion segments as elbow flexion versus extension, due to variability in EMG signals throughout movements. Motion characteristics can also be divided into a greater number of levels or classes to further breakdown the effects of motion characteristics on EMG signals and as inputs to classification models.

- Collect EMG from pectoral muscles and kinematic data of torso or the other side of the body in addition to arm and shoulder muscles. This would give more insight into compensatory movements due to impairment in movement and control elsewhere in the arm (wrist, elbow, shoulder), possibly indicative of difficulty moving the arm. As well, with higher forces being resisted, qualitatively it looked and felt like muscles on the opposite side of the body were being activated, this information could be useful if measured. During this investigation, the feet placement and body stance was free to vary slightly as participants were instructed to face a certain direction and hold their arm in a specified starting position, yet they could put their feet in whatever position and stance felt comfortable. This could have introduced some variability, and is another factor that could be constrained or varied as another indepedent variable, or observed and noted in future investigations. Since natural unconstrained movements were the focus of this investigation, recording of the body position rather than constraining it would be preferred.

- Repeat portions of the study with participants with a musculoskeletal injury, movement impairment, or going through the process of rehabilitation and physical therapy. Investigate differences in EMG metrics between participants with and without impairment, and indicators of the need for motion assistance. During testing with healthy participants, higher force movements at slow speeds with the shoulder abducted induced some fatigue, noted subjectively in qualitative comments. For patients with musculoskeletal conditions, the testing protocol may need to be adjusted to better fit their current capabilities to ensure they can complete the measurement activities without too much effort or causing more injury.

- Investigate the effects of tuning control models and choosing control pathways using motion characteristic classes, such as knowledge of contact with the environment, as inputs. Force classification could be part of a multi-tiered approach for control of wearable devices, with identification of intended levels of motion characteristics paired up with mathematical models in real time. These models used to control wearable devices can be optimized by setting multiple parameters, however, parameter settings may control devices best under constrained conditions of a single arm orientation, a specific motion velocity, or no interaction with the environment. Knowing if the device user is in contact with the environment, and with which force level, could guide the tuning of control model parameters or control the device at a high level by adjusting the device stiffness/compliance or level of assistance provided by the device to the user.

The purpose of this thesis was to gain insight into the effects of arm position, interaction forces, and velocity motion characteristics changing simultaneously on muscle activations measured as EMG signals. This knowledge was used to inform the calibration of models using EMG signals as inputs to detect intended motion, working towards the control of a wearable mechatronic device for rehabilitation. These objectives were accomplished first by developing software to use a KUKA robot to precisely control types of environmental interaction and measure motion kinematics and dynamics as EMG signals from a variety of upper extremity muscles were collected. Effects of motion factors on EMG signals were highlighted, then a pilot test, informed by the significant effects of motion characteristics on EMG signals, calibrated training sets of an LDA model for force classification. Continued work in using motion characteristic information to tune control systems, or in detecting intended motion characteristics, instead of merely gross motions, could lead to more accurately controlled wearable mechatronic devices for elbow rehabilitation, and possibly for devices actuating other joints.

References

[1] Canadian Institutes of Health Research and Institute of Musculoskeletal Health and Arthritis, "Institute of Musculoskeletal Health and Arthritis, Strategic Plan 2014-2018: Enhancing Musculoskeletal, Skin and Oral Health," tech. rep., 2014.

[2] Public Health Agency of Canada, "Life with arthritis in Canada: A personal and public health challenge," tech. rep., 2010.

[3] Public Services Health and Safety Association, "Musculoskeletal disorders," tech. rep., 2010. Available from: `https://www.scopus.com/inward/record.uri?eid=2-s2.0-0034580655{\&}partnerID=40{\&}md5=5af20e85a72ecb7ae232cd7b8e176e5f`.

[4] Infrastructure Health and Safety Association, "Musculoskeletal disorders and ergonomics.". Available from: `http://www.ihsa.ca/msd`.

[5] A. D. Woolf and B. Pfleger, "Burden of major musculoskeletal conditions," *Bulletin of the World Health Organization*, vol. 81, no. 9, pp. 646–656, 2003.

[6] K. E. Wilk, L. C. Macrina, E. L. Cain, J. R. Dugas, and J. R. Andrews, "Rehabilitation of the Overhead Athlete's Elbow," *Sports Health: A Multidisciplinary Approach*, vol. 4, no. 5, pp. 404–414, 2012. Available from: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3435939/pdf/10.1177{\_}1941738112455006.pdf`.

[7] I. Fusaro, S. Orsini, T. Sforza, R. Rotini, and M. G. Benedetti, "The use of braces in the rehabilitation treatment of post-traumatic elbow," *Joints*, vol. 2, no. 2, pp. 81–86, 2014.

[8] S. F. Bassett, "The assessment of patient adherence to physiotherapy rehabilitation," *New Zealand Journal of Physiotherapy*, vol. 31, no. 2, pp. 60–66, 2003.

[9] H. Zhou and H. Hu, "Human motion tracking for rehabilitation-A survey," *Biomedical Signal Processing and Control*, vol. 3, no. 1, pp. 1–18, 2008. Available from: `http://ac.els-cdn.com/S1746809407000778/1-s2.0-S1746809407000778-main.pdf?{\_}tid=3e40095a-0378-11e7-af3d-00000aacb362{\&}acdnat=1488920243{\_}45bf261d704e752ce06d0c61dabc90d3`.

[10] J. Mehrholz, M. Pohl, T. Platz, J. Kugler, and B. Elsner, "Electromechanical and robot-assisted arm training for improving activities of daily living, arm function, and arm muscle strength after stroke.," *The Cochrane database of systematic reviews*, vol. 11, no. 11, pp. 1–126, 2015. Available from: `http://doi.wiley.com/10.1002/14651858.CD006876.pub4{\%}5Cnhttp://onlinelibrary.wiley.com/doi/10.1002/14651858.CD006876.pub4/full`.

[11] R. A. R. C. Gopura, D. S. V. Bandara, K. Kiguchi, and G. K. I. Mann, "Developments in hardware systems of active upper-limb exoskeleton robots: A review," *Robotics and Autonomous Systems*, vol. 75, pp. 203–220, 2016. Available from: http://dx.doi.org/10.1016/j.robot.2015.10.001.

[12] C. Shirota, J. Jansa, J. Diaz, S. Balasubramanian, S. Mazzoleni, N. A. Borghese, and A. Melendez-calderon, "On the assessment of coordination between upper extremities : towards a common language between rehabilitation engineers , clinicians and neuroscientists," *Journal of NeuroEngineering and Rehabilitation*, pp. 1–14, 2016. Available from: http://dx.doi.org/10.1186/s12984-016-0186-x.

[13] J. Kristensen and A. Franklyn-Miller, "Resistance training in musculoskeletal rehabilitation: a systematic review," *British Journal of Sports Medicine*, vol. 46, no. 10, pp. 719–726, 2012. Available from: http://bjsm.bmj.com/content/bjsports/early/2011/07/25/bjsm.2010.079376.full.pdf.

[14] S. L. Wolf, P. A. Thompson, D. M. Morris, D. K. Rose, C. J. Winstein, E. Taub, C. Giuliani, and S. L. Pearson, "The EXCITE trial: Attributes of the Wolf Motor Function Test in patients with subacute stroke," *Neurorehabilitation and Neural Repair*, vol. 19, no. 3, pp. 194–205, 2005.

[15] C. J. van Andel, N. Wolterbeek, C. A. M. Doorenbosch, D. H. E. J. . Veeger, and J. Harlaar, "Complete 3D kinematics of upper extremity functional tasks," *Gait and Posture*, vol. 27, no. 1, pp. 120–127, 2008. Available from: http://ac.els-cdn.com/S0966636207000665/1-s2.0-S0966636207000665-main.pdf?{_}tid=c52b9a0a-8a4d-11e6-a8be-00000aacb361{\&}acdnat=1475597910{_}107c2442fef9963820e9474ddf30ddc6.

[16] I. A. Murray and G. R. Johnson, "A study of the external forces and moments at the shoulder and elbow while performing every day tasks.," *Clinical biomechanics (Bristol, Avon)*, vol. 19, pp. 586–94, jul 2004. Available from: http://resolver.scholarsportal.info/resolve/02680033/v19i0006/586{_}asotefewpedt.xmlhttp://www.ncbi.nlm.nih.gov/pubmed/15234482.

[17] D. J. Magermans, E. K. J. Chadwick, H. E. J. Veeger, and F. C. T. Van Der Helm, "Requirements for upper extremity motions during activities of daily living," *Clinical Biomechanics*, vol. 20, no. 6, pp. 591–599, 2005.

[18] C. Disselhorst-Klug, T. Schmitz-Rode, and G. Rau, "Surface electromyography and muscle force: Limits in sEMG–force relationship and new approaches for applications," *Clinical Biomechanics*, vol. 24, no. 3, pp. 225–235, 2009. Available from: http://dx.doi.org/10.1016/j.clinbiomech.2008.08.003.

[19] D. R. Bueno and L. Montano, "An optimized model for estimation of muscle contribution and human joint torques from sEMG information," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, pp. 3364–3367, 2012.

[20] S. Guo, M. Pang, B. Gao, H. Hirata, and H. Ishihara, "Comparison of sEMG-based feature extraction and motion classification methods for upper-limb movement," *Sensors (Basel, Switzerland)*, vol. 15, no. 4, pp. 9022–9038, 2015.

[21] K. Kiguchi and Y. Hayashi, "Motion estimation based on EMG and EEG signals to control wearable robots," in *Proceedings - 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013*, pp. 4213–4218, 2013.

[22] M. Coscia, V. C. Cheung, P. Tropea, A. Koenig, V. Monaco, C. Bennis, S. Micera, and P. Bonato, "The effect of arm weight support on upper limb muscle synergies during reaching movements," *Journal of Neuroengineering and Rehabilitation*, vol. 11, no. 22, pp. 1–15, 2014. Available from: `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3996016{\&}tool=pmcentrez{\&}rendertype=abstract{\%}5Cnhttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3996016{\%}7B{\&}{\%}7Dtool=pmcentrez{\%}7B{\&}{\%}7Drendertype=abstract`.

[23] D. A. Funk, K. N. An, B. F. Morrey, and J. R. Daube, "Electromyographic Analysis of Muscles Across the Elbow Joint," *Journal of Orthopaedic Research*, no. 2, pp. 5529–538, 1987.

[24] T. K. K. Koo and A. F. T. Mak, "Feasibility of using EMG driven neuromusculoskeletal model for prediction of dynamic movement of the elbow," *Journal of Electromyography and Kinesiology*, vol. 15, no. 1, pp. 12–26, 2005.

[25] T. Lenzi, S. M. M. De Rossi, N. Vitiello, and M. C. Carrozza, "Intention-based EMG control for powered exoskeletons," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 8, pp. 2180–2190, 2012.

[26] Z. O. Khokhar, Z. G. Xiao, C. Menon, and Others, "Surface EMG pattern recognition for real-time control of a wrist exoskeleton," *Biomedical engineering online*, vol. 9, no. 1, pp. 41–57, 2010. Available from: `http://dx.doi.org/10.1186/1475-925X-9-41{\%}5Cnhttps://biomedical-engineering-online.biomedcentral.com/track/pdf/10.1186/1475-925X-9-41?site=biomedical-engineering-online.biomedcentral.com{\%}5Cnhttp://www.biomedcentral.com/content/pdf/1475-925x-9-41.pdf`.

[27] J. L. G. Nielsen, S. Holmgaard, N. Jiang, K. B. Englehart, D. Farina, and P. A. Parker, "Simultaneous and proportional force estimation for multifunction myoelectric prostheses using mirrored bilateral training," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 3, pp. 681–688, 2011.

[28] R. Koeppen, M. E. Huber, D. Sternad, and N. Hogan, "Controlling physical interactions: Humans do not minimize muscle effort," in *ASME Dynamic Systems and Control (DSC) Conference*, pp. 1–8, 2017.

[29] K. Saber-Sheikh, E. C. Bryant, C. Glazzard, A. Hamel, and R. Y. W. Lee, "Feasibility of using inertial sensors to assess human movement," *Manual Therapy*, vol. 15, no. 1, pp. 122–125, 2010. Available from: `http://dx.doi.org/10.1016/j.math.2009.05.009`.

[30] H. Zhou, T. Stone, H. Hu, and N. Harris, "Use of multiple wearable inertial sensors in upper limb motion tracking," *Medical Engineering and Physics*, vol. 30, no. 1, pp. 123–133, 2008.

[31] S. Obdrzalek, G. Kurillo, F. Ofli, R. Bajcsy, E. Seto, H. Jimison, and M. Pavel, "Accuracy and robustness of Kinect pose estimation in the context of coaching of elderly population," *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, pp. 1188–1193, 2012.

[32] X. Xu and R. W. McGorry, "The validity of the first and second generation Microsoft Kinect for identifying joint center locations during static postures," *Applied Ergonomics*, vol. 49, pp. 47–54, 2015.

[33] R. J. Adams, M. D. Lichter, E. T. Krepkovich, A. Ellington, M. White, and P. T. Diamond, "Assessing Upper Extremity Motor Function in Practice of Virtual Activities of Daily Living," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 2, pp. 287–296, 2015.

[34] H. J. Hermens, B. Freriks, C. Disselhorst-Klug, and G. Rau, "Development of recommendations for SEMG sensors and sensor placement procedures," *Journal of Electromyography and Kinesiology*, vol. 10, no. 5, pp. 361–374, 2000.

[35] M. Hakonen, H. Piitulainen, and A. Visala, "Current state of digital signal processing in myoelectric interfaces and related applications," *Biomedical Signal Processing and Control*, vol. 18, pp. 334–359, 2015. Available from: `http://dx.doi.org/10.1016/j.bspc.2015.02.009`.

[36] E. Scheme and K. Englehart, "Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use," *Journal of Rehabilitation Research and Development*, vol. 48, no. 6, pp. 643–659, 2016.

[37] D. Yang, Y. Gu, L. Jiang, L. Osborn, and H. Liu, "Dynamic training protocol improves the robustness of PR-based myoelectric control," *Biomedical Signal Processing and Control*, vol. 31, pp. 249–256, 2017. Available from: `http://dx.doi.org/10.1016/j.bspc.2016.08.017`.

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

[39] D. Farina, "Interpretation of the surface electromyogram in dynamic contractions," *Exercise and Sport Sciences Reviews*, vol. 34, no. 3, pp. 121–127, 2006.

[40] R. Boostani and M. H. Moradi, "Evaluation of the forearm EMG signal features for the control of a prosthetic hand," *Physiol. Meas*, vol. 24, pp. 309–319, 2003. Available from: `https://pdfs.semanticscholar.org/6236/dbce3909cbb7d6bf45af3fb307f9f3da81f8.pdf`.

[41] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for EMG signal classification," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7420–7431, 2012. Available from: `http://dx.doi.org/10.1016/j.eswa.2012.01.102`.

[42] E. Scheme and K. Englehart, "Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use," *Journal of Rehabilitation Research and Development*, vol. 48, no. 6, pp. 643–660, 2011. Available from: `http://www.rehab.research.va.gov/jour/11/486/pdf/scheme486.pdf`.

[43] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau, "EMG feature evaluation for improving myoelectric pattern recognition robustness," *Expert Systems with Applications*, 2013.

[44] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE Trans Biomed Eng*, vol. 50, no. 7, pp. 848–854, 2003. Available from: `http://www.ncbi.nlm.nih.gov/pubmed/12848352{\%}5Cnhttp://ieeexplore.ieee.org/ielx5/10/27145/01206493.pdf?tp={\&}arnumber=1206493{\&}isnumber=27145`.

[45] R. H. Chowdhury, M. B. I. Reaz, M. A. B. M. Ali, A. A. A. Bakar, K. Chellappan, and T. G. Chang, "Surface electromyography signal processing and classification techniques.," *Sensors (Basel, Switzerland)*, vol. 13, no. 9, pp. 12431–12466, 2013.

[46] B. Hudgins, P. Parker, and R. N. Scott, "A New Strategy for Multifunction Myoelectric Control," *IEEE Transactions on Biomedical Engineering*, vol. 40, no. 1, pp. 82–94, 1993.

[47] R. Merletti, M. Aventaggiato, A. Botter, A. Holobar, H. Marateb, and T. M. M. Vieira, "Advances in surface EMG: recent progress in detection and processing techniques.," *Critical reviews in biomedical engineering*, vol. 38, no. 4, pp. 305–345, 2010. Available from: `http://www.ncbi.nlm.nih.gov/pubmed/21133837`.

[48] T. Lorrain, N. Jiang, and D. Farina, "Influence of the training set on the accuracy of surface EMG classification in dynamic contractions for the control of multifunction prostheses," *Journal of NeuroEngineering and Rehabilitation*, vol. 8, no. 1, p. 25, 2011. Available from: `http://www.jneuroengrehab.com/content/8/1/25{\%}5Cnhttp://jneuroengrehab.biomedcentral.com/articles/10.1186/1743-0003-8-25`.

[49] A. Fougner, E. Scheme, A. D. C. Chan, K. Englehart, and Ø. Stavdahl, "Resolving the limb position effect in myoelectric pattern recognition," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 6, pp. 644–651, 2011.

[50] S. C. F. A. V. Werder, T. Kleiber, and C. Disselhorst-klug, "A method for a categorized and probabilistic analysis of the surface electromyogram in dynamic contractions," *Frontiers in Physiology*, vol. 6, no. February, pp. 1–8, 2015.

[51] K.-h. Park, H.-i. Suk, and S.-w. Lee, "Position-independent decoding of movement intention for proportional myoelectric interfaces," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 9, pp. 928–939, 2016.

[52] S. C. F. A. von Werder and C. Disselhorst-Klug, "The role of biceps brachii and brachioradialis for the control of elbow flexion and extension movements," *Journal of Electromyography and Kinesiology*, vol. 28, pp. 67–75, 2016. Available from: `http://dx.doi.org/10.1016/j.jelekin.2016.03.004`.

[53] M. Farooq and A. A. Khan, "Effects of shoulder rotation combined with elbow flexion on discomfort and EMG activity of ECRB muscle," *International Journal of Industrial Ergonomics*, vol. 44, no. 6, pp. 882–891, 2014. Available from: `http://dx.doi.org/10.1016/j.ergon.2013.10.010`.

[54] N. Jiang, S. Muceli, B. Graimann, and D. Farina, "Effect of arm position on the prediction of kinematics from EMG in amputees," *Medical and Biological Engineering and Computing*, vol. 51, no. 1, pp. 143–151, 2013.

[55] J. Hashemi, E. Morin, P. Mousavi, and K. Hashtrudi-Zaad, "Enhanced dynamic EMG-force estimation through calibration and PCI modeling," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 1, pp. 41–50, 2015.

[56] N. Popovic, S. Williams, T. Schmitz-Rode, G. Rau, and C. Disselhorst-Klug, "Robot-based methodology for a kinematic and kinetic analysis of unconstrained, but reproducible upper extremity movement," *Journal of Biomechanics*, vol. 42, no. 10, pp. 1570–1573, 2009. Available from: `http://ac.els-cdn.com/S0021929009001985/1-s2.0-S0021929009001985-main.pdf?{\_}tid=67a429b2-8a4b-11e6-86fb-00000aacb35e{\&}acdnat=1475596894{\_}59d5eaf7a2901eab138b9f427799c120`.

[57] F. Oscari, C. Finetto, S. A. Kautz, and G. Rosati, "Changes in muscle coordination patterns induced by exposure to a viscous force field.," *Journal of neuroengineering and rehabilitation*, vol. 13, no. 1, pp. 58–73, 2016. Available from: `http://www.ncbi.nlm.nih.gov/pubmed/27305944{\%}5Cnhttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4910356`.

[58] D. Blana, T. Kyriacou, J. M. Lambrecht, and E. K. Chadwick, "Feasibility of using combined EMG and kinematic signals for prosthesis control: A simulation study using a virtual reality environment," *Journal of Electromyography and Kinesiology*, vol. 29, pp. 21–27, 2016. Available from: `http://dx.doi.org/10.1016/j.jelekin.2015.06.010http://ac.els-cdn.com/S1050641115001467/1-s2.0-S1050641115001467-main.pdf?{\_}tid=1b5f840e-9ac5-11e6-a907-00000aab0f27{\&}acdnat=1477408384{\_}53f9972701e0152cc46abe5ed6808fda`.

[59] R. Feeney, "Specific anthropometric and strength data for people with dexterity disability.," tech. rep., 2002.

[60] K. H. Liao, "Optimal Handle Grip Span for Maximum Hand Grip Strength and Accurate Grip Control Strength Exertion According to Individual Hand Size," *Journal of Osteoporosis and Physical Activity*, vol. 4, no. 2, pp. 2–7, 2016. Available from: `http://www.esciencecentral.org/journals/optimal-handle-grip-span-for-maximum-hand-grip-strength-and-accurate-gripcontrol-strength` `php?aid=75749`.

[61] C. C. for Occupational Health and Safety, "Hand tool ergonomics - tool design : Osh answers." `https://www.ccohs.ca/oshanswers/ergonomics/handtools/tooldesign.html`, jan 2018. (Accessed on 01/12/2018).

[62] M. Patkin, "A Check-List for Handle Design," tech. rep., 2001. Available from: `http://ergonomics.uq.edu.au/eaol/handle.pdf`.

[63] Delsys Incorporated, "Trigno Wireless System User's Guide," 2014. Available from: `http://delsys.com/Attachments{\_}pdf/download/users-guides/trigno-wireless-systems.pdf`.

[64] KUKA Laboratories GmbH, "KUKA Sunrise.OS 1.5 KUKA Sunrise.Workbench 1.5 V1," 2014.

[65] E. Scheme, B. Lock, L. Hargrove, W. Hill, U. Kuruganti, and K. Englehart, "Motion normalized proportional control for improved pattern recognition-based myoelectric control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 1, pp. 149–157, 2014.

[66] E. Farago, *Development of an EMG-based muscle health model for elbow trauma patients.* mongraph, The University of Western Ontario, 2018.

[67] M. A. Oskoei and H. Hu, "Support Vector Machine-Based Classification Scheme for Myoelectric Control Applied to Upper Limb," *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, vol. 55, no. 8, pp. 1956–1965, 2008.

[68] M. Atzori, A. Gijsberts, I. Kuzborskij, S. Elsig, A. G. Mittaz Hager, O. Deriaz, C. Castellini, H. Müller, and B. Caputo, "Characterization of a benchmark database for myoelectric movement classification," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 1, pp. 73–83, 2015.

[69] L. H. Smith, L. J. Hargrove, B. A. Lock, and T. A. Kuiken, "Determining the Optimal Window Length for Pattern Recognition-Based Myoelectric Control: Balancing the Competing Effects of Classification Error and Controller Delay," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 2, pp. 186–192, 2014.

[70] A. Phinyomark, C. Limsakul, and P. Phukpattaranont, "A Novel Feature Extraction for Robust EMG Pattern Recognition," *JOURNAL OF COMPUTING*, vol. 1, no. 1, pp. 71–80, 2009. Available from: `https://arxiv.org/ftp/arxiv/papers/0912/0912.3973.pdf`.

[71] K.-h. Park and S.-w. Lee, "Movement intention decoding based on deep learning for multiuser myoelectric interfaces," in *4th International Winter Conference on Brain-Computer Interface, BCI 2016*, pp. 7–8, 2016.

[72] J. Yousefi and A. Hamilton-Wright, "Characterizing EMG data using machine-learning tools," *Computers in Biology and Medicine*, vol. 51, pp. 1–13, 2014.

[73] N. Nazmi, M. Abdul Rahman, S.-I. Yamamoto, S. Ahmad, H. Zamzuri, and S. Mazlan, "A Review of Classification Techniques of EMG Signals during Isotonic and Isometric Contractions," *Sensors*, vol. 16, no. 8, pp. 1–28, 2016. Available from: `http://www.mdpi.com/1424-8220/16/8/1304`.

[74] B. Lock, K. B. Englehart, and B. Hudgins, "Real-time myoelectric control in a virtual environment to relate usability vs. accuracy," in *Proceedings of the 2005 MyoElectric Controls/Powered Prosthetics Symposium*, pp. 17–20, 2005. Available from: `http://dukespace.lib.duke.edu/dspace/handle/10161/2721`.

Appendix A

Code Used

A.1 KUKA Robotic Programs

SetStartPositions.java

```java
package application;


import java.util.concurrent.TimeUnit;

import javax.inject.Inject;

import com.kuka.generated.ioAccess.MediaFlangeIOGroup;
import com.kuka.roboticsAPI.applicationModel.RoboticsAPIApplication;
import static com.kuka.roboticsAPI.motionModel.BasicMotions.*;

import com.kuka.roboticsAPI.conditionModel.BooleanIOCondition;
import com.kuka.roboticsAPI.conditionModel.MotionPathCondition;
import com.kuka.roboticsAPI.conditionModel.ReferenceType;
import com.kuka.roboticsAPI.controllerModel.Controller;
import com.kuka.roboticsAPI.deviceModel.LBR;
import com.kuka.roboticsAPI.geometricModel.CartDOF;
import com.kuka.roboticsAPI.geometricModel.Tool;
import com.kuka.roboticsAPI.geometricModel.World;
import com.kuka.roboticsAPI.geometricModel.math.MathUtils;
import com.kuka.roboticsAPI.ioModel.AbstractIO;
import com.kuka.roboticsAPI.motionModel.controlModeModel.CartesianImpedanceControlMode;
import com.kuka.roboticsAPI.motionModel.controlModeModel.CartesianSineImpedanceControlMode;
import com.kuka.roboticsAPI.sensorModel.DataRecorder;
import com.kuka.roboticsAPI.sensorModel.DataRecorder.AngleUnit;
import com.kuka.roboticsAPI.sensorModel.StartRecordingAction;
import com.kuka.roboticsAPI.uiModel.ApplicationDialogType;
import com.sun.org.apache.xml.internal.utils.StringVector;

/**
 * Implementation of a robot application.
 * <p>
 * The application provides a {@link RoboticsAPITask#initialize()} and a
 * {@link RoboticsAPITask#run()} method, which will be called successively in
 * the application lifecycle. The application will terminate automatically after
 * the {@link RoboticsAPITask#run()} method has finished or after stopping the
 * task. The {@link RoboticsAPITask#dispose()} method will be called, even if an
 * exception is thrown during initialization or run.
 * <p>
 * <b>It is imperative to call <code>super.dispose()</code> when overriding the
 * {@link RoboticsAPITask#dispose()} method.</b>
 *
 * @see UseRoboticsAPIContext
 * @see #initialize()
 * @see #run()
 * @see #dispose()
 */
public class SetStartPositions extends RoboticsAPIApplication {
    private Controller kuka_Sunrise_Cabinet_1;
    private LBR lbr_iiwa_14_R820_1;

    private Tool handle;

    private String[] recNameD = {"RF","RE"}; // RF = resist flexion, RE = resist extension
    private String[] recNameP = {"P1","P2","P3","P4","P5"}; //arm position P1=down, P2=front, P3=side, P4=book, P5=cup
    private String[] recNameA = {"A1","A2","A3","A4","A5", "A6"}; //angle A1=0, A2=90, A3=150, A4=book, A5=cup, A6=flex/ext
    private String[] recNameF = {"F1","F2","F3"}; //force
    private String[] recNameV = {"V1","V2","V3"}; //velocity v1 = stationary, v2 = slow, v3 = fast
    private String[] recNameR = {"R1","R2","R3","R4","R5"}; //rep
    private int numReps = 3; //set the total number of reps
```

```java
    private String recName;
    private String MVCrecName;
    private int recTime;

    private String ADLStartPos;
    private String SingleMotionStartPos;
    private String StatStartPos;

    private int direction;
    private int position;
    private int angle;
    private int force;
    private int vel;
    private int rep;

    private int adjust;

    private int set;
    private int[] forceVal = {0,22,11};
    private int[] XYstiffness = {0, 0, 0}; //XYstiffnesses to match with forceVal

    private int forceAbs;

    private int choice;
    private int choiceMVC;

    @Inject
    private MediaFlangeIOGroup flangeIO; //using media flange IO

    @Override
    public void initialize() {
        kuka_Sunrise_Cabinet_1 = getController("KUKA_Sunrise_Cabinet_1");
        lbr_iiwa_14_R820_1 = (LBR) getDevice(kuka_Sunrise_Cabinet_1,
                "LBR_iiwa_14_R820_1");
    }

    @Override
    public void run() {
        lbr_iiwa_14_R820_1.setESMState("1");

        //attach handle tool to robot
        handle = getApplicationData().createFromTemplate("Handle1");
        getLogger().info("Attaching handle tool");
        handle.attachTo(lbr_iiwa_14_R820_1.getFlange());

        flangeIO = new MediaFlangeIOGroup(kuka_Sunrise_Cabinet_1);
        flangeIO.setLEDBlue(false);

        AbstractIO greenButton = flangeIO.getInput("UserButton");
        BooleanIOCondition greenButton_active = new BooleanIOCondition(greenButton, true);

        //holding mode
        CartesianImpedanceControlMode holdImpCon = new CartesianImpedanceControlMode();
        holdImpCon.parametrize(CartDOF.TRANSL).setStiffness(5000);
        holdImpCon.parametrize(CartDOF.ROT).setStiffness(300);
        holdImpCon.parametrize(CartDOF.Z).setAdditionalControlForce(0);


        int Hand = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Dominant hand?", "Right", "Left");

        switch(Hand){
```

```
        case 0: //right hand
            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/ADLPounce")).setJointVelocityRel(0.3));
            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initADL1bookStart")).setJointVelocityRel(0.3));
            getLogger().info("Holding force ADL1bookStart, update, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));

            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initADL2cupStart")).setJointVelocityRel(0.3));
            getLogger().info("Holding force ADL2cupStart, update, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));

            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initMVCStart")).setJointVelocityRel(0.3));
            getLogger().info("Holding force MVCStart and P1A2, update both, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));

            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initSingleMotionStartP1Down")).setJointVelocityRel(0.3));
            getLogger().info("Holding force P1Down, update both, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));

            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/SingleMotionPounce")).setJointVelocityRel(0.3));
            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initSingleMotionStartP2Front")).setJointVelocityRel(0.3));
            getLogger().info("Holding force P2Front, update both, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));

            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/SingleMotionPounce")).setJointVelocityRel(0.3));
            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initSingleMotionStartP3Side")).setJointVelocityRel(0.3));
            getLogger().info("Holding force P3Side, update both, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));


            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/SingleMotionPounce")).setJointVelocityRel(0.3));
            getLogger().info("Done");
            break;
        case 1: //left hand
            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/ADLPounce")).setJointVelocityRel(0.3));
            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initLADL1bookStart")).setJointVelocityRel(0.3));
            getLogger().info("Holding force ADL1bookStart, update, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
```

```java
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));

            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initLADL2cupStart")).setJointVelocityRel(0.3));
            getLogger().info("Holding force ADL2cupStart, update, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));

            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initLMVCStart")).setJointVelocityRel(0.3));
            getLogger().info("Holding force MVCStart and P1A2, update both, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));

            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initLSingleMotionStartP1Down")).setJointVelocityRel(0.3));
            getLogger().info("Holding force P1Down, update both, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));

            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/SingleMotionPounce")).setJointVelocityRel(0.3));
            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initLSingleMotionStartP2Front")).setJointVelocityRel(0.3));
            getLogger().info("Holding force P2Front, update both, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));

            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/SingleMotionPounce")).setJointVelocityRel(0.3));
            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/initLSingleMotionStartP3Side")).setJointVelocityRel(0.3));
            getLogger().info("Holding force P3Side, update both, then press green button");
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, -1, TimeUnit.SECONDS).breakWhen(greenButton_active));
            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS));


            handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/SingleMotionPounce")).setJointVelocityRel(0.3));
            getLogger().info("Done");
            break;
        }

        handle.detach();

    }

    /**
     * Auto-generated method stub. Do not modify the contents of this method.
     */
    public static void main(String[] args) {
        SetStartPositions app = new SetStartPositions();
        app.runApplication();
    }
}
```

MeasuringForDynamicCalibration.java

```java
package application;


import java.util.concurrent.TimeUnit;

import javax.inject.Inject;

import com.kuka.generated.ioAccess.MediaFlangeIOGroup;
import com.kuka.roboticsAPI.applicationModel.RoboticsAPIApplication;
import static com.kuka.roboticsAPI.motionModel.BasicMotions.*;

import com.kuka.roboticsAPI.conditionModel.BooleanIOCondition;
import com.kuka.roboticsAPI.conditionModel.MotionPathCondition;
import com.kuka.roboticsAPI.conditionModel.ReferenceType;
import com.kuka.roboticsAPI.controllerModel.Controller;
import com.kuka.roboticsAPI.deviceModel.LBR;
import com.kuka.roboticsAPI.geometricModel.CartDOF;
import com.kuka.roboticsAPI.geometricModel.Tool;
import com.kuka.roboticsAPI.geometricModel.World;
import com.kuka.roboticsAPI.geometricModel.math.MathUtils;
import com.kuka.roboticsAPI.ioModel.AbstractIO;
import com.kuka.roboticsAPI.motionModel.controlModeModel.CartesianImpedanceControlMode;
import com.kuka.roboticsAPI.motionModel.controlModeModel.CartesianSineImpedanceControlMode;
import com.kuka.roboticsAPI.sensorModel.DataRecorder;
import com.kuka.roboticsAPI.sensorModel.DataRecorder.AngleUnit;
import com.kuka.roboticsAPI.sensorModel.StartRecordingAction;
import com.kuka.roboticsAPI.uiModel.ApplicationDialogType;
import com.sun.org.apache.xml.internal.utils.StringVector;

/**
 * Implementation of a robot application.
 * <p>
 * The application provides a {@link RoboticsAPITask#initialize()} and a
 * {@link RoboticsAPITask#run()} method, which will be called successively in
 * the application lifecycle. The application will terminate automatically after
 * the {@link RoboticsAPITask#run()} method has finished or after stopping the
 * task. The {@link RoboticsAPITask#dispose()} method will be called, even if an
 * exception is thrown during initialization or run.
 * <p>
 * <b>It is imperative to call <code>super.dispose()</code> when overriding the
 * {@link RoboticsAPITask#dispose()} method.</b>
 *
 * @see UseRoboticsAPIContext
 * @see #initialize()
 * @see #run()
 * @see #dispose()
 */
public class MeasuringForDynamicCalibration extends RoboticsAPIApplication {
    private Controller kuka_Sunrise_Cabinet_1;
    private LBR lbr_iiwa_14_R820_1;

    private Tool handle;

    private String[] recNameD = {"RF","RE"}; // RF = resist flexion, RE = resist extension
    private String[] recNameP = {"P1","P2","P3","P4","P5"}; //arm position P1=down, P2=front, P3=side, P4=book, P5=cup
    private String[] recNameA = {"A1","A2","A3","A4","A5", "A6"}; //angle A1=0, A2=90, A3=150, A4=book, A5=cup, A6=flex/ext
    private String[] recNameF = {"F1","F2","F3"}; //force
    private String[] recNameV = {"V1","V2","V3"}; //velocity v1 = stationary, v2 = slow, v3 = fast
    private String[] recNameR = {"R1","R2","R3","R4","R5"}; //rep
    private int numReps = 3; //set the total number of reps
```

```
    private String recName;
    private String MVCrecName;
    private int recTime;

    private String ADLStartPos;
    private String SingleMotionStartPos;
    private String StatStartPos;

    private int direction;
    private int position;
    private int angle;
    private int force;
    private int vel;
    private int rep;

    private int adjust;
    private int trial;
    private int trialType; // 0 = stationary, 1 = flexion/extension, 2 = ADL
    //trial labels for force direction, position, angle, force level, velocity
    private int[] trialDir = {0,0,1,0,0,
            1,0,0,1,0,
            0,1,0,0,0,
            0,1,1,0,0,
            0,0,1,1,0,
            0,0,0,1,1,
            0,0,0,0,0,
            0,0,0};
    private int[] trialPos = {0,0,0,0,0,
            0,1,1,1,2,
            2,2,0,0,0,
            0,0,0,1,1,
            1,1,1,1,2,
            2,2,2,2,2,
            3,3,3,3,4,
            4,4,4};
    private int[] trialAng = {0,0,0,1,1,
            1,0,0,0,0,
            0,0,5,5,5,
            5,5,5,5,5,
            5,5,5,5,5,
            5,5,5,5,5,
            3,3,3,3,4,
            4,4,4};
    private int[] trialForce = {0,1,1,0,1,
            1,0,1,1,0,
            1,1,0,0,1,
            1,1,1,0,0,
            1,1,1,1,0,
            0,1,1,1,1,
            2,2,1,1,2,
            2,1,1};
    private int[] trialVel = {0,0,0,0,0,
            0,0,0,0,0,
            0,0,1,2,1,
            2,1,2,1,2,
            1,2,1,2,1,
            2,1,2,1,2,
            1,2,1,2,1,
            2,1,2};

    private int[] orderDir = {0,0,0,0,0,
            0,0,0,0,0,
```

```java
            0,0,0,0,0};
    private int[] orderPos = {0,0,0,0,0,
            0,0,0,0,0,
            0,0,0,0,0};
    private int[] orderAng = {0,1,2,3,4,
            0,1,2,3,4,
            0,1,2,3,4};
    private int[] orderForce = {1,1,1,1,1,
            2,2,2,2,2,
            0,0,0,0,0};
    private int[] orderVel = {0,0,0,0,0,
            0,0,0,0,0,
            0,0,0,0,0};
    private int set;
    private int[] forceVal = {0,22,11};
    private int[] XYstiffness = {0, 0, 0}; //XYstiffnesses to match with forceVal

    private int forceAbs;

    //private boolean continueTest = true;
    private int choice;
    private int choiceMVC;

    @Inject
    private MediaFlangeIOGroup flangeIO; //using media flange IO

    @Override
    public void initialize() {
        kuka_Sunrise_Cabinet_1 = getController("KUKA_Sunrise_Cabinet_1");
        lbr_iiwa_14_R820_1 = (LBR) getDevice(kuka_Sunrise_Cabinet_1,
                "LBR_iiwa_14_R820_1");
    }

    @Override
    public void run() {
        /*
         *
         * Initialization
         *
         */
        lbr_iiwa_14_R820_1.setESMState("1");

        //attach handle tool to robot
        handle = getApplicationData().createFromTemplate("Handle1");
        getLogger().info("Attaching handle tool");
        handle.attachTo(lbr_iiwa_14_R820_1.getFlange());

        flangeIO = new MediaFlangeIOGroup(kuka_Sunrise_Cabinet_1);
        flangeIO.setLEDBlue(false);

        // set starting values of some variables
        int size = orderDir.length;
        choice = 0;
        set = -1;
        choiceMVC = 0;

        //setting up recording data to log file (SI manual 15.25), enabled and started later below
        recName = new String(recNameD[direction]+recNameP[position]+recNameA[angle]+recNameF[force]+recNameV
[vel]+recNameR[rep]);
        DataRecorder rec = new DataRecorder(recName, -1, TimeUnit.SECONDS, 1);
        rec.addCartesianForce(handle.getFrame("/Handle1Con"), null); //force at handle middle, orientation
frame of handle middle
```

```java
        rec.addCartesianForce(handle.getFrame("/Handle1Con"), World.Current.getRootFrame()); //force at handle middle, orientation frame of world root
        rec.addCurrentCartesianPositionXYZ(handle.getFrame("/Handle1Con"), World.Current.getRootFrame()); //cartesian position of handle middle, orientation of world root
        rec.addCurrentJointPosition(lbr_iiwa_14_R820_1, AngleUnit.Degree); //joint positions, angles in degrees

        if(direction == 0){
            forceVal[0] = 0;
            forceVal[1] = 22;
            forceVal[2] = 11;
        }else if(direction == 1){
            forceVal[0] = 0;
            forceVal[1] = -22;
            forceVal[2] = -11;
        }

        forceAbs = Math.abs(forceVal[force]);

        //mode for holding stationary
        CartesianImpedanceControlMode softImpCon = new CartesianImpedanceControlMode();
        softImpCon.parametrize(CartDOF.X).setStiffness(XYstiffness[force]);
        softImpCon.parametrize(CartDOF.Y).setStiffness(XYstiffness[force]);
        softImpCon.parametrize(CartDOF.Z).setStiffness(0);
        softImpCon.parametrize(CartDOF.ALL).setDamping(0.7); //recommended default value for general use given in manual
        softImpCon.parametrize(CartDOF.ROT).setStiffness(300);
        softImpCon.parametrize(CartDOF.Z).setAdditionalControlForce(forceVal[force]);

        // mode for stationary tests
        CartesianSineImpedanceControlMode softSineImpCon = new CartesianSineImpedanceControlMode();
        softSineImpCon.parametrize(CartDOF.X).setStiffness(XYstiffness[force]);
        softSineImpCon.parametrize(CartDOF.Y).setStiffness(XYstiffness[force]);
        softSineImpCon.parametrize(CartDOF.Z).setStiffness(0);
        softSineImpCon.parametrize(CartDOF.ALL).setDamping(0.7); //recommended default value for general use given in manual
        softSineImpCon.parametrize(CartDOF.ROT).setStiffness(0);
        softSineImpCon.parametrize(CartDOF.Z).setBias(forceVal[force]);
        softSineImpCon.parametrize(CartDOF.Z).setForceLimit(forceAbs);
        softSineImpCon.setNullSpaceDamping(0.7);
        softSineImpCon.setNullSpaceStiffness(50);
        softSineImpCon.setRiseTime(2.0);
        softSineImpCon.setHoldTime(5.0);
        softSineImpCon.setFallTime(2.0);

        // mode for flexion extension tests
        CartesianSineImpedanceControlMode softSineImpConFE = new CartesianSineImpedanceControlMode();
        softSineImpConFE.parametrize(CartDOF.X).setStiffness(0);
        softSineImpConFE.parametrize(CartDOF.Y).setStiffness(0);
        softSineImpConFE.parametrize(CartDOF.Z).setStiffness(0);
        softSineImpConFE.parametrize(CartDOF.ALL).setDamping(0.7);
        softSineImpConFE.parametrize(CartDOF.ROT).setStiffness(0);
        softSineImpConFE.parametrize(CartDOF.Z).setBias(forceVal[force]);
        softSineImpConFE.parametrize(CartDOF.Z).setForceLimit(forceAbs);
        softSineImpConFE.setNullSpaceDamping(0.7);
        softSineImpConFE.setNullSpaceStiffness(50);
        softSineImpConFE.setRiseTime(2);

        //mode for ADL tests
        CartesianSineImpedanceControlMode softSineImpConADL = new CartesianSineImpedanceControlMode();
        softSineImpConADL.parametrize(CartDOF.X).setStiffness(0);
        softSineImpConADL.parametrize(CartDOF.Y).setStiffness(0);
```

```java
        softSineImpConADL.parametrize(CartDOF.Z).setStiffness(0);
        softSineImpConADL.parametrize(CartDOF.ALL).setDamping(0.7);
        softSineImpConADL.parametrize(CartDOF.A).setStiffness(0);
        softSineImpConADL.parametrize(CartDOF.B).setStiffness(300);
        softSineImpConADL.parametrize(CartDOF.C).setStiffness(300);
        softSineImpConADL.parametrize(CartDOF.Z).setBias(forceVal[force]);
        softSineImpConADL.parametrize(CartDOF.Z).setForceLimit(forceAbs);
        softSineImpConADL.setNullSpaceDamping(0.7);
        softSineImpConADL.setNullSpaceStiffness(50);
        softSineImpConADL.setRiseTime(2);

        //holding mode
        CartesianImpedanceControlMode holdImpCon = new CartesianImpedanceControlMode();
        holdImpCon.parametrize(CartDOF.TRANSL).setStiffness(5000);
        holdImpCon.parametrize(CartDOF.ROT).setStiffness(300);
        holdImpCon.parametrize(CartDOF.Z).setAdditionalControlForce(0);
        holdImpCon.setMaxControlForce(5000.0, 5000.0, 5000.0, 300.0, 300.0, 300.0, false);
        holdImpCon.setNullSpaceDamping(0.7);
        holdImpCon.setNullSpaceStiffness(200);

        //soft motion mode
        CartesianImpedanceControlMode moveImpCon = new CartesianImpedanceControlMode();
        moveImpCon.parametrize(CartDOF.TRANSL).setStiffness(3000);
        moveImpCon.parametrize(CartDOF.ROT).setStiffness(200);
        //moveImpCon.setNullSpaceDamping(0.7);
        //moveImpCon.setNullSpaceStiffness(150);

        //motion path condition used in triggering start of recording at start of motion
        MotionPathCondition startCond = new MotionPathCondition(ReferenceType.START, 0.0, 0);

        getLogger().info("Going home");
        lbr_iiwa_14_R820_1.move(ptpHome().setJointAccelerationRel(0.5).setJointVelocityRel(0.1));

        /*
         *
         * MVC
         *
         */

        while(choiceMVC != 2){
            choiceMVC = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Measure MVC?", "Biceps (flexion)", "Triceps (extension)", "Done MVC");
            switch (choiceMVC){
            case 0:
                MVCrecName = "MVCflex";
                doMVC(rec, holdImpCon);
                break;
            case 1:
                MVCrecName = "MVCext";
                doMVC(rec, holdImpCon);
                break;
            case 2:
                break;
            }
        }

        /*
         *
         * Motion Trials
         *
         */
```

```
        if(set == (size-2)){
            choice = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Continue testing?","No","Trial 1-12, stationary",
                        "Trial 13-21, flex/ext","Trial 22-30, flex/ext","Trial 31-38, ADL");
        }else{
            choice = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Continue testing?","No", "Trial 1-12, stationary",
                        "Trial 13-21, flex/ext","Trial 22-30, flex/ext","Trial 31-38, ADL","Yes, from programmed set");
        }

        while(choice != 0){
            switch (choice){
            case 0:
                break;
            case 1: //Yes, input values

                do{
                trial = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Trial?",
                        "1-RFP1A1F1V1","2-RFP1A1F2V1","3-REP1A1F2V1","4-RFP1A2F1V1",
                        "5-RFP1A2F2V1","6-REP1A2F2V1","7-RFP2A1F1V1","8-RFP2A1F2V1",
                        "9-REP2A1F2V1","10-RFP3A1F1V1","11-RFP3A1F2V1","12-REP3A1F2V1");

                setTrialVals();

                adjust = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Test correct? Trial: "+(trial+1)+", direction: "
                        +recNameD[direction]+", position: "+recNameP[position]+", angle: "+recNameA[angle]+", force: "
                                +recNameF[force]+", velocity: "+recNameV[vel]+"?", "No, adjust", "Yes, continue");
                }while(adjust == 0);
                break;
            case 2:
                do{
                trial = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Trial?",
                        "13-RFP1A6F1V2","14-RFP1A6F1V3","15-RFP1A6F2V2","16-RFP1A6F2V3",
                        "17-REP1A6F2V2","18-REP1A6F2V3","19-RFP2A6F1V2","20-RFP2A6F1V3",
                        "21-RFP2A6F2V2");
                trial = trial + 12;
                setTrialVals();

                adjust = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Test correct? Trial: "+(trial+1)+", direction: "
                        +recNameD[direction]+", position: "+recNameP[position]+", angle: "+recNameA[angle]+", force: "
                                +recNameF[force]+", velocity: "+recNameV[vel]+"?", "No, adjust", "Yes, continue");
                }while(adjust == 0);
                break;
            case 3:
                do{
                trial = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Trial?",
                        "22-RFP2A6F2V3","23-REP2A6F2V2","24-REP2A6F2V3","25-RFP3A6F1V2",
                        "26-RFP3A6F1V3","27-RFP3A6F2V2","28-RFP3A6F2V3","29-REP3A6F2V2",
                        "30-REP3A6F2V3");
                trial = trial + 21;
                setTrialVals();

                adjust = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Test correct? Trial: "+(trial+1)+", direction: "
                        +recNameD[direction]+", position: "+recNameP[position]+", angle: "+recNameA[angle]+"
```

```
, force: "
                                +recNameF[force]+", velocity: "+recNameV[vel]+"?", "No, adjust", "Yes, continue");
                }while(adjust == 0);

                break;
            case 4:
                do{
                trial = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Trial?",
                        "31-RFP4A4F3V2","32-RFP4A4F3V3","33-RFP4A4F2V2","34-RFP4A4F2V3",
                        "35-RFP5A5F3V2","36-RFP5A5F3V3","37-RFP5A5F2V2","38-RFP5A5F2V3");
                trial = trial + 30;
                setTrialVals();

                adjust = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Test correct? Trial: "+(trial+1)+", direction: "
                        +recNameD[direction]+", position: "+recNameP[position]+", angle: "+recNameA[angle]+", force: "
                                +recNameF[force]+", velocity: "+recNameV[vel]+"?", "No, adjust", "Yes, continue");
                }while(adjust == 0);
                break;
            case 5: //Yes, from programmed set
                set = set +1;
                direction = orderDir[set];
                position = orderPos[set];
                angle = orderAng[set];
                force = orderForce[set];
                vel = orderVel[set];
                rep = 0;
                break;
            }

            if(direction == 0){
                forceVal[0] = 0;
                forceVal[1] = 22;
                forceVal[2] = 11;
            }else if(direction == 1){
                forceVal[0] = 0;
                forceVal[1] = -22;
                forceVal[2] = -11;
            }

            if((position == 2)&&(angle==5)){
                if(direction == 0){
                    forceVal[0] = 0;
                    forceVal[1] = -22;
                    forceVal[2] = -11;
                }else if(direction == 1){
                    forceVal[0] = 0;
                    forceVal[1] = 22;
                    forceVal[2] = 11;
                }
            }

            if((position == 2)&&(angle==0)){
                if(direction == 0){
                    forceVal[0] = 0;
                    forceVal[1] = -22;
                    forceVal[2] = -11;
                }else if(direction == 1){
                    forceVal[0] = 0;
```

```
                    forceVal[1] = 22;
                    forceVal[2] = 11;
                }
            }

            forceAbs = Math.abs(forceVal[force]);

            getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Frame set for direction: "
            +recNameD[direction]+", position: "+recNameP[position]+", angle: "+recNameA[angle]+", force: "
                    +recNameF[force]+", velocity: "+recNameV[vel]+"?", "Yes");
            getLogger().info("Frame set for direction: "
            +recNameD[direction]+", position: "+recNameP[position]+", angle: "+recNameA[angle]+", force: "
                    +recNameF[force]+", velocity: "+recNameV[vel]+"?");

            handle.getFrame("/Handle1Con").move(positionHold(holdImpCon, 2, TimeUnit.SECONDS)); //changed from 5 to 2 seconds wait

            switch(angle){
            case 0:
                recTime = 19;
                doStatReps(rec, softImpCon, softSineImpCon, holdImpCon, startCond, moveImpCon);
                break;
            case 1:
                recTime = 19;
                doStatReps(rec, softImpCon, softSineImpCon, holdImpCon, startCond, moveImpCon);
                break;
            case 2:
                recTime = 19;
                doStatReps(rec, softImpCon, softSineImpCon, holdImpCon, startCond, moveImpCon);
                break;
            case 3: // ADL1 - book
                ADLStartPos = "/ADL1bookStart";
                doADLReps(rec, softSineImpConADL, holdImpCon, moveImpCon);
                break;
            case 4: // ADL2 - cup
                ADLStartPos = "/ADL2cupStart";
                doADLReps(rec, softSineImpConADL, holdImpCon, moveImpCon);
                break;
            case 5: //flexion extension
                if(position == 0){
                    SingleMotionStartPos = "/SingleMotionStartP1Down";
                }else if(position == 1){
                    SingleMotionStartPos = "/SingleMotionStartP2Front";
                }else if(position == 2){
                    SingleMotionStartPos = "/SingleMotionStartP3Side";
                }

                doFlexExtReps(rec, softSineImpConFE, holdImpCon, moveImpCon);
                break;
            }

            if(set == (size-2)){
                choice = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Continue testing?","No","Trial 1-12, stationary",
                        "Trial 13-21, flex/ext","Trial 22-30, flex/ext","Trial 31-38, ADL");
            }else{
                choice = getApplicationUI().displayModalDialog(ApplicationDialogType.QUESTION, "Continue testing?","No", "Trial 1-12, stationary",
                        "Trial 13-21, flex/ext","Trial 22-30, flex/ext","Trial 31-38, ADL","Yes, from programmed set");
            }
```

```java
        }

        getLogger().info("Done");

        handle.detach();

    }

    /**
     *
     */
    private void setTrialVals() {
        direction = trialDir[trial];
        position = trialPos[trial];
        angle = trialAng[trial];
        force = trialForce[trial];
        vel = trialVel[trial];
        rep = 0;
    }

    /**
     *
     */
    private void doMVC(DataRecorder recMVC, CartesianImpedanceControlMode holdMode) {
        handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/MVCStart")).setJointVelocityRel(0.1));
        getLogger().info("5 seconds until MVC");
        handle.getFrame("/Handle1Con").move(positionHold(holdMode, 5, TimeUnit.SECONDS));

        recMVC.setFileName(MVCrecName);

        flangeIO.setLEDBlue(true); //turn on flange LED
        recMVC.enable();

        getLogger().info("Holding force for MVC"); //print to pendant


        lbr_iiwa_14_R820_1.setESMState("3");
        recMVC.startRecording();
        handle.getFrame("/Handle1Con").move(positionHold(holdMode, 5, TimeUnit.SECONDS));
        lbr_iiwa_14_R820_1.setESMState("1");
        flangeIO.setLEDBlue(false);
        recMVC.stopRecording();

        getLogger().info("Done MVC.");
        return;
    }

    /**
     *
     */
    private void doADLReps(DataRecorder rec,
            CartesianSineImpedanceControlMode softSineModeADL, CartesianImpedanceControlMode holdMode, CartesianImpedanceControlMode moveMode) {
        AbstractIO greenButton = flangeIO.getInput("UserButton");
        BooleanIOCondition greenButton_active = new BooleanIOCondition(greenButton, true);

        handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/ADLPounce")).setJointVelocityRel(0.1));
        handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame(ADLStartPos)).setJointVelocityRel(0.1));

```

```
        for(rep=0;rep<numReps;rep++){

            //handle.getFrame("/Handle1Con").move(positionHold(holdMode, 5, TimeUnit.SECONDS));
            getLogger().info("5 seconds until rep");
            handle.getFrame("/Handle1Con").move(positionHold(holdMode, 5, TimeUnit.SECONDS));

            //update softMode
            softSineModeADL.parametrize(CartDOF.X).setStiffness(0);
            softSineModeADL.parametrize(CartDOF.Y).setStiffness(0);
            softSineModeADL.parametrize(CartDOF.Z).setBias(forceAbs);
            softSineModeADL.parametrize(CartDOF.Z).setForceLimit(forceAbs);

            //update recording name
            recName = (recNameD[direction]+recNameP[position]+recNameA[angle]+recNameF[force]+recNameV[vel]+recNameR[rep]);
            rec.setFileName(recName);

            flangeIO.setLEDBlue(true); //turn on flange LED
            rec.enable();

            getLogger().info("Holding force, rep: "+recNameR[rep]); //print to pendant

            if(forceAbs == 22){
                lbr_iiwa_14_R820_1.setESMState("3");
            }

            rec.startRecording();

            handle.getFrame("/Handle1Con").move(positionHold(softSineModeADL, -1, TimeUnit.SECONDS).breakWhen(greenButton_active)); // <0 value gives endless timeout
            lbr_iiwa_14_R820_1.setESMState("1");

            //stop recording
            rec.stopRecording();

            flangeIO.setLEDBlue(false);
            getLogger().info("5 seconds until returning to start");
            handle.getFrame("/Handle1Con").move(positionHold(holdMode, 5, TimeUnit.SECONDS));

            if(rep==(numReps-1)){
                rec.stopRecording();
                getLogger().info("Done "+numReps+" reps of direction: "+recNameD[direction]+", position: "+recNameP[position]+", angle: "+recNameA[angle]+", force: "+recNameF[force]+", velocity: "+recNameV[vel]+".");
            return;
            }
        }
    }

    /**
     * @param rec
     * @param softSineImpCon
     */
    private void doFlexExtReps(DataRecorder rec,
            CartesianSineImpedanceControlMode softSineModeFE, CartesianImpedanceControlMode holdMode, CartesianImpedanceControlMode moveMode) {

        AbstractIO greenButton = flangeIO.getInput("UserButton");
        BooleanIOCondition greenButton_active = new BooleanIOCondition(greenButton, true);

        handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame("/SingleMotionPounce")).setJointVelocityRel(0.1));
```

```
        handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame(SingleMotionStartPos)).setJointVelocityRel(0.1));

        for(rep=0;rep<numReps;rep++){

            getLogger().info("10 seconds until rep");

            handle.getFrame("/Handle1Con").move(positionHold(holdMode, 10, TimeUnit.SECONDS));

            //update softMode
            softSineModeFE.parametrize(CartDOF.X).setStiffness(0);
            softSineModeFE.parametrize(CartDOF.Y).setStiffness(0);
            softSineModeFE.parametrize(CartDOF.Z).setBias(forceVal[force]);
            softSineModeFE.parametrize(CartDOF.Z).setForceLimit(forceAbs);

            //update recording name
            recName = (recNameD[direction]+recNameP[position]+recNameA[angle]+recNameF[force]+recNameV[vel]+recNameR[rep]);
            rec.setFileName(recName);

            flangeIO.setLEDBlue(true); //turn on flange LED
            rec.enable();

            getLogger().info("Holding force, rep: "+recNameR[rep]); //print to pendant

            if(forceAbs == 22){
                lbr_iiwa_14_R820_1.setESMState("3");
            }

            rec.startRecording();

            handle.getFrame("/Handle1Con").move(positionHold(softSineModeFE, -1, TimeUnit.SECONDS).breakWhen(greenButton_active)); // <0 value gives endless timeout
            lbr_iiwa_14_R820_1.setESMState("1");

            //stop recording
            rec.stopRecording();

            handle.getFrame("/Handle1Con").move(positionHold(holdMode, 1, TimeUnit.NANOSECONDS));
            flangeIO.setLEDBlue(false);

            if(rep==(numReps-1)){
                rec.stopRecording();
                getLogger().info("Done "+numReps+" reps of direction: "+recNameD[direction]+", position: "+recNameP[position]+", angle: "+recNameA[angle]+", force: "+recNameF[force]+", velocity: "+recNameV[vel]+".");
            return;
            }
        }
    }

    /**
     * @param rec
     * @param soft22
     * @param holdMode
     */
    private void doStatReps(DataRecorder rec, CartesianImpedanceControlMode softMode, CartesianSineImpedanceControlMode softSineModeStat,
            CartesianImpedanceControlMode holdMode, MotionPathCondition startCondRec, CartesianImpedanceControlMode moveMode) {

        if(position == 0){
```

```
            if(angle == 0){
                StatStartPos = "/StatStartP1A1down";
            }else if(angle ==1){
                StatStartPos = "/StatStartP1A2down90";
            }
        }else if(position == 1){
            StatStartPos = "/StatStartP2Front";
        }else if(position == 2){
            StatStartPos = "/StatStartP3Side";
        }

        handle.getFrame("/Handle1Con").move(ptp(getApplicationData().getFrame(StatStartPos)).setJointAccelerationRel(0.5).setJointVelocityRel(0.1));


        for(rep=0;rep<numReps;rep++){

            getLogger().info("10 seconds until rep");

            handle.getFrame("/Handle1Con").move(positionHold(holdMode, 10, TimeUnit.SECONDS));

            //update softMode
            softMode.parametrize(CartDOF.X).setStiffness(XYstiffness[force]);
            softMode.parametrize(CartDOF.Y).setStiffness(XYstiffness[force]);
            softMode.parametrize(CartDOF.Z).setAdditionalControlForce(forceVal[force]);

            softSineModeStat.parametrize(CartDOF.X).setStiffness(XYstiffness[force]);
            softSineModeStat.parametrize(CartDOF.Y).setStiffness(XYstiffness[force]);
            softSineModeStat.parametrize(CartDOF.Z).setBias(forceVal[force]);
            softSineModeStat.parametrize(CartDOF.Z).setForceLimit(forceAbs);

            //update recording name
            recName = (recNameD[direction]+recNameP[position]+recNameA[angle]+recNameF[force]+recNameV[vel]+recNameR[rep]);
            rec.setFileName(recName);

            flangeIO.setLEDBlue(true);
            rec.enable();
            getLogger().info("Holding position, compliant, 5 seconds, rep: "+recNameR[rep]);

            if(forceAbs == 45){
                lbr_iiwa_14_R820_1.setESMState("3");
            }

            rec.startRecording();

            handle.getFrame("/Handle1Con").move(positionHold(softSineModeStat, 9, TimeUnit.SECONDS)); // recording starts with motion, .triggerWhen(startCondRec, startAct) caused error -not supported for positionHold
            lbr_iiwa_14_R820_1.setESMState("1");

            //stop recording
            rec.stopRecording();

            handle.getFrame("/Handle1Con").move(positionHold(holdMode, 1, TimeUnit.NANOSECONDS));
            flangeIO.setLEDBlue(false);

            if(rep==(numReps-1)){
                //rec.stopRecording();
                getLogger().info("Done "+numReps+" reps of direction: "+recNameD[direction]+", position: "+recNameP[position]+", angle: "+recNameA[angle]+", force: "+recNameF[force]+", velocity: "+recNameV[vel]+"."
);
```

```java
                return;
            }
        }
    }

    /**
     * Auto-generated method stub. Do not modify the contents of this method.
     */
    public static void main(String[] args) {
        MeasuringForDynamicCalibration app = new MeasuringForDynamicCalibration();
        app.runApplication();
    }
}
```

A.2 MATLAB

getKUKA.m

```
%% 20180223
%% loading data into matlab from log files
clear all
close all
clc

%% assigning variables
subjects = {'testS1','S2','S3'};
testType = {'stat','sing','adl'}; % 1=stationary, 2=single motion, 3=ADL
testSetD = {'RF','RE'}; % direction resisting motion
testSetP = {'P1','P2','P3','P4','P5'}; % position
testSetA = {'A1','A2','A3','A4','A5'}; % angle
testSetF = {'F1','F2','F3'}; % force
testSetV = {'V1','V2','V3'}; % velocity
testRep = {'R1','R2','R3','R4','R5'}; % repetition

numReps = 3;

testSet = {1,'RFP1A1F1V1'
    2,'RFP1A1F2V1'
    3,'REP1A1F2V1'
    4,'RFP1A2F1V1'
    5,'RFP1A2F2V1'
    6,'REP1A2F2V1'
    7,'RFP2A1F1V1'
    8,'RFP2A1F2V1'
    9,'REP2A1F2V1'
    10,'RFP3A1F1V1'
    11,'RFP3A1F2V1'
    12,'REP3A1F2V1'
    13,'RFP1A6F1V2'
    14,'RFP1A6F1V3'
    15,'RFP1A6F2V2'
    16,'RFP1A6F2V3'
    17,'REP1A6F2V2'
    18,'REP1A6F2V3'
    19,'RFP2A6F1V2'
    20,'RFP2A6F1V3'
```

```
    21,'RFP2A6F2V2'
    22,'RFP2A6F2V3'
    23,'REP2A6F2V2'
    24,'REP2A6F2V3'
    25,'RFP3A6F1V2'
    26,'RFP3A6F1V3'
    27,'RFP3A6F2V2'
    28,'RFP3A6F2V3'
    29,'REP3A6F2V2'
    30,'REP3A6F2V3'
    31,'RFP4A4F3V2'
    32,'RFP4A4F3V3'
    33,'RFP4A4F2V2'
    34,'RFP4A4F2V3'
    35,'RFP5A5F3V2'
    36,'RFP5A5F3V3'
    37,'RFP5A5F2V2'
    38,'RFP5A5F2V3'
    39,'MVCflex'
    40,'MVCext'
    };



%% load Kuka data, get timestamps for start and end of each trial

subNum = 24; % set as subject currently being processed

startTrial = 1;
numTrials = 40;
timestampRepStartK = zeros(numReps,40);
timestampRepEndK = zeros(numReps,40);
TimeStartKFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\','
    timesStartK.csv');
TimeEndKFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\','
    timesEndK.csv');

for trial=startTrial:(startTrial - 1 + numTrials)
    if(trial < 39)
        for rep=1:numReps
```

```
            fileName = strcat('D:\TaylorMasters2\Data\S',int2str(subNum),'\KUKA\',testSet{
                trial,2},testRep{rep});

            dataKuka = load(fileName);
            KTimeCol = getKukaTimestampCol(dataKuka(:,1),dataKuka(:,2));

            % collect timestamps
            timestampRepStartK(rep,trial) = KTimeCol(2001);
            if (trial < 13)
                timestampRepEndK(rep,trial) = KTimeCol(7001);
            elseif (trial > 12)
             timestampRepEndK(rep,trial) = KTimeCol(size(KTimeCol,1));
            end

        end
    end

    if (trial > 38)
        fileName = strcat('D:\TaylorMasters2\Data\S',int2str(subNum),'\KUKA\',testSet{trial
            ,2});
        dataKuka = load(fileName);
        KTimeCol = getKukaTimestampCol(dataKuka(:,1),dataKuka(:,2));
        % collect timestamps
        timestampRepStartK(1,trial) = KTimeCol(1);
        timestampRepEndK(1,trial) = KTimeCol(size(KTimeCol,1));
    end

end

%% save Kuka timestamps to files

dlmwrite(TimeStartKFilename, timestampRepStartK,'precision',16);
dlmwrite(TimeEndKFilename, timestampRepEndK,'precision',16);
%saveFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\',testSet{
    set,2},testRep{rep},'CondensedData');
%dlmwrite(saveFilename,condensedData, 'precision',16);
```

getKUKATimestampCol.m

```
function [ KTimestampCol ] = getKukaTimestampCol( dataTime_1 , dataTime_2 )
% this function combines two timestamp values given by KUKA into one
% number , returns a column of timestamps
for j=1:size(dataTime_1 ,1)
    if(dataTime_2(j,1) == 0)
        tempTime = strcat(num2str(dataTime_1(j,1)),'.',num2str(0),num2str(0),num2str(0),
            num2str(0),num2str(0),num2str(0),num2str(0),num2str(0),num2str(0));
    elseif (dataTime_2(j,1) < 10000000)
        tempTime = strcat(num2str(dataTime_1(j,1)),'.',num2str(0),num2str(0),num2str(
            dataTime_2(j,1)));
    elseif(dataTime_2(j,1) < 100000000)
        tempTime = strcat(num2str(dataTime_1(j,1)),'.',num2str(0),num2str(dataTime_2(j,1)))
            ;

    elseif (dataTime_2(j,1) >= 100000000)
        tempTime = strcat(num2str(dataTime_1(j,1)),'.',num2str(dataTime_2(j,1)));
    end
    KTimestampCol(j,1) = str2double(tempTime);
end

end
```

getTrigno.m

```
%% 20180306
% this script loads Trigno data files and segments data into repetitions
% based on KUKA timestamps
%% assigning variables
subNum = 1; % set as subject currently being processed

startTrial = 1;
numTrials = 40; % includes MVC files
timeOffsetRepStartT = zeros(numReps, 40);
timeOffsetRepEndT = zeros(numReps, 40);
%matOffsetRepStartT = zeros(numReps, 40);
%matOffsetRepEndTEMG = zeros(numReps, 40);
offset = 26; % offset of trigno to get to real time move 2 seconds back, kuka to real time 28 seconds back, between is 26 seconds
[offsetK, offsetKTimezone] = getKOffset(subNum); % time offset of Kuka, specific for subject
offsetKTimezone = offsetKTimezone{1,1};
offsetT = 0; %S1+0, S2+0, S3+0, S4+0

%% get time offsets
% run assign variables section first
fileName = strcat('D:\TaylorMasters2\Data\S',int2str(subNum),'\Trigno\EMGRecTimestamps.xlsx');
[numT, txtT, rawT] = xlsread(fileName,'B1:C40');
timestampFileT = datetime(strcat(txtT(:,1),txtT(:,2)),'InputFormat','yyyy/MM/ddHH:mm:ss.SSSSSSS');
timestampFileT.TimeZone = offsetKTimezone; %-4:00 for S1, S2, -3:00 S3,
timestampFileT.Format = 'yyyy/MM/dd HH:mm:ss.SSSSSSS';

timeRepStartFromKfile = csvread(TimeStartKFilename);
timestampRepStartFromKfile = datetime(timeRepStartFromKfile,'ConvertFrom','epochtime');
timestampRepStartFromKfile.TimeZone = '+00:00';
timestampRepStartFromKfile.Format = 'yyyy/MM/dd HH:mm:ss.SSSSSSS';

timeRepEndFromKfile = csvread(TimeEndKFilename);
timestampRepEndFromKfile = datetime(timeRepEndFromKfile,'ConvertFrom','epochtime');
timestampRepEndFromKfile.TimeZone = '+00:00';
timestampRepEndFromKfile.Format = 'yyyy/MM/dd HH:mm:ss.SSSSSSS';

for trial=startTrial:(startTrial - 1 + numTrials)
```

```
        for rep=1:numReps
            tempTimeOffsetStart = (timestampRepStartFromKfile(rep,trial)-seconds(offsetK))
                - (timestampFileT(trial)-seconds(offsetT));
            timeOffsetRepStartT(rep,trial) = seconds(duration(tempTimeOffsetStart));

            tempTimeOffsetEnd = (timestampRepEndFromKfile(rep,trial)-seconds(offsetK)) - (
                timestampFileT(trial)-seconds(offsetT));
            timeOffsetRepEndT(rep,trial) = seconds(duration(tempTimeOffsetEnd));
        end
end

matOffsetRepStartTEMG = round(timeOffsetRepStartT./(1/1925.926),0) + 1;
matOffsetRepEndTEMG = round(timeOffsetRepEndT./(1/1925.926)) + 1;

%% get Trigno data, save as separate reps
% run assign variables section first

for trial=startTrial:(startTrial - 1 + numTrials)
    if(trial < 39)
        for rep=1:numReps

            fileName = strcat('D:\TaylorMasters2\Data\S',int2str(subNum),'\Trigno\EMG',
                int2str(trial),'.csv');

            dataTrigno = csvread(fileName,453,0);
            %get portion of Trigno data for specific rep, specific trial
            tempDataTrignoEMGRep = [dataTrigno(matOffsetRepStartTEMG(rep,trial):
                matOffsetRepEndTEMG(rep,trial),1:2),... %BicepsBrachiiShortHead
                dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
                    ,10),... %BicepsBrachiiLongHead
                dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
                    ,18),... %Brachialis
                dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
                    ,26),... %Brachioradialis
                dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
                    ,34),... %TricepsBrachiiLongHead
                dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
                    ,42),... %TricepsLateralHead
                dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
                    ,50),... %TricepsMedialHead
                dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
```

```
            ,58),... %PronatorTeres
        dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
            ,66),... %Infraspinatus
        dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
            ,74),... %AnteriorDeltoid
        dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
            ,82),... %LateralDeltoid
        dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
            ,90),... %PosteriorDeltoid
        dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
            ,98),... %ExtCarpiUlnaris
        dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
            ,106),... %ExtCarpiRadialis
        dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
            ,114),... %FlexCarpiUlnaris
        dataTrigno(matOffsetRepStartTEMG(rep,trial):matOffsetRepEndTEMG(rep,trial)
            ,122) %FlexCarpiRadialis
    ];

    saveTrignoEmgRepFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str
        (subNum),'\EMGReps\EMGT',int2str(trial),'R',int2str(rep),'.csv');
    dlmwrite(saveTrignoEmgRepFilename,tempDataTrignoEMGRep,'precision',16);
  end
end

if(trial > 38)
  if(trial == 39)
    fileName = strcat('D:\TaylorMasters2\Data\S',int2str(subNum),'\Trigno\
        EMGMVCflex.csv');
  elseif(trial == 40)
    fileName = strcat('D:\TaylorMasters2\Data\S',int2str(subNum),'\Trigno\EMGMVCext
        .csv');
  end
    dataTrigno = csvread(fileName,453,0);
    %get portion of Trigno data for specific rep, specific trial
    tempDataTrignoEMGRep = [dataTrigno(matOffsetRepStartTEMG(1,trial):
        matOffsetRepEndTEMG(1,trial),1:2),... %BicepsBrachiiShortHead
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),10),
            ... %BicepsBrachiiLongHead
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),18),
            ... %Brachialis
```

```
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),26),
            ... %Brachioradialis
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),34),
            ... %TricepsBrachiiLongHead
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),42),
            ... %TricepsLateralHead
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),50),
            ... %TricepsMedialHead
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),58),
            ... %PronatorTeres
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),66),
            ... %Infraspinatus
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),74),
            ... %AnteriorDeltoid
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),82),
            ... %LateralDeltoid
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),90),
            ... %PosteriorDeltoid
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),98),
            ... %ExtCarpiUlnaris
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),106)
            ,... %ExtCarpiRadialis
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),114)
            ,... %FlexCarpiUlnaris
        dataTrigno(matOffsetRepStartTEMG(1,trial):matOffsetRepEndTEMG(1,trial),122)
             %FlexCarpiRadialis
    ];

    saveTrignoEmgRepFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str
        (subNum),'\EMGReps\EMGT',int2str(trial),'.csv');
    dlmwrite(saveTrignoEmgRepFilename,tempDataTrignoEMGRep,'precision',16);
  end
end
```

getKOffset.m

```
function [ subKOffset,subKOffsetTimezone ] = getKOffset( sub )
% input is subject number
% KOffsets is matrix, col 1 is subject number, col 2 is Kuka time offset
% each row is for a subject
KOffsets = [1,28
    2,28
    3,30.2
    4,30.2
    5,31.7
    6,31.1
    7,33.2
    8,32.7
    9,33.6
    10,34
    11,34.2
    12,35.4
    13,36
    14,36.2
    15,37.8
    16,37.5
    17,38
    18,40
    19,40
    20,39.4
    21,40.2
    22,39.8
    23,40
    24,41.3
    ];
KOffsetsTimezones = {1,'-04:00'
    2,'-04:00'
    3,'-03:00'
    4,'-04:00'
    5,'-04:00'
    6,'-05:00'
    7,'-05:00'
    8,'-05:00'
    9,'-05:00'
    10,'-05:00'
    11,'-05:00'
```

```
    12,'-05:00'
    13,'-05:00'
    14,'-05:00'
    15,'-05:00'
    16,'-05:00'
    17,'-05:00'
    18,'-05:00'
    19,'-05:00'
    20,'-05:00'
    21,'-05:00'
    22,'-05:00'
    23,'-05:00'
    24,'-05:00'
    };
subKOffset = KOffsets(sub,2);
subKOffsetTimezone = KOffsetsTimezones(sub,2);
end
```

preprocessTrigno.m

```
%% pre-processing Trigno
% this script filters and normalizes EMG signals, extracts features, saves
% .csv files
%% assigning variables
subNum = 24;

startTrial = 1;
numTrials = 40; % filter 40 including MVC, normalize and extract features 38 trials
numReps = 3;
numMuscles = 16;
numFeatsHudgins = 4;
numFeatsOskoei = 2;
numFeatsFreq = 2;

repsStartEnd = getSubReps(subNum);

% prior to 20180618 used window length 100 samples, increment 10
% 20180618 used window length 500 samples, increment 250

mywinsize = 500;
mywininc = 250;

HudginsFeat = {'mav'
    'ssc'
    'wl'
    'zc'};
OskoeiFeat = {'rms'
    'ar'};
freqFeat = {'mnf'
    'mdf'};

%% filter
% run assign variables section first

for trial = startTrial:(startTrial - 1 + numTrials)
    if(trial < 39)
        for rep=repsStartEnd(trial,1):repsStartEnd(trial,2)
            filename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\
                EMGReps\EMGT',int2str(trial),'R',int2str(rep),'.csv');
            dataTrignoRep = csvread(filename);
```

```
            dataTrignoRepFilt = zeros(size(dataTrignoRep,1),numMuscles);
            for muscle=2:1+numMuscles
                dataTrignoRepFilt(:,muscle-1) = emgfilter_trigno(dataTrignoRep(:,muscle));
            end

            saveTrignoEmgRepFiltFilename = strcat('D:\TaylorMasters2\Data\Processing\S',
                int2str(subNum),'\EMGRepsFilt\EMGFiltT',int2str(trial),'R',int2str(rep),'.
                csv');
            dlmwrite(saveTrignoEmgRepFiltFilename,dataTrignoRepFilt,'precision',16);
        end
    end

    if(trial > 38)
        filename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\EMGReps\
            EMGT',int2str(trial),'.csv');
            dataTrignoRep = csvread(filename);
            dataTrignoRepFilt = zeros(size(dataTrignoRep,1),numMuscles);
            for muscle=2:1+numMuscles
                dataTrignoRepFilt(:,muscle-1) = emgfilter_trigno(dataTrignoRep(:,muscle));
            end

            saveTrignoEmgRepFiltFilename = strcat('D:\TaylorMasters2\Data\Processing\S',
                int2str(subNum),'\EMGRepsFilt\EMGFiltT',int2str(trial),'.csv');
            dlmwrite(saveTrignoEmgRepFiltFilename,dataTrignoRepFilt,'precision',16);
    end
end

%% normalize filtered EMG
% run assign variables section first

MVCflexFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\
    EMGRepsFilt\EMGFiltT39.csv');
MVCflexEMG = csvread(MVCflexFilename);
MVCextFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\EMGRepsFilt
    \EMGFiltT40.csv');
MVCextEMG = csvread(MVCextFilename);

%find max EMG amplitude for each muscle
    [MVCmax(1,:),I] = max(MVCflexEMG,[],1);
    [MVCmax(2,:),I] = min(MVCflexEMG,[],1);
    [MVCmax(3,:),I] = max(MVCextEMG,[],1);
```

```
    [MVCmax(4,:),I] = min(MVCextEMG,[],1);
    MVCmax = abs(MVCmax);
    [maxEMG, I] = max(MVCmax,[],1);


for trial = startTrial:(startTrial - 1 + numTrials) % for just the 38 trials, not MVC
    because normalizing relative to MVC
    for rep=repsStartEnd(trial,1):repsStartEnd(trial,2)
        filename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\
            EMGRepsFilt\EMGFiltT',int2str(trial),'R',int2str(rep),'.csv');
        tempEMG = csvread(filename);
        tempEMGnorm = tempEMG./maxEMG;

        saveEMGRepFiltNormFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(
            subNum),'\EMGRepsFiltNorm\EMGFiltNormT',int2str(trial),'R',int2str(rep),'.csv')
            ;
        dlmwrite(saveEMGRepFiltNormFilename,tempEMGnorm,'precision',16);
    end

end

%% extract features (not normalized)
% run assign variables section first
% extract 4 Hudgins set features

for trial = startTrial:(startTrial - 1 + numTrials) % for just the 38 trials, not MVC
    for rep=repsStartEnd(trial,1):repsStartEnd(trial,2)
        filename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\
            EMGRepsFilt\EMGFiltT',int2str(trial),'R',int2str(rep),'.csv');
        dataTrignoRepFiltered = csvread(filename);
        clear dataTrignoRepFeatMAV dataTrignoRepFeatSSC dataTrignoRepFeatWL
            dataTrignoRepFeatZC
       for muscle=1:numMuscles
           dataTrignoRepFeatMAV(:,muscle) = mavfeat(dataTrignoRepFiltered(:,muscle),
               mywinsize,mywininc);
           dataTrignoRepFeatSSC(:,muscle) = sscfeat(dataTrignoRepFiltered(:,muscle),
               mywinsize,mywininc);
           dataTrignoRepFeatWL(:,muscle) = wlfeat(dataTrignoRepFiltered(:,muscle),mywinsize
               ,mywininc);
           dataTrignoRepFeatZC(:,muscle) = zcfeat(dataTrignoRepFiltered(:,muscle),mywinsize
               ,mywininc);
```

```
        end
        for feat = 1:numFeatsHudgins %write to files
            saveTrignoEmgRepFeatFilename = strcat('D:\TaylorMasters2\Data\Processing\S',
                int2str(subNum),'\EMGRepsHudginsFeat\EMGFeatT',int2str(trial),'R',int2str(
                rep),HudginsFeat{feat},'.csv');
            if (feat==1)
               dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatMAV,'precision',16);
            elseif (feat == 2)
                dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatSSC,'precision',16);
            elseif (feat == 3)
                dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatWL,'precision',16);
            elseif (feat == 4)
                dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatZC,'precision',16);
            end
        end
    end
end

%% extract features from normalized emg
% run assign variables section first
% extract 4 Hudgins set features

for trial = startTrial:(startTrial - 1 + numTrials) % for just the 38 trials, not MVC
    for rep=repsStartEnd(trial,1):repsStartEnd(trial,2)
        filename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\
            EMGRepsFiltNorm\EMGFiltNormT',int2str(trial),'R',int2str(rep),'.csv');
        dataTrignoRepFiltered = csvread(filename);
        clear dataTrignoRepFeatMAV dataTrignoRepFeatSSC dataTrignoRepFeatWL
            dataTrignoRepFeatZC
        for muscle=1:numMuscles
            dataTrignoRepFeatMAV(:,muscle) = mavfeat(dataTrignoRepFiltered(:,muscle),
                mywinsize,mywininc);
            dataTrignoRepFeatSSC(:,muscle) = sscfeat(dataTrignoRepFiltered(:,muscle),
                mywinsize,mywininc);
            dataTrignoRepFeatWL(:,muscle) = wlfeat(dataTrignoRepFiltered(:,muscle),mywinsize
                ,mywininc);
            dataTrignoRepFeatZC(:,muscle) = zcfeat(dataTrignoRepFiltered(:,muscle),mywinsize
                ,mywininc);
        end
        for feat = 1:numFeatsHudgins %write to files
            saveTrignoEmgRepFeatFilename = strcat('D:\TaylorMasters2\Data\Processing\S',
```

```
                int2str(subNum),'\EMGRepsHudginsFeatOfNorm500_250\EMGFeatT',int2str(trial),'
                R',int2str(rep),HudginsFeat{feat},'500_250.csv');
            if (feat==1)
               dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatMAV,'precision',16);
            elseif (feat == 2)
                dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatSSC,'precision',16);
            elseif (feat == 3)
                dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatWL,'precision',16);
            elseif (feat == 4)
                dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatZC,'precision',16);
            end
        end
    end
end

%% extract features from normalized emg
% run assign variables section first
% extract 2 Oskoei set features

for trial = startTrial:(startTrial - 1 + numTrials) % for just the 38 trials, not MVC
    for rep=repsStartEnd(trial,1):repsStartEnd(trial,2)
        filename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\
            EMGRepsFiltNorm\EMGFiltNormT',int2str(trial),'R',int2str(rep),'.csv');
        dataTrignoRepFiltered = csvread(filename);
        clear dataTrignoRepFeatRMS dataTrignoRepFeatAR
       for muscle=1:numMuscles
           dataTrignoRepFeatRMS(:,muscle) = rmsfeat(dataTrignoRepFiltered(:,muscle),
               mywinsize,mywininc);
       end
       if subNum == 7
           dataTrignoRepFiltered(:,9)=0; % only for subject 7, muscle 9 normalized filtered
                values were NaN because MVC absolute max was 0 because of faulty sensor
           % only for AR feature because cannot accept NaN
       end
       if subNum == 9
           dataTrignoRepFiltered(:,7)=0; % only for subject 9, muscle 7 normalized filtered
                values were NaN because MVC absolute max was 0 because of faulty sensor
           % only for AR feature because cannot accept NaN
       end
       dataTrignoRepFeatAR = arfeat4(dataTrignoRepFiltered,mywinsize,mywininc,4);
       for feat = 1:numFeatsOskoei %write to files
```

```
            saveTrignoEmgRepFeatFilename = strcat('D:\TaylorMasters2\Data\Processing\S',
                int2str(subNum),'\EMGRepsOskoeiFeatOfNorm500_250\EMGFeatT',int2str(trial),'R
                ',int2str(rep),OskoeiFeat{feat},'500_250.csv');
            if (feat==1)
               dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatRMS,'precision',16);
            elseif (feat == 2)
               dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatAR,'precision',16);
            end
        end
    end
end

%% extract features from normalized emg
% run assign variables section first
% extract 2 frequency features

for trial = startTrial:(startTrial - 1 + numTrials) % for just the 38 trials, not MVC
    for rep=repsStartEnd(trial,1):repsStartEnd(trial,2)
        filename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\
            EMGRepsFiltNorm\EMGFiltNormT',int2str(trial),'R',int2str(rep),'.csv');
        dataTrignoRepFiltered = csvread(filename);
        clear dataTrignoRepFeatMNF dataTrignoRepFeatMDF

        if subNum == 7
            dataTrignoRepFiltered(:,9)=0; % only for subject 7, muscle 9 normalized filtered
                 values were NaN because MVC absolute max was 0 because of faulty sensor
            % freq features cannot accept NaN
        end

        if subNum == 9
            dataTrignoRepFiltered(:,7)=0; % only for subject 9, muscle 7 normalized filtered
                 values were NaN because MVC absolute max was 0 because of faulty sensor
            % freq features cannot accept NaN
        end

        for muscle=1:numMuscles
            dataTrignoRepFeatMNF(:,muscle) = mnffeat(dataTrignoRepFiltered(:,muscle),
                mywinsize,mywininc);
            dataTrignoRepFeatMDF(:,muscle) = mdffeat(dataTrignoRepFiltered(:,muscle),
                mywinsize,mywininc);
        end
```

```
        for feat = 1:numFeatsFreq %write to files
            saveTrignoEmgRepFeatFilename = strcat('D:\TaylorMasters2\Data\Processing\S',
                int2str(subNum),'\EMGRepsFreqFeatOfNorm500_250\EMGFeatT',int2str(trial),'R',
                int2str(rep),freqFeat{feat},'500_250.csv');
            if (feat==1)
               dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatMNF,'precision',16);
            elseif (feat == 2)
               dlmwrite(saveTrignoEmgRepFeatFilename,dataTrignoRepFeatMDF,'precision',16);
            end
        end
    end
end
```

getSubReps.m

```
function [ subReps ] = getSubReps( sub )
% input is subject number
% returns matrix, two columns for specified subject, S1.1 is first rep of S1 reps, S1.2
% is last rep of subject reps
% rows are each trial, 1-38
startEndReps =
    [1,2,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    4,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    5,2,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    6,2,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    7,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    8,1,3,1,3,1,3,1,3,2,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    9,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    10,2,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    11,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    12,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    13,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    14,2,3,2,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    15,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    16,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

    17,2,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3
```

```
18,2,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

19,1,3,1,3,2,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

20,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

21,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

22,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

23,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

24,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

25,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

26,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

27,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

28,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

29,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

30,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

31,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

32,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

33,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

34,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

35,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

36,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

37,1,3,2,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3

38,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2,1,3,1,2,1,2,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3
```

```
    ];

subReps = startEndReps(:,(sub*2):(sub*2+1));
end
```

aveFeats.m

```
%% average features for each motion
% for each separate motion trial and feature, average features over windows for one
% feature value per rep, average features over reps for one feature value
% per motion trial

%% clear
clear all
close all
clc

%% run variable setup for feature averaging
startTrial = 1;
endTrial = 38;
numReps = 3;
numFeatsHudgins = 4;
numFeatsOskoei = 2;
numFeatsFreq = 2;

HudginsFeat = {'mav500_250'
    'ssc500_250'
    'wl500_250'
    'zc500_250'};
OskoeiFeat = {'rms500_250'
    'ar500_250'};
freqFeat = {'mnf500_250'
    'mdf500_250'};

openFolders={'EMGRepsHudginsFeatOfNorm500_250'
    'EMGRepsOskoeiFeatOfNorm500_250'
    'EMGRepsFreqFeatOfNorm500_250'};
saveFolders={'EMGMeanHudginsFeatOfNorm500_250'
    'EMGMeanOskoeiFeatOfNorm500_250'
    'EMGMeanFreqFeatOfNorm500_250'};

%% run averaging of features over reps for each trial
for sub=1:24
    for featH=1:numFeatsHudgins
        getAveFeats(startTrial,endTrial,openFolders{1},saveFolders{1},HudginsFeat{featH},
            sub);
    end
```

```
    for featO=1:numFeatsOskoei
        getAveFeats(startTrial,endTrial,openFolders{2},saveFolders{2},OskoeiFeat{featO},sub
            );
    end

    for featF=1:numFeatsFreq
        getAveFeats(startTrial,endTrial,openFolders{3},saveFolders{3},freqFeat{featF},sub);
    end
end
```

getAveFeats.m

```
function [ ] = getAveFeats( sTrial,eTrial, openFold, saveFold, featName, subNumber)
%averages feature values of given feature for given subject
%   inputs are start and end trial, folder names, feature, subject number
%   average specified feature over trials for given subject, save csv file,
%   rows are trial, columns are muscle
  clear EMGAveFeat EMGAveFeatAll
    for trial = sTrial:eTrial
        clear tempDataEMGFeat EMGFeat
        repsStartEnd = getSubReps(subNumber);
        numRepsToAve = 0;
        for rep=repsStartEnd(trial,1):repsStartEnd(trial,2)
            numRepsToAve = numRepsToAve +1;
            filename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNumber),'\',
                openFold,'\EMGFeatT',int2str(trial),'R',int2str(rep),featName,'.csv');
            tempDataEMGFeat = csvread(filename);
            EMGFeat(numRepsToAve,:) = mean(tempDataEMGFeat,1);
        end
        EMGAveFeat = mean(EMGFeat,1);
        EMGAveFeatAll(trial,:) = EMGAveFeat;
    end
    saveAveFeatAllTrial = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNumber),'
        \',saveFold,'\',featName,'.csv');
    dlmwrite(saveAveFeatAllTrial,EMGAveFeatAll,'precision',16);
end
```

getKukaVelocity.m

```
%% 20180718
% this gets retrieves the KUKA data and calculates velocities (speed components)
clear all
close all
clc

%% Assigning variables
testSet = {1,'RFP1A1F1V1'
    2,'RFP1A1F2V1'
    3,'REP1A1F2V1'
    4,'RFP1A2F1V1'
    5,'RFP1A2F2V1'
    6,'REP1A2F2V1'
    7,'RFP2A1F1V1'
    8,'RFP2A1F2V1'
    9,'REP2A1F2V1'
    10,'RFP3A1F1V1'
    11,'RFP3A1F2V1'
    12,'REP3A1F2V1'
    13,'RFP1A6F1V2'
    14,'RFP1A6F1V3'
    15,'RFP1A6F2V2'
    16,'RFP1A6F2V3'
    17,'REP1A6F2V2'
    18,'REP1A6F2V3'
    19,'RFP2A6F1V2'
    20,'RFP2A6F1V3'
    21,'RFP2A6F2V2'
    22,'RFP2A6F2V3'
    23,'REP2A6F2V2'
    24,'REP2A6F2V3'
    25,'RFP3A6F1V2'
    26,'RFP3A6F1V3'
    27,'RFP3A6F2V2'
    28,'RFP3A6F2V3'
    29,'REP3A6F2V2'
    30,'REP3A6F2V3'
    31,'RFP4A4F3V2'
    32,'RFP4A4F3V3'
    33,'RFP4A4F2V2'
```

```
    34,'RFP4A4F2V3'
    35,'RFP5A5F3V2'
    36,'RFP5A5F3V3'
    37,'RFP5A5F2V2'
    38,'RFP5A5F2V3'
    39,'MVCflex'
    40,'MVCext'
    };

%%
subNum = 24; % set as subject currently being processed
startTrial = 1;
numTrials = 40;
% 1 millisecond period
period = 0.001; % 0.001 seconds (1 millisecond)
rowMotionStart = 2001;
rowMotionEndStat = 7001;

clear p_handle_ave_all v_handle_ave_all
for trial = startTrial:numTrials
    if (trial < 39) %motion trials not including MVC
        clear temp_p_handle_reps_ave temp_v_handle_reps_ave
        repsStartEnd = getSubReps(subNum);
        numRepsToAve = 0;
        for rep = repsStartEnd(trial,1):repsStartEnd(trial,2) %using reps matching reps
            used for feature extraction
            clear dataKuka p_handle v_handle
            numRepsToAve = numRepsToAve + 1;
            fileName = strcat('D:\TaylorMasters2\Data\S',int2str(subNum),'\KUKA\',testSet{
                trial,2},'R',int2str(rep));

            dataKuka = load(fileName);

            if (trial < 13)
                rowMotionEnd = rowMotionEndStat;
            end
            if (trial > 12)
                rowMotionEnd = size(dataKuka,1); % KUKA2 added the '-2000'
            end
            rowVel = 1;
            for rowData = rowMotionStart:rowMotionEnd-1
```

```
                %calculate velocity
                p_handle(rowVel,1:3) = [(dataKuka((rowData + 1),22) - dataKuka(rowData,22))
                    ,(dataKuka((rowData + 1),23) - dataKuka(rowData,23)),(dataKuka((rowData
                     + 1),24) - dataKuka(rowData,24))];
                p_handle(rowVel,4) = sqrt((p_handle(rowVel,1)^2)+(p_handle(rowVel,2)^2)+(
                    p_handle(rowVel,3)^2));
                v_handle(rowVel,1:3) = [(dataKuka((rowData + 1),22) - dataKuka(rowData,22))
                    ,(dataKuka((rowData + 1),23) - dataKuka(rowData,23)),(dataKuka((rowData
                     + 1),24) - dataKuka(rowData,24))]/period;
                v_handle(rowVel,4) = sqrt((v_handle(rowVel,1)^2)+(v_handle(rowVel,2)^2)+(
                    v_handle(rowVel,3)^2));
                rowVel = rowVel + 1;
            end
            savePositionRepFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(
                subNum),'\KUKA\PositionT',int2str(trial),'R',int2str(rep),'.csv');
            saveVelocityRepFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(
                subNum),'\KUKA\VelocityT',int2str(trial),'R',int2str(rep),'.csv');
            dlmwrite(savePositionRepFilename,p_handle,'precision',8);
            dlmwrite(saveVelocityRepFilename,v_handle,'precision',8);
            temp_p_handle_reps_ave(numRepsToAve,:) = mean(abs(p_handle),1);
            temp_v_handle_reps_ave(numRepsToAve,:) = mean(abs(v_handle),1);
        end
        p_handle_ave_all(trial,:) = mean(temp_p_handle_reps_ave,1);
        v_handle_ave_all(trial,:) = mean(temp_v_handle_reps_ave,1);
    end

    if(trial > 38)
        clear temp_p_handle_reps_ave temp_v_handle_reps_ave
        clear dataKuka p_handle v_handle
        repsStartEnd = getSubReps(subNum);
        fileName = strcat('D:\TaylorMasters2\Data\S',int2str(subNum),'\KUKA\',testSet{trial
            ,2});
        dataKuka = load(fileName);
        rowMotionEnd = size(dataKuka,1); % KUKA2 added the '-2000';
        rowVel = 1;
        for rowData = rowMotionStart:rowMotionEnd-1
            %calculate velocity
            p_handle(rowVel,1:3) = [(dataKuka((rowData + 1),22) - dataKuka(rowData,22)),(
                dataKuka((rowData + 1),23) - dataKuka(rowData,23)),(dataKuka((rowData + 1)
                ,24) - dataKuka(rowData,24))];
            p_handle(rowVel,4) = sqrt((p_handle(rowVel,1)^2)+(p_handle(rowVel,2)^2)+(
```

```
            p_handle(rowVel,3)^2));
        v_handle(rowVel,1:3) = [(dataKuka((rowData + 1),22) - dataKuka(rowData,22)),(
            dataKuka((rowData + 1),23) - dataKuka(rowData,23)),(dataKuka((rowData + 1)
            ,24) - dataKuka(rowData,24))]/period;
        v_handle(rowVel,4) = sqrt((v_handle(rowVel,1)^2)+(v_handle(rowVel,2)^2)+(
            v_handle(rowVel,3)^2));
        rowVel = rowVel + 1;
    end
    savePositionRepFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(
        subNum),'\KUKA\PositionT',int2str(trial),'R',int2str(rep),'.csv');
    saveVelocityRepFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(
        subNum),'\KUKA\VelocityT',int2str(trial),'R',int2str(rep),'.csv');
    dlmwrite(savePositionRepFilename, p_handle,'precision',8);
    dlmwrite(saveVelocityRepFilename, v_handle,'precision',8);
    p_handle_ave_all(trial,:) = mean(p_handle,1);
    v_handle_ave_all(trial,:) = mean(v_handle,1);
end



end

saveAvePosAllTrialsFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),
    '\KUKA\MeanPosition.csv');
saveAveSpeedAllTrialsFilename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum
    ),'\KUKA\MeanSpeed.csv');
dlmwrite(saveAvePosAllTrialsFilename, p_handle_ave_all,'precision',8);
dlmwrite(saveAveSpeedAllTrialsFilename, v_handle_ave_all,'precision',8);

subNum %print subject number to screen to see progress of script running
```

A.2.1 Classification Implemented Using MATLAB

classificationLOSO.m

```
%% classification
% this script generates matrixes of predictors, vectors of labels, runs
% LOSO LDA or SVM classification with specified predictors, labels, sets of
% trials, saves accuracies and results
%% clear
clear all
close all
clc

%% labeling
%% run variable setup

numSubjects = 24;
numTrials = 38;

HudginsFeat = {'mav500_250'
    'ssc500_250'
    'wl500_250'
    'zc500_250'};
OskoeiFeat = {'rms500_250'
    'ar500_250'};
freqFeat = {'mnf500_250'
    'mdf500_250'};

featNames = {'mav500_250'
    'ssc500_250'
    'wl500_250'
    'zc500_250'
    'rms500_250'
    'ar500_250'
    'ar500_250'
    'ar500_250'
    'ar500_250'
    'mnf500_250'
    'mdf500_250'};

openFolders={'EMGMeanHudginsFeatOfNorm500_250'
    'EMGMeanOskoeiFeatOfNorm500_250'
```

```
    'EMGMeanFreqFeatOfNorm500_250'};

%% setting up trial labels
% labels of trials
% column 1: data trial number
% column 2: flexion/extension position 1=down, 2=front, 3=side (0=down 90 deg flex, 4=ADL1,
    5=ADL2)
% column 3: flexion/extension force 1=0 N, 2=22 N RF, 3=22 N RE (4=11 N)
% column 4: flexion/extension velocity 1=stationary, 2=slow, 3=fast
% column 5: ADL force 1=11 N, 2=22 N (3=22 N RE, 4=0 N)
% column 6: ADL velocity 1=slow, 2=fast (3=stationary)
% column 7: flexion/extension velocity 1=stationary, 2=moving
% column 8: flexion/extension position 1=down, 2=front, 3=side (1=down 90
% deg flex, 4=ADL1, 5=ADL2) **** only change to col 2 is 1=down 90 deg flex
% as well
% column 9: flexion/extension force 1=0 N, 2= > 0 N
trial_labels = [
1,1,1,1,4,0,1,1,1
2,1,2,1,2,0,1,1,2
3,1,3,1,3,0,1,1,2
4,0,1,1,4,0,1,1,1
5,0,2,1,2,0,1,1,2
6,0,3,1,3,0,1,1,2
7,2,1,1,4,0,1,2,1
8,2,2,1,2,0,1,2,2
9,2,3,1,3,0,1,2,2
10,3,1,1,4,0,1,3,1
11,3,2,1,2,0,1,3,2
12,3,3,1,3,0,1,3,2
13,1,1,2,4,1,2,1,1
14,1,1,3,4,2,2,1,1
15,1,2,2,2,1,2,1,2
16,1,2,3,2,2,2,1,2
17,1,3,2,4,1,2,1,2
18,1,3,3,4,2,2,1,2
19,2,1,2,4,1,2,2,1
20,2,1,3,4,2,2,2,1
21,2,2,2,2,1,2,2,2
22,2,2,3,2,2,2,2,2
23,2,3,2,3,1,2,2,2
24,2,3,3,3,2,2,2,2
```

```
25,3,1,2,4,1,2,3,1
26,3,1,3,4,2,2,3,1
27,3,2,2,2,1,2,3,2
28,3,2,3,2,2,2,3,2
29,3,3,2,3,1,2,3,2
30,3,3,3,3,2,2,3,2
31,4,4,2,1,1,2,4,2
32,4,4,3,1,2,2,4,2
33,4,2,2,2,1,2,4,2
34,4,2,3,2,2,2,4,2
35,5,4,2,1,1,2,5,2
36,5,4,3,1,2,2,5,2
37,5,2,2,2,1,2,5,2
38,5,2,3,2,2,2,5,2
];

%% setting up consolidated feature file
%   featuresALL contains all features for all muscles (columns), for all
%   subjects for all trials (rows)
%   columns: features MAV, SSC, ZC, RMS, AR1, AR2, AR3, AR4, MNF, MDF, all
%   muscles M1-15 for each (excluding muscle 8 from data file)
%   rows: subjects 1-24, all trials 1-38 for each
clear featuresAll
for subNum = 1:numSubjects
    for feat = 1:size(featNames,1)
        if feat < 5
            folder = openFolders{1};
        elseif feat < 10
            folder = openFolders{2};
        else
            folder = openFolders{3};
        end
        filename = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\',folder,
            '\',featNames{feat},'.csv');
        tempDataFeat = csvread(filename);

        if feat < 6
            % put in zeros for S7 M9, S9 M7 (unreliable data because of disconnecting
                sensor)
            if subNum == 7
                tempDataFeat(:,9)=0;
```

```
        elseif subNum == 9
            tempDataFeat(:,7)=0;
        end

        featuresAll(((subNum-1)*38 +1):subNum*38,((feat-1)*15 +1):feat*15) = [
            tempDataFeat(:,1:7),tempDataFeat(:,9:16)];
    elseif feat < 10
        subFeat = feat - 5;
        tempSubFeat(:,1:15) = [tempDataFeat(:,subFeat),tempDataFeat(:,subFeat+4),
            tempDataFeat(:,subFeat+(2*4)),tempDataFeat(:,subFeat+(3*4)),tempDataFeat(:,
            subFeat+(4*4)),tempDataFeat(:,subFeat+(5*4)),tempDataFeat(:,subFeat+(6*4)),
            tempDataFeat(:,subFeat+(8*4)),tempDataFeat(:,subFeat+(9*4)),tempDataFeat(:,
            subFeat+(10*4)),tempDataFeat(:,subFeat+(11*4)),tempDataFeat(:,subFeat
            +(12*4)),tempDataFeat(:,subFeat+(13*4)),tempDataFeat(:,subFeat+(14*4)),
            tempDataFeat(:,subFeat+(15*4))];

        % put in zeros for S7 M9, S9 M7 (unreliable data because of disconnecting
            sensor)
        if subNum == 7
            tempSubFeat(:,8)=0; % data muscle 8 was already excluded
        elseif subNum == 9
            tempSubFeat(:,7)=0;
        end

        featuresAll(((subNum-1)*38 +1):subNum*38,((feat-1)*15 +1):feat*15) =
            tempSubFeat;

    else
        % put in zeros for S7 M9, S9 M7 (unreliable data because of disconnecting
            sensor)
        if subNum == 7
            tempDataFeat(:,9)=0;
        elseif subNum == 9
            tempDataFeat(:,7)=0;
        end

        featuresAll(((subNum-1)*38 +1):subNum*38,((feat-1)*15 +1):feat*15) = [
            tempDataFeat(:,1:7),tempDataFeat(:,9:16)];
    end
end
end
```

```
%% get position labels and actual averaged velocity
clear pos_vel_all
for subNum = 1:numSubjects

    pos_vel_all(((subNum-1)*38 +1):subNum*38,1) = trial_labels(:,8); %position 1, 2, 3

    filenameSpeed = strcat('D:\TaylorMasters2\Data\Processing\S',int2str(subNum),'\KUKA\
        MeanSpeed.csv');
    dataSpeed = csvread(filenameSpeed);
    pos_vel_all(((subNum-1)*38 +1):subNum*38,2) = dataSpeed(1:38,4); %actual average hand
        speed

    pos_vel_all(((subNum-1)*38 +1):subNum*38,3) = trial_labels(:,7); %speed label
        stationary, moving

    pos_vel_all(((subNum-1)*38 +1):subNum*38,4) = trial_labels(:,4); %speed label
        stationary, slow, fast
end


%% setup desired predictor subset
muscles_feats_factors = zeros(15,15);
% fill in matrix with desired feature/muscle combinations for predictor
% subset
% 1 = include, 2 = exclude
% row 12 indicates include position, row 13 indicates include average speed
%% continue setting up predictor subset
% column is trial set, rows are trials during data collection
% replace 0 with 1 if want that data trial in the set of trials included
% for predictors
% column 2: flexion/extension
% column 3: ADL 1
% column 4: ADL 2
% column 5: felxion/extension, exclude stationary
% column 6: all trials
% column 7: only P1
% column 8: ADL 1 and ADL 2
trials = [
1,1,0,0,0,1,1,0
2,1,0,0,0,1,1,0
```

```
3,1,0,0,0,1,1,0
4,0,0,0,0,1,0,0
5,0,0,0,0,1,0,0
6,0,0,0,0,1,0,0
7,1,0,0,0,1,0,0
8,1,0,0,0,1,0,0
9,1,0,0,0,1,0,0
10,1,0,0,0,1,0,0
11,1,0,0,0,1,0,0
12,1,0,0,0,1,0,0
13,1,0,0,1,1,1,0
14,1,0,0,1,1,1,0
15,1,0,0,1,1,1,0
16,1,0,0,1,1,1,0
17,1,0,0,1,1,1,0
18,1,0,0,1,1,1,0
19,1,0,0,1,1,0,0
20,1,0,0,1,1,0,0
21,1,0,0,1,1,0,0
22,1,0,0,1,1,0,0
23,1,0,0,1,1,0,0
24,1,0,0,1,1,0,0
25,1,0,0,1,1,0,0
26,1,0,0,1,1,0,0
27,1,0,0,1,1,0,0
28,1,0,0,1,1,0,0
29,1,0,0,1,1,0,0
30,1,0,0,1,1,0,0
31,0,1,0,0,1,0,1
32,0,1,0,0,1,0,1
33,0,1,0,0,1,0,1
34,0,1,0,0,1,0,1
35,0,0,1,0,1,0,1
36,0,0,1,0,1,0,1
37,0,0,1,0,1,0,1
38,0,0,1,0,1,0,1
];

%% get feature / muscle subset of predictors, position and velocity added here too
% from the matrix with all features gathered, get only desired
% feature/muscles combinations
```

```
numMusc_per_feat = sum(muscles_feats_factors,2);
clear predictors_v1
column = 1;
for feature = 1:11
    for muscle = 1:15
        if muscles_feats_factors(feature,muscle) == 1
            predictors_v1(:,column) =  featuresAll(:,(feature-1)*15 +muscle);
            column = column+1;
        end
    end

end
    if muscles_feats_factors(12,1) == 1
        predictors_v1(:,column) = pos_vel_all(:,1);
        column = column+1;
    end
    if muscles_feats_factors(13,1) == 1
        predictors_v1(:,column) = pos_vel_all(:,2);
        column = column+1;
    end
    if muscles_feats_factors(14,1) == 1
        predictors_v1(:,column) = pos_vel_all(:,3);
        column = column+1;
    end
    if muscles_feats_factors(15,1) == 1
        predictors_v1(:,column) = pos_vel_all(:,4);
    end
%%
% specify model type, test number, test set, and labels and trials to
% include for the specified test number
model = 'LDA'; % 'LDA' 'SVM'
test = 77;
set = 16;
labels_trials = [
    2,2,1
    3,2,2
    4,2,3
    5,3,4
    6,3,5
    5,4,6
    6,4,7
```

```
7,2,8
2,6,9
3,2,10
3,2,11
3,2,12
3,7,13
3,7,14
5,3,15
5,4,16
5,8,17
5,8,18
3,2,19
3,2,20
3,2,21
3,2,22
3,2,23
3,2,24
3,2,25
3,2,26
5,3,27
5,3,28
5,4,29
5,4,30
5,8,31
5,8,32
5,3,33
5,4,34
3,2,35
3,2,36
3,2,37
3,2,38
3,2,39
3,2,40
3,2,41
3,2,42
3,2,43
3,2,44
3,2,45
3,2,46
5,3,47
5,4,48
```

```
    5,8,49
    9,2,50
    9,2,51
    9,2,52
    9,2,53
    3,2,54
    3,2,55
    3,2,56
    3,2,57
    3,2,58
    5,3,59
    5,3,60
    5,3,61
    5,8,62
    6,8,63
    6,8,64
    5,4,65
    5,4,66
    5,4,67
    5,8,68
    5,8,69
    5,8,70
    3,2,71
    3,2,72
    3,2,73
    3,2,74
    3,2,75
    3,2,76
    3,2,77
    ];

%% get trial subset of predictors
% from matrix with desired feature/muscle combinations, get only desired
% trials
trialSet = labels_trials(test,2); % column number in trials variable
clear predictors_v2
for sub = 1:numSubjects
    trialCount = 1;
    for trialCounter = 1:38
        if trials(trialCounter,trialSet) == 1
```

```
            predictors_v2((sub-1)*sum(trials(:,trialSet))+trialCount,:)= predictors_v1((sub
                -1)*38 + trialCounter,:);
            trialCount = trialCount +1;
        end
    end
end

%% get trial subset of trial labels

% trialSet = 3; % column of trials variable (trials to include)
labelSet = labels_trials(test,1); % column of trial_labels variable (type of label to use)
clear labels_v1
for sub = 1:numSubjects
    trialCount = 1;
    for trialCounter = 1:38
        if trials(trialCounter,trialSet) == 1
            labels_v1((sub-1)*sum(trials(:,trialSet))+trialCount,:)=trial_labels(
                trialCounter,labelSet);
            trialCount = trialCount +1;
        end
    end
end

%% classify v01
clear mdl ldaResubErr cp cvlda ldaCVErr
% classify with LDA
% mdl = fitcdiscr(predictors_v2,labels_v1);
 mdl = fitcecoc(predictors_v2,labels_v1);

ldaResubErr = resubLoss(mdl);

cplda = cvpartition(labels_v1,'KFold',10);
cvlda = crossval(mdl,'CVPartition',cp);
ldaCVErr = kfoldLoss(cvlda);

%% classify v02
clear predictors_full labels_full train_predictors train_labels test_predictors test_labels
     labels_mdl
clear Mdl
matching(1,test) = 0;
for sub = 1:numSubjects
```

```
    sub
    trials_per_sub = sum(trials(:,trialSet));
    predictors_full = predictors_v2;
    labels_full = labels_v1;

    train_predictors = predictors_full;
    train_predictors((sub-1)*trials_per_sub+1:sub*trials_per_sub,:)=[];
    train_labels = labels_v1;
    train_labels((sub-1)*trials_per_sub+1:sub*trials_per_sub,:)=[];

    test_predictors = predictors_full((sub-1)*trials_per_sub+1:sub*trials_per_sub,:);
    test_labels = labels_full((sub-1)*trials_per_sub+1:sub*trials_per_sub,1);

    switch model
        case 'LDA'
            Mdl = fitcdiscr(train_predictors,train_labels);
        case 'SVM'
            Mdl = fitcecoc(train_predictors,train_labels);
    end

    labels_mdl(:,sub) = predict(Mdl,test_predictors);
    labels_result(sub,test) = sum(eq(labels_mdl(:,sub), test_labels));
    matching(1,test) = matching(1,test) + labels_result(sub,test);
    A(sub,test) = labels_result(sub,test)/trials_per_sub;

    cp = cvpartition(train_labels,'KFold',10);
    cvmdl = crossval(Mdl,'CVPartition',cp);
    CVErr(sub,test) = kfoldLoss(cvmdl);

    a_cv(sub,test) = 1 - CVErr(sub,test);
end
labels_mdl(:,sub+1) = test_labels;
Accuracy(test) = (matching(1,test)/size(predictors_full,1))*100;

filename_01 = strcat('C:\Users\Taylor\Documents\Taylor\Masters1\classification\set',int2str
    (set),'\',model,'\T',int2str(test),'.csv');
csvwrite(filename_01,labels_mdl);

%% save
filename_02 = strcat('C:\Users\Taylor\Documents\Taylor\Masters1\classification\set',int2str
    (set),'\',model,'\labels_result.csv');
```

```
csvwrite(filename_02,labels_result);

filename_03 = strcat('C:\Users\Taylor\Documents\Taylor\Masters1\classification\set',int2str
    (set),'\',model,'\matching.csv');
csvwrite(filename_03,matching);

filename_04 = strcat('C:\Users\Taylor\Documents\Taylor\Masters1\classification\set',int2str
    (set),'\',model,'\A.csv');
csvwrite(filename_04,A);

filename_05 = strcat('C:\Users\Taylor\Documents\Taylor\Masters1\classification\set',int2str
    (set),'\',model,'\a_cv.csv');
csvwrite(filename_05,a_cv);

filename_06 = strcat('C:\Users\Taylor\Documents\Taylor\Masters1\classification\set',int2str
    (set),'\',model,'\Accuracy.csv');
csvwrite(filename_06,Accuracy);
```

Appendix B

Ethics Permissions and Approvals, Forms

B.1 Ethics Approval

Western Research

Research Ethics

Western University Health Science Research Ethics Board
HSREB Delegated Initial Approval Notice

Principal Investigator: Dr. Ana Luisa Trejos
Department & Institution: Unknown, London Health Sciences Centre

Review Type: Delegated
HSREB File Number: 109356
Study Title: Dynamic Calibration of EMG Signals

HSREB Initial Approval Date: August 25, 2017
HSREB Expiry Date: August 25, 2018

Documents Approved and/or Received for Information:

Document Name	Comments	Version Date
Western University Protocol		2017/08/25
Letter of Information & Consent		2017/08/03
Data Collection Form/Case Report Form	Trial Form	2017/05/26
Instruments	List of Instruments	2017/08/03

The Western University Health Science Research Ethics Board (HSREB) has reviewed and approved the above named study, as of the HSREB Initial Approval Date noted above.

HSREB approval for this study remains valid until the HSREB Expiry Date noted above, conditional to timely submission and acceptance of HSREB Continuing Ethics Review.

The Western University HSREB operates in compliance with the Tri-Council Policy Statement Ethical Conduct for Research Involving Humans (TCPS2), the International Conference on Harmonization of Technical Requirements for Registration of Pharmaceuticals for Human Use Guideline for Good Clinical Practice Practices (ICH E6 R1), the Ontario Personal Health Information Protection Act (PHIPA, 2004), Part 4 of the Natural Health Product Regulations, Health Canada Medical Device Regulations and Part C, Division 5, of the Food and Drug Regulations of Health Canada.

Members of the HSREB who are named as Investigators in research studies do not participate in discussions related to, nor vote on such studies when they are presented to the REB.

The HSREB is registered with the U.S. Department of Health & Human Services under the IRB registration number IRB 00000940.

B.2 Letter of Information/Consent

Letter of Information
Title: Dynamic Calibration of EMG Signals
Principal Investigator: Dr. Ana Luisa Trejos

You are being invited to participate in a research study directed by Dr. Ana Luisa Trejos to increase our understanding of factors affecting electromyography (EMG) signals during daily arm movements. At this initial visit, one of the collaborators working on this project will read through this consent form with you, describe the procedure in detail and answer any questions you may have. This study is being conducted by the following researchers:

Dr. Ana Luisa Trejos, Ph.D. (Principal Investigator)
Assistant Professor, Department of Electrical and Computer Engineering
The University of Western Ontario[redacted]
Associate Scientist, Canadian Surgical Technologies & Advanced Robotics (CSTAR)
[redacted]

Shrikant Chinchalkar, (Co-Investigator)
Therapist, Schulich School of Medicine and Dentistry
Roth-McFarlane Hand and Upper Limb Centre
[redacted]

S. Jayne Garland, Ph.D. (Co-Investigator)
Dean, Faculty of Health Sciences
Professor, School of Physical Therapy
The University of Western Ontario[redacted]

Taylor Stanbury, B.E.Sc. (Coordinator)
Graduate Student, Biomedical Engineering Program
The University of Western Ontario[redacted]

Western

Details of the Study

The overall goal of this study is to collect muscle activation data during arm motions to gain a better understanding of factors affecting the muscle activation measured by surface electromyography (EMG). This data will be used to observe muscle activation patterns and detect participants' arm motion based on muscle activations. This data will be used to further develop a wearable arm brace to assist during arm movements.

The experiments will be conducted at the Wearable Biomechatronics Laboratory in the Spencer Engineering Building, after the consent form is signed by you and the study investigator or coordinator. The research coordinator will fill out a Trial Form. You will be asked questions (age, hand dominance, weight, height, level of activity) and the coordinator will measure sections of your dominant arm, waist, and hip.

Activity of arm muscles will be recorded by attaching small sensors on up to 16 muscle groups, located around your shoulder, arm and forearm. The skin where the electrodes will be placed will be cleaned with alcohol, the alcohol will evaporate and the sensors will be attached with a sticky tape. The sensors are not invasive and will not obstruct normal movement. Arm movements will be recorded by an optical sensor. Video of the session will also be recorded. The video recording will not include the face of the participant, and will not be linked to any personal identifiable information.

You will be asked to perform all or a subset of 3 trial sets (isometric, single motion, activities of daily living, as described below). While performing motions, a safe robot, i.e., one that is safe to interact with in all circumstances, will be applying resistive or assistive forces to your hand. These forces are comparable to holding 0 pounds, 5 pounds, or 10 pounds. Prior to the motion trials, you will be asked to perform maximum muscle contractions of select muscles of interest.

Isometric

With your upper arm in each of 5 different orientations, you will be asked to hold your arm still with your elbow bent at 5 different angles in your normal range of motion. Breaks will be given between contractions. The robot arm will apply external forces equivalent to holding 0 pounds, 5 pounds, and 10 pounds. You will perform 5 repetitions at 5 different angles with your arm in 5 different orientations with 3 different force levels.

Single Motion

You will be asked to perform flexion/extension (bending/straightening) motions with your elbow. While holding on to the end of the robot arm, you will bend and straighten your arm through your full range of motion. The robot will resist/assist movement equivalent to you holding 0 pounds, 5 pounds, and 10 pounds. You will watch a timer to guide you while performing the motions at a very slow speed and at a quicker speed. You will perform 5 repetitions with 3 different forces at 2 different speeds with your arm in 5 different orientations.

Activities of Daily Living
You will be asked to perform up to 5 different activities of daily living. Activities of daily living will include: simulating eating (hand to mouth); hand to back pocket; simulating answering telephone (hand to ear); simulating opening and closing a door; and reach above shoulder level in front of body (lifting and extending arm at 45° above horizontal). You will complete the tasks while holding on to the end of the robot arm. The robot will resist/assist movement equivalent to you holding 0 pounds, 5 pounds, and 10 pounds. You will watch a timer to guide you while performing the motions at a very slow speed and at a quicker speed. You will perform 5 repetitions with 3 different forces at 2 different speeds.

Risks
There is the potential for temporary muscle discomfort due to the repetitive tasks being performed. You will be asked to perform motions within your normal range of motion. Slight skin irritation could occur temporarily at the sites of sticky tape attaching EMG sensors to the skin if the skin is very sensitive. There is a risk for harm from the robot. Emergency safety stops are available for you and the investigator to press at any time. You will be holding on to the end of the robot and can let go during any trial if you become uncomfortable/concerned. The robot is only enabled to be moved with varying levels of stiffness during the trial time. The robot will not move when it senses resistance levels outside the expected range. The trials can be stopped immediately at any time if you wish. The loads for one trial can range from no load to loads comparable to the weight of two textbooks. There is a risk of privacy breach, the following confidentiality section outlines precautions taken to avoid this.
NOTE: The participation in this study is voluntary. You may withdraw from the experiments at any time. Data cannot be withdrawn after completing the trials.

Benefits:
Although you may not benefit directly from this study, your participation may contribute to our knowledge of human mechanics and human muscle activation during daily activities, and how to incorporate this knowledge into the design of EMG-driven control systems for assistive devices.

Confidentiality:
All data and video recordings will be stored in a password protected computer (University of Western Ontario, Spencer Engineering Building). Identifiable information will not be linked to video recordings and faces of participants will not be visible in the recordings. Hard copies of any documents will be stored in locked cabinets in a locked office. The only documents containing your name will be the Consent Forms, which will not be linked to any of the recorded data. Consent Forms will be stored separately from other data in a locked cabinet in a locked office. Access to records and data is limited to authorized persons. Your anonymity will be protected at all times by using numeric codes when analyzing your experimental data. Data will be retained for 15 years (in accordance with Lawson policy), then destroyed.

Representatives of the University of Western Ontario Health Sciences Research Ethics Board may require access to your study-related documents to oversee the ethical conduct of this study.

Representatives of Lawson Quality Assurance Education Program may require access to your study-related documents to ensure that proper laws and guidelines are being followed.

Rights:
You do not waive any legal right by consenting to this study.

If you have any questions or concerns regarding participation in our study, please contact Dr. Ana Luisa Trejos [redacted]
If you have any questions about the conduct of this study or your rights as a research subject you may contact The Office of Human Research Ethics [redacted]. A copy of this information package is yours to keep for your personal records.

CONSENT FORM

Title of Research: **Dynamic Calibration of EMG Signals**

Principal Investigator: Dr. Ana Luisa Trejos

Co-Investigators: Shrikant Chinchalkar, Dr. S. Jayne Garland

Collaborators: Taylor Stanbury

For the Participant:

I have read and understand the above information describing this study. I have had the purposes, procedures and technical language of this study explained to me. I have been given sufficient time to consider the above information and to seek advice if I chose to do so. I have had the opportunity to ask questions which have been answered to my satisfaction. I am voluntarily signing this form. I will receive a copy of this consent form for my information.

If at any time I have further questions, problems, or adverse events, I can contact Dr. Ana Luisa Trejos, the principal investigator of the project [redacted] or any of the investigators and collaborators on the project.

If I have any questions about my rights as a research participant or the conduct of this study, I may contact The Office of Human Research Ethics [redacted].

By signing this consent form, I am indicating that I agree to participate in this study.

Name of Participant (please print)	**Signature of Participant**	**Date**
Name of Person Obtaining Informed Consent	**Signature of Person Obtaining Informed Consent**	**Date**

B.3 Trial Form

TRIAL FORM

Title of Research: **Dynamic Calibration of EMG Signals**

Principal Investigator: Dr. Ana Luisa Trejos

Co-Investigators: Shrikant Chinchalkar, Dr. S. Jayne Garland

Coordinator: Taylor Stanbury

To be entered by the Coordinator:

If the participant is not comfortable answering any of these questions, they do not have to respond.

Participant Information		
Subject code:		
Age:		years
Dominant hand:	R L	
Gender:	M F Other	
Weight:		kg
Height:		cm
Waist circumference:		cm
Wrist circumference:		cm
Hip circumference:		cm
Forearm circumference:		cm
Forearm length:		cm
Upper arm length:		cm
Environment Information		
Room temperature:		°C
Time of day:		AM PM

Level of activity (sports, # of times / week exercising): ______________________________________

Notes on trial performance:

Appendix C

Statistical Analyses Tables

C.1 Consolidated statistical analysis of EMG signals during flexion–extension motions

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

Flexion–Extension				Level Mean			Std Error			Significance		
Muscle	Feature	Factor	Significance	L1	L2	L3	L1	L2	L3	SF 1-2	SF 1-3	SF 2-3
BB_S (ch1)	MAV	Position	**0.038**	0.018	0.020	0.015	0.003	0.003	0.003	0.242	**0.063**	**0.010**
		Force	**< 0.001**	0.009	0.034	0.011	0.001	0.006	0.002	**0.005**	**0.001**	**0.004**
		Velocity	**0.011**	0.016	0.020	0.018	0.003	0.003	0.003	**0.003**	0.051	**0.031**
	SSC	Position	**< 0.001**	55.313	51.051	54.106	0.846	0.745	0.730	**< 0.001**	0.092	**< 0.001**
		Force	**< 0.001**	57.517	45.454	57.500	0.694	0.928	0.766	**< 0.001**	0.981	**< 0.001**
		Velocity	**< 0.001**	54.815	52.173	53.483	0.716	0.773	0.722	**< 0.001**	**0.024**	**< 0.001**
	WL	Position	0.059	2.928	3.087	2.483	0.467	0.457	0.515			
		Force	**< 0.001**	1.494	5.140	1.864	0.238	0.802	0.435	**< 0.001**	0.187	**< 0.001**
		Velocity	**0.004**	2.622	3.039	2.837	0.452	0.501	0.452	**0.001**	**0.042**	**0.007**
	ZC	Position	**< 0.001**	60.278	53.923	57.363	1.308	1.259	1.315	**< 0.001**	0.005	**0.001**
		Force	**< 0.001**	60.417	49.500	61.647	1.289	1.297	1.527	**< 0.001**	0.258	**< 0.001**
		Velocity	**< 0.001**	59.166	55.303	57.095	1.330	1.171	1.246	**< 0.001**	**0.028**	**0.001**
	RMS	Position	**0.030**	0.025	0.027	0.021	0.004	0.004	0.004	0.194	0.050	**0.008**
		Force	**< 0.001**	0.012	0.046	0.014	0.002	0.008	0.003	**< 0.001**	0.255	**< 0.001**
		Velocity	**0.008**	0.022	0.026	0.024	0.004	0.004	0.004	**0.002**	**0.038**	**0.049**
	AR1	Position	0.309	2.357	2.367	2.369	0.011	0.016	0.013			
		Force	0.330	2.370	2.354	2.370	0.012	0.020	0.012			
		Velocity	0.252	2.364	2.361	2.369	0.013	0.014	0.013			
	AR2	Position	0.117	-2.418	-2.361	-2.420	0.037	0.052	0.037			
		Force	**< 0.001**	-2.466	-2.261	-2.472	0.034	0.062	0.034	**< 0.001**	0.780	**< 0.001**
		Velocity	**0.022**	-2.419	-2.369	-2.411	0.040	0.044	0.038	**0.024**	0.649	**0.008**
	AR3	Position	**0.006**	1.315	1.214	1.306	0.038	0.050	0.034	**0.008**	0.695	**0.001**
		Force	**< 0.001**	1.380	1.074	1.382	0.032	0.060	0.034	**< 0.001**	0.911	**< 0.001**
		Velocity	**0.002**	1.310	1.237	1.288	0.039	0.043	0.038	**0.003**	0.273	**0.001**
	AR4	Position	**0.001**	-0.336	-0.291	-0.330	0.014	0.018	0.012	**0.002**	0.523	**< 0.001**
		Force	**< 0.001**	-0.363	-0.232	-0.361	0.012	0.020	0.013	**< 0.001**	0.808	**< 0.001**
		Velocity	**< 0.001**	-0.332	-0.302	-0.322	0.014	0.015	0.013	**0.001**	0.165	**< 0.001**
	MNF	Position	**0.002**	106.402	98.110	100.639	2.681	3.144	3.023	**0.001**	**0.005**	0.213
		Force	**0.008**	104.076	96.084	104.991	2.586	3.761	2.868	**0.002**	0.615	**0.016**
		Velocity	**0.029**	103.544	99.865	101.742	2.891	2.769	2.771	**0.012**	0.179	0.050
	MDF	Position	**0.038**	87.950	82.224	83.444	2.795	3.293	3.376	**0.018**	**0.037**	0.595
		Force	0.577	85.229	82.302	86.086	2.768	4.075	3.094			

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

		Velocity	0.272	85.519	83.344	84.755	3.042	2.970	2.955			
BB_L (ch2)	MAV	Position	**< 0.001**	0.019	0.029	0.018	0.002	0.004	0.003	**0.001**	0.726	**< 0.001**
		Force	**< 0.001**	0.013	0.043	0.010	0.002	0.006	0.002	**< 0.001**	**0.006**	**< 0.001**
		Velocity	**0.020**	0.021	0.024	0.022	0.003	0.003	0.002	0.065	0.688	**0.007**
	SSC	Position	**< 0.001**	54.627	48.152	51.130	0.719	0.860	0.838	**< 0.001**	**0.001**	**< 0.001**
		Force	**< 0.001**	53.748	43.148	57.013	0.666	0.995	0.722	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**0.001**	52.379	50.123	51.407	0.821	0.632	0.678	**0.003**	0.110	**< 0.001**
	WL	Position	**< 0.001**	2.748	4.261	2.820	0.300	0.516	0.376	**< 0.001**	0.682	**< 0.001**
		Force	**< 0.001**	2.016	6.056	1.757	0.217	0.721	0.249	**< 0.001**	**0.039**	**< 0.001**
		Velocity	**0.008**	3.133	3.497	3.199	0.421	0.389	0.338	**0.025**	0.637	**0.002**
	ZC	Position	**< 0.001**	59.791	52.945	56.325	1.584	1.326	1.779	**< 0.001**	**0.009**	**0.010**
		Force	**< 0.001**	57.953	47.689	63.420	1.427	1.438	1.757	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**0.006**	57.890	54.785	56.386	1.507	1.360	1.520	**0.004**	0.082	**0.004**
	RMS	Position	**< 0.001**	0.025	0.039	0.025	0.003	0.005	0.003	**0.001**	0.724	**< 0.001**
		Force	**< 0.001**	0.018	0.058	0.014	0.002	0.007	0.002	**< 0.001**	**0.007**	**< 0.001**
		Velocity	**0.015**	0.028	0.032	0.029	0.004	0.004	0.003	**0.046**	0.578	**0.007**
	AR1	Position	**< 0.001**	2.314	2.338	2.356	0.013	0.016	0.014	**0.023**	**< 0.001**	0.086
		Force	**0.029**	2.344	2.305	2.359	0.013	0.022	0.010	**0.007**	0.131	**0.012**
		Velocity	0.692	2.339	2.334	2.335	0.013	0.014	0.014			
	AR2	Position	**0.018**	-2.312	-2.251	-2.349	0.036	0.049	0.043	0.105	0.213	**0.004**
		Force	**< 0.001**	-2.355	-2.111	-2.446	0.036	0.065	0.032	**< 0.001**	**0.003**	**< 0.001**
		Velocity	0.180	-2.324	-2.281	-2.307	0.041	0.041	0.040			
	AR3	Position	**0.002**	1.235	1.095	1.216	0.032	0.047	0.041	**0.002**	0.581	**0.001**
		Force	**< 0.001**	1.250	0.935	1.361	0.033	0.059	0.031	**< 0.001**	**0.001**	**< 0.001**
		Velocity	0.061	1.206	1.151	1.188	0.040	0.036	0.036			
	AR4	Position	**< 0.001**	-0.318	-0.251	-0.298	0.010	0.016	0.015	**< 0.001**	0.131	**< 0.001**
		Force	**< 0.001**	-0.317	-0.190	-0.360	0.012	0.019	0.011	**< 0.001**	**0.001**	**< 0.001**
		Velocity	**0.024**	-0.298	-0.276	-0.292	0.014	0.011	0.012	**0.046**	0.466	**0.006**
	MNF	Position	**0.015**	102.660	95.706	99.968	2.937	2.966	3.293	**0.004**	0.172	**0.022**
		Force	**< 0.001**	101.127	89.263	107.945	2.955	3.147	3.172	**< 0.001**	**0.004**	**< 0.001**
		Velocity	0.058	101.108	97.647	99.579	2.989	2.845	3.011			
	MDF	Position	0.104	84.013	81.217	84.715	3.078	3.117	3.298			
		Force	**< 0.001**	84.121	75.201	90.622	3.168	3.188	3.283	**< 0.001**	**0.010**	**< 0.001**
		Velocity	0.196	84.465	82.008	83.472	3.110	2.966	3.112			

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

BRA (ch3)	MAV	Position	**< 0.001**	0.032	0.032	0.025	0.010	0.003	0.004	0.933	0.224	**0.035**
		Force	**< 0.001**	0.017	0.053	0.018	0.004	0.008	0.005	**< 0.001**	0.670	**< 0.001**
		Velocity	0.090	0.026	0.030	0.033	0.004	0.005	0.008			
	SSC	Position	**< 0.001**	53.693	48.719	52.772	1.336	0.874	0.948	**< 0.001**	0.192	**< 0.001**
		Force	**< 0.001**	55.814	42.041	57.328	1.034	1.138	1.153	**< 0.001**	**0.045**	**< 0.001**
		Velocity	**< 0.001**	53.195	50.197	51.792	0.966	1.063	1.115	**0.002**	0.073	**< 0.001**
	WL	Position	**0.003**	3.466	3.989	3.156	0.575	0.391	0.380	0.347	0.362	**0.009**
		Force	**< 0.001**	2.228	5.823	2.561	0.276	0.606	0.400	**< 0.001**	0.112	**< 0.001**
		Velocity	0.107	3.261	3.614	3.736	0.361	0.376	0.515			
	ZC	Position	**< 0.001**	56.584	48.945	55.707	2.226	1.960	2.244	**< 0.001**	0.323	**< 0.001**
		Force	**< 0.001**	56.804	41.092	63.340	2.179	1.440	2.887	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	56.138	51.462	53.637	2.054	2.080	2.245	**< 0.001**	**0.028**	**< 0.001**
	RMS	Position	**< 0.001**	0.041	0.041	0.032	0.012	0.004	0.006	0.998	0.235	**0.028**
		Force	**< 0.001**	0.023	0.068	0.024	0.005	0.009	0.007	**< 0.001**	0.658	**< 0.001**
		Velocity	0.106	0.033	0.039	0.042	0.005	0.006	0.010			
	AR1	Position	**< 0.001**	2.260	2.267	2.298	0.023	0.027	0.024	0.642	**< 0.001**	**0.014**
		Force	**< 0.001**	2.297	2.192	2.336	0.024	0.023	0.027	**< 0.001**	**0.002**	**< 0.001**
		Velocity	0.199	2.280	2.270	2.275	0.023	0.024	0.025			
	AR2	Position	**< 0.001**	-2.178	-2.097	-2.242	0.057	0.064	0.060	**0.042**	**0.007**	**< 0.001**
		Force	**< 0.001**	-2.278	-1.822	-2.418	0.062	0.055	0.070	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	-2.202	-2.138	-2.178	0.057	0.060	0.062	**0.029**	0.400	**< 0.001**
	AR3	Position	**< 0.001**	1.140	1.003	1.165	0.044	0.049	0.048	**< 0.001**	0.236	**< 0.001**
		Force	**< 0.001**	1.226	0.719	1.364	0.051	0.043	0.056	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	1.133	1.062	1.113	0.047	0.045	0.046	**0.009**	0.395	**< 0.001**
	AR4	Position	**< 0.001**	-0.296	-0.234	-0.294	0.012	0.015	0.014	**< 0.001**	0.862	**< 0.001**
		Force	**< 0.001**	-0.317	-0.137	-0.370	0.016	0.013	0.017	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	-0.286	-0.259	-0.279	0.016	0.012	0.013	**0.009**	0.486	**< 0.001**
	MNF	Position	**< 0.001**	94.828	85.630	95.085	3.792	3.789	4.147	**< 0.001**	0.861	**< 0.001**
		Force	**< 0.001**	94.799	73.185	107.559	3.933	2.890	5.155	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	95.555	88.414	91.574	3.911	3.732	4.003	**0.001**	**0.040**	**< 0.001**
	MDF	Position	**0.001**	76.521	70.567	78.815	3.672	3.662	4.006	**0.001**	0.102	**< 0.001**
		Force	**< 0.001**	75.959	58.849	91.096	3.839	2.748	5.125	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	78.584	72.225	75.095	3.904	3.560	3.842	**0.002**	0.063	**< 0.001**
BRD	MAV	Position	0.268	0.025	0.029	0.037	0.005	0.007	0.014			

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

(ch4)		Force	**< 0.001**	0.024	0.038	0.028	0.008	0.008	0.010	**< 0.001**	0.191	**0.005**
		Velocity	0.055	0.019	0.033	0.039	0.005	0.009	0.012			
	SSC	Position	0.103	58.362	57.548	57.065	1.334	1.351	1.377			
		Force	**< 0.001**	60.951	51.808	60.216	1.328	1.523	1.356	**< 0.001**	0.066	**< 0.001**
		Velocity	**0.004**	60.380	56.173	56.421	1.538	1.260	1.471	**0.001**	**0.001**	0.622
	WL	Position	0.565	3.411	3.611	4.137	0.605	0.745	1.227			
		Force	**< 0.001**	3.017	4.909	3.234	0.866	0.810	0.923	**< 0.001**	0.056	**< 0.001**
		Velocity	**0.015**	2.756	4.064	4.339	0.529	0.923	1.120	**0.006**	**0.019**	0.284
	ZC	Position	**0.012**	60.741	58.460	59.092	2.472	2.418	2.750	**0.003**	0.133	0.549
		Force	**< 0.001**	63.667	50.698	63.928	2.693	1.993	3.226	**< 0.001**	0.817	**< 0.001**
		Velocity	**0.001**	64.812	56.365	57.117	2.636	2.567	2.757	**< 0.001**	**< 0.001**	0.096
	RMS	Position	0.192	0.032	0.038	0.048	0.007	0.009	0.017			
		Force	**< 0.001**	0.031	0.050	0.037	0.010	0.010	0.013	**< 0.001**	0.162	**0.003**
		Velocity	**0.044**	0.024	0.043	0.051	0.006	0.011	0.016	**0.012**	**0.022**	0.171
	AR1	Position	0.608	2.296	2.291	2.294	0.030	0.029	0.032			
		Force	0.548	2.294	2.287	2.299	0.031	0.030	0.032			
		Velocity	0.116	2.299	2.294	2.288	0.033	0.031	0.031			
	AR2	Position	0.151	-2.352	-2.320	-2.327	0.069	0.068	0.076			
		Force	**0.003**	-2.387	-2.220	-2.392	0.072	0.068	0.079	0.000	0.767	0.001
		Velocity	0.201	-2.393	-2.305	-2.301	0.075	0.073	0.075			
	AR3	Position	0.115	1.339	1.302	1.311	0.051	0.050	0.055			
		Force	**< 0.001**	1.400	1.155	1.397	0.053	0.054	0.061	**< 0.001**	0.891	**< 0.001**
		Velocity	0.050	1.392	1.277	1.282	0.053	0.055	0.057			
	AR4	Position	0.065	-0.366	-0.350	-0.356	0.016	0.015	0.016			
		Force	**< 0.001**	-0.394	-0.287	-0.391	0.015	0.018	0.018	**< 0.001**	0.719	**< 0.001**
		Velocity	**0.003**	-0.389	-0.339	-0.344	0.015	0.017	0.017	**< 0.001**	**0.001**	0.217
	MNF	Position	**0.005**	102.945	98.021	99.251	4.470	4.387	4.948	**0.001**	0.121	0.550
		Force	**0.001**	105.150	89.607	105.460	4.414	4.544	5.681	**< 0.001**	0.898	**0.001**
		Velocity	**0.002**	109.195	95.065	95.957	4.437	4.836	4.971	**< 0.001**	**0.001**	0.159
	MDF	Position	**0.001**	81.388	76.750	79.340	4.657	4.596	4.780	**< 0.001**	0.185	0.101
		Force	**0.037**	81.608	71.432	84.438	4.524	4.904	5.696	**0.013**	0.166	**0.010**
		Velocity	**0.002**	86.338	75.240	75.901	4.492	4.863	4.955	**< 0.001**	**< 0.001**	0.151
TRI_LO	MAV	Position	**0.020**	0.016	0.014	0.017	0.002	0.002	0.002	**0.046**	0.419	**0.007**
(ch5)		Force	**< 0.001**	0.009	0.009	0.028	0.001	0.001	0.003	0.683	**0.000**	**0.000**

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

		Velocity	**< 0.001**	0.014	0.015	0.017	0.001	0.002	0.002	0.102	**0.004**	**< 0.001**
	SSC	Position	**< 0.001**	61.730	58.667	58.846	1.011	1.178	1.037	**< 0.001**	**< 0.001**	0.673
		Force	**< 0.001**	63.291	59.515	56.438	0.944	1.255	1.108	**< 0.001**	**< 0.001**	**0.002**
		Velocity	**0.001**	60.630	59.182	59.432	1.028	1.091	1.014	**< 0.001**	**0.002**	0.376
	WL	Position	**< 0.001**	3.168	2.489	3.418	0.404	0.291	0.349	**0.003**	0.217	**< 0.001**
		Force	**< 0.001**	1.957	1.593	5.525	0.242	0.203	0.612	**0.002**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	2.976	2.876	3.222	0.303	0.346	0.374	0.478	0.164	**< 0.001**
	ZC	Position	**< 0.001**	65.146	60.721	62.639	1.519	1.468	1.756	**< 0.001**	**0.037**	**0.021**
		Force	**< 0.001**	69.029	58.167	61.310	1.391	1.495	1.990	**< 0.001**	**< 0.001**	**0.049**
		Velocity	**< 0.001**	65.566	60.756	62.184	1.512	1.390	1.683	**< 0.001**	**< 0.001**	**0.003**
	RMS	Position	**0.018**	0.021	0.018	0.022	0.003	0.002	0.002	**0.041**	0.438	**0.006**
		Force	**< 0.001**	0.012	0.012	0.038	0.001	0.002	0.004	0.844	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	0.019	0.021	0.022	0.002	0.003	0.003	0.170	**0.006**	**< 0.001**
	AR1	Position	**0.006**	2.337	2.347	2.363	0.017	0.020	0.021	0.073	**0.002**	**0.004**
		Force	**< 0.001**	2.354	2.324	2.368	0.019	0.019	0.020	**< 0.001**	0.097	**< 0.001**
		Velocity	**0.017**	2.357	2.343	2.347	0.019	0.019	0.020	**0.005**	0.061	0.266
	AR2	Position	**0.008**	-2.496	-2.468	-2.522	0.036	0.040	0.045	0.123	0.239	**0.002**
		Force	**< 0.001**	-2.571	-2.411	-2.505	0.037	0.039	0.049	**< 0.001**	**0.018**	**0.008**
		Velocity	**< 0.001**	-2.544	-2.460	-2.483	0.039	0.037	0.043	**< 0.001**	**< 0.001**	**0.028**
	AR3	Position	**0.005**	1.486	1.427	1.476	0.029	0.031	0.036	**0.004**	0.655	**0.011**
		Force	**< 0.001**	1.564	1.389	1.436	0.024	0.032	0.044	**< 0.001**	**< 0.001**	0.201
		Velocity	**< 0.001**	1.518	1.423	1.448	0.031	0.028	0.033	**< 0.001**	**< 0.001**	**0.026**
	AR4	Position	**0.001**	-0.421	-0.390	-0.408	0.011	0.012	0.014	**< 0.001**	0.189	**0.029**
		Force	**< 0.001**	-0.450	-0.378	-0.391	0.009	0.012	0.017	**< 0.001**	**< 0.001**	0.367
		Velocity	**< 0.001**	-0.431	-0.389	-0.400	0.013	0.010	0.013	**< 0.001**	**< 0.001**	**0.018**
	MNF	Position	**0.001**	111.234	104.768	110.548	3.718	3.753	4.480	**0.004**	0.769	**0.002**
		Force	**< 0.001**	118.766	96.747	111.037	3.762	3.496	4.913	**< 0.001**	**0.004**	**< 0.001**
		Velocity	**< 0.001**	115.743	103.441	107.366	4.158	3.491	4.071	**< 0.001**	**< 0.001**	**< 0.001**
	MDF	Position	**0.001**	88.452	83.819	90.780	4.102	4.101	4.695	**0.046**	0.318	**< 0.001**
		Force	**< 0.001**	96.003	74.334	92.715	4.307	3.650	5.119	**< 0.001**	0.187	**< 0.001**
		Velocity	**< 0.001**	94.583	82.124	86.345	4.626	3.797	4.272	**< 0.001**	**< 0.001**	**< 0.001**
TRI_LAT (ch6)	MAV	Position	**< 0.001**	0.017	0.018	0.028	0.003	0.002	0.003	0.354	**< 0.001**	**< 0.001**
		Force	**< 0.001**	0.012	0.014	0.037	0.001	0.002	0.005	0.378	**< 0.001**	**< 0.001**
		Velocity	**0.002**	0.019	0.021	0.022	0.003	0.003	0.003	0.074	**0.001**	**0.039**

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

	SSC	Position	**< 0.001**	58.702	53.973	53.461	0.730	0.613	0.680	**< 0.001**	**< 0.001**	0.168
		Force	**< 0.001**	58.286	53.257	54.594	0.606	0.671	0.916	**< 0.001**	**< 0.001**	0.145
		Velocity	**< 0.001**	56.882	54.178	55.076	0.522	0.715	0.717	**< 0.001**	**< 0.001**	**< 0.001**
	WL	Position	**< 0.001**	3.709	3.696	5.907	0.583	0.474	0.728	0.969	**< 0.001**	**< 0.001**
		Force	**< 0.001**	2.777	2.303	8.231	0.369	0.321	1.059	**0.029**	**< 0.001**	**< 0.001**
		Velocity	**0.001**	4.237	4.338	4.736	0.550	0.580	0.595	0.610	**0.022**	**< 0.001**
	ZC	Position	**< 0.001**	65.095	60.139	61.749	1.183	1.098	1.094	**< 0.001**	**< 0.001**	**0.010**
		Force	**< 0.001**	66.529	54.654	65.800	1.057	1.076	1.590	**< 0.001**	0.371	**< 0.001**
		Velocity	**< 0.001**	64.758	59.972	62.253	1.075	1.053	1.186	**< 0.001**	**0.003**	**< 0.001**
	RMS	Position	**< 0.001**	0.023	0.025	0.038	0.004	0.003	0.005	0.326	**< 0.001**	**< 0.001**
		Force	**< 0.001**	0.017	0.018	0.051	0.002	0.003	0.006	0.646	**< 0.001**	**< 0.001**
		Velocity	**0.001**	0.027	0.029	0.031	0.004	0.004	0.004	0.074	**0.001**	**0.028**
	AR1	Position	**< 0.001**	2.365	2.403	2.405	0.006	0.006	0.007	**< 0.001**	**< 0.001**	0.535
		Force	**< 0.001**	2.397	2.355	2.420	0.007	0.009	0.007	**< 0.001**	**0.001**	**< 0.001**
		Velocity	**0.046**	2.382	2.394	2.396	0.007	0.006	0.007	**0.020**	**0.019**	0.540
	AR2	Position	0.061	-2.538	-2.551	-2.568	0.015	0.017	0.016			
		Force	**< 0.001**	-2.615	-2.393	-2.649	0.015	0.029	0.020	**< 0.001**	0.082	**< 0.001**
		Velocity	**0.001**	-2.573	-2.524	-2.559	0.021	0.015	0.013	**0.012**	0.440	**< 0.001**
	AR3	Position	0.080	1.490	1.450	1.465	0.018	0.020	0.020			
		Force	**< 0.001**	1.552	1.300	1.553	0.017	0.030	0.027	**< 0.001**	0.976	**< 0.001**
		Velocity	**< 0.001**	1.516	1.423	1.467	0.024	0.019	0.017	**< 0.001**	**0.036**	**< 0.001**
	AR4	Position	**0.004**	-0.421	-0.394	-0.401	0.008	0.010	0.009	**0.002**	**0.003**	0.262
		Force	**< 0.001**	-0.441	-0.337	-0.438	0.008	0.012	0.013	**< 0.001**	0.725	**< 0.001**
		Velocity	**< 0.001**	-0.431	-0.383	-0.402	0.010	0.009	0.009	**< 0.001**	**0.006**	**< 0.001**
	MNF	Position	**0.013**	117.049	112.330	116.583	2.333	2.338	2.279	**0.009**	0.772	**0.008**
		Force	**< 0.001**	121.438	97.064	127.459	2.363	2.257	3.162	**< 0.001**	**0.005**	**< 0.001**
		Velocity	**< 0.001**	119.868	110.551	115.542	2.676	1.877	2.201	**< 0.001**	**0.018**	**< 0.001**
	MDF	Position	**0.009**	97.748	96.431	101.399	2.371	2.362	2.225	0.459	0.055	**0.002**
		Force	**< 0.001**	103.648	79.181	112.749	2.527	2.285	3.086	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	102.267	94.206	99.105	2.755	1.805	2.208	**< 0.001**	0.105	**< 0.001**
TRI_M (ch7)	MAV	Position	**< 0.001**	0.019	0.017	0.024	0.003	0.002	0.003	0.074	**0.002**	**< 0.001**
		Force	**< 0.001**	0.013	0.011	0.037	0.002	0.001	0.005	0.091	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	0.018	0.020	0.023	0.003	0.003	0.003	**0.027**	**< 0.001**	**< 0.001**
	SSC	Position	**< 0.001**	58.519	57.723	55.933	0.951	0.801	0.796	0.050	**< 0.001**	**< 0.001**

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

		Force	**< 0.001**	60.991	57.115	54.069	0.827	1.297	0.891	**< 0.001**	**< 0.001**	**0.038**
		Velocity	**< 0.001**	59.427	56.277	56.471	0.800	0.938	0.881	**< 0.001**	**< 0.001**	0.606
	WL	Position	**< 0.001**	4.178	3.648	5.151	0.695	0.593	0.748	0.093	**0.004**	**< 0.001**
		Force	**< 0.001**	3.086	1.963	7.928	0.607	0.271	1.158	**0.008**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	4.067	4.162	4.748	0.683	0.641	0.697	0.695	**0.017**	**< 0.001**
	ZC	Position	0.232	63.794	63.904	62.726	1.402	1.441	1.535			
		Force	**< 0.001**	70.405	55.850	64.169	1.331	1.711	1.854	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	68.109	60.244	62.072	1.281	1.438	1.679	**< 0.001**	**< 0.001**	**0.005**
	RMS	Position	**< 0.001**	0.027	0.023	0.034	0.004	0.004	0.004	0.095	**0.001**	**< 0.001**
		Force	**< 0.001**	0.018	0.014	0.051	0.003	0.002	0.007	0.058	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	0.025	0.028	0.031	0.004	0.004	0.004	0.064	**0.001**	**< 0.001**
	AR1	Position	**< 0.001**	2.362	2.390	2.395	0.016	0.013	0.012	**< 0.001**	**< 0.001**	0.335
		Force	**< 0.001**	2.393	2.334	2.420	0.015	0.018	0.011	**< 0.001**	**0.003**	**< 0.001**
		Velocity	**0.048**	2.388	2.375	2.383	0.011	0.014	0.016	0.075	0.592	0.091
	AR2	Position	**0.013**	-2.521	-2.579	-2.571	0.037	0.033	0.034	**0.006**	**0.010**	0.682
		Force	**< 0.001**	-2.648	-2.409	-2.614	0.031	0.047	0.037	**< 0.001**	0.110	**< 0.001**
		Velocity	**< 0.001**	-2.619	-2.510	-2.542	0.027	0.036	0.041	**< 0.001**	**0.002**	**0.038**
	AR3	Position	0.237	1.474	1.507	1.487	0.031	0.031	0.033			
		Force	**< 0.001**	1.602	1.367	1.498	0.023	0.043	0.040	**< 0.001**	**< 0.001**	**0.008**
		Velocity	**< 0.001**	1.567	1.436	1.465	0.024	0.033	0.037	**< 0.001**	**< 0.001**	0.061
	AR4	Position	0.566	-0.411	-0.417	-0.410	0.012	0.013	0.013			
		Force	**< 0.001**	-0.461	-0.369	-0.408	0.010	0.016	0.016	**< 0.001**	**< 0.001**	**0.039**
		Velocity	**< 0.001**	-0.448	-0.389	-0.400	0.010	0.013	0.014	**< 0.001**	**< 0.001**	0.063
	MNF	Position	0.184	114.101	117.627	116.610	3.330	3.577	3.615			
		Force	**< 0.001**	128.026	98.447	121.866	3.422	3.607	4.208	**< 0.001**	**0.009**	**< 0.001**
		Velocity	**< 0.001**	125.271	109.210	113.857	3.300	3.249	3.834	**< 0.001**	**< 0.001**	**< 0.001**
	MDF	Position	**0.025**	95.132	99.597	100.086	3.711	4.053	3.933	**0.043**	**0.008**	0.807
		Force	**< 0.001**	109.200	77.983	107.632	4.125	3.850	4.310	**< 0.001**	0.478	**< 0.001**
		Velocity	**< 0.001**	107.485	90.934	96.397	3.923	3.517	4.096	**< 0.001**	**< 0.001**	**< 0.001**
ISPI (ch8)	MAV	Position	**< 0.001**	0.022	0.050	0.043	0.003	0.007	0.006	**< 0.001**	**< 0.001**	0.201
		Force	**< 0.001**	0.028	0.050	0.037	0.004	0.006	0.005	**< 0.001**	**< 0.001**	**0.001**
		Velocity	**0.022**	0.036	0.041	0.037	0.004	0.005	0.005	**0.007**	0.243	**0.013**
	SSC	Position	**0.001**	56.938	54.743	53.644	0.979	1.018	1.086	0.051	**< 0.001**	0.195
		Force	**< 0.001**	57.388	53.410	54.526	0.869	1.092	0.832	**< 0.001**	**< 0.001**	0.062

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

		Velocity	**0.001**	55.523	54.532	55.270	0.834	0.937	0.955	**0.009**	0.466	**0.001**
	WL	Position	**< 0.001**	3.789	9.799	7.698	0.456	1.295	1.183	**< 0.001**	**< 0.001**	**0.025**
		Force	**< 0.001**	5.561	9.488	6.238	0.767	1.232	0.839	**< 0.001**	**0.015**	**< 0.001**
		Velocity	**0.026**	6.686	7.622	6.979	0.819	1.078	0.877	**0.007**	0.177	**0.019**
	ZC	Position	**< 0.001**	58.329	64.219	59.080	1.206	1.475	1.218	**0.002**	0.453	**< 0.001**
		Force	**< 0.001**	63.994	60.826	56.809	1.058	1.266	1.131	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	0.311	60.917	60.190	60.522	1.122	1.062	1.136			
	RMS	Position	**< 0.001**	0.029	0.065	0.056	0.003	0.009	0.008	**< 0.001**	**< 0.001**	0.185
		Force	**< 0.001**	0.037	0.065	0.048	0.005	0.008	0.006	**< 0.001**	**< 0.001**	**0.002**
		Velocity	**0.028**	0.047	0.054	0.049	0.005	0.007	0.006	**0.008**	0.193	**0.017**
	AR1	Position	**< 0.001**	2.346	2.390	2.385	0.007	0.010	0.013	**< 0.001**	**< 0.001**	0.521
		Force	**< 0.001**	2.377	2.386	2.359	0.008	0.012	0.009	0.071	**< 0.001**	**< 0.001**
		Velocity	0.221	2.376	2.374	2.371	0.010	0.010	0.009			
	AR2	Position	**0.020**	-2.401	-2.504	-2.457	0.028	0.040	0.045	**0.005**	0.076	0.074
		Force	**< 0.001**	-2.503	-2.465	-2.394	0.031	0.042	0.033	**0.024**	**< 0.001**	**0.001**
		Velocity	0.132	-2.468	-2.446	-2.448	0.034	0.035	0.035			
	AR3	Position	0.176	1.332	1.390	1.334	0.034	0.043	0.046			
		Force	**< 0.001**	1.420	1.342	1.294	0.034	0.046	0.034	**< 0.001**	**< 0.001**	**0.043**
		Velocity	0.085	1.369	1.339	1.347	0.036	0.038	0.038			
	AR4	Position	0.063	-0.348	-0.365	-0.338	0.014	0.016	0.016			
		Force	**< 0.001**	-0.378	-0.344	-0.329	0.013	0.017	0.012	**< 0.001**	**< 0.001**	0.108
		Velocity	0.095	-0.357	-0.345	-0.349	0.013	0.014	0.014			
	MNF	Position	**< 0.001**	95.402	111.587	102.041	2.219	2.893	2.363	**< 0.001**	**0.003**	**< 0.001**
		Force	**< 0.001**	107.623	105.967	95.439	2.089	2.509	1.967	0.085	**< 0.001**	**< 0.001**
		Velocity	0.738	103.412	102.789	102.828	2.195	2.019	2.120			
	MDF	Position	**< 0.001**	74.008	96.197	84.958	2.288	3.050	2.705	**< 0.001**	**< 0.001**	**< 0.001**
		Force	**< 0.001**	88.433	89.984	76.746	2.368	2.698	2.068	**0.125**	**< 0.001**	**< 0.001**
		Velocity	0.579	84.822	85.439	84.903	2.458	2.158	2.248			
AD (ch9)	MAV	Position	**< 0.001**	0.011	0.063	0.058	0.002	0.005	0.005	**< 0.001**	**< 0.001**	0.138
		Force	**< 0.001**	0.040	0.063	0.030	0.003	0.006	0.003	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**0.026**	0.046	0.044	0.042	0.004	0.004	0.003	0.087	**0.006**	0.142
	SSC	Position	**< 0.001**	61.771	48.713	45.472	0.925	0.727	0.713	**< 0.001**	**< 0.001**	**< 0.001**
		Force	**< 0.001**	53.428	48.106	54.422	0.568	0.803	0.640	**< 0.001**	**0.026**	**< 0.001**
		Velocity	0.127	52.198	51.654	52.105	0.618	0.656	0.650			

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

	WL	Position	**< 0.001**	2.083	11.486	10.035	0.248	0.942	0.945	**< 0.001**	**< 0.001**	0.054
		Force	**< 0.001**	7.302	11.144	5.157	0.580	0.987	0.476	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**0.003**	8.300	7.827	7.477	0.647	0.664	0.595	**0.035**	**0.001**	**0.035**
	ZC	Position	**< 0.001**	65.467	55.988	53.606	1.550	1.383	1.155	**< 0.001**	**< 0.001**	**0.019**
		Force	**< 0.001**	61.337	55.532	58.193	1.194	1.190	1.201	**< 0.001**	**< 0.001**	**0.003**
		Velocity	**0.004**	59.472	57.885	57.704	1.163	1.098	1.255	**0.001**	**0.005**	0.708
	RMS	Position	**< 0.001**	0.015	0.085	0.077	0.002	0.007	0.007	**< 0.001**	**< 0.001**	0.118
		Force	**< 0.001**	0.053	0.084	0.041	0.004	0.007	0.004	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**0.021**	0.062	0.059	0.057	0.005	0.005	0.005	0.059	**0.005**	0.157
	AR1	Position	**< 0.001**	2.335	2.398	2.385	0.007	0.012	0.013	**< 0.001**	**< 0.001**	0.114
		Force	**0.004**	2.368	2.388	2.361	0.009	0.012	0.009	**0.001**	0.269	**0.006**
		Velocity	0.266	2.378	2.371	2.368	0.011	0.009	0.010			
	AR2	Position	**0.013**	-2.456	-2.424	-2.350	0.023	0.042	0.044	0.351	**0.008**	**0.010**
		Force	**0.006**	-2.429	-2.386	-2.415	0.032	0.041	0.030	**0.008**	0.417	0.301
		Velocity	**0.038**	-2.431	-2.398	-2.401	0.035	0.032	0.033	**0.010**	0.062	0.841
	AR3	Position	**< 0.001**	1.426	1.250	1.159	0.027	0.042	0.044	**< 0.001**	**< 0.001**	**0.003**
		Force	**< 0.001**	1.313	1.204	1.318	0.032	0.041	0.030	**< 0.001**	0.797	**0.001**
		Velocity	**0.020**	1.301	1.261	1.273	0.034	0.032	0.033	**0.005**	0.075	0.222
	AR4	Position	**< 0.001**	-0.388	-0.298	-0.264	0.011	0.014	0.015	**< 0.001**	**< 0.001**	**0.003**
		Force	**< 0.001**	-0.334	-0.278	-0.339	0.012	0.015	0.011	**< 0.001**	0.406	**< 0.001**
		Velocity	**0.006**	-0.326	-0.309	-0.316	0.012	0.012	0.012	**0.002**	0.074	**0.036**
	MNF	Position	**0.044**	106.910	103.236	100.137	2.597	3.027	2.577	0.177	**0.015**	0.123
		Force	**< 0.001**	107.319	102.785	100.179	2.469	2.712	2.269	**< 0.001**	**< 0.001**	0.146
		Velocity	**0.001**	105.479	102.725	102.078	2.398	2.332	2.486	**< 0.001**	**0.001**	0.393
	MDF	Position	**0.009**	82.016	90.121	89.070	2.637	3.168	2.512	**0.004**	**0.005**	0.594
		Force	**< 0.001**	90.327	89.607	81.272	2.594	2.933	2.350	0.604	**< 0.001**	**0.001**
		Velocity	**< 0.001**	89.161	86.447	85.598	2.480	2.444	2.567	**< 0.001**	**< 0.001**	0.179
LD (ch10)	MAV	Position	**< 0.001**	0.022	0.071	0.174	0.003	0.008	0.016	**< 0.001**	**< 0.001**	**< 0.001**
		Force	**< 0.001**	0.073	0.083	0.110	0.007	0.009	0.010	**0.012**	**< 0.001**	**0.001**
		Velocity	**0.006**	0.096	0.085	0.086	0.009	0.008	0.008	**0.002**	**0.002**	0.253
	SSC	Position	**< 0.001**	60.617	50.955	47.703	1.392	1.097	1.066	**< 0.001**	**< 0.001**	**< 0.001**
		Force	**< 0.001**	55.045	51.334	52.895	0.989	1.253	1.034	**< 0.001**	**< 0.001**	**0.009**
		Velocity	**0.026**	52.805	52.980	53.489	1.022	1.091	1.114	0.545	**0.027**	**0.016**
	WL	Position	**< 0.001**	3.710	12.394	31.948	0.407	1.361	3.071	**< 0.001**	**< 0.001**	**< 0.001**

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

		Force	**< 0.001**	13.739	14.678	19.635	1.324	1.570	1.937	0.177	**< 0.001**	**0.003**
		Velocity	**0.003**	17.402	15.013	15.637	1.650	1.409	1.511	**0.001**	**0.003**	0.067
	ZC	Position	**< 0.001**	61.640	54.691	56.711	1.739	1.541	1.714	**< 0.001**	**0.004**	0.068
		Force	**< 0.001**	61.787	55.562	55.694	1.496	1.499	1.499	**< 0.001**	**< 0.001**	0.783
		Velocity	**0.005**	58.375	56.935	57.733	1.458	1.435	1.568	**0.002**	0.167	**0.022**
	RMS	Position	**< 0.001**	0.028	0.095	0.233	0.004	0.010	0.022	**< 0.001**	**< 0.001**	**< 0.001**
		Force	**< 0.001**	0.097	0.111	0.148	0.009	0.012	0.014	**0.009**	**< 0.001**	**0.002**
		Velocity	**0.004**	0.128	0.113	0.115	0.012	0.011	0.011	**0.001**	**0.002**	0.253
	AR1	Position	**< 0.001**	2.320	2.370	2.397	0.007	0.010	0.009	**< 0.001**	**< 0.001**	**0.006**
		Force	0.733	2.360	2.363	2.364	0.006	0.008	0.008			
		Velocity	0.169	2.366	2.360	2.361	0.007	0.007	0.007			
	AR2	Position	0.248	-2.396	-2.383	-2.431	0.032	0.037	0.038			
		Force	**0.002**	-2.436	-2.377	-2.397	0.029	0.038	0.033	**< 0.001**	**0.033**	0.275
		Velocity	**0.014**	-2.413	-2.391	-2.406	0.032	0.032	0.033	**0.019**	0.443	**0.018**
	AR3	Position	**0.011**	1.374	1.262	1.274	0.042	0.041	0.044	**0.003**	**0.013**	0.640
		Force	**< 0.001**	1.352	1.262	1.296	0.035	0.046	0.037	**< 0.001**	**0.002**	0.105
		Velocity	**0.011**	1.310	1.291	1.310	0.037	0.039	0.039	0.053	0.974	**0.006**
	AR4	Position	**0.001**	-0.374	-0.318	-0.319	0.016	0.015	0.017	**< 0.001**	**0.002**	0.903
		Force	**< 0.001**	-0.359	-0.320	-0.333	0.014	0.018	0.014	**< 0.001**	**< 0.001**	0.081
		Velocity	**0.011**	-0.339	-0.332	-0.340	0.015	0.015	0.015	0.062	0.730	**0.005**
	MNF	Position	**0.002**	98.347	94.778	103.468	2.393	2.457	2.924	0.109	0.059	**< 0.001**
		Force	**< 0.001**	104.760	96.156	95.678	2.345	2.248	2.295	**< 0.001**	**< 0.001**	0.565
		Velocity	**< 0.001**	100.587	97.368	98.638	2.306	2.142	2.357	**< 0.001**	**0.009**	**0.013**
	MDF	Position	**< 0.001**	73.617	79.676	92.257	1.845	2.399	2.786	**0.002**	**< 0.001**	**< 0.001**
		Force	**< 0.001**	86.789	80.373	78.387	2.094	2.098	2.120	**< 0.001**	**< 0.001**	0.060
		Velocity	**< 0.001**	83.923	80.322	81.305	2.130	1.927	2.109	**< 0.001**	**0.001**	**0.031**
PD (ch11)	MAV	Position	**< 0.001**	0.023	0.021	0.101	0.003	0.003	0.013	0.353	**< 0.001**	**< 0.001**
		Force	**< 0.001**	0.028	0.026	0.091	0.004	0.004	0.011	0.233	**< 0.001**	**< 0.001**
		Velocity	0.129	0.050	0.046	0.048	0.006	0.006	0.006			
	SSC	Position	**< 0.001**	57.097	55.432	48.237	1.017	1.198	0.867	**0.047**	**< 0.001**	**< 0.001**
		Force	**< 0.001**	56.801	55.203	48.763	0.850	1.073	1.029	**0.005**	**< 0.001**	**< 0.001**
		Velocity	0.473	53.370	53.579	53.818	0.936	0.958	0.962			
	WL	Position	**< 0.001**	3.342	3.456	15.846	0.420	0.508	1.992	0.645	**< 0.001**	**< 0.001**
		Force	**< 0.001**	4.820	4.277	13.547	0.579	0.620	1.650	**0.031**	**< 0.001**	**< 0.001**

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

		Velocity	**0.019**	7.921	7.196	7.527	0.992	0.894	0.922	**0.046**	0.262	**0.013**
	ZC	Position	**< 0.001**	57.005	57.005	52.782	1.418	1.671	1.346	1.000	**0.002**	**< 0.001**
		Force	**< 0.001**	60.885	57.191	48.716	1.396	1.544	1.316	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	0.680	55.840	55.337	55.615	1.386	1.377	1.445			
	RMS	Position	**< 0.001**	0.030	0.028	0.136	0.004	0.005	0.017	0.306	**< 0.001**	**< 0.001**
		Force	**< 0.001**	0.037	0.034	0.123	0.005	0.005	0.015	0.283	**< 0.001**	**< 0.001**
		Velocity	0.088	0.067	0.062	0.064	0.009	0.008	0.008			
	AR1	Position	**< 0.001**	2.313	2.345	2.379	0.006	0.008	0.008	**< 0.001**	**< 0.001**	**< 0.001**
		Force	**< 0.001**	2.361	2.360	2.315	0.006	0.006	0.009	0.739	**< 0.001**	**< 0.001**
		Velocity	0.098	2.352	2.341	2.343	0.008	0.006	0.007			
	AR2	Position	0.069	-2.325	-2.374	-2.365	0.023	0.036	0.033			
		Force	**< 0.001**	-2.446	-2.410	-2.207	0.025	0.031	0.033	**0.011**	**< 0.001**	**< 0.001**
		Velocity	0.201	-2.369	-2.343	-2.351	0.030	0.028	0.029			
	AR3	Position	**0.026**	1.284	1.294	1.217	0.028	0.043	0.037	0.674	**0.034**	**0.007**
		Force	**< 0.001**	1.374	1.327	1.095	0.031	0.038	0.036	**0.005**	**< 0.001**	**< 0.001**
		Velocity	0.358	1.276	1.255	1.265	0.035	0.033	0.034			
	AR4	Position	**0.003**	-0.341	-0.335	-0.298	0.011	0.017	0.014	0.568	**0.002**	**0.001**
		Force	**< 0.001**	-0.367	-0.346	-0.261	0.012	0.015	0.013	**0.002**	**< 0.001**	**< 0.001**
		Velocity	0.425	-0.327	-0.321	-0.325	0.013	0.012	0.013			
	MNF	Position	0.411	91.350	93.464	93.643	2.156	2.364	2.205			
		Force	**< 0.001**	100.495	94.421	83.541	2.168	2.221	2.016	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	0.361	93.550	92.230	92.678	2.090	2.037	2.173			
	MDF	Position	**< 0.001**	69.303	74.113	80.646	1.997	1.956	2.379	**0.002**	**< 0.001**	**< 0.001**
		Force	**< 0.001**	81.129	75.650	67.284	2.045	1.954	1.996	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	0.258	75.557	74.109	74.397	2.023	1.875	2.012			
ECU (ch12)	MAV	Position	**< 0.001**	0.053	0.069	0.079	0.005	0.007	0.008	**0.001**	**< 0.001**	**0.006**
		Force	**< 0.001**	0.051	0.062	0.088	0.005	0.005	0.009	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	0.051	0.074	0.075	0.006	0.007	0.007	**< 0.001**	**< 0.001**	0.744
	SSC	Position	**0.028**	58.805	57.349	57.759	1.118	1.303	1.356	**0.012**	0.095	0.186
		Force	**< 0.001**	59.087	57.098	57.728	1.204	1.279	1.301	**< 0.001**	**0.027**	0.213
		Velocity	**0.008**	59.406	56.730	57.777	0.990	1.490	1.344	**0.002**	**0.026**	0.096
	WL	Position	**< 0.001**	12.215	15.925	18.866	1.361	1.980	2.294	**0.001**	**< 0.001**	**< 0.001**
		Force	**< 0.001**	11.841	14.256	20.909	1.391	1.545	2.722	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	12.082	17.225	17.699	1.608	1.999	2.064	**< 0.001**	**< 0.001**	0.440

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

	ZC	Position	**0.042**	70.823	69.684	71.228	2.434	2.655	2.775	0.148	0.597	**0.012**
		Force	**0.002**	71.406	68.884	71.445	2.344	2.728	2.852	**0.001**	0.972	**0.022**
		Velocity	**0.008**	71.824	69.067	70.844	2.431	3.008	2.528	**0.011**	0.196	0.215
	RMS	Position	**< 0.001**	0.073	0.095	0.109	0.007	0.010	0.012	**0.001**	**< 0.001**	**0.004**
		Force	**< 0.001**	0.070	0.085	0.121	0.007	0.008	0.014	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	0.070	0.103	0.104	0.008	0.010	0.011	**< 0.001**	**< 0.001**	0.668
	AR1	Position	0.490	2.358	2.367	2.369	0.024	0.028	0.027			
		Force	0.450	2.359	2.370	2.364	0.023	0.028	0.030			
		Velocity	0.426	2.358	2.360	2.375	0.023	0.031	0.026			
	AR2	Position	0.068	-2.567	-2.577	-2.600	0.064	0.078	0.077			
		Force	0.797	-2.583	-2.572	-2.589	0.061	0.078	0.082			
		Velocity	0.468	-2.590	-2.554	-2.600	0.062	0.088	0.073			
	AR3	Position	**0.014**	1.540	1.540	1.569	0.053	0.063	0.062	0.962	0.083	**0.009**
		Force	0.253	1.560	1.532	1.557	0.050	0.064	0.065			
		Velocity	0.125	1.571	1.519	1.559	0.052	0.071	0.058			
	AR4	Position	**0.005**	-0.450	-0.450	-0.466	0.017	0.019	0.019	0.989	**0.012**	**0.004**
		Force	0.079	-0.460	-0.446	-0.460	0.016	0.020	0.020			
		Velocity	**0.013**	-0.465	-0.444	-0.457	0.018	0.021	0.017	**0.008**	0.142	0.192
	MNF	Position	**0.003**	129.071	128.957	133.340	5.210	5.905	5.907	0.952	**0.006**	**0.006**
		Force	**0.020**	131.241	126.932	133.194	5.232	5.864	6.035	**0.009**	0.326	**0.014**
		Velocity	0.115	132.453	127.946	130.968	5.579	6.279	5.341	**0.041**	0.416	0.254
	MDF	Position	**< 0.001**	112.325	112.922	118.376	5.309	6.041	5.963	0.767	**< 0.001**	**0.003**
		Force	**0.034**	114.432	111.312	117.878	5.450	5.905	6.018	**0.048**	0.058	**0.009**
		Velocity	0.458	115.457	112.804	115.361	5.812	6.224	5.425			
ECR (ch13)	MAV	Position	0.504	0.037	0.036	0.040	0.007	0.006	0.007			
		Force	**< 0.001**	0.030	0.046	0.036	0.004	0.008	0.009	**< 0.001**	0.211	**0.006**
		Velocity	**0.001**	0.030	0.041	0.041	0.006	0.007	0.007	**0.004**	**< 0.001**	0.823
	SSC	Position	**0.001**	59.219	59.120	57.420	2.025	1.962	1.812	0.828	**< 0.001**	**0.001**
		Force	**< 0.001**	59.528	56.200	60.031	1.887	1.994	1.961	**< 0.001**	0.356	**< 0.001**
		Velocity	0.051	59.559	57.892	58.308	1.982	1.896	1.949			
	WL	Position	0.240	6.961	6.990	7.659	1.086	1.144	1.231			
		Force	**< 0.001**	6.180	8.924	6.507	1.057	1.363	1.063	**< 0.001**	0.354	**< 0.001**
		Velocity	**0.001**	6.101	7.895	7.614	1.076	1.258	1.124	**< 0.001**	**< 0.001**	0.288
	ZC	Position	**0.047**	69.024	68.689	66.348	3.840	3.650	3.175	0.678	**0.016**	**0.029**

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

		Force	**< 0.001**	69.260	65.334	69.467	3.546	3.466	3.677	**< 0.001**	0.837	**< 0.001**
		Velocity	0.075	70.628	66.786	66.647	3.853	3.409	3.544			
	RMS	Position	0.506	0.049	0.048	0.053	0.009	0.008	0.009			
		Force	**< 0.001**	0.040	0.062	0.049	0.006	0.010	0.011	**< 0.001**	0.202	**0.007**
		Velocity	**< 0.001**	0.040	0.055	0.056	0.008	0.009	0.010	**0.003**	**< 0.001**	0.806
	AR1	Position	0.054	2.332	2.343	2.361	0.033	0.030	0.026			
		Force	**0.008**	2.349	2.354	2.333	0.030	0.028	0.032	0.546	0.113	**0.002**
		Velocity	0.502	2.344	2.348	2.344	0.032	0.028	0.030			
	AR2	Position	0.209	-2.473	-2.491	-2.502	0.061	0.057	0.047			
		Force	0.091	-2.516	-2.468	-2.483	0.047	0.054	0.064			
		Velocity	0.470	-2.511	-2.478	-2.477	0.049	0.056	0.061			
	AR3	Position	0.591	1.432	1.444	1.436	0.050	0.049	0.042			
		Force	**0.006**	1.469	1.393	1.451	0.038	0.049	0.055	**0.002**	0.401	**0.016**
		Velocity	0.128	1.467	1.420	1.426	0.039	0.049	0.053			
	AR4	Position	0.582	-0.400	-0.402	-0.395	0.018	0.019	0.018			
		Force	**0.001**	-0.413	-0.377	-0.407	0.016	0.020	0.020	**< 0.001**	0.394	**0.003**
		Velocity	**0.012**	-0.415	-0.390	-0.394	0.017	0.019	0.019	**0.005**	**0.022**	0.182
	MNF	Position	0.144	121.248	120.242	116.528	7.387	6.820	5.579			
		Force	**0.001**	121.850	115.515	120.653	6.850	6.251	6.722	**0.001**	0.519	**0.002**
		Velocity	0.124	124.647	116.927	116.444	7.478	6.274	6.503			
	MDF	Position	0.172	104.246	103.463	99.636	7.022	6.588	5.140			
		Force	**0.022**	104.741	99.598	103.006	6.531	5.934	6.330	0.014	0.331	0.015
		Velocity	0.115	108.212	100.004	99.129	7.267	6.021	6.192			
FCU	MAV	Position	**< 0.001**	0.044	0.057	0.050	0.005	0.006	0.006	**< 0.001**	**0.011**	**0.008**
(ch14)		Force	**< 0.001**	0.039	0.052	0.060	0.004	0.006	0.007	**< 0.001**	**< 0.001**	**0.003**
		Velocity	**< 0.001**	0.034	0.057	0.059	0.004	0.007	0.007	**< 0.001**	**< 0.001**	0.327
	SSC	Position	**0.001**	57.991	57.846	59.184	0.558	0.628	0.653	0.736	**0.018**	**< 0.001**
		Force	**< 0.001**	59.371	57.337	58.312	0.675	0.615	0.482	**< 0.001**	**0.010**	**0.005**
		Velocity	**< 0.001**	59.881	57.225	57.916	0.793	0.487	0.552	**< 0.001**	**0.001**	**< 0.001**
	WL	Position	**< 0.001**	9.482	12.798	11.458	1.014	1.493	1.391	**< 0.001**	**0.002**	**0.020**
		Force	**< 0.001**	8.729	11.277	13.733	0.997	1.242	1.647	**< 0.001**	**< 0.001**	**< 0.001**
		Velocity	**< 0.001**	7.770	12.767	13.201	0.925	1.548	1.539	**< 0.001**	**< 0.001**	0.318
	ZC	Position	**< 0.001**	67.696	68.435	71.347	1.025	1.428	1.439	0.339	**< 0.001**	**< 0.001**
		Force	**< 0.001**	69.451	67.674	70.354	1.275	1.317	1.246	**0.001**	0.078	**< 0.001**

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

		Velocity	0.180	70.123	68.517	68.838	1.409	1.228	1.279			
	RMS	Position	**< 0.001**	0.059	0.078	0.068	0.006	0.009	0.008	**< 0.001**	**0.009**	**0.006**
		Force	**< 0.001**	0.054	0.070	0.081	0.006	0.008	0.009	**< 0.001**	**< 0.001**	**0.003**
		Velocity	**< 0.001**	0.046	0.078	0.081	0.005	0.009	0.009	**< 0.001**	**< 0.001**	0.261
	AR1	Position	0.438	2.416	2.423	2.419	0.008	0.006	0.007			
		Force	0.062	2.412	2.420	2.426	0.008	0.007	0.005			
		Velocity	**0.014**	2.407	2.425	2.426	0.009	0.006	0.006	**0.030**	**0.004**	0.672
	AR2	Position	**0.036**	-2.652	-2.677	-2.694	0.022	0.025	0.026	0.097	**0.011**	0.123
		Force	**0.029**	-2.665	-2.658	-2.700	0.026	0.027	0.017	0.416	**0.014**	**0.008**
		Velocity	**0.036**	-2.664	-2.674	-2.686	0.026	0.023	0.022	0.444	**0.032**	0.142
	AR3	Position	**0.008**	1.574	1.597	1.624	0.020	0.026	0.028	0.109	**0.002**	**0.024**
		Force	**0.005**	1.598	1.573	1.623	0.027	0.028	0.018	**0.004**	0.059	**0.003**
		Velocity	**0.046**	1.605	1.586	1.604	0.026	0.024	0.023	0.102	0.921	**0.012**
	AR4	Position	**0.004**	-0.441	-0.449	-0.463	0.008	0.011	0.012	0.199	**0.002**	**0.006**
		Force	**0.001**	-0.452	-0.439	-0.462	0.011	0.012	0.008	**< 0.001**	0.081	**0.001**
		Velocity	**0.006**	-0.457	-0.444	-0.453	0.011	0.010	0.010	**0.023**	0.368	**0.002**
	MNF	Position	**< 0.001**	122.325	126.219	130.994	3.197	3.970	3.839	0.079	**< 0.001**	**< 0.001**
		Force	**< 0.001**	125.790	124.095	129.654	3.516	3.916	3.319	0.088	**< 0.001**	**< 0.001**
		Velocity	0.844	126.415	126.394	126.730	3.731	3.523	3.579			
	MDF	Position	**< 0.001**	104.006	108.170	112.965	3.715	4.258	4.160	0.086	**< 0.001**	**< 0.001**
		Force	**< 0.001**	106.541	106.532	112.069	3.843	4.260	3.727	0.994	**< 0.001**	**< 0.001**
		Velocity	0.239	106.767	109.242	109.133	4.077	3.893	3.929			
FCR (ch15)	MAV	Position	**0.003**	0.040	0.049	0.040	0.005	0.006	0.005	**0.001**	0.926	**0.003**
		Force	**< 0.001**	0.032	0.049	0.047	0.004	0.006	0.006	**< 0.001**	**< 0.001**	0.146
		Velocity	**< 0.001**	0.028	0.048	0.052	0.003	0.006	0.007	**< 0.001**	**< 0.001**	0.059
	SSC	Position	**0.016**	56.935	56.379	57.690	0.958	1.030	0.981	0.104	0.087	0.004
		Force	**< 0.001**	58.605	55.679	56.719	1.066	0.814	1.093	0.000	0.000	0.061
		Velocity	**0.001**	58.604	55.892	56.508	1.212	0.900	0.897	0.000	0.003	0.003
	WL	Position	**0.003**	8.291	10.297	8.593	0.960	1.265	1.068	0.001	0.521	0.009
		Force	**< 0.001**	6.862	10.385	9.934	0.810	1.234	1.206	0.000	0.000	0.262
		Velocity	**< 0.001**	5.947	10.234	11.001	0.695	1.261	1.354	0.000	0.000	0.066
	ZC	Position	**0.019**	66.094	66.699	68.266	1.385	1.425	1.329	0.372	**0.005**	**0.032**
		Force	**0.007**	68.231	65.619	67.208	1.452	1.335	1.313	**0.002**	**0.016**	**0.019**
		Velocity	0.066	68.616	66.015	66.427	1.590	1.303	1.336			

Table C.1: Mean, std error, and significance of 11 EMG feature values for 15 muscles for 3 levels of position, force, and velocity during elbow flexion–extension.

	RMS	Position	**0.004**	0.054	0.066	0.054	0.006	0.008	0.007	**0.001**	0.892	**0.006**
		Force	**< 0.001**	0.044	0.067	0.064	0.005	0.008	0.008	**< 0.001**	**< 0.001**	0.153
		Velocity	**< 0.001**	0.038	0.066	0.071	0.005	0.008	0.009	**< 0.001**	**< 0.001**	0.046
	AR1	Position	0.210	2.401	2.401	2.409	0.012	0.011	0.010			
		Force	**0.001**	2.393	2.410	2.407	0.012	0.009	0.012	**0.006**	**< 0.001**	0.699
		Velocity	**0.010**	2.392	2.409	2.410	0.012	0.010	0.010	**0.017**	**0.003**	0.784
	AR2	Position	0.071	-2.583	-2.582	-2.615	0.028	0.024	0.021			
		Force	0.313	-2.590	-2.589	-2.601	0.023	0.025	0.023			
		Velocity	0.050	-2.584	-2.593	-2.602	0.021	0.025	0.025			
	AR3	Position	0.052	1.493	1.485	1.525	0.026	0.024	0.021			
		Force	**0.031**	1.514	1.483	1.505	0.022	0.025	0.022	**0.010**	0.260	0.111
		Velocity	0.123	1.509	1.490	1.504	0.021	0.025	0.023			
	AR4	Position	0.051	-0.408	-0.405	-0.421	0.011	0.011	0.009			
		Force	**0.004**	-0.419	-0.402	-0.411	0.011	0.011	0.010	**0.002**	**0.036**	0.102
		Velocity	0.071	-0.417	-0.405	-0.411	0.010	0.011	0.010			
	MNF	Position	0.089	117.724	119.681	121.462	3.619	3.548	3.048			
		Force	0.541	119.967	118.791	120.109	3.363	3.582	3.152			
		Velocity	0.973	119.886	119.486	119.495	3.213	3.511	3.493			
	MDF	Position	0.071	100.510	103.145	104.557	4.359	4.191	3.672			
		Force	0.256	101.863	103.017	103.332	4.038	4.157	3.896			
		Velocity	0.612	101.786	103.354	103.072	3.755	4.224	4.193			

C.2 Consolidated statistical analysis of EMG signals during ADL 1 motions

Table C.2: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 1.

ADL 1				Level Mean		Std Error	
Muscle	Feature	Factor	Significance	L1	L2	L1	L2
BB_S (ch1)	MAV	Force	**< 0.001**	0.021	0.031	0.003	0.004
		Velocity	0.676	0.026	0.026	0.004	0.004
	SSC	Force	**< 0.001**	47.568	45.220	0.965	0.788
		Velocity	**0.007**	45.874	46.914	0.876	0.889
	WL	Force	**< 0.001**	3.231	4.803	0.476	0.676
		Velocity	0.542	4.047	3.987	0.588	0.564
	ZC	Force	0.597	50.293	50.003	1.235	1.159
		Velocity	0.101	49.706	50.590	1.194	1.195
	RMS	Force	**< 0.001**	0.029	0.042	0.004	0.006
		Velocity	0.526	0.035	0.036	0.005	0.005
	AR1	Force	0.552	2.363	2.358	0.019	0.019
		Velocity	0.703	2.361	2.359	0.018	0.020
	AR2	Force	0.072	-2.303	-2.263	0.058	0.058
		Velocity	0.500	-2.277	-2.289	0.056	0.059
	AR3	Force	**0.004**	1.128	1.070	0.055	0.054
		Velocity	0.165	1.086	1.112	0.053	0.056
	AR4	Force	**< 0.001**	-0.255	-0.230	0.019	0.018
		Velocity	0.083	-0.236	-0.248	0.018	0.019
	MNF	Force	0.579	95.834	96.350	3.437	3.401
		Velocity	0.802	96.209	95.975	3.477	3.362
	MDF	Force	0.224	81.979	83.437	3.769	3.732
		Velocity	0.255	83.314	82.102	3.781	3.701
BB_L (ch2)	MAV	Force	**< 0.001**	0.034	0.049	0.004	0.006
		Velocity	0.074	0.040	0.043	0.005	0.005
	SSC	Force	**< 0.001**	44.213	42.526	0.807	0.773
		Velocity	0.080	42.952	43.788	0.826	0.793
	WL	Force	**< 0.001**	5.045	7.144	0.610	0.861
		Velocity	0.051	5.945	6.244	0.721	0.748
	ZC	Force	**0.011**	49.066	48.088	1.427	1.318
		Velocity	0.121	48.059	49.095	1.358	1.440
	RMS	Force	**< 0.001**	0.046	0.065	0.006	0.008
		Velocity	0.060	0.054	0.057	0.007	0.007

Table C.2: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 1.

	AR1	Force	0.079	2.320	2.308	0.019	0.018
		Velocity	0.897	2.314	2.315	0.019	0.019
	AR2	Force	**0.002**	-2.163	-2.113	0.054	0.054
		Velocity	0.622	-2.132	-2.145	0.057	0.054
	AR3	Force	**< 0.001**	0.989	0.931	0.049	0.050
		Velocity	0.516	0.952	0.968	0.053	0.048
	AR4	Force	**< 0.001**	-0.210	-0.188	0.016	0.016
		Velocity	0.511	-0.196	-0.202	0.017	0.015
	MNF	Force	**0.026**	92.484	91.022	3.433	3.156
		Velocity	0.437	91.333	92.173	3.437	3.211
	MDF	Force	0.077	79.189	78.068	3.709	3.432
		Velocity	0.696	78.416	78.841	3.714	3.484
BRA (ch3)	MAV	Force	**< 0.001**	0.043	0.062	0.009	0.012
		Velocity	**0.045**	0.050	0.056	0.009	0.012
	SSC	Force	**< 0.001**	45.524	42.559	1.034	0.955
		Velocity	0.169	43.772	44.310	0.956	1.013
	WL	Force	**< 0.001**	4.881	6.780	0.631	0.818
		Velocity	**0.012**	5.559	6.102	0.660	0.790
	ZC	Force	**0.001**	43.636	41.903	1.387	1.566
		Velocity	**0.047**	42.066	43.473	1.418	1.577
	RMS	Force	**< 0.001**	0.057	0.080	0.012	0.015
		Velocity	**0.035**	0.065	0.072	0.013	0.015
	AR1	Force	**< 0.001**	2.244	2.221	0.020	0.022
		Velocity	0.388	2.237	2.228	0.021	0.021
	AR2	Force	**< 0.001**	-1.995	-1.902	0.046	0.051
		Velocity	0.672	-1.954	-1.943	0.050	0.050
	AR3	Force	**< 0.001**	0.891	0.789	0.037	0.039
		Velocity	0.726	0.844	0.836	0.038	0.040
	AR4	Force	**< 0.001**	-0.191	-0.156	0.012	0.012
		Velocity	0.759	-0.175	-0.172	0.011	0.013
	MNF	Force	**0.006**	76.699	74.658	2.887	3.023
		Velocity	0.066	74.516	76.841	2.940	3.054
	MDF	Force	0.172	61.472	60.593	3.072	3.082
		Velocity	0.085	59.980	62.085	3.031	3.200

Table C.2: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 1.

BRD (ch4)	MAV	Force	**0.003**	0.031	0.041	0.009	0.011
		Velocity	**0.025**	0.034	0.038	0.010	0.011
	SSC	Force	**< 0.001**	55.503	52.590	1.524	1.625
		Velocity	0.773	53.949	54.144	1.479	1.706
	WL	Force	**< 0.001**	3.751	4.887	0.994	1.058
		Velocity	0.074	4.199	4.439	1.012	1.036
	ZC	Force	**0.026**	54.184	51.749	2.494	2.362
		Velocity	0.264	52.532	53.401	2.218	2.577
	RMS	Force	**0.002**	0.041	0.054	0.012	0.015
		Velocity	**0.012**	0.045	0.050	0.013	0.014
	AR1	Force	0.589	2.287	2.292	0.027	0.027
		Velocity	0.076	2.295	2.283	0.028	0.026
	AR2	Force	0.216	-2.273	-2.244	0.068	0.068
		Velocity	0.333	-2.267	-2.250	0.069	0.067
	AR3	Force	**0.015**	1.242	1.185	0.057	0.058
		Velocity	0.753	1.217	1.211	0.058	0.056
	AR4	Force	**0.003**	-0.325	-0.300	0.019	0.019
		Velocity	0.809	-0.311	-0.313	0.019	0.019
	MNF	Force	0.386	91.738	90.105	4.763	4.690
		Velocity	0.687	90.683	91.160	4.508	4.831
	MDF	Force	0.910	71.822	71.612	4.840	4.935
		Velocity	0.637	71.425	72.009	4.743	4.932
TRI_LO (ch5)	MAV	Force	0.087	0.009	0.009	0.001	0.001
		Velocity	**< 0.001**	0.008	0.010	0.001	0.001
	SSC	Force	**< 0.001**	62.556	60.424	1.152	1.219
		Velocity	**< 0.001**	62.167	60.813	1.201	1.150
	WL	Force	0.249	1.704	1.803	0.265	0.254
		Velocity	**< 0.001**	1.604	1.903	0.245	0.270
	ZC	Force	**< 0.001**	64.056	60.657	1.473	1.423
		Velocity	0.230	62.874	61.839	1.564	1.385
	RMS	Force	0.218	0.011	0.012	0.002	0.002
		Velocity	**< 0.001**	0.011	0.013	0.002	0.002
	AR1	Force	0.087	2.325	2.332	0.019	0.020
		Velocity	**0.014**	2.322	2.335	0.019	0.021

Table C.2: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 1.

	AR2	Force	0.093	-2.470	-2.451	0.036	0.037
		Velocity	0.250	-2.452	-2.469	0.034	0.039
	AR3	Force	**0.006**	1.471	1.434	0.027	0.026
		Velocity	0.828	1.451	1.454	0.026	0.028
	AR4	Force	**0.001**	-0.413	-0.395	0.010	0.010
		Velocity	0.883	-0.404	-0.403	0.011	0.010
	MNF	Force	**< 0.001**	105.681	100.494	3.337	3.295
		Velocity	0.703	102.771	103.404	3.379	3.375
	MDF	Force	**0.001**	81.122	77.146	3.526	3.638
		Velocity	0.152	77.963	80.306	3.639	3.617
TRI_LAT (ch6)	MAV	Force	**0.002**	0.011	0.013	0.002	0.002
		Velocity	**< 0.001**	0.011	0.013	0.002	0.002
	SSC	Force	**< 0.001**	57.506	54.787	0.489	0.579
		Velocity	**< 0.001**	56.712	55.581	0.511	0.538
	WL	Force	**0.013**	2.196	2.553	0.338	0.393
		Velocity	**< 0.001**	2.154	2.596	0.339	0.388
	ZC	Force	**< 0.001**	60.863	57.938	1.017	1.028
		Velocity	0.593	59.499	59.302	1.084	0.958
	RMS	Force	**0.002**	0.014	0.018	0.002	0.003
		Velocity	**< 0.001**	0.015	0.018	0.002	0.003
	AR1	Force	**0.009**	2.359	2.373	0.007	0.008
		Velocity	0.187	2.362	2.370	0.008	0.007
	AR2	Force	0.408	-2.478	-2.467	0.019	0.023
		Velocity	0.442	-2.467	-2.478	0.023	0.019
	AR3	Force	**0.009**	1.422	1.384	0.019	0.024
		Velocity	0.957	1.403	1.403	0.023	0.020
	AR4	Force	**0.001**	-0.389	-0.370	0.008	0.010
		Velocity	0.996	-0.380	-0.380	0.009	0.008
	MNF	Force	**< 0.001**	105.510	101.360	2.263	2.138
		Velocity	**0.045**	102.432	104.438	2.391	2.007
	MDF	Force	**0.018**	85.524	82.838	2.410	2.177
		Velocity	**0.007**	82.555	85.807	2.546	2.029
TRI_M (ch7)	MAV	Force	0.154	0.011	0.012	0.002	0.001
		Velocity	**0.020**	0.010	0.012	0.002	0.001

Table C.2: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 1.

	SSC	Force	**< 0.001**	59.584	57.574	1.189	1.109
		Velocity	0.169	58.882	58.276	1.228	1.062
	WL	Force	0.718	2.246	2.297	0.371	0.337
		Velocity	**0.010**	2.065	2.478	0.369	0.340
	ZC	Force	**< 0.001**	61.689	58.340	1.830	2.193
		Velocity	**0.005**	59.238	60.790	2.047	1.950
	RMS	Force	0.213	0.015	0.016	0.002	0.002
		Velocity	**0.017**	0.014	0.016	0.002	0.002
	AR1	Force	0.523	2.345	2.339	0.017	0.018
		Velocity	0.072	2.335	2.349	0.018	0.017
	AR2	Force	**0.049**	-2.485	-2.440	0.038	0.047
		Velocity	**0.037**	-2.446	-2.479	0.042	0.042
	AR3	Force	**0.012**	1.454	1.400	0.032	0.044
		Velocity	0.181	1.418	1.436	0.037	0.038
	AR4	Force	**0.010**	-0.404	-0.384	0.013	0.017
		Velocity	0.315	-0.391	-0.397	0.014	0.015
	MNF	Force	**0.008**	108.024	103.385	3.718	4.347
		Velocity	**< 0.001**	102.896	108.513	3.929	4.086
	MDF	Force	0.076	87.080	83.681	4.089	4.361
		Velocity	**< 0.001**	81.699	89.061	4.032	4.307
ISPI (ch8)	MAV	Force	**< 0.001**	0.053	0.077	0.009	0.013
		Velocity	0.329	0.064	0.066	0.011	0.011
	SSC	Force	**< 0.001**	52.601	50.573	1.179	1.201
		Velocity	0.109	51.290	51.883	1.187	1.204
	WL	Force	**< 0.001**	10.263	14.551	1.748	2.424
		Velocity	**0.011**	12.080	12.734	2.077	2.088
	ZC	Force	**< 0.001**	61.541	59.737	1.395	1.540
		Velocity	0.058	60.118	61.160	1.410	1.543
	RMS	Force	**< 0.001**	0.069	0.101	0.012	0.018
		Velocity	0.303	0.084	0.086	0.015	0.014
	AR1	Force	0.305	2.383	2.377	0.013	0.012
		Velocity	0.945	2.380	2.380	0.012	0.014
	AR2	Force	**0.005**	-2.452	-2.406	0.049	0.049
		Velocity	0.637	-2.424	-2.434	0.049	0.050

Table C.2: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 1.

	AR3	Force	**< 0.001**	1.325	1.261	0.053	0.054
		Velocity	0.535	1.286	1.300	0.054	0.054
	AR4	Force	**< 0.001**	-0.339	-0.312	0.019	0.020
		Velocity	0.449	-0.323	-0.328	0.020	0.019
	MNF	Force	**0.003**	107.385	105.096	2.713	2.952
		Velocity	**0.034**	105.162	107.319	2.764	2.942
	MDF	Force	**0.016**	92.541	91.002	2.862	3.029
		Velocity	0.055	90.841	92.702	2.814	3.112
AD (ch9)	MAV	Force	**< 0.001**	0.070	0.090	0.006	0.008
		Velocity	**< 0.001**	0.073	0.087	0.007	0.008
	SSC	Force	**0.001**	48.403	47.258	0.645	0.721
		Velocity	0.782	47.778	47.883	0.688	0.701
	WL	Force	**< 0.001**	12.741	16.715	1.260	1.663
		Velocity	**< 0.001**	13.386	16.070	1.372	1.552
	ZC	Force	0.289	56.006	55.533	1.435	1.431
		Velocity	0.852	55.705	55.834	1.343	1.564
	RMS	Force	**< 0.001**	0.094	0.120	0.009	0.011
		Velocity	**< 0.001**	0.097	0.117	0.009	0.010
	AR1	Force	0.290	2.385	2.378	0.013	0.013
		Velocity	0.773	2.383	2.380	0.012	0.015
	AR2	Force	0.052	-2.385	-2.351	0.040	0.040
		Velocity	0.929	-2.369	-2.367	0.037	0.045
	AR3	Force	**0.007**	1.210	1.162	0.037	0.039
		Velocity	0.904	1.185	1.187	0.035	0.042
	AR4	Force	**0.001**	-0.283	-0.262	0.013	0.013
		Velocity	0.725	-0.271	-0.273	0.012	0.014
	MNF	Force	0.677	102.831	103.196	3.175	3.188
		Velocity	0.999	103.015	103.013	3.044	3.444
	MDF	Force	0.397	89.751	90.549	3.428	3.442
		Velocity	0.970	90.184	90.116	3.339	3.696
LD (ch10)	MAV	Force	**< 0.001**	0.082	0.113	0.013	0.017
		Velocity	**< 0.001**	0.087	0.108	0.014	0.016
	SSC	Force	**< 0.001**	52.084	49.845	1.220	1.196
		Velocity	0.210	51.195	50.733	1.261	1.159

Table C.2: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 1.

	WL	Force	**< 0.001**	14.821	19.999	2.205	2.904
		Velocity	**< 0.001**	15.471	19.350	2.295	2.813
	ZC	Force	**< 0.001**	56.251	54.305	1.501	1.360
		Velocity	0.499	55.145	55.410	1.420	1.446
	RMS	Force	**< 0.001**	0.108	0.148	0.017	0.023
		Velocity	**< 0.001**	0.115	0.142	0.018	0.022
	AR1	Force	0.989	2.356	2.356	0.009	0.011
		Velocity	**0.033**	2.348	2.363	0.010	0.010
	AR2	Force	0.071	-2.367	-2.334	0.034	0.040
		Velocity	0.079	-2.333	-2.368	0.038	0.037
	AR3	Force	**0.004**	1.259	1.208	0.039	0.044
		Velocity	0.181	1.220	1.247	0.043	0.041
	AR4	Force	**0.001**	-0.319	-0.298	0.014	0.016
		Velocity	0.236	-0.304	-0.313	0.016	0.015
	MNF	Force	**0.002**	96.349	94.376	2.267	2.199
		Velocity	**0.023**	94.582	96.143	2.184	2.292
	MDF	Force	0.231	80.384	79.649	2.171	2.160
		Velocity	**0.002**	78.831	81.203	2.086	2.255
PD (ch11)	MAV	Force	**< 0.001**	0.022	0.033	0.005	0.007
		Velocity	**0.002**	0.024	0.031	0.005	0.007
	SSC	Force	**< 0.001**	56.671	53.613	1.217	1.220
		Velocity	**0.001**	55.673	54.611	1.207	1.229
	WL	Force	**< 0.001**	3.812	5.270	0.673	0.926
		Velocity	**0.001**	4.001	5.080	0.687	0.915
	ZC	Force	**< 0.001**	59.187	56.036	1.667	1.547
		Velocity	0.149	57.888	57.335	1.588	1.630
	RMS	Force	**< 0.001**	0.029	0.043	0.006	0.009
		Velocity	**0.002**	0.031	0.041	0.006	0.009
	AR1	Force	0.696	2.353	2.351	0.006	0.006
		Velocity	0.210	2.347	2.357	0.008	0.006
	AR2	Force	**< 0.001**	-2.420	-2.367	0.031	0.031
		Velocity	0.690	-2.390	-2.397	0.036	0.027
	AR3	Force	**< 0.001**	1.351	1.274	0.039	0.039
		Velocity	0.680	1.316	1.309	0.043	0.036

Table C.2: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 1.

	AR4	Force	**< 0.001**	-0.357	-0.326	0.016	0.016
		Velocity	0.374	-0.344	-0.339	0.017	0.015
	MNF	Force	**< 0.001**	96.359	92.617	2.431	2.140
		Velocity	0.841	94.545	94.431	2.285	2.289
	MDF	Force	**0.017**	76.103	74.581	2.168	1.865
		Velocity	0.159	74.890	75.793	2.002	2.045
ECU (ch12)	MAV	Force	0.311	0.066	0.072	0.011	0.012
		Velocity	0.347	0.068	0.071	0.011	0.010
	SSC	Force	0.758	57.765	57.977	1.504	1.473
		Velocity	0.168	57.425	58.318	1.647	1.299
	WL	Force	**0.026**	15.029	17.641	2.683	3.379
		Velocity	0.091	15.726	16.944	3.118	2.920
	ZC	Force	0.107	69.699	71.593	3.648	3.137
		Velocity	**0.049**	69.477	71.816	3.554	3.243
	RMS	Force	0.298	0.092	0.101	0.016	0.018
		Velocity	0.287	0.095	0.099	0.017	0.016
	AR1	Force	0.495	2.346	2.354	0.031	0.029
		Velocity	0.734	2.348	2.352	0.032	0.028
	AR2	Force	0.451	-2.543	-2.568	0.087	0.081
		Velocity	0.438	-2.541	-2.570	0.093	0.076
	AR3	Force	0.405	1.519	1.542	0.072	0.066
		Velocity	0.376	1.516	1.545	0.077	0.062
	AR4	Force	0.330	-0.447	-0.456	0.023	0.021
		Velocity	0.253	-0.445	-0.457	0.024	0.020
	MNF	Force	0.272	130.247	132.757	7.518	6.612
		Velocity	0.056	129.296	133.708	7.373	6.766
	MDF	Force	0.291	115.687	118.084	7.454	6.735
		Velocity	**0.034**	114.553	119.219	7.364	6.810
ECR (ch13)	MAV	Force	0.106	0.039	0.053	0.009	0.016
		Velocity	0.535	0.046	0.046	0.013	0.012
	SSC	Force	**0.008**	58.342	57.273	2.137	2.121
		Velocity	**0.018**	57.479	58.136	2.158	2.092
	WL	Force	**0.001**	6.764	8.293	1.256	1.439
		Velocity	0.058	7.395	7.662	1.347	1.325

Table C.2: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 1.

	ZC	Force	0.148	67.071	66.389	3.750	3.776
		Velocity	0.462	66.498	66.962	3.804	3.733
	RMS	Force	0.101	0.052	0.070	0.012	0.020
		Velocity	0.433	0.060	0.061	0.017	0.016
	AR1	Force	0.721	2.332	2.328	0.033	0.033
		Velocity	0.823	2.331	2.329	0.033	0.033
	AR2	Force	0.371	-2.460	-2.439	0.067	0.070
		Velocity	0.682	-2.447	-2.452	0.068	0.068
	AR3	Force	0.155	1.421	1.394	0.057	0.058
		Velocity	0.455	1.403	1.412	0.057	0.057
	AR4	Force	0.081	-0.394	-0.385	0.020	0.021
		Velocity	0.360	-0.388	-0.391	0.020	0.020
	MNF	Force	0.251	117.081	116.293	6.807	6.804
		Velocity	0.980	116.670	116.703	6.823	6.837
	MDF	Force	0.438	99.794	99.320	6.567	6.596
		Velocity	0.884	99.678	99.436	6.655	6.597
FCU (ch14)	MAV	Force	**< 0.001**	0.035	0.046	0.005	0.006
		Velocity	0.245	0.040	0.041	0.005	0.006
	SSC	Force	**0.007**	58.723	57.522	0.857	0.694
		Velocity	**0.026**	57.831	58.414	0.788	0.736
	WL	Force	**< 0.001**	7.802	10.175	1.221	1.447
		Velocity	0.147	8.789	9.189	1.301	1.346
	ZC	Force	0.487	67.964	67.537	1.318	1.380
		Velocity	0.060	67.300	68.201	1.278	1.388
	RMS	Force	**< 0.001**	0.047	0.063	0.007	0.008
		Velocity	0.209	0.054	0.056	0.007	0.007
	AR1	Force	0.723	2.409	2.411	0.010	0.009
		Velocity	0.428	2.412	2.408	0.008	0.010
	AR2	Force	0.510	-2.647	-2.638	0.028	0.031
		Velocity	0.959	-2.643	-2.642	0.026	0.032
	AR3	Force	0.090	1.577	1.554	0.028	0.032
		Velocity	0.735	1.563	1.568	0.028	0.032
	AR4	Force	**0.037**	-0.444	-0.432	0.012	0.013
		Velocity	0.645	-0.437	-0.439	0.012	0.013

Table C.2: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 1.

	MNF	Force	0.853	124.184	123.978	3.830	4.115
		Velocity	0.133	123.421	124.742	3.878	4.039
	MDF	Force	0.450	105.169	106.208	4.204	4.468
		Velocity	0.223	105.166	106.211	4.236	4.373
FCR (ch15)	MAV	Force	**< 0.001**	0.029	0.041	0.004	0.005
		Velocity	0.926	0.035	0.035	0.004	0.004
	SSC	Force	**0.001**	56.935	55.193	1.014	0.847
		Velocity	**0.006**	55.689	56.439	0.961	0.868
	WL	Force	**< 0.001**	5.937	8.393	0.702	1.006
		Velocity	0.856	7.192	7.138	0.867	0.826
	ZC	Force	0.062	64.928	63.697	1.102	1.381
		Velocity	**0.028**	63.796	64.829	1.271	1.185
	RMS	Force	**< 0.001**	0.039	0.056	0.005	0.007
		Velocity	0.985	0.047	0.048	0.006	0.006
	AR1	Force	0.591	2.396	2.400	0.011	0.010
		Velocity	0.175	2.402	2.394	0.011	0.009
	AR2	Force	0.601	-2.562	-2.553	0.025	0.031
		Velocity	0.711	-2.560	-2.555	0.029	0.026
	AR3	Force	0.246	1.468	1.445	0.026	0.034
		Velocity	0.835	1.455	1.458	0.030	0.028
	AR4	Force	0.178	-0.397	-0.387	0.012	0.015
		Velocity	0.611	-0.391	-0.393	0.013	0.013
	MNF	Force	0.569	116.357	115.753	3.194	3.753
		Velocity	0.299	115.517	116.593	3.546	3.418
	MDF	Force	0.570	99.077	99.621	3.955	4.293
		Velocity	0.676	99.099	99.599	4.200	4.085

C.3 Consolidated statistical analysis of EMG signals during ADL 2 motions

Table C.3: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 2.

ADL 2				Level Mean		Std Error		Significance
Muscle	Feature	Factor	Significance	L1	L2	L1	L2	SF 1-2
BB_S (ch1)	MAV	Force	**< 0.001**	0.021	0.030	0.003	0.004	**< 0.001**
		Velocity	**0.001**	0.027	0.023	0.004	0.004	**0.001**
	SSC	Force	**< 0.001**	48.926	46.105	1.213	0.928	**< 0.001**
		Velocity	**< 0.001**	46.504	48.527	1.068	1.038	**< 0.001**
	WL	Force	**< 0.001**	3.279	4.749	0.509	0.713	**< 0.001**
		Velocity	**0.002**	4.335	3.693	0.661	0.560	**0.002**
	ZC	Force	**0.039**	52.303	50.500	1.184	1.030	**0.039**
		Velocity	**0.029**	50.593	52.211	1.141	1.031	**0.029**
	RMS	Force	**< 0.001**	0.028	0.041	0.004	0.006	**< 0.001**
		Velocity	**0.001**	0.037	0.032	0.005	0.005	**0.001**
	AR1	Force	**0.037**	2.389	2.365	0.017	0.021	**0.037**
		Velocity	0.139	2.370	2.384	0.019	0.019	
	AR2	Force	**0.002**	-2.390	-2.301	0.052	0.059	**0.002**
		Velocity	**0.011**	-2.313	-2.377	0.056	0.055	**0.011**
	AR3	Force	**< 0.001**	1.218	1.113	0.050	0.055	**< 0.001**
		Velocity	**0.002**	1.128	1.204	0.054	0.051	**0.002**
	AR4	Force	**< 0.001**	-0.288	-0.245	0.018	0.019	**< 0.001**
		Velocity	**0.001**	-0.252	-0.281	0.018	0.018	**0.001**
	MNF	Force	0.337	99.709	98.301	3.081	3.006	
		Velocity	0.136	98.026	99.985	3.182	2.859	
	MDF	Force	0.501	85.644	84.458	3.522	3.400	
		Velocity	0.531	84.613	85.488	3.542	3.297	
BB_L (ch2)	MAV	Force	**< 0.001**	0.027	0.037	0.003	0.004	**< 0.001**
		Velocity	**0.012**	0.033	0.031	0.004	0.003	**0.012**
	SSC	Force	**0.002**	46.164	44.401	0.897	0.749	**0.002**
		Velocity	**0.003**	44.461	46.104	0.852	0.793	**0.003**
	WL	Force	**< 0.001**	4.011	5.510	0.458	0.573	**< 0.001**
		Velocity	**0.001**	4.999	4.521	0.545	0.477	**0.001**
	ZC	Force	0.171	49.561	48.706	1.239	1.278	
		Velocity	**0.030**	48.426	49.840	1.305	1.212	**0.030**
	RMS	Force	**< 0.001**	0.036	0.050	0.004	0.005	**< 0.001**
		Velocity	**0.024**	0.045	0.042	0.005	0.005	**0.024**

Table C.3: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 2.

	AR1	Force	**< 0.001**	2.338	2.304	0.020	0.021	**< 0.001**
		Velocity	0.067	2.314	2.329	0.021	0.020	
	AR2	Force	**< 0.001**	-2.238	-2.133	0.055	0.056	**< 0.001**
		Velocity	**0.020**	-2.158	-2.212	0.058	0.054	**0.020**
	AR3	Force	**< 0.001**	1.080	0.975	0.049	0.047	**< 0.001**
		Velocity	**0.012**	0.999	1.056	0.050	0.046	**0.012**
	AR4	Force	**< 0.001**	-0.246	-0.209	0.015	0.014	**< 0.001**
		Velocity	**0.011**	-0.218	-0.238	0.015	0.014	**0.011**
	MNF	Force	**0.022**	93.279	91.268	3.221	3.273	**0.022**
		Velocity	0.312	91.753	92.794	3.500	3.001	
	MDF	Force	**0.032**	79.016	77.379	3.585	3.535	**0.032**
		Velocity	0.797	78.050	78.345	3.814	3.345	
BRA (ch3)	MAV	Force	**< 0.001**	0.036	0.048	0.009	0.011	**< 0.001**
		Velocity	0.411	0.042	0.041	0.010	0.010	
	SSC	Force	**< 0.001**	47.302	44.703	1.271	1.103	**< 0.001**
		Velocity	0.068	45.359	46.646	1.057	1.334	
	WL	Force	**< 0.001**	3.794	5.027	0.569	0.680	**< 0.001**
		Velocity	**0.006**	4.547	4.274	0.603	0.636	**0.006**
	ZC	Force	0.151	43.779	42.703	1.426	1.654	
		Velocity	**0.040**	42.429	44.054	1.387	1.691	**0.040**
	RMS	Force	**< 0.001**	0.046	0.062	0.011	0.014	**< 0.001**
		Velocity	0.289	0.055	0.053	0.013	0.013	
	AR1	Force	0.212	2.237	2.228	0.025	0.024	
		Velocity	0.314	2.238	2.227	0.022	0.027	
	AR2	Force	**0.007**	-2.011	-1.952	0.054	0.053	**0.007**
		Velocity	0.984	-1.982	-1.981	0.048	0.060	
	AR3	Force	**0.001**	0.939	0.865	0.039	0.038	**0.001**
		Velocity	0.425	0.891	0.913	0.035	0.044	
	AR4	Force	**0.003**	-0.216	-0.190	0.011	0.012	**0.003**
		Velocity	0.118	-0.196	-0.209	0.010	0.013	
	MNF	Force	0.418	75.405	74.619	2.739	3.019	
		Velocity	0.170	74.284	75.740	2.763	3.010	
	MDF	Force	0.599	59.766	60.264	2.953	2.963	
		Velocity	0.316	59.532	60.498	2.929	2.989	

Table C.3: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 2.

BRD (ch4)	MAV	Force	0.302	0.029	0.034	0.008	0.008	
		Velocity	0.228	0.030	0.033	0.007	0.008	
	SSC	Force	**< 0.001**	56.755	53.715	1.375	1.525	**< 0.001**
		Velocity	0.271	54.889	55.582	1.456	1.437	
	WL	Force	0.249	3.646	4.083	0.915	0.692	
		Velocity	0.834	3.846	3.883	0.740	0.845	
	ZC	Force	**0.005**	55.887	52.594	2.047	2.123	**0.005**
		Velocity	0.723	54.487	53.995	2.078	2.180	
	RMS	Force	0.252	0.039	0.045	0.011	0.010	
		Velocity	0.187	0.040	0.044	0.009	0.011	
	AR1	Force	0.398	2.305	2.296	0.027	0.030	
		Velocity	0.221	2.306	2.294	0.030	0.027	
	AR2	Force	**0.046**	-2.334	-2.270	0.063	0.070	**0.046**
		Velocity	0.329	-2.316	-2.289	0.069	0.063	
	AR3	Force	**0.008**	1.303	1.222	0.050	0.056	**0.008**
		Velocity	0.362	1.275	1.251	0.055	0.051	
	AR4	Force	**0.003**	-0.346	-0.315	0.016	0.018	**0.003**
		Velocity	0.248	-0.336	-0.325	0.017	0.017	
	MNF	Force	0.068	94.600	90.993	3.858	4.544	
		Velocity	0.287	93.992	91.601	4.136	4.365	
	MDF	Force	0.172	73.791	71.741	4.122	4.586	
		Velocity	0.210	73.935	71.597	4.204	4.575	
TRI_LO (ch5)	MAV	Force	0.835	0.006	0.006	0.001	0.001	
		Velocity	0.345	0.006	0.006	0.001	0.001	
	SSC	Force	**< 0.001**	66.071	64.377	1.155	1.257	**< 0.001**
		Velocity	0.628	65.314	65.135	1.250	1.159	
	WL	Force	0.191	1.251	1.192	0.201	0.197	
		Velocity	0.151	1.173	1.271	0.194	0.206	
	ZC	Force	**< 0.001**	68.470	64.282	1.558	1.435	**< 0.001**
		Velocity	0.544	66.677	66.076	1.558	1.562	
	RMS	Force	0.978	0.008	0.008	0.002	0.001	
		Velocity	0.330	0.008	0.008	0.002	0.001	
	AR1	Force	**0.019**	2.320	2.307	0.020	0.019	**0.019**
		Velocity	0.727	2.313	2.314	0.019	0.020	

Table C.3: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 2.

	AR2	Force	**< 0.001**	-2.510	-2.444	0.036	0.034	**< 0.001**
		Velocity	0.813	-2.475	-2.479	0.033	0.038	
	AR3	Force	**< 0.001**	1.539	1.467	0.026	0.026	**< 0.001**
		Velocity	0.963	1.503	1.503	0.024	0.029	
	AR4	Force	**< 0.001**	-0.443	-0.413	0.011	0.010	**< 0.001**
		Velocity	0.966	-0.428	-0.428	0.010	0.012	
	MNF	Force	**< 0.001**	111.078	102.805	3.421	3.087	**< 0.001**
		Velocity	0.790	106.689	107.193	3.143	3.549	
	MDF	Force	**< 0.001**	83.152	75.412	3.719	3.307	**< 0.001**
		Velocity	0.447	78.527	80.036	3.287	3.864	
TRI_LAT (ch6)	MAV	Force	0.348	0.007	0.007	0.001	0.001	
		Velocity	0.343	0.007	0.007	0.001	0.001	
	SSC	Force	**0.003**	60.977	59.172	0.718	0.752	**0.003**
		Velocity	0.827	60.041	60.108	0.717	0.686	
	WL	Force	0.663	1.392	1.359	0.208	0.218	
		Velocity	0.256	1.332	1.420	0.215	0.212	
	ZC	Force	**< 0.001**	64.370	60.866	1.106	1.171	**< 0.001**
		Velocity	0.256	62.185	63.051	1.045	1.274	
	RMS	Force	0.552	0.009	0.009	0.001	0.001	
		Velocity	0.274	0.009	0.009	0.001	0.001	
	AR1	Force	0.076	2.368	2.358	0.009	0.009	
		Velocity	0.639	2.361	2.365	0.007	0.011	
	AR2	Force	**< 0.001**	-2.542	-2.479	0.020	0.024	**< 0.001**
		Velocity	0.309	-2.500	-2.521	0.019	0.027	
	AR3	Force	**< 0.001**	1.509	1.432	0.020	0.024	**< 0.001**
		Velocity	0.238	1.459	1.481	0.019	0.026	
	AR4	Force	**< 0.001**	-0.425	-0.392	0.009	0.009	**< 0.001**
		Velocity	0.169	-0.403	-0.413	0.008	0.010	
	MNF	Force	**< 0.001**	108.946	101.898	2.193	2.218	**< 0.001**
		Velocity	0.067	103.847	106.997	1.917	2.664	
	MDF	Force	**< 0.001**	86.848	80.407	2.506	2.209	**< 0.001**
		Velocity	0.051	81.726	85.529	2.042	2.837	
TRI_M (ch7)	MAV	Force	0.846	0.008	0.008	0.001	0.001	
		Velocity	0.050	0.007	0.009	0.001	0.001	

Table C.3: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 2.

	SSC	Force	**0.015**	61.476	59.881	1.056	1.304	**0.015**
		Velocity	0.101	59.869	61.488	1.001	1.442	
	WL	Force	0.051	1.674	1.424	0.284	0.201	
		Velocity	**0.037**	1.400	1.697	0.245	0.251	**0.037**
	ZC	Force	**< 0.001**	62.664	57.202	2.003	2.186	**< 0.001**
		Velocity	**0.006**	58.626	61.239	1.872	2.218	**0.006**
	RMS	Force	0.640	0.011	0.011	0.002	0.001	
		Velocity	**0.039**	0.010	0.012	0.002	0.002	**0.039**
	AR1	Force	**0.003**	2.348	2.306	0.016	0.022	**0.003**
		Velocity	0.221	2.334	2.321	0.017	0.021	
	AR2	Force	**< 0.001**	-2.507	-2.376	0.042	0.052	**< 0.001**
		Velocity	0.937	-2.441	-2.443	0.044	0.049	
	AR3	Force	**< 0.001**	1.488	1.373	0.037	0.042	**< 0.001**
		Velocity	0.350	1.423	1.438	0.037	0.041	
	AR4	Force	**< 0.001**	-0.416	-0.378	0.014	0.014	**< 0.001**
		Velocity	0.230	-0.393	-0.402	0.013	0.016	
	MNF	Force	**< 0.001**	106.852	95.817	4.228	4.324	**< 0.001**
		Velocity	**0.003**	98.689	103.980	3.765	4.622	**0.003**
	MDF	Force	**< 0.001**	84.040	73.581	4.556	4.143	**< 0.001**
		Velocity	**0.029**	76.434	81.187	3.815	4.816	**0.029**
ISPI (ch8)	MAV	Force	**< 0.001**	0.035	0.047	0.004	0.005	**< 0.001**
		Velocity	**0.044**	0.043	0.039	0.005	0.004	**0.044**
	SSC	Force	**< 0.001**	51.880	50.626	1.253	1.255	**< 0.001**
		Velocity	**0.032**	50.800	51.706	1.237	1.283	**0.032**
	WL	Force	**< 0.001**	6.343	8.501	0.726	0.902	**< 0.001**
		Velocity	0.067	7.708	7.137	0.902	0.723	
	ZC	Force	0.997	58.084	58.086	1.489	1.468	
		Velocity	0.060	57.596	58.573	1.474	1.492	
	RMS	Force	**< 0.001**	0.046	0.061	0.005	0.007	**< 0.001**
		Velocity	0.051	0.056	0.051	0.007	0.005	
	AR1	Force	0.994	2.364	2.364	0.011	0.015	
		Velocity	0.402	2.360	2.367	0.013	0.013	
	AR2	Force	0.251	-2.385	-2.365	0.045	0.052	
		Velocity	0.261	-2.362	-2.388	0.049	0.049	

Table C.3: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 2.

	AR3	Force	0.054	1.262	1.229	0.050	0.057	
		Velocity	0.243	1.232	1.259	0.054	0.054	
	AR4	Force	**0.030**	-0.316	-0.303	0.019	0.021	**0.030**
		Velocity	0.279	-0.305	-0.314	0.020	0.020	
	MNF	Force	0.649	100.271	100.611	2.872	2.728	
		Velocity	0.180	99.938	100.943	2.847	2.753	
	MDF	Force	0.097	84.800	86.182	2.839	2.637	
		Velocity	0.396	85.212	85.770	2.730	2.729	
AD (ch9)	MAV	Force	**< 0.001**	0.035	0.043	0.003	0.005	**< 0.001**
		Velocity	0.159	0.040	0.038	0.004	0.004	
	SSC	Force	0.081	49.386	48.785	0.774	0.825	
		Velocity	**0.037**	48.694	49.476	0.789	0.815	**0.037**
	WL	Force	**< 0.001**	6.448	7.942	0.657	0.866	**< 0.001**
		Velocity	0.125	7.405	6.984	0.763	0.769	
	ZC	Force	0.860	55.775	55.879	1.288	1.531	
		Velocity	0.858	55.774	55.880	1.337	1.488	
	RMS	Force	**< 0.001**	0.047	0.058	0.005	0.006	**< 0.001**
		Velocity	0.180	0.053	0.051	0.005	0.005	
	AR1	Force	**0.003**	2.395	2.373	0.013	0.015	**0.003**
		Velocity	0.236	2.389	2.378	0.014	0.014	
	AR2	Force	**0.002**	-2.418	-2.357	0.037	0.044	**0.002**
		Velocity	0.490	-2.395	-2.380	0.040	0.043	
	AR3	Force	**0.001**	1.244	1.186	0.035	0.041	**0.001**
		Velocity	0.984	1.215	1.215	0.036	0.040	
	AR4	Force	**0.002**	-0.294	-0.274	0.012	0.014	**0.002**
		Velocity	0.639	-0.282	-0.286	0.012	0.014	
	MNF	Force	0.477	102.663	101.856	3.259	3.604	
		Velocity	0.376	102.796	101.723	3.408	3.475	
	MDF	Force	0.663	88.903	88.464	3.725	3.970	
		Velocity	0.193	89.621	87.746	3.861	3.901	
LD (ch10)	MAV	Force	**< 0.001**	0.030	0.038	0.004	0.005	**< 0.001**
		Velocity	0.597	0.035	0.034	0.004	0.004	
	SSC	Force	**< 0.001**	55.584	53.681	1.381	1.339	**< 0.001**
		Velocity	0.794	54.558	54.707	1.362	1.381	

Table C.3: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 2.

	WL	Force	**< 0.001**	5.337	6.648	0.670	0.821	**< 0.001**
		Velocity	0.672	6.055	5.930	0.760	0.743	
	ZC	Force	**0.022**	57.345	56.053	1.560	1.502	**0.022**
		Velocity	0.251	56.410	56.987	1.456	1.597	
	RMS	Force	**< 0.001**	0.040	0.050	0.005	0.006	**< 0.001**
		Velocity	0.600	0.046	0.045	0.006	0.005	
	AR1	Force	**0.025**	2.357	2.345	0.010	0.009	**0.025**
		Velocity	0.811	2.350	2.352	0.010	0.009	
	AR2	Force	**< 0.001**	-2.409	-2.353	0.035	0.034	**< 0.001**
		Velocity	0.713	-2.378	-2.383	0.032	0.038	
	AR3	Force	**< 0.001**	1.330	1.261	0.042	0.042	**< 0.001**
		Velocity	0.708	1.292	1.299	0.038	0.046	
	AR4	Force	**< 0.001**	-0.349	-0.322	0.016	0.016	**< 0.001**
		Velocity	0.622	-0.334	-0.338	0.015	0.017	
	MNF	Force	0.141	94.660	93.690	2.245	2.185	
		Velocity	0.068	93.648	94.701	2.164	2.254	
	MDF	Force	0.499	75.690	76.033	2.160	2.075	
		Velocity	0.135	75.382	76.341	2.137	2.114	
PD (ch11)	MAV	Force	**0.006**	0.013	0.016	0.002	0.003	**0.006**
		Velocity	0.732	0.014	0.015	0.002	0.003	
	SSC	Force	**< 0.001**	59.715	56.848	1.203	1.220	**< 0.001**
		Velocity	0.463	58.097	58.467	1.220	1.233	
	WL	Force	**0.005**	2.220	2.655	0.346	0.426	**0.005**
		Velocity	0.768	2.416	2.459	0.389	0.388	
	ZC	Force	**< 0.001**	60.776	57.880	1.847	1.616	**< 0.001**
		Velocity	0.076	58.829	59.827	1.701	1.758	
	RMS	Force	**0.009**	0.017	0.021	0.003	0.003	**0.009**
		Velocity	0.685	0.018	0.019	0.003	0.003	
	AR1	Force	0.681	2.333	2.330	0.007	0.009	
		Velocity	0.810	2.331	2.333	0.008	0.007	
	AR2	Force	**0.007**	-2.418	-2.367	0.031	0.038	**0.007**
		Velocity	0.464	-2.387	-2.398	0.035	0.033	
	AR3	Force	**< 0.001**	1.389	1.317	0.038	0.044	**< 0.001**
		Velocity	0.441	1.346	1.359	0.042	0.040	

Table C.3: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 2.

	AR4	Force	**< 0.001**	-0.378	-0.350	0.015	0.016	**< 0.001**
		Velocity	0.438	-0.362	-0.367	0.016	0.015	
	MNF	Force	**0.002**	96.639	93.162	2.716	2.348	**0.002**
		Velocity	**0.039**	94.130	95.671	2.461	2.565	**0.039**
	MDF	Force	0.383	73.184	72.315	2.211	2.093	
		Velocity	**0.029**	72.123	73.376	2.000	2.221	**0.029**
ECU (ch12)	MAV	Force	0.391	0.052	0.045	0.011	0.005	
		Velocity	**0.037**	0.056	0.040	0.011	0.005	**0.037**
	SSC	Force	0.614	58.575	58.993	1.887	1.257	
		Velocity	**0.028**	58.097	59.471	1.725	1.415	**0.028**
	WL	Force	0.403	12.445	10.582	2.942	1.419	
		Velocity	0.077	13.470	9.557	2.978	1.279	
	ZC	Force	0.913	70.720	70.551	3.156	2.540	
		Velocity	0.327	69.878	71.392	3.289	2.360	
	RMS	Force	0.373	0.073	0.062	0.016	0.008	
		Velocity	**0.047**	0.078	0.057	0.016	0.007	**0.047**
	AR1	Force	0.178	2.340	2.361	0.031	0.022	
		Velocity	0.363	2.345	2.356	0.030	0.022	
	AR2	Force	0.329	-2.532	-2.573	0.087	0.058	
		Velocity	0.304	-2.535	-2.569	0.084	0.060	
	AR3	Force	0.437	1.513	1.538	0.071	0.048	
		Velocity	0.257	1.510	1.542	0.068	0.050	
	AR4	Force	0.825	-0.443	-0.445	0.021	0.017	
		Velocity	0.366	-0.440	-0.449	0.021	0.017	
	MNF	Force	0.596	129.368	127.857	6.298	5.372	
		Velocity	0.828	128.310	128.915	6.665	4.891	
	MDF	Force	0.490	112.975	111.181	6.421	5.745	
		Velocity	0.635	112.701	111.454	6.827	5.264	
ECR (ch13)	MAV	Force	**0.003**	0.032	0.038	0.006	0.007	**0.003**
		Velocity	0.413	0.037	0.033	0.008	0.006	
	SSC	Force	0.099	59.073	58.475	2.076	2.129	
		Velocity	**< 0.001**	58.098	59.451	2.170	2.032	**< 0.001**
	WL	Force	**0.001**	6.097	7.654	1.085	1.391	**0.001**
		Velocity	0.359	7.056	6.695	1.283	1.203	

Table C.3: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 2.

	ZC	Force	0.819	68.478	68.255	3.857	4.056	
		Velocity	**0.008**	67.699	69.034	4.030	3.838	**0.008**
	RMS	Force	**0.002**	0.042	0.051	0.008	0.009	**0.002**
		Velocity	0.427	0.049	0.045	0.010	0.008	
	AR1	Force	0.459	2.334	2.340	0.031	0.034	
		Velocity	0.801	2.336	2.338	0.034	0.032	
	AR2	Force	0.729	-2.477	-2.483	0.059	0.063	
		Velocity	0.309	-2.469	-2.491	0.064	0.058	
	AR3	Force	0.787	1.437	1.432	0.050	0.053	
		Velocity	0.154	1.421	1.449	0.054	0.049	
	AR4	Force	0.539	-0.402	-0.398	0.019	0.021	
		Velocity	0.105	-0.395	-0.406	0.020	0.019	
	MNF	Force	0.956	120.676	120.798	7.411	8.150	
		Velocity	0.059	119.704	121.770	7.837	7.620	
	MDF	Force	0.983	103.884	103.925	7.327	7.902	
		Velocity	0.460	103.405	104.404	7.762	7.412	
FCU (ch14)	MAV	Force	**0.001**	0.029	0.033	0.004	0.005	**0.001**
		Velocity	**0.030**	0.033	0.030	0.005	0.005	**0.030**
	SSC	Force	**0.004**	59.417	58.412	0.788	0.765	**0.004**
		Velocity	**< 0.001**	58.311	59.518	0.743	0.800	**< 0.001**
	WL	Force	**0.003**	6.360	7.271	0.985	1.099	**0.003**
		Velocity	**0.049**	7.161	6.470	1.069	1.025	**0.049**
	ZC	Force	0.074	68.038	67.169	1.229	1.168	
		Velocity	0.168	67.215	67.992	1.211	1.204	
	RMS	Force	**0.001**	0.040	0.046	0.006	0.007	**0.001**
		Velocity	0.061	0.045	0.041	0.006	0.006	
	AR1	Force	0.173	2.413	2.407	0.009	0.010	
		Velocity	0.351	2.413	2.407	0.009	0.011	
	AR2	Force	**0.039**	-2.651	-2.627	0.028	0.029	**0.039**
		Velocity	0.788	-2.641	-2.637	0.029	0.029	
	AR3	Force	**0.009**	1.578	1.549	0.029	0.029	**0.009**
		Velocity	0.777	1.561	1.566	0.030	0.028	
	AR4	Force	**0.012**	-0.442	-0.431	0.012	0.012	**0.012**
		Velocity	0.679	-0.435	-0.437	0.013	0.012	

Table C.3: Significance, mean, and std error of 11 EMG feature values for 15 muscles for 2 levels of force and velocity during ADL 2.

	MNF	Force	0.843	121.839	121.660	3.740	3.723	
		Velocity	0.765	121.636	121.863	3.767	3.679	
	MDF	Force	0.619	102.583	103.121	4.139	4.112	
		Velocity	0.528	103.152	102.552	4.193	4.041	
FCR (ch15)	MAV	Force	**0.003**	0.025	0.032	0.003	0.005	**0.003**
		Velocity	0.249	0.029	0.027	0.004	0.004	
	SSC	Force	**0.007**	57.631	56.336	1.051	0.861	**0.007**
		Velocity	0.057	56.577	57.390	0.884	1.025	
	WL	Force	**0.004**	5.157	6.380	0.604	0.861	**0.004**
		Velocity	0.124	6.014	5.523	0.777	0.690	
	ZC	Force	0.054	65.116	63.678	1.273	1.258	
		Velocity	0.814	64.324	64.471	1.185	1.318	
	RMS	Force	**0.003**	0.034	0.043	0.004	0.006	**0.003**
		Velocity	0.315	0.040	0.038	0.005	0.005	
	AR1	Force	0.942	2.397	2.397	0.010	0.010	
		Velocity	0.599	2.399	2.396	0.010	0.010	
	AR2	Force	0.280	-2.570	-2.551	0.027	0.029	
		Velocity	0.882	-2.562	-2.559	0.029	0.027	
	AR3	Force	0.120	1.481	1.451	0.029	0.029	
		Velocity	0.783	1.464	1.469	0.029	0.028	
	AR4	Force	0.083	-0.403	-0.390	0.012	0.012	
		Velocity	0.581	-0.394	-0.398	0.012	0.012	
	MNF	Force	0.055	115.917	113.658	3.471	3.818	
		Velocity	0.459	115.209	114.366	3.749	3.546	
	MDF	Force	0.090	98.154	96.683	4.277	4.610	
		Velocity	0.332	97.995	96.841	4.566	4.361	

VITA

Name: Taylor Stanbury

Post-secondary Education and Degrees:
The University of Western Ontario
London, Ontario, Canada
2011–2016 B.E.Sc.,
Mechatronic Systems Engineering

The University of Western Ontario
London, Ontario, Canada
2016–2018 M.E.Sc.,
Biomedical Engineering

Honours and Awards:
NSERC Canada Graduate Scholarships-Master's (CGS M)
2016–2017

Related Work Experience:
Teaching Assistant
MSE 2202 – Introduction to Mechatronic Design
MSE 3302 – Sensors and Actuators
ECE 9053 – Robotic Manipulators
The University of Western Ontario
2016–2018

Research Assistant
The University of Western Ontario
2016–2018

Canadian Bone and Joint Conference (Poster Presentation)
London, Ontario
2018

Biomedical Engineering Department Seminar (Oral Presentation)
The University of Western Ontario
2018

Biomedical Engineering Department Seminar (Oral Presentation)
The University of Western Ontario
2017