Machine Learning Based Approach Using Electromyography to Predict Joint Angles of the Knee

by

Jordan C Coker

A thesis submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Auburn, Alabama
August 8, 2020

Keywords: electromyography, intent prediction, exoskeleton, machine learning

Approved by

Michael Zabala, Chair, Assistant Professor of Mechanical Engineering
Mark Schall, Associate Professor of Industrial and Systems Engineering
Howard Chen, Assistant Research Professor of Mechanical Engineering

Abstract

In order for an active exoskeleton and its user to achieve synchronous motion, the intended motion of the user needs to be detected with enough lead-time to process and move the exoskeleton accordingly. This must also happen with a level of accuracy such that the exoskeleton does not impedes the motion of the user. Synchronous motion is difficult to achieve because human musculoskeletal motion is extremely complex with multiple muscles controlling multiple degrees of freedom of the joints. One promising method of reading human motion intent is with the detectible electrical signal that results from muscle activation measured via electromyography. This signal can be measured noninvasively on the surface of the skin, and is detectable approximately 100 ms before movement ensues. For the work presented in this thesis, a control scheme that associates muscle activation to future knee flexion was developed using artificial neural network machine learning algorithms. Artificial neural networks are designed to function much like the human brain. Inversely to how the brain decides on the movement that the body will take and then tells the muscles to activate accordingly, the algorithms will read the muscle activation signal and make informed estimations of the joint angles that the brain is trying to achieve. This method was used to create a model for anticipating error versus prediction time. Furthermore, the method was used to assess the necessary inputs for an algorithm to make accurate knee flexion predictions on a user independent from the algorithm's training data.

Acknowledgments

I would like to express my sincere gratitude to my supervisor Dr. Michael Zabala for the constant support, mentorship, and engagement through this master's thesis. I would also like to thank my committee, Dr. Mark Schall and Dr. Howard Chen for their support and assistance through the various projects that I worked on during my master's program. I would like to thank Taylor Oldfather, Scott Kennedy, and Raju Gupta for their guidance. Finally, I would like to thank all members of the Auburn University Biomechanical Engineering Lab for their support.

Many people helped me along my journey through the master's program outside of the academic world. My parents are absolutely incredible and no amount of thanks will ever compare to what they deserve. My brothers were always there, or at most, a phone call away. Finally, I could not have done it without my friends, both old and new. Cinda and Cody believed in me and convinced me to pursue my passion, which ultimately led me to this point. Mohamed took a chance to work with me, which turned into a blossoming friendship. Pablo, David, Emily, Aura, Kevin, Chad, Sierra, Jeff, Aaron, Jake, Chris, Jess, Scott, Kaitlin, and the countless more who provided support along every step of the way, I cannot give enough thanks.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Since the beginning of civilization, humankind has sought to increase physical performance. Impaired individuals have adopted technology such as glasses and hearing aids to restore full function. Healthy individuals lift weights and take supplements to push human limits and compete for notoriety. Recently a push has been made for exoskeletons to increase human potential beyond the natural limitations of the human body [1]. Exoskeletons can be used for rehabilitation after an injury, allowing the user to regain strength and mobility. They can also be used for enhancement, allowing workers to lift and hold heavy objects that they otherwise could not [2].

Naturalistic and synchronized movement of an exoskeleton with its operator is crucial for widespread adoption of the technology since lagging or cumbersome designs could increase the metabolic costs and reduce the potential benefits to the user [3]. Furthermore, the powerful actuation mechanisms could cause injury to the user's joints if desynchronization occurs.

Various designs and control methods have been proposed for exoskeletons to achieve naturalistic and synchronized movement. Passive exoskeleton designs can involve a spring at the ankle that stores energy on impact and releasing the energy to propel the user during walking or

running gait [4]. More complex powered exoskeleton designs involve sensors, control boards, and actuators to read the intentions and then mimic or assist the joint at which they are mounted [5]. Providing a naturalistic and synchronized movement for a powered exoskeleton based on the intention of the user is an active area of research. The use of machine learning to assess current joint angles and associated muscle activation to predict future joint angles is a potential solution for providing naturalistic and synchronized movement for a powered exoskeleton. By making predictions rather than estimations as seen in previous designs, the system will be able to counter any delays caused by data transmission, computation, and actuation of the exoskeleton.

The overall objective of this thesis is to increase the knowledge base of machine learning algorithms as a means to control exoskeletons. This will be accomplished by providing an assessment for various prediction times compared to the accuracy of said prediction. It will be important to understand the tradeoffs between time saved and accuracy lost. Secondly, assessing the necessary inputs for a machine learning algorithm to make accurate knee flexion predictions for a population independent of the algorithm's training population. The ability to predict joint angles for an independent population will negate the need for individually tuning an exoskeleton to each user. This could lead to a wider adaptation of exoskeleton technology due to a removal of costly and time prohibitive barriers.

Chapter 2

Background and Lit Review

2.1 Exoskeleton

Over the past several decades, wearable robotics have developed at a rapid rate due to the size reduction of electronics, increases in computing power, and improved battery technology [6]. Exoskeleton technology, in particular, continues to demonstrate exponential improvements for the user [2]. While technological advances in exoskeletons have been made, further improvements are required before a truly seamless connection between the user and the wearable device can occur.

Exoskeletons are, by definition, any structure that provides rigidity and/or protection outside of a living organism. In the field of biomechanics, the term exoskeleton normally refers to a device outside of the human body that moves with the user and often assists the user in movement of either their internal body structure or an external load [7]. Moving the internal body structure is largely important for restoring functionality to the user who may have a neurological disorder or sustained a musculoskeletal injury. Most of the time, these exoskeletons are used for rehabilitative purposes but can also be assistive and used in place of mobility devices such as wheelchairs. A rehabilitative exoskeleton can reduce the required load for the muscles to move one's own body to a point where muscle strengthening can occur through assisted gait. As the muscles strengthen

over time, the reliance on the exoskeleton can be reduced to a point where it is no longer required. For the case of neurological disorders and severe musculoskeletal injuries, the assistive exoskeleton becomes an integral part of the user's ability to function and move without the reliance on assistance from other human such as nurses. Examples of assistive exoskeletons include the HAL, ReWalk, and Ekso [8] [9] [10].

An exoskeleton can also augments a user's strength, which will allow them to move an external load with less strain on the musculoskeletal system. This is currently seen in many industrial settings where lifting heavy objects for extended periods of time above the worker's head is required. Many automotive manufacturers in the United States are implementing exoskeleton technologies to reduce the wear and tear on their workers' shoulders and backs. Examples of strength augmentation exoskeletons include BLEEX, XOS, and HULC [11] [12] [13].

2.2 Exoskeleton Designs

Arguably, the most important aspect of an exoskeleton is its means of simulating the human body's mode of movement. The human body uses muscles that span over joints and shorten to produce a force on the bone. Muscles provide torque about joints which allows for the body to control a sum total of 244 degrees of freedom (DOF) [14]. Exoskeleton technology currently lacks the precision of control for sufficient synergy between user and exoskeleton resulting in limited cooperation between the two. Consequently, designs are often made in a way that attempt to minimize complexity without sacrificing naturalistic motion [15]. The most common ways of simplification are to remove DOFs, under-actuate the system, or only actuate through portions of the overall movement. All exoskeleton designs seek to find a balance between necessary complexities associated with naturalistic motion and the ability to match the speed and force output

of human motion. This can be accomplished with both passive and active exoskeleton designs [1].

2.2.1 Passive Exoskeletons

Passive exoskeletons utilize mechanical manipulations of force transfer in order to lessen the work to be accomplished by the user. Many unpowered exoskeletons, such as the Lockheed FORTIS, are utilized in industry to transmit forces around the user to reduce loads for physiologically taxing tasks [7]. Other designs are used to improve human efficiencies by recycling energy lost through motions such as gait. An example of a lower body unpowered ankle exoskeleton is one that Collins et al. built to reduce the metabolic rate of the user while walking on a treadmill [4]. The design utilizes a spring and clutch to store and return energy during the gait cycle. It was shown to reduce metabolic rates by an average of 7.2% over their nine subjects when compared to walking without an exoskeleton. A major limitation of the passive exoskeleton is that mechanical work cannot be created and, therefore, the user is responsible for creating the necessary work to maintain motion. Another major limitation is the lack of variability in the design. Collins et al. looked to vary spring stiffness in order to maximize metabolic efficiency in normal walking. However, the optimal spring stiffness could vary drastically for walking with increased loads or running and, with this design, there is no way to make the changes instantaneously. There would either need to be specific mechanisms to be changed when starting a different task, or to implement a generalized design that can handle every task without optimization for any task individually.

2.2.2 Active Exoskeletons

Active exoskeletons use an input of mechanical energy to actuate the system, requiring

power sources. As such, they are much more complex than their unpowered counterparts. An active exoskeleton consists of three main components: actuators, sensors, and controllers. With data transmission and processing times, there is a time gap between detecting human intent (e.g. via EMG) and when the exoskeleton movement occurs. It is not entirely clear how fast a system must operate for lag to be completely unnoticeable to the user, but currently, the clinically accepted response time delay to the user's movement is 300 ms [16]. Furthermore, work by Petrella et al. (1997) shows that through proprioception, healthy young adults are able to distinguish and reproduce knee flexion angle positions to within two degrees of accuracy [17]. Although the need for empirical validation remains, it can be inferred that from the study by Petrella et al. any movement between the user and the exoskeleton should result in a discrepancy of less than two degrees to ensure seamless interaction between the two.

2.2.2.1 Actuators

Active exoskeletons can rely on gears and cable drives, electric motors, linear actuators, pneumatics, or hydraulics for force transmission. Designs often face the challenge of generating sufficient force at rates similar to muscular movement without becoming awkward or cumbersome to the user. Currently, electric motors are used to actuate approximately 72% of exoskeletons designs [7]. Electric motors provide a mature technology that is easily controlled; however, there are discrepancies in the movement of human motions and the optimal operation of an electric motor. Electric motors often are measured in revolutions per minute (rpm) while no joint in the human body is designed to make an entire rotation. Some options such as gear reduction techniques can increase the output torque at the sacrifice of travel velocity in an attempt to better simulate the human body. Another problem with electric motors is that the size and weight of the motor is

correlated with the power output. For a active exoskeleton to lift heavy objects, bulky motors are required to be designed into the system. For this reason, many augmentation exoskeletons rely on pneumatics or hydraulics for actuation [3]. Both are subject to the problem of leakage, although, pneumatics can have an onboard compressor to regain any lost air. Another problem with pneumatics is that since air is a compressible fluid, fine control of heavy loads are difficult to manage. Much work is being done on the development of soft actuators such as shape memory allows (SMA) [18]. Soft actuators will actuate in a manner similar to human muscles, theoretically allowing exoskeletons to increase DOF while maintaining a light profile. SMAs currently hold a number of limitations that deter from wide scale adoption. They have an inverse relationship of force output to actuation velocity and can only shorten on the scale of 4-8% [19]. Further advancements are required before SMAs are a viable option on exoskeleton designs.

2.2.2.2 Sensors

In order for an exoskeleton to move synchronously with the user, it must fully understand the movement intentions of the user. Exoskeletons can utilize single signals or implement sensor fusion algorithms that allow for utilization of inputs from multiple sensors in order to achieve intention inferences. Sensors are able to pick up on both non-biological and biological signals that are indicative of human motion. Common non-biological signals include eye tracking, speech, kinematics (positions, orientations, velocities, accelerations), and kinetics (force outputs of a segment). Kinematic and kinetic parameters can be provided using inertial measurement units (IMUs), encoders, and force sensors. These sensors are often used in commercial exoskeletons due to the cost, size, simplicity, and reliability. Many augmentation exoskeletons such as BLEEX or the Nurse Robotic Suit rely on strictly kinematic and kinetic sensors [20] [21] [22] [23]. The

biggest drawback to non-biological sensors is that they measure motion that has already occurred, which eliminates lead time for an exoskeleton to move synchronously with a user. Exoskeletons that rely heavily on the use of non-biological sensors, particularly force sensors, often require the user to come into contact with the exoskeleton in order for movement to occur. This reactive response is less than ideal as it can result in delays that cause inefficiencies of use as well as potential injury from contact stress.

In contrast, biological signals are records that can be tracked during a biological event. Signals that are used for exoskeleton intention inference are magnetoencephalogram (MEG) which record magnetic fields around the head to track brain activity, mechanomyogram (MMG) which record mechanical vibrations of a muscle to detect contraction, sonomyogram (SMG) which produces a sonogram image of a muscle using ultrasound echo, electrocorticogram (ECoG) and electroencephalogram (EEG) which are ways of monitoring electrical activities of the brain, and electromyogram (EMG) which reads electrical activity of a muscle [24]. Surface EMG is the most commonly used biological signal because it is relatively inexpensive, noninvasive, read in close proximity to the exoskeleton, and generates a well understood signal.

2.2.2.2.1 EMG

EMG measures the electrical action potential in a muscle that occurs when a neurological signal is sent from the brain for the purpose of activation. This signal can be measured up to 100ms before activation of the skeletal muscle ensues [25]. Surface EMG is a method of placing electrodes on the skin above the belly of the muscle to record its activation. The measurement waveform from surface EMG sensors is considered irregular due to crosstalk from multiple muscle fibers. Irregularities in EMG waveform is mitigated though standardized data collection methods,

filtering, feature extraction, and normalization to allow for easy comparisons of EMG signals.

Collection standards for EMG include skin preparation, electrode placement, and adherence. Before placing the electrode, the skin should be shaved and cleaned to remove oils and dead skin. A safety razor and alcohol swabs are most commonly used. It is then important for researchers to accurately identify the muscle belly and place the electrodes where there will be little interference from neighboring muscle groups. There are international standards, such as Surface ElectroMyoGraphy for the Non-Invasive Assessment of Muscles (SENIAM), to find ideal locations for commonly collected muscles [26]. It is also vital that electrodes do not shift during data collection as changes in position will read new muscle fibers and comparison with previous tests will become more difficult or, in severe cases, impossible. Double sided tape and external wraps can help to minimize sensor shift.

Once the raw EMG signals are collected, they are filtered through a bandpass filter to remove frequencies of data outside of the desired range. Typically, the discarded frequencies are under 20 Hz, as they can be attributed to mechanical artifacts, and above 450 Hz, as high-frequency aliasing can corrupt the signal [27]. When surrounded by other exoskeleton components such as motors or controller boards, notch filtering at frequencies around 50 or 60 Hz can remove electrical noise.

The EMG signal is considered to be a combination of features all describing different aspects of the muscle firing patterns [28] [29]. The three types of feature extractions typically used to decode EMG signals are 1) time domain features, 2) frequency domain features, and 3) time-frequency domain features.

Time domain features generally deal with the overall shape of the signal and are most frequently used in continuous motion estimation problems since these features are often correlated

with the level of activation or “effort” exerted by the muscle. Examples of time domain features include root mean square (RMS) (Figure 2.1), slope sign change (SSC), zero crossings, waveform length (WL), and mean absolute value (MAV) [30] [31]. Frequency Domain Features are extracted by using the estimated power spectrum density and are computationally more taxing than Time Domain Features. Examples include power spectrum (PS), mean frequency (MNF), and median frequency (MDF) [5]. There is evidence that Frequency Domain Features allow for the detection of muscular fatigue [29].

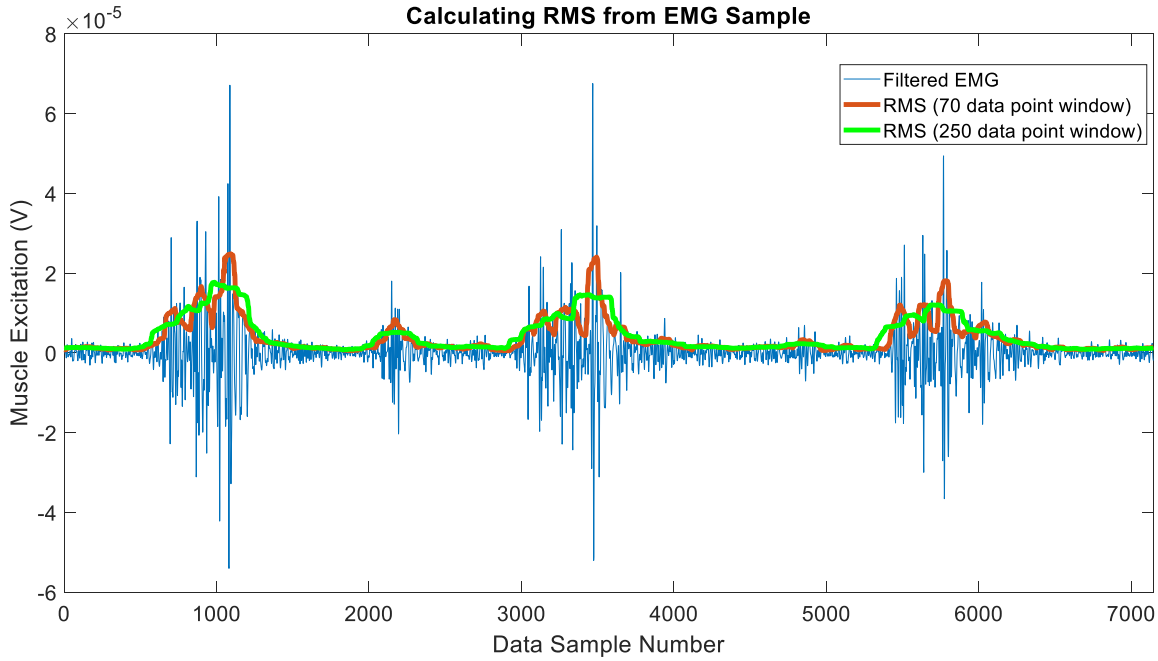

Figure 2.1: Example EMG data and corresponding RMS values calculated with 70 and 250 data point moving windows.

Time-Frequency Domain Features help assess the signal in both the time and frequency domains through a single set of computations. However, given the combination of time and frequency, this domain is generally the most computationally expensive and time consuming,

making it a less practical method than calculating the time domain and frequency domain features individually if they are to both be used [5].

After extracting the desired features, normalization techniques can be used to compare the muscle firing patterns between muscles, subjects, or even the same subject collected at a different time [32] [33]. Because EMG relies on reading the muscle activation through the surface of the skin, skin impedance can play a large role in the shape of the collected signal [32]. This means that testing a subject at various levels of hydration will affect their EMG readings. One of the most common normalization methods is to compare everything as a percentage of the maximum voluntary isometric contraction (MVIC). At the beginning of collection, the subject is asked to contract each muscle as hard as they can while the tester restricts motion of the subject [34]. Another method is a reference voluntary contraction (RVC) where a known weight, similar to the weights that would be moved during collection, is lifted [35]. The RVC is designed to set the calibration point much closer to the desired collection data range than the MVIC. Finally, there are methods such as mean or peak activation levels that normalize the EMG signals during the desired task. The subject performs the desired task, the mean or peak EMG value is determined, and then all subsequent testing is divided by that value. These last methods have been shown to reduce variability between subjects when compared to raw EMG data, MVIC, and RVC methods [36] [37] [38].

2.2.2.3 Controllers

Similar to brains in humans, exoskeleton controllers input data from sensors and then output a response to the actuators to achieve a certain outcome. The chosen sensors in part dictate the choice of controller design. In some cases, a handheld selection device is used. For others, the

exoskeleton tries to use the sensors available to determine user intention. For exoskeleton systems, user intention can be inferred using classification and regression-based algorithms. Classification designates a distinct label chosen from a list of possible outcomes that it has deemed to be the statistically most likely based on the given input data. One way classification can be visualized is through an example of a k-nearest neighbor (KNN) algorithm (Figure 2.2). KNN algorithms attempt to classify new data by measuring its distance to the existing data and classifying as the group with the shortest average distance. Classification is commonly used for sorting algorithms such as shown in Figure 2.3. Many assistive and rehabilitative exoskeletons such as Ekso, ReWalk, AUSTIN, Mina, LOPES, PAM, and ALTACRO rely on classifying high-level commands (e.g. walk, sit down) to drive the user to the predefined motion profiles [10] [39] [9] [40] [41] [42] [43] [44]. Exoskeletons used for repetitive movement tasks can be controlled with a simple timing-based, mechanically intrinsic controller on a loop, but do not allow for variability of tasks and often lead to users relying excessively on the exoskeleton [45]. Even laboratory-based exoskeletons are driven by labelling actions, poses, or even parts of a repetitive action such as portions of the gait cycle [46] [47] [48]. Classification alone could run into problems when a similar action requires varying velocity or torque outputs to the actuators. For example, a user with a fast gait might have adjust their walking speed if the programmed gait cycle is slower.

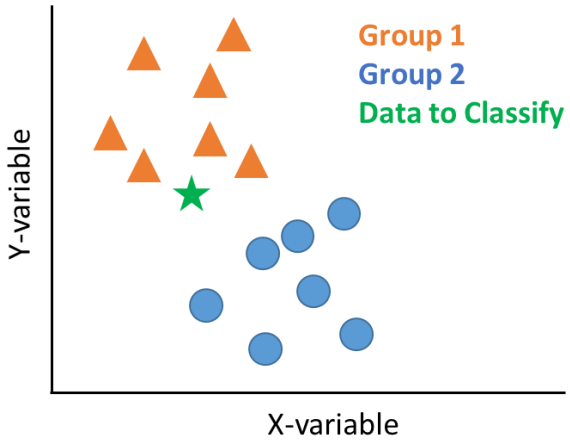

Figure 2.2: Example KNN Classification Algorithm

Figure 2.3: Example Classification Sorting Algorithm

Alternative to classification, regression is the method for continuously outputting analog signal throughout the entire motion. A regression approach removes the need for preprogrammed actions and delivers velocity or torque on a needed basis to produce a more naturalistic motion. Regression techniques for continuous motion can be categorized as static model-based or adaptive model-based. In a static model-based approach, a relationship is formed between the input signals to an estimated output signal through established equations or experimentation. Kinematic, dynamic, and musculoskeletal models have all been proposed. Kinematic models look at the human body as a chain and use techniques such as forward or inverse kinematics to calculate joint position, orientation, velocity, or acceleration [5]. Borbély & Szolgay have used an inverse kinematic model for real time estimations of joints in the arm for an OpenSim model [49].

Dynamic models use similar methods as the kinematic models but expand upon them by adding calculations of inertia, Coriolis, centrifugal, or gravity vector calculations in order to estimate a torque or force output. Koike et al. was able to mimic arm movements from EMG channels by using a forward dynamics model [50]. Musculoskeletal models use EMG signals combined with joint kinematics to estimate forces of muscles using understood muscle models such as Hill-type muscle model [51] [52]. Adaptive model-based approaches look to develop a relationship between inputs and a desired output by utilizing "black box" techniques, which mostly fall into the categorization of machine learning. Machine learning approaches operate by using large data sets to optimize a mapping function that can often find relationships not easily detectible through conventional methods.

2.2.2.3.1 Machine Learning

Machine learning methods can be separated into unsupervised methods, supervised methods, and reinforcement learning methods. In a supervised learning model, both the inputs and desired outputs are presented as part of the training data set. The algorithm produced will be a function that will link every given input to every given output by altering the equations used between them. Increasing the number of data points in the training set will generally lead to increased accuracy of the estimated output. Another method of machine learning is unsupervised learning, for which the training set only includes input data, and the algorithm finds patterns in the input dataset itself. Lastly, is a method of machine learning known as reinforcement learning, in which models learn by interacting with their environment, similar to humans learning a skill. A supervised learning method is typically pursued for exoskeleton applications since unsupervised learning may have limited association between inputs and outputs and reinforcement learning

would require a complex simulated environment.

Supervised machine learning is commonly investigated as a way of generating controls for exoskeletons. One method of supervised machine learning is Hidden Markov Models (HMM), which are statistical models that use a stochastic process to, first, describe how the system may transition from one state to another. Secondly, they provide the statistical probabilities for the outcome from each state [15]. Work by Chan and Englehart showed that a HMM built on EMG data of the arm could be used to classify six unique static poses of the hand and wrist [53]. Another example of a supervised machine learning technique is Gaussian Mixture Models (GMM), which are combinations of normal Gaussian distributions. Kilicarslan et al. has used EEG signals for real-time classification of six unique motions in paraplegic subjects [54]. Support Vector Machines (SVM) are another type of supervised machine learning, which are used to create a separation of data sets that maximizes the overall margins between the line or plane, and the data classifications. Khokhar et al. showed that EMG and force data could be used to classify 19 classes of torque at the wrist [30]. A method of supervised machine learning that is common in fields such as navigation systems or computer vision is Kalman Filters (KF). KF are recursive solutions that work by estimating current state variables using sets of understood equations, observing the accuracy of the estimation, and updating the weighted averages used to compute the estimation. A method of using a state space model built on a Hill-based muscle model and modified with an extended Kalman filter has been used to estimate joint angles and velocities at the elbow [51]. Finally, there are Artificial Neural Networks (ANN). ANN are mathematical models that are designed in a similar manner to the biological neural network comprising animal brains. The general structure of ANN can be broken down into three layers: 1) the input layer which receives the various signals, 2) the hidden layer(s) which perform various transformations, and 3) the output

layer which quantifies the results of the hidden layer(s) into outputs for the system [55]. EMG based ANN has been used for continuous motion estimation by continuously estimating joint angles, in some cases more than one joint angle simultaneously [56] [57] [58]. Figure 2.4 shows an example ANN. The input to the ANN could be EMG signals of the leg while the overall output could be a joint angle prediction. By training the algorithms in this way, a real time system could be built to control the exoskeleton.

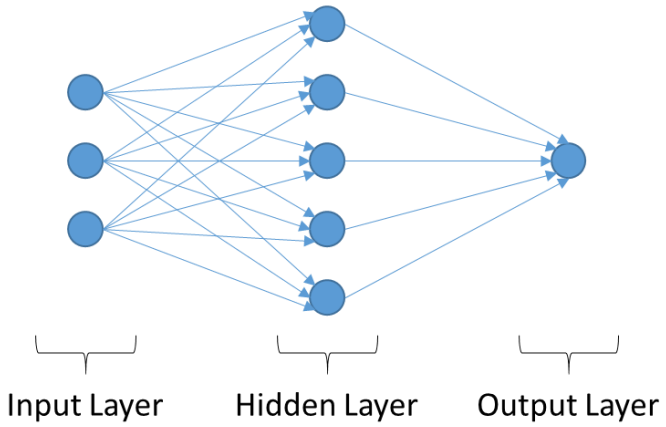

Figure 2.4: Example ANN Layer Structure

ANN layers are made up of nodes (circles in Figure 2.4), which are connected to the previous layer by a series of weighting functions (arrows in Figure 2.4). The number of hidden layers determine if it is a single-layer neural network (zero hidden layers), a shallow multi-layer neural network (one hidden layer), or a deep neural network (multiple hidden layers) [55]. The data streams come in through the input nodes without any transformation. The transition layer nodes are then calculated by the following equation:

$$v = (w_1 \times x_1) + (w_2 \times x_2) + \cdots + (w_n \times x_n) + b \tag{2.1}$$

Where v is the weighted sum for the node, w_n are the weighting functions (expressed in a $1 \times n$ matrix, w), x_n are the input values for the node (expressed in a $n \times 1$ matrix, x), and b is

the bias added to each node [55]. The weighted sum is then calculated through an activation function as follows:

$$y = \varphi(v) \tag{2.2}$$

Where y is the output for the given node and φ is the activation function. Activation functions used for machine learning are often three types of transfer functions. 1) Logistic sigmoid transfer function (log-sig):

$$\varphi(x) = \frac{1}{1 + e^{-x}} \tag{2.3}$$

2) Hyperbolic tangent sigmoid transfer function (tan-sig):

$$\varphi(x) = \tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.4}$$

3) Linear transfer function:

$$\varphi(x) = Cx \tag{2.5}$$

C is defined as a constant. The log-sig function will generate φ values apostolically between 0 and 1 while tan-sig generates values apostolically between -1 and 1 (Figure 2.5).

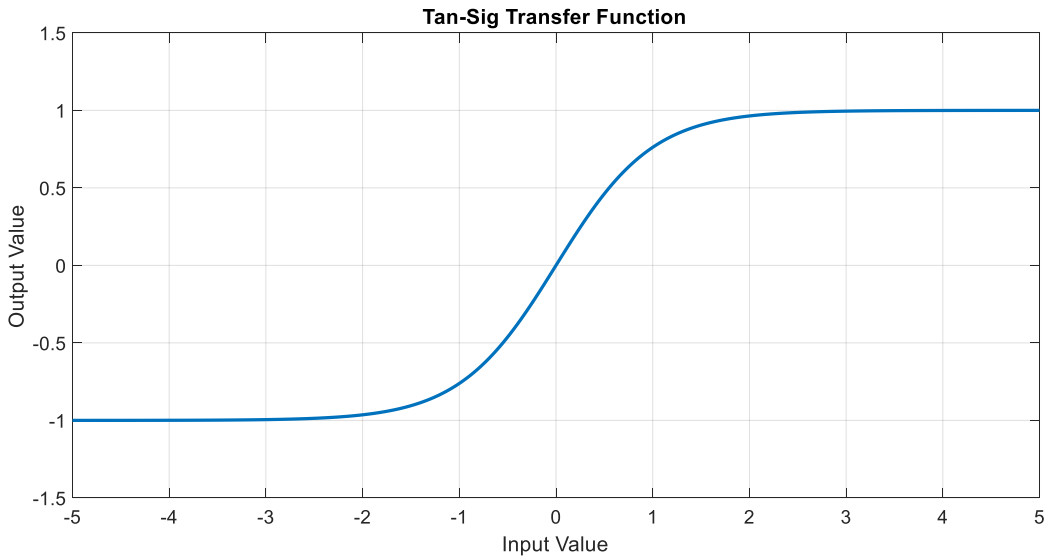

Figure 2.5: Example Tan-Sig Transfer Function

The concept of supervised learning is that an error is calculated for the estimated output against the actual output. The weighting functions are then updated for all of the nodes and the process repeated until the error is minimized [59]. The math is often simple enough to be completed by hand, but a single iteration would take minutes to calculate. In the same amount of time, the machine learning algorithm could have gone through a thousand iterations and converged on a result. This process benefits from large datasets to compute the weighting functions, often resulting in the need for long data streams. Delays are introduced mathematically into time series problem functions by the following equation:

$$y(t) = \varphi(x(t-1), \dots, x(t-d)) \tag{2.6}$$

Where d is the delay and correlates to the number of terms in the transfer.

The larger the delay, the greater number of past inputs will be used in the calculations. Figure 2.6 shows an example ANN that is used in the work of this thesis. This network is designed

with an input of seven data streams ($x_n(t)$ where $n = 7$). The delay value is two data points. The weighting and bias are added together into a tan-sig transfer function in the hidden layer of ten nodes. Finally, the weighting and bias are added into a linear transfer function of the output layer of a single node to calculate a single output stream ($y_n(t)$ where $n = 1$).

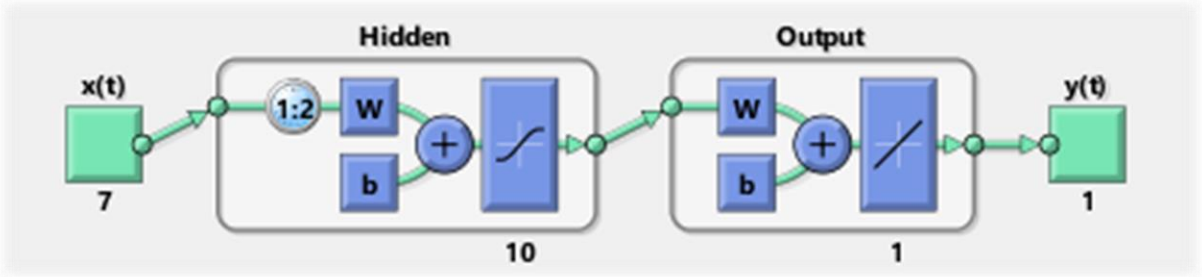

Figure 2.6: Figure of ANN used in this thesis, generated from Matlab®

To illustrate the capabilities of the algorithm described in Figure 2.6, example data has been used to predict knee flexion angles during gait (Figure 2.7).

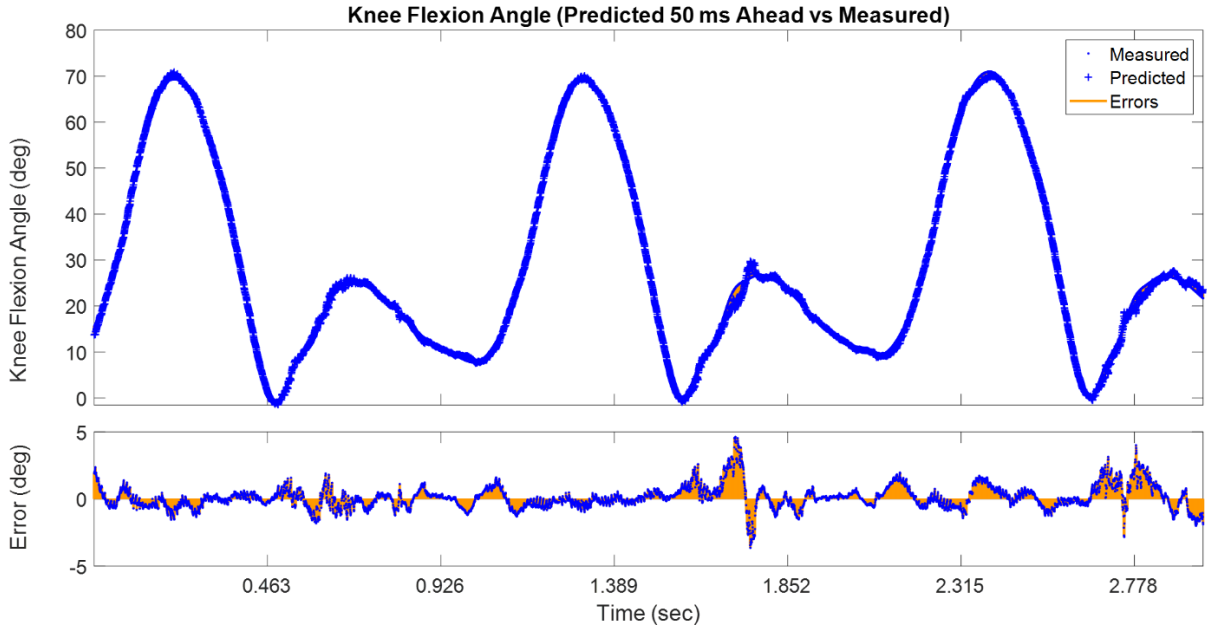

Figure 2.7: Example Output Data of an ANN

In this example, six streams of EMG data and a single stream of knee flexion angles were the inputs used to predict a stream of knee flexion angles 50 ms into the future. For clarity, the error has been enlarged in scale relative to the knee flexion angle plot.

2.3 Purpose

The following work employs the described ANNs to make predictions of knee flexion angles. These predictions will utilize the ability to detect EMG signals roughly 100 ms before bodily motion occurs to create a control algorithm capable of reducing lag time between the exoskeleton and the user. Furthermore, the following work will assess the necessary inputs for an ANN to make accurate knee flexion predictions on a population independent of the population used to train such ANN.

Chapter 3

Subjects and Data Collection Methods

This chapter describes the combined methods used in the two studies (Chapters 4 and 5) presented in this thesis. Subjects and data collection methods overlapped while data processing distinguishes between the two and will be discussed in their respective chapters. Overall, there were ten subjects that participated in this study (5 males, age = 21.5 ± 2.0 yrs, weight = 64.5 ± 9.8 kg, height = 166.9 ± 14.5 cm). All subjects reported no history of chronic pain in the spine or lower extremities in the six months prior to participating in the study. All study procedures were approved by the Auburn University Institutional Review Board (IRB), and subjects provided informed written consent before participating. The experiment took place at the Auburn University Biomechanical Engineering Laboratory.

Twelve surface EMG electrodes (Delsys Trigno IM, Delsys Inc.) were placed bilaterally on six muscles along the thigh. These muscles were the right and left tensor fasciae latae, rectus femoris, vastus medialis, vastus lateralis, biceps femoris, and semitendinosus. The placement of these electrodes was consistent with SENIAM guidelines [26]. Before placing the electrodes, excessive hair was removed with a small electric hair trimmer and the skin was cleansed of oils and debris with an alcohol swab to improve the quality of the recorded signals. A double-sided

adhesive was used to secure the sensors in their desired locations and sports wrap was added over the electrode to help prevent loss of signal connection. Before testing, the subjects were allowed time to acclimate to the equipment. Raw EMG signals were collected at 1111 Hz for each of the twelve channels and fed through a Butterworth filter to remove motion artifacts (< 20 Hz) and high frequency aliasing effects (> 450 Hz). These signals were then detrended and rectified so that the RMS for each muscle could be calculated.

A ten-camera Vicon motion capture system was used to track a 79 retroreflective marker set consistent with the work of Andriacchi et al. [60]. A Vicon Lock+ box was used to ensure synchronicity between the Vicon motion capture and the Delsys Trigno sensor signals. Nexus software was used to collect the marker position at 120 Hz (Version 2.6.1; Vicon Motion Systems Ltd, Oxford Industrial Park, Oxford, UK). Marker positional data was transferred to Visual3D, where it was filtered with a 15 Hz low pass Butterworth filter to remove noise. Body segments were created using the marker positions following the International Society of Biomechanics recommendations [61]. Grood and Stunay's joint coordinate system was used to calculate knee flexion [62]. The Visual 3D model of the knee was made with six degrees of freedom, but because the knee flexion angle is much greater than other rotations and translations in the knee, the focus of the predictive algorithms was on a single degree: flexion.

Participants performed 15 walking trials over a distance of approximately 30 feet at a self-selected pace. Minimal feedback was provided in order to capture naturalistic movements. Collected trials were randomly split into training and testing categories in order to counter learning effects over the duration of the trials. Ten trials were chosen as training trials to train the algorithms. The ten chosen training trials were trials one, two, four, five, seven, eight, ten, eleven, thirteen, and fourteen. The remaining five testing trials were used to test the algorithm accuracy.

This study used the time domain feature, amplitude, to compute root mean squared (RMS) of the EMG for analysis with a moving window set to 70 data points. MATLAB was used to create Nonlinear Input-Output Time Series Neural Network algorithms trained using Bayesian Regularization with a single hidden layer of ten nodes and a feedback delay set to two. The inputs and outputs of the algorithms were study dependent and will expanded upon accordingly.

Chapter 4

EMG and Joint Angle-Based Machine Learning to Predict Future Joint Angles at the Knee

4.1 Introduction

In order for an exoskeleton to feel naturalistic to a user, the motion of the actuator must match the intended motion of the person. Delays in control mechanisms are likely to lead to desynchronization with the user and increased metabolic cost. It is not entirely clear how fast a system must operate, but currently, the clinically accepted response time delay to the user's movement is 300 ms [16]. Furthermore, work by Petrella et al. (1997) shows that through proprioception, healthy young adults are able to distinguish and reproduce knee flexion angle positions within two degrees of accuracy [17]. Although the need for empirical validation remains, the work by Petrella et al. provide a reasonable target of a discrepancy less than two degrees for a naturalistic feel.

The largest gaps in active exoskeleton technology include controls that are limited in the number of actions they can perform and delays between the user and the actuation of the exoskeleton. EMG can be read noninvasively and approximately 100 ms before human motion ensues [63]. Therefore, a theoretical window exists between the readings of EMG signal to the

realization of motion in which a prediction can be made of a biomechanical parameter, such as a future joint angle. Regression algorithms can create an advantage that involves the allowance for any motion to occur if trained correctly, rather than the user of the exoskeleton being limited to a predetermined path.

The purpose of this study was to assess the feasibility of employing a predictive ANN algorithm trained using supervised learning to accurately counter any delays that may be caused by computation time or the transmission of data from sensors to an active exoskeleton actuator. The study also explored the effects of various time delays on predictive accuracy.

4.2 Methods

The seven input variables for the machine learning algorithms were the six EMG signals on a single leg and the knee flexion angle calculated post hoc with Visual 3D. The algorithms output predictions of that same knee's flexion angle estimated at 50, 100, 150, and 200 ms into the future.

Ten algorithms were trained for a varying number of randomly assigned training trials (1-10) for each of the ten subject's two legs and for all four prediction time intervals. This resulted in a total of 800 trained algorithms. RMS error was calculated by comparing the algorithms' output angle against the motion capture-based calculation of the knee flexion angle for each data point of that subject's five testing trials. The averages and standard deviations of RMS error were calculated for each algorithm for each prediction time and number of training trials.

Analysis of variance (ANOVA) was used to test for the significance of each prediction variable (subject, gender, leg, and time interval) and their interactions. Time interval was treated

as a repeated measures variable. The unadjusted algorithm error data were found to have fan-shaped residuals (indicating unequal variances by time interval), thus a log transformation of the algorithm predictions was used for the ANOVA. This resolved the equal variance assumption violation. Tukey Honestly Significant Difference (HSD) post hoc tests were used to evaluate significant differences between conditions for main effects or interactions that were significant. Linear regression was used to develop a prediction model for error based on variables identified as significant in the ANOVA model. The Type I error rate (alpha) was set at 0.05 for all tests.

4.3 Results

Figures 3.1 and 3.2 show the average RMS errors of all the subjects at various prediction times and the various number of training trials separated by the right and left leg. Both legs saw a significant decrease in the average prediction error as the number of training trials increased and as the prediction times were decreased. The largest decrease in error from a single trial to ten trials was observed in the right leg being predicted at 100 ms with a 93% reduction. The smallest decrease was observed in the right leg 200 ms prediction with a reduction of 74%. Similarly, Figures 3.3 and 3.4 show that standard deviation of the error also decreased with an increase in the number of training trials.

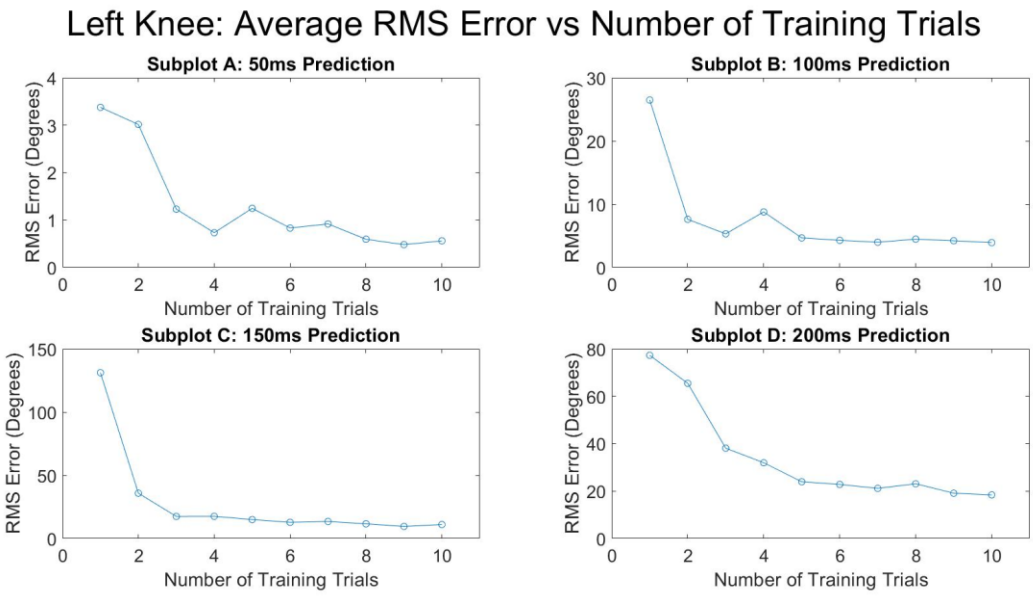

Figure 4.1: Number of training trials effects on average degrees of RMS error in left knee flexion prediction

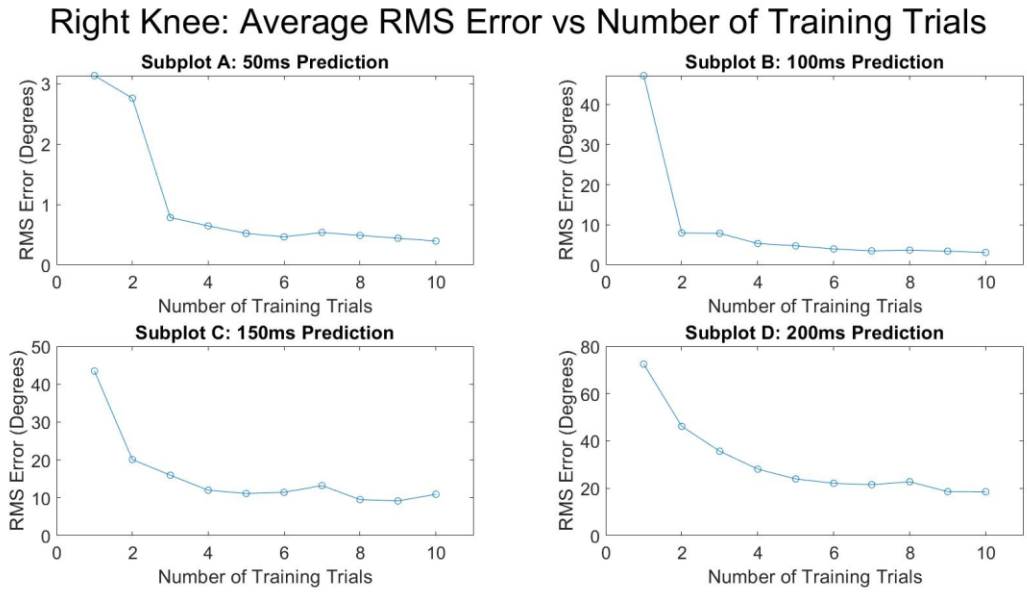

Figure 4.2: Number of training trials effects on average degrees of RMS error in right knee flexion prediction

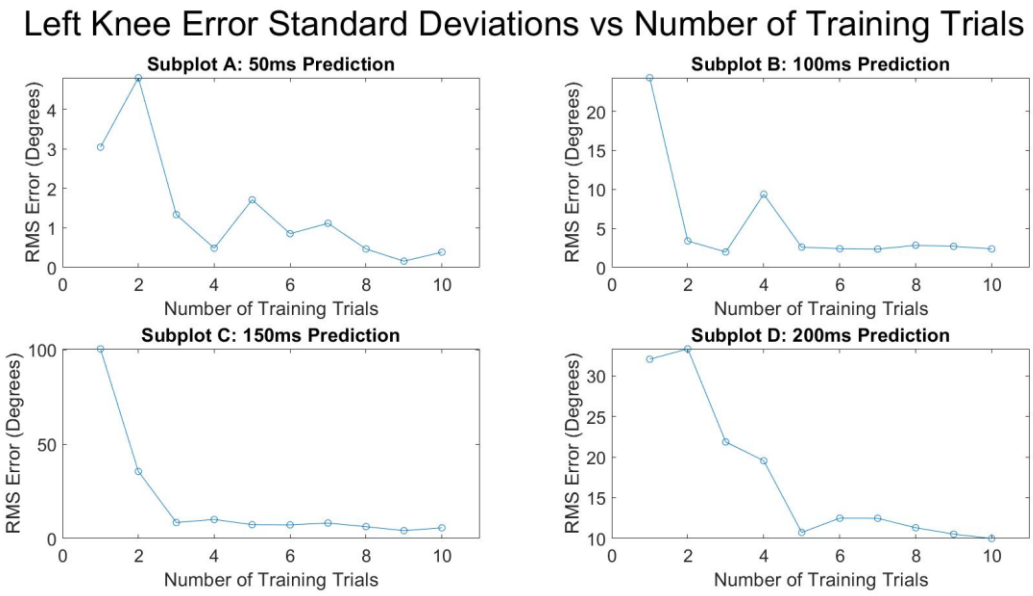

Figure 4.3: Number of training trials effects on average standard deviation of degrees of RMS error in left knee flexion prediction

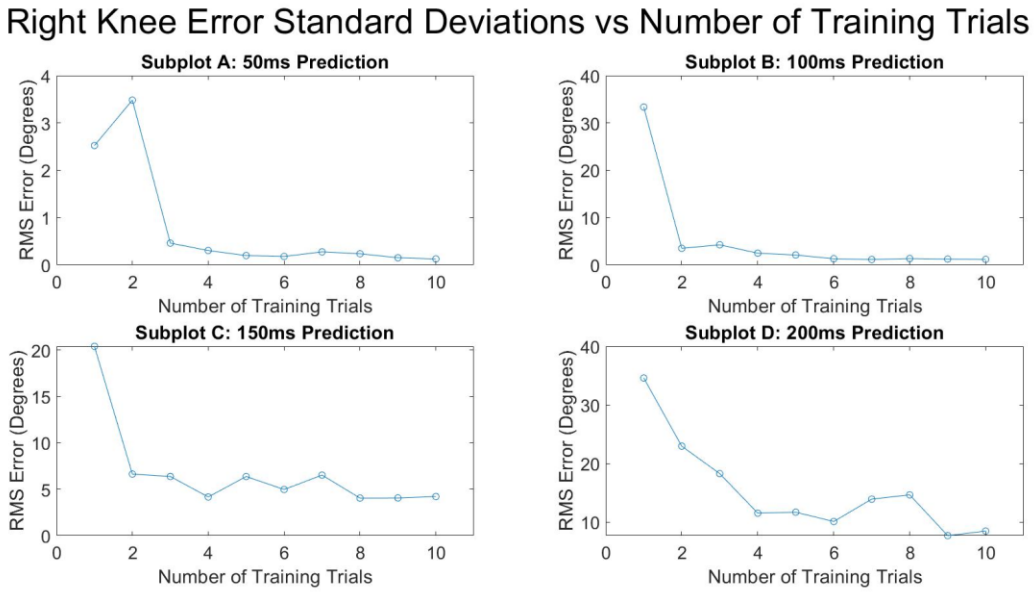

Figure 4.4: Number of training trials effects on average standard deviation of degrees of RMS error in right knee flexion prediction

Figure 4.5 shows a box and whisker plot of the RMS error for each of the subjects' algorithms trained with one (top) as well as all ten (bottom) available training trials. A similar trend is seen in both the right and the left leg that the error and variation of predictions increase the further into the future the algorithm is attempting to predict. The median errors for both the right and left leg were reduced at rates of 97% when comparing 200 ms to 50 ms at ten training trials.

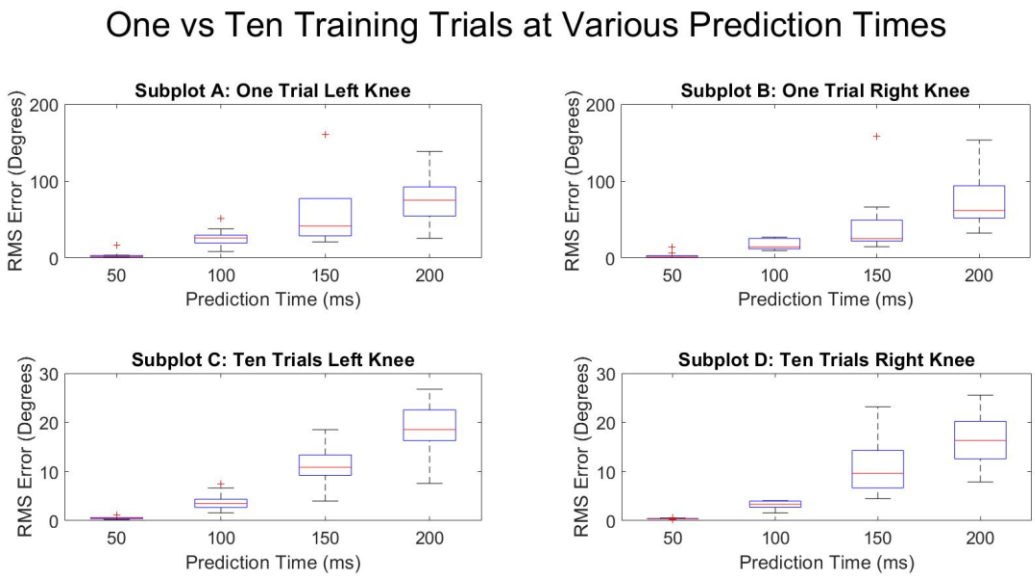

Figure 4.5: Prediction times effects on average degrees of RMS error in knee flexion prediction for one training trial and for ten training trials. Outlier in Subplot A, 150 ms prediction time of value 810.85 degrees. Outlier in Subplot B, 100 ms prediction time of value 157.70 degrees. Outlier in Subplot D, 200 ms prediction time of value 42.02 degrees.

Table 4.1 shows the results of the Tukey HSD All-Pairwise Comparisons Test with a calculated Standard Error for Comparison of 0.03, Critical Q Value of 3.76, and Critical Value for Comparison of 0.086. ANOVA results demonstrated that the main effect of time was statistically significant with respect to algorithm predictions ($F_{3,48} = 953.88$, $p < 0.0001$). Tukey HSD post hoc

tests demonstrated that algorithm predictions for each time period were significantly different from one another (Table 4.2).

Table 4.1: Analysis of Variance performed for subject, gender, leg, and prediction time

Analysis of Variance Table for Algorithm					
Source	**DF**	**SS**	**MS**	**F**	***p***
Subject	4	0.7523	0.18807		
Gender	1	0.5241	0.52411	7.38	0.0532
Error Subject*Gender	4	0.2840	0.07100		
Leg	1	0.0701	0.07007	1.19	0.3076
Gender*Leg	1	0.0634	0.06341	1.07	0.3303
Error Subject*Gender*Leg	8	0.4721	0.05902		
Times	3	29.7298	9.90993	953.88	0.0000
Gender*Times	3	0.0049	0.00164	0.16	0.9243
Leg*Times	3	0.0369	0.01230	1.18	0.3256
Gender*Leg*Times	3	0.0094	0.00312	0.30	0.8250
Error Subject*Gender*leg*Times	48	0.4987	0.01039		
Total	79				

Note: SS are marginal (type III) sums of squares

Table 4.2: Tukey HSD All-Pairwise Comparisons Test for time intervals showing significance between each time interval.

Prediction Time	Mean RMS Error
50 ms	0.48
100 ms	3.51
150 ms	10.99
200 ms	18.47

ANOVA results reveal that subject, leg, or gender differences were not statistically significantly different in terms of the algorithm predictions, nor were any interactions among these variables. It may be noted that the main effect of gender demonstrated a non-significant trend.

Thus, the final model indicates that the only significant factor relating to algorithm predictions was time interval.

Figure 4.6 shows the regression model with slope = 10.522 log(RMS error)/seconds, $p<0.0001$ and y-intercept = -0.721 log(RMS error), $p<0.0001$. The adjusted R^2 fit was equivalent to 0.85. Figure 3.7 shows the same data without logarithmic transformation.

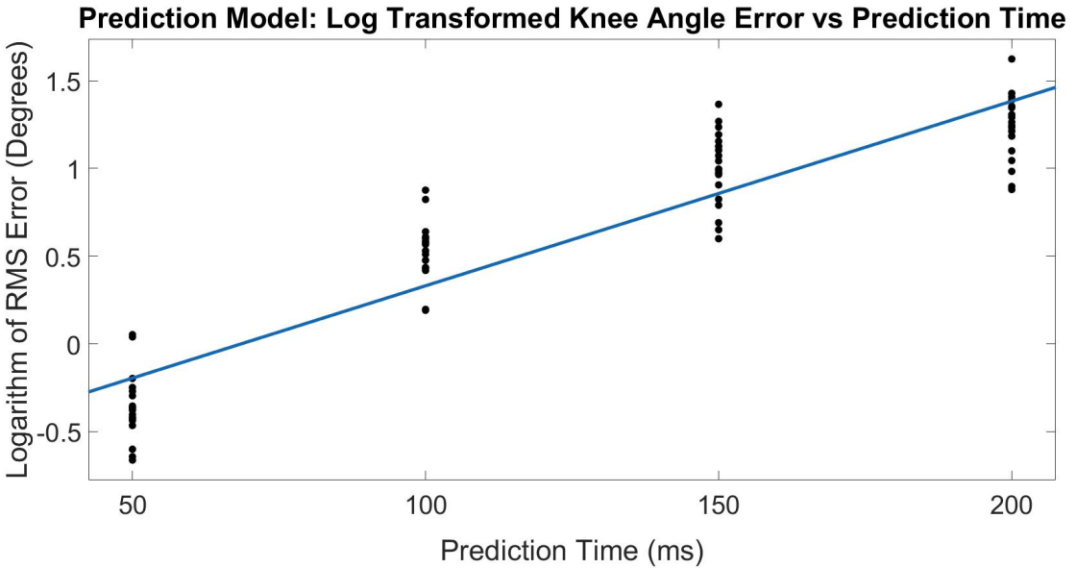

Figure 4.6: Regression model of logarithmically transformed error for prediction times.

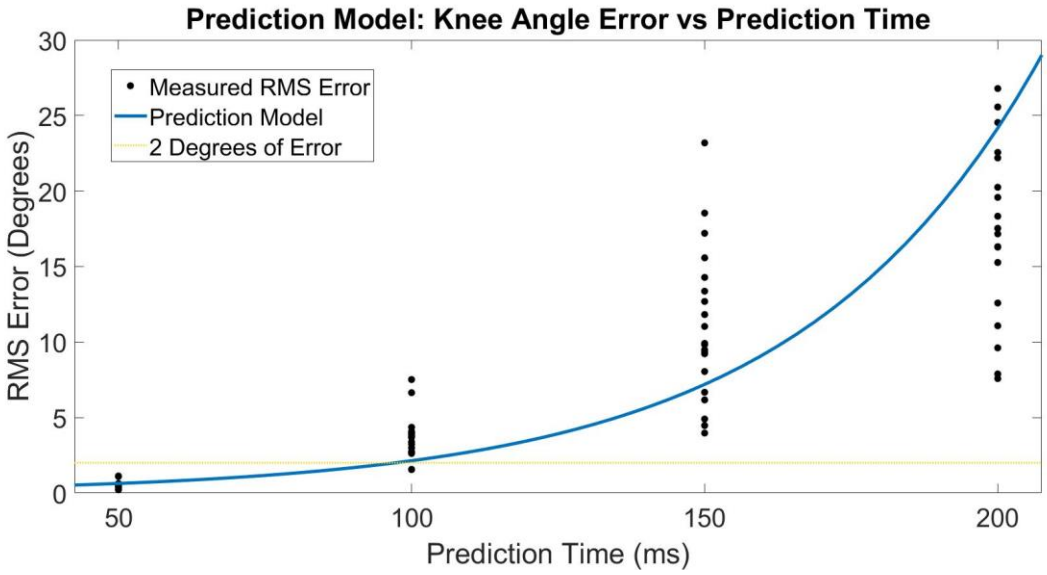

Figure 4.7: Regression model of error for prediction times compared to 2 degrees of error

4.4 Discussion

This study investigated the accuracy of an ANN designed to predict future flexion angles of the knee using EMG measurements and past measurements of knee flexion. Using a two-degree error metric as the measure for success, based upon the study by Petrella et al. [17], it can be concluded that the algorithms succeeded in predicting the knee flexion angle 50 ms into the future for all subjects, and succeeded in predicting the knee flexion angle 100 ms into the future for some subjects (Figure 4.7). These predictive algorithms were successful for both legs and both genders. The generated regression prediction model calculated that times less than 97.2 ms (Figure 4.7) have the greatest potential for generating undetectable errors. This is directly in line with the time allowable for reading muscle activation before motion occurs [63]. Predictions made further out than 100 ms could be detrimentally affected by the inputs of EMG signals that are not associated with the impending motions. In addition, the finding that time is the only significant factor opens the possibility to exploring opportunities of using independent sets of data for training and testing.

Training and testing on the same subject and same leg may not always be feasible, for example, if the exoskeleton will be used for rehabilitating a musculoskeletal injury. Predicting future angles is essential for synchronized movement between the user and exoskeleton even with the computational abilities of modern exoskeletons. The bigger the gap in prediction time, the more buffer the exoskeleton has to process signals and actuate the given joints to match the motion of the user.

A limitation of this study is the small sample size and narrow range of demographic characteristics of the subjects. Age, percentage body fat, and activity level could impact the quality of predictions. Another limitation is the focus on subjects engaged in an established gait cycle. Expanding the predictions to less repeatable actions will most likely cause the accuracies to decrease if using the same machine learning method. Further testing should be conducted to determine if a single walking-based regression algorithm is sufficient for multiple actions or if a more complex solution is required. Examples of complex solutions would be deep learning models or multilayer networks that can classify the movement and then used a specific regression algorithm to predict angles of that action.

Further work should also explore if a single algorithm can be utilized across an independent population. With the inclusion of multiple subjects or both of an individual's legs into a single algorithm, various patterns can be learned in hopes of making predictions on a subject without the need for their specific individual training data. Additional future work could include assessing the number of training trials required to reach the point of diminishing returns to prediction accuracy, introducing transitional movements into the training sets, or using techniques such as hyper-parameter tuning in order to design the optimal machine-learning algorithm parameters for the desired prediction time.

4.5 Conclusion

This study successfully demonstrated the feasibility of employing an ANN to accurately predict knee flexion angle. This can be used to counter delays caused by transmission of data and computation times of powered exoskeletons. A prediction model has been created which draws the correlation of prediction time against the expected accuracy. It also outlines the basis for understanding the amount of data needed to accurately train an algorithm for a given subject.

Chapter 5

Population and Training Source Influence on Machine Learning Ability to Predict Future Joint Angles at the Knee

5.1 Introduction

Currently, many assistive and rehabilitative exoskeletons such as Ekso, ReWalk, AUSTIN, Mina, LOPES, PAM, and ALTACRO rely on predefined motion profiles to drive the user to high-level commands (e.g. walk, sit down) [10] [39] [9] [40] [41] [42] [43] [44]. Predefined motion patterns lack full mobility, especially for healthy users where low-level control is desired (e.g. walking speed, stride length). Many augmentation exoskeletons such as BLEEX or the Nurse Robotic Suit rely strictly on kinematic and kinetic sensors such as inertial measurement units (IMUs), encoders, and force sensors to initiation a movement response [20] [21] [22] [23]. The biggest drawback of such mechanical sensors is that detection of motion intent can only occur once the motion has begun, resulting in lack sufficient synergy between user and exoskeleton and limiting cooperation between the two [15].

In response to this problem, the work presented in Chapter 4 utilized EMG to create predictive algorithms that could accurately estimate knee flexion angles (< 2° error) as far as 97 ms into the future. The results indicated that accurate predictive algorithms could be created

regardless of subject, leg, and gender. The next steps were to see if the distinguishing of subject, leg and gender could be removed to achieve accurate predictions, thus, creating a generalized algorithm that could work for anyone. A gap exists in current regression methods found in the literature whereas algorithms must be created for each individual subject to predict joint motion [56] [58]. This is often not feasible as training individualized algorithms can be cost, time, and physically prohibitive. The regimen needed for training individualized algorithms would currently require every task to be performed in a laboratory setting by the user without the aid of such exoskeletons.

One of the main reasons a generalized algorithm has not yet been developed is because EMG signals are often noisier than non-biological signals and fluctuate between users [32]. Many studies that utilize EMG to make comparisons between subjects will apply a form of normalization based on the given task [32] [33]. The normalized signals often yield similar signal profiles for the given task, therefore, removing any natural subject-to-subject variability [37] [38]. Removal of variability is theorized to be necessary for the creation of generalized exoskeleton control algorithms.

The purpose of this study was to employ a predictive ANN algorithm to explore the effects of various training sources on predictive accuracy for theoretical control of an active knee exoskeleton. This study will attempt to create a generalized control algorithm that would be ideal for situations when the user is unable to perform the required trials for an individual fit. For this study, the following hypotheses were developed: 1.) Training and testing on the same leg would yield the highest accuracy when compared to training and testing on the contralateral leg or independent populations; 2.) EMG signals without knee flexion angles do not provide enough information to accurately predict future knee flexion angles; 3.) Using only past flexion angles will

be able to produce accurate predictions for knee flexion angles during a repetitive gait cycle; and 4.) The combination of EMG signals and flexion angles will improve the accuracy when compared to only using knee flexion angles only.

5.2 Methods

In this chapter, MATLAB was used to create Nonlinear Input-Output Time Series ANN algorithms trained using Bayesian Regularization with a single hidden layer of ten nodes and a feedback delay set to two. Input variables for the algorithms included all six EMG channel signals (either RMS normalized to mean walking values or raw, not normalized RMS) from a single leg and/or the simultaneous knee flexion angle of the same leg that was calculated post hoc with Visual 3D. As a result, five testing input conditions were created based on the type of training data provided to the algorithm: 1.) Not normalized RMS + flexion angle (NNR+FA), 2.) Normalized RMS + flexion angle (NR+FA), 3.) Flexion angle (FA) only, 4.) Normalized RMS (NR) only, and 5.) Not normalized RMS (NNR) only. The target output for the predictive algorithms was the knee flexion angle predicted 50 ms into the future. Results of these predictions were compared to results from Visual 3D at the corresponding 50 ms into the future.

Further, all five testing input conditions were tested for three subgroups of subjects: 1.) Trained on a single user, single leg and tested on that user's same leg (ipsilateral); 2.) Trained on a single user, single leg and tested on that user's alternate leg (contralateral); and 3.) Trained on nine subjects, both legs and tested on the remaining subject's two legs (independent population). Moreover, ten algorithms were trained for each permutation of testing resulting in 150 predictive algorithms total (5 training data types, 3 training subgroups, 10 subjects). Each algorithm generated two data points because right and left legs of each subject were used independent of one

another. Figure 5.1 provides a breakdown for each testing input condition to how the subgroups were formed. RMS error was calculated by comparing the algorithms' output angle against the motion capture-based calculation of the knee flexion angle for each data point of that subject's five testing trials. Means and standard deviations were calculated from the 20 data points in each subgroup.

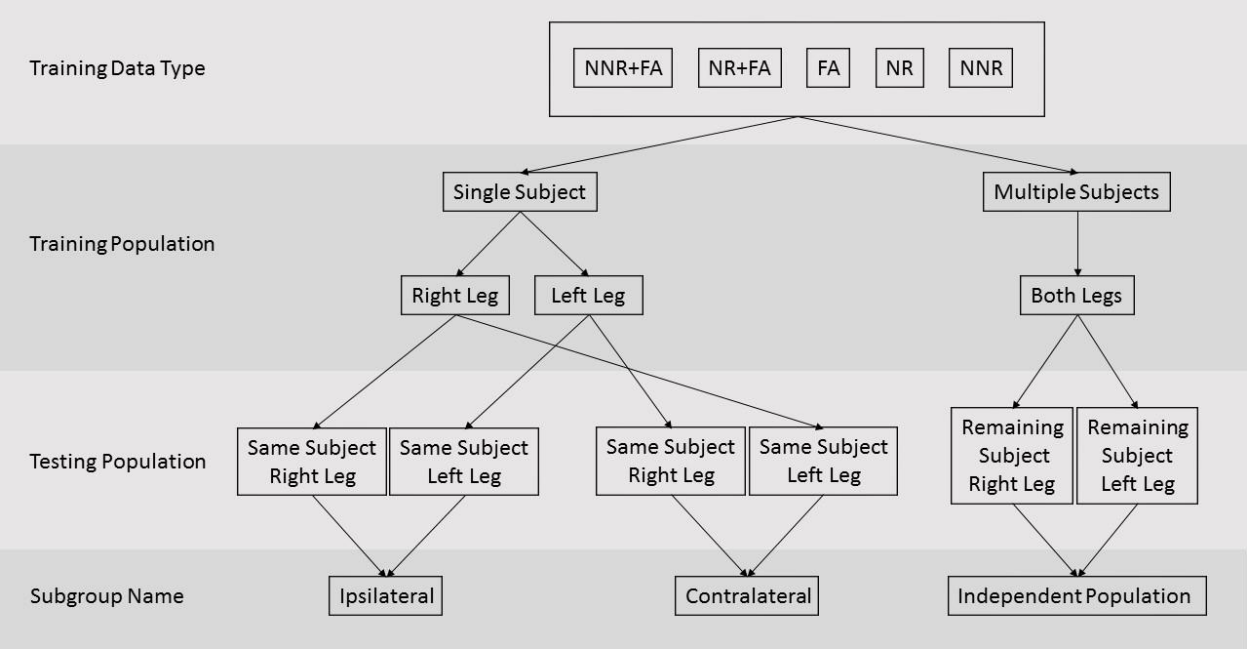

Figure 5.1: Breakdown of Testing Data Type into Subgroups

A power analysis was conducted to find the number of subjects required to assess the success metric of 2 degrees RMS error given a 0.8 power. The power analysis indicated that 10 legs were required. By using both right and left legs of the ten subjects, a sufficient sample size was obtained. Analysis of variance (ANOVA) was used to test for significant differences in resulting RMS error between each test condition (type of training data and the training/testing population). The threshold for outcomes to be included in the ANOVA was set at <10 degrees of

mean RMS error as larger error values indicate unrealistic utility for a knee exoskeleton control system. Following the ANOVA, Tukey Honestly Significant Difference (HSD) post hoc tests were used to further evaluate significant differences between conditions for main effects or interactions that were significant. The Type I error rate (alpha) was set at 0.05 for all tests.

5.3 Results

Seven out of the 15 groupings successfully produced results with <10 degrees of RMS error (Table 5.1). All of the subgroupings of NR+FA and FA combined for six of the seven while zero of the algorithms for NR or NNR were able to predict knee flexion angles with <10 degrees RMS error.

Table 5.1: Mean and Standard Deviation Root Mean Squared (RMS) Error in Degrees for All Groups

		Mean RMS Error (Degrees)	Standard Deviation RMS Error (Degrees)
Not Normalized RMS + Flexion Angle (NNR+FA)	Ipsilateral	0.58	0.39
	Contralateral	36.61	149.88
	Independent Population	321.31	961.94
Normalized RMS + Flexion Angle (NR+FA)	Ipsilateral	0.50	0.25
	Contralateral	1.16	0.41
	Independent Population	1.69	1.01
Flexion Angle (FA)	Ipsilateral	1.06	0.35
	Contralateral	1.55	0.53
	Independent Population	2.17	0.91
Normalized RMS (NR)	Ipsilateral	239.25	111.06
	Contralateral	402.15	81.53
	Independent Population	352.23	106.27
Not Normalized RMS (NNR)	Ipsilateral	238.15	106.22
	Contralateral	541.45	403.22
	Independent Population	716.39	879.61

Table 5.2 shows the individual results for the algorithms trained for NNR+FA. The results of this specific test condition were isolated because the results for contralateral and independent population exhibited a standard deviation much larger than the mean RMS error, indicating outliers. It was found that a small number of algorithms produced large errors, as much as 3,488 degrees RMS error, while most of the outcomes were close to, if not within, the desired range of less than two degrees RMS error. NR+FA is included to highlight improvement of normalization.

Table 5.2: Mean Output Values for Algorithms Trained Not Normalized RMS + Flexion Angle (NNR+FA) and Normalized RMS +Flexion Angle (NR+FA)

Test Population	Not Normalized RMS + Flexion Angle (NNR+FA) RMS Error (Degrees)			Normalized RMS + Flexion Angle (NR+FA) RMS Error (Degrees)		
	Ipsilateral	Contralateral	Independent Population	Ipsilateral	Contralateral	Independent Population
F01 Left	0.41	1.03	1.22	0.45	0.57	0.66
F01 Right	0.43	0.87	1.47	0.43	0.81	0.83
F02 Left	0.48	3.01	1.58	0.62	1.26	1.58
F02 Right	0.42	1.61	2.04	0.37	1.21	1.84
F03 Left	0.81	1.09	2.48	0.71	0.95	1.48
F03 Right	0.48	1.08	2.96	0.48	0.67	1.99
F04 Left	1.75	1.75	2.92	0.50	1.70	1.83
F04 Right	0.4	1.97	2.12	0.40	1.48	1.29
F05 Left	0.37	1.49	1.12	0.32	1.23	0.85
F05 Right	0.37	0.69	1.29	0.36	0.51	0.81
M01 Left	0.31	24.29	2.65	0.35	1.81	0.96
M01 Right	1.01	673.09	3.16	0.46	1.56	1.56
M02 Left	1.2	8.06	3488.27	1.16	1.49	2.42
M02 Right	1.06	2.26	2898.84	1.15	1.55	3.06
M03 Left	0.29	1.42	4.72	0.27	1.76	5.32
M03 Right	0.36	3.89	1.61	0.35	0.80	1.77
M04 Left	0.41	1.16	1.88	0.48	0.83	1.30
M04 Right	0.39	0.47	1.82	0.38	0.70	1.18
M05 Left	0.33	2.17	2.29	0.38	0.98	1.57
M05 Right	0.28	0.86	1.78	0.29	1.22	1.52

Figure 5.2 shows a box and whisker plot comparing mean RMS Error for all of the test conditions with mean error <10 degrees. ANOVA results indicated significance between the seven groupings (p <0.0001). Figure 5.3 and Table 5.3 shows results of the Tukey HSD post hoc tests and demonstrate groups that are significantly different from one another by whether or not the vertical bounds lines overlap.

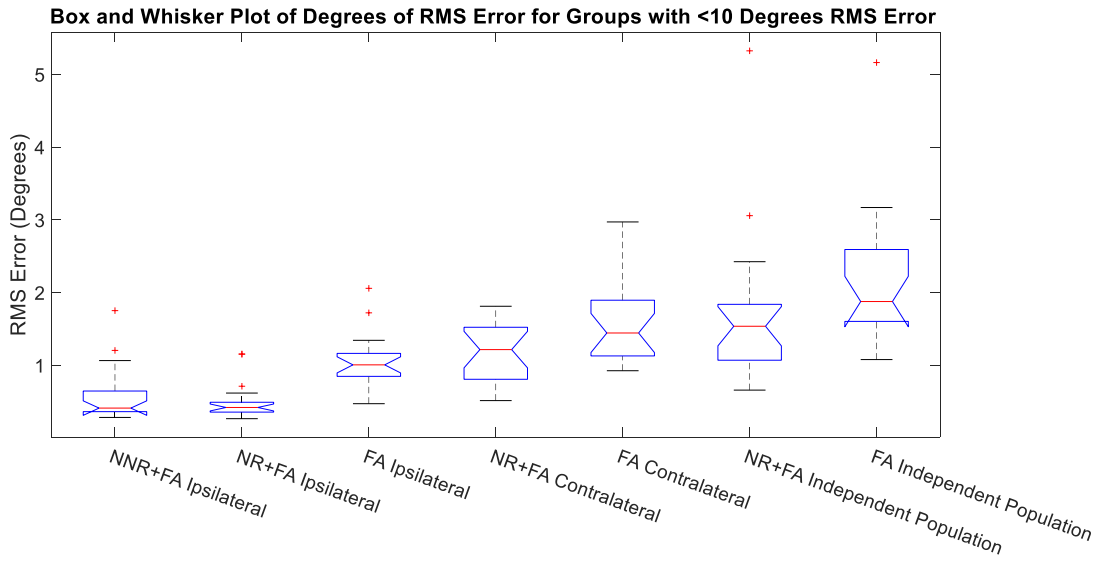

Figure 5.2: Box and Whisker Plot of Degrees of Root Mean Squared (RMS) Error for Groups with <10 Degrees RMS Error

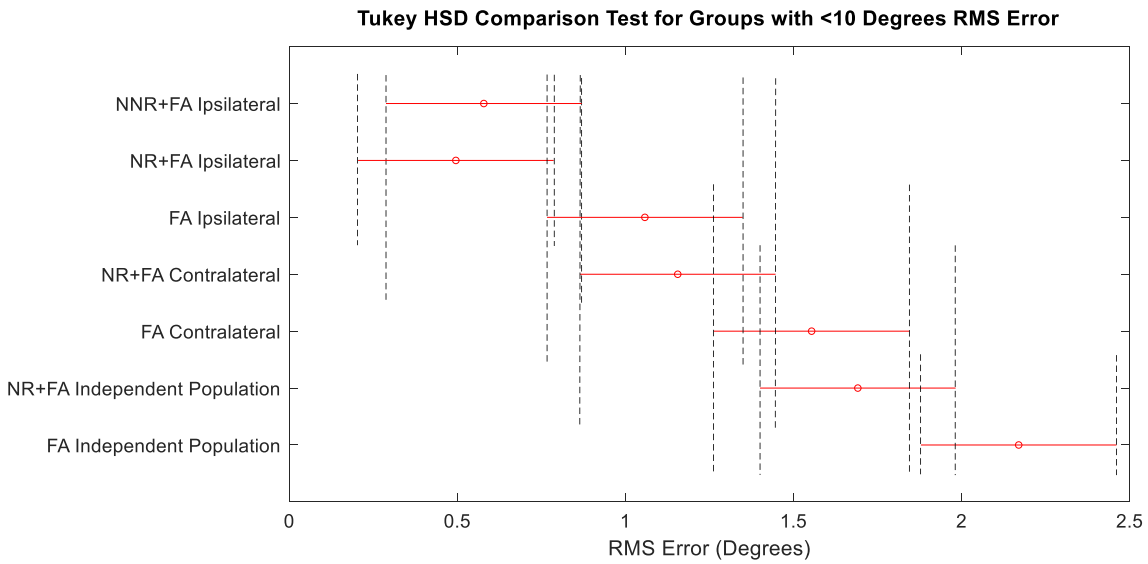

Figure 5.3: Tukey Honestly Significant Difference (HSD) Comparison Test for Groups with <10 Degrees Root Mean Squared (RMS) Error. Groups that do not overlap are significantly different from each other (indicated with the dotted lines).

Table 5.3: Tukey Honestly Significant Difference (HSD) Comparison Test for Groups with <10 Degrees Root Mean Squared (RMS) Error. Emboldened indicates significance ($p<0.05$)

First Group	Second Group	Lower Limit 95% Confidence Interval	Difference in Mean (deg)	Upper Limit 95% Confidence Interval	*p*-Value
NNR+FA (Ips)	NR+FA (Ips)	-0.4987	0.0832	0.6650	0.9996
NNR+FA (Ips)	FA (Ips)	-1.0609	-0.4791	0.1028	0.1870
NNR+FA (Ips)	NR+FA (Con)	-1.1587	-0.5768	0.0050	0.0538
NNR+FA (Ips)	**FA (Con)**	**-1.5569**	**-0.9751**	**-0.3932**	**0.0000**
NNR+FA (Ips)	**NR+FA (Ind)**	**-1.6944**	**-1.1125**	**-0.5307**	**0.0000**
NNR+FA (Ips)	**FA (Ind)**	**-2.1729**	**-1.5911**	**-1.0092**	**0.0000**
NR+FA (Ips)	FA (Ips)	-1.1441	-0.5623	0.0196	0.0661
NR+FA (Ips)	**NR+FA (Con)**	**-1.2419**	**-0.6600**	**-0.0781**	**0.0145**
NR+FA (Ips)	**FA (Con)**	**-1.6401**	**-1.0583**	**-0.4764**	**0.0000**
NR+FA (Ips)	**NR+FA (Ind)**	**-1.7776**	**-1.1957**	**-0.6138**	**0.0000**
NR+FA (Ips)	**FA (Ind)**	**-2.2561**	**-1.6743**	**-1.0924**	**0.0000**
FA (Ips)	NR+FA (Con)	-0.6796	-0.0977	0.4841	0.9989
FA (Ips)	FA (Con)	-1.0779	-0.4960	0.0859	0.1543
FA (Ips)	**NR+FA (Ind)**	**-1.2153**	**-0.6334**	**-0.0516**	**0.0226**
FA (Ips)	**FA (Ind)**	**-1.6939**	**-1.1120**	**-0.5301**	**0.0000**
NR+FA (Con)	FA (Con)	-0.9801	-0.3983	0.1836	0.4030
NR+FA (Con)	NR+FA (Ind)	-1.1176	-0.5357	0.0461	0.0946
NR+FA (Con)	**FA (Ind)**	**-1.5961**	**-1.0143**	**-0.4324**	**0.0000**
FA (Con)	NR+FA (Ind)	-0.7193	-0.1374	0.4444	0.9928
FA (Con)	**FA (Ind)**	**-1.1978**	**-0.6160**	**-0.0341**	**0.0298**
NR+FA (Ind)	FA (Ind)	-1.0604	-0.4785	0.1033	0.1880

5.4 Discussion

This study investigated the accuracy of an ANN built from various training sources and designed to predict future knee flexion angles using combinations of EMG and current knee flexion angles. For all of the scenarios tested, six subgroupings were able to predict knee flexion angles under the target goal of two degrees of error (Table 5.1). Three of these six successful algorithms were trained using contralateral or independent population subgroup conditions. This suggests that it may be possible to build a predictive control algorithm that would work for an

exoskeleton user outside of this study, thus removing the need to develop an individual algorithm for each user.

In a practical sense, the contralateral subgroup may be considered representative of a user who lost the full ability of a single leg. The findings of this study suggest that a control algorithm trained using data from the other, healthy leg sufficiently simulates knee flexion necessary to allow an exoskeleton to operate on the impaired leg to increase maneuverability. The independent population subgroup may be considered representative of a generalized algorithm that could be used for any new user. More development would be required for a true generalized algorithm to be created; however, the results indicate promise in such an approach.

Consistent with the first hypothesis, training and testing performed on the same leg was the best performing scenario for every data type (Table 5.1). The ANN algorithms were able to discern the movement patterns of each individual user and associate muscle activation to those patterns. However, as per hypothesis 2, EMG alone, whether normalized or not normalized, was unable to produce errors within the acceptable range. This is likely because muscle activation alone, even during a cyclic action like walking, does not provide enough information to allow for an accurate prediction of joint angles. For example, muscles can theoretically produce similar activation levels throughout the entire range of knee flexion angles. Additionally, the angles outputted by the ANN were not limited to the physical bounds of the knee (e.g. 0-140° of flexion), and in every NR and NNR case, the error greatly exceeded angles that would be safe for human knees. However, by allowing the algorithm to exceed anatomical limits, an accurate assessment of the performance was achieved. If simulated stops were put in place at the bounds, the algorithms may have artificially inflated performance numbers as the bounds could be closer to the measured value than the predicted value.

Regarding the third hypothesis, flexion angle-only based algorithms trained for the ipsilateral or contralateral case generated predictions with less than 2 degrees of RMS error. However, for an independent population, RMS errors were calculated at just over 2 degrees (Table 5.1). With the introduction of "not normalized" EMG signals, however, NNR+FA algorithms were shown to be unsuccessful, on average, for contralateral and independent population conditions based on both the two-degree and ten-degree metrics of error (Table 5.1). The NNR+FA condition did work on average when the training and testing was performed on the same leg, which is in line with previous studies using EMG to predict joint angles for various joints of the arm [56] [58]. Testing on the contralateral and on independent population conditions resulted in 10% of algorithms (both legs for M01 contralateral and both legs for M02 independent population) producing RMS errors so large that the group averages exceeded 10 degrees (Table 5.2). Despite the outliers, the NNR+FA algorithms trained on the independent population showed that nine out of the twenty legs tested were under the two-degree success metric and that 18 were under the ten-degree success metric (Table 5.2). EMG normalization (NR+FA) drastically improved the results, with 17 of the legs under the two-degree metric, all 20 legs under the ten-degree metric, and a max error of 5.3 degrees (Figure 5.2). It is noteworthy that it was especially challenging for researchers to locate lower-limb musculature for subjects M01 and M02 due to higher BMI and less muscular definition. Therefore, sensor placement error likely contributed to increased errors among these particular subjects during not normalized conditions. Nonetheless, these large errors were reduced greatly following EMG normalization (Table 5.2), providing additional support to the value of normalization with this technique.

The Tukey HSD Comparison Test showed 10 overlaps between the tested groups, meaning that 11 from the possible 21 interactions produced statistically significant differences from one

another (Figure 5.3). Ideally, the results would show that algorithms tested on an independent population would not be significantly different from results of the ipsilateral condition because it would suggest that the two sources of training data might be able to be used interchangeably. However, although statistically significant, the differences were small (<0.5° for NR+FA) and may not be practically significant. It is not yet clear if differences of this magnitude would be problematic or if indeed an independent population may be used to train the predictive algorithm used in exoskeleton control since the error still remains quite small.

Although the fourth hypothesis, stating NR+FA would produce algorithms being more accurate than only FA, was not statistically significant for any case, it is possible that an increase in accuracy of ~0.5° for every case could indicate a noticeable increase in synchronization and comfort for the user (Table 5.3). More testing should be completed to fully assess the practical benefits to the user when adding normalized EMG signals.

This study was subject to a number of limitations. For example, the tested action, walking gait, is cyclical and largely universal between subjects. The cyclic nature likely contributed heavily to the predictive ability of the algorithms. It is believed that by testing similar algorithms trained on more complicated movements, such as initiation and conclusion of gait or dynamic lateral turns, errors will increase. As such, this style of training and testing should be expanded to actions less repeatable as the algorithms will then lose the ability to rely heavily on past knee flexion angles and will have to rely more heavily on EMG to predict future joint angles. Less repeatable actions would also provide more information as to ipsilateral, contralateral, and independent population algorithms ability to succeed for real world use.

Another limitation of this study was that all testing was completed theoretically rather than empirically with a user wearing an actual exoskeleton. It is likely that users would change their

muscle firing patterns or walking gait patterns to adapt to delays or tendencies of a worn exoskeleton. The perceived size and weight added by the exoskeleton could also cause the user to alter gait characteristics in unanticipated ways. Exoskeletons that are assistive or rehabilitative can change muscle firing signals based on the level of assistance, with zero muscle activation meaning full assistance [3]. This limitation can be countered in future studies by providing training muscle signals with the user donning the exoskeleton.

A final limitation to this study is the small sample size and the narrow range of demographic characteristics of the subjects. Future work with a greater number of subjects with wider demographics would provide a deeper understanding of the utility of ANN algorithms to predict future knee joint angles in a population at large.

5.5 Conclusion

This study demonstrated the feasibility of employing an EMG and knee flexion angle-based ANN to accurately predict the knee flexion angle for a population independent of the algorithm's training population 50 ms into the future and within two degrees of error. It was found that a simple mean walking EMG normalization greatly increases the chance of success to make accurate knee flexion predictions for an independent population. It was also found that EMG alone is not a sufficient input for training a predictive algorithm, while flexion angle alone could be for cyclic movements such as walking. The results of this work will inform future investigations with the aim of training an algorithm with an independent population to be used to pair a user to an exoskeleton.

Chapter 6

Conclusions and Future Work

Various methods of controlling exoskeletons have been pursued and are evidenced throughout the literature [2] [3] [5] [7] [13] [15]. The use of EMG poses many challenges that, if overcome, could potentially produce an exoskeleton that is able to synchronize with user intent. The work presented in this thesis provides a step towards a fully capable exoskeleton. Understanding that predictions can be made as far as ~100 ms into the future and that a population independent of the population used to generate control algorithms for exoskeleton users may benefit impaired populations, industry workers, and even military soldiers. The improvement include quicker response time of the exoskeleton, shortened or eliminated calibration trials, and better accuracy of angle prediction than current exoskeleton designs.

There are many avenues for future work in this area. The first and foremost opportunity should be focused on expanding the testing cohort. Incorporation of more subjects, actions, or sensors could continue to expand knowledge in this area. Chapter 5 illustrated that algorithms trained strictly from past motion data could perform a repetitive task such as gait with high levels of accuracy. The inclusion of EMG improved upon those results as it tailored the gait to the individuals based on muscle firing patterns better than kinematic information alone. EMG results are anticipated to have an even larger impact for motions that are non-repetitive or non-

symmetrical. Actions that involve altering speeds or directions could use EMG as the main predictor to the upcoming change.

Additional control methods should also be pursued. It should be determined whether or not a single regression algorithm is enough for predicting multiple actions. By incorporating additional actions into a study design, a better framework for the controls algorithm could be developed. Testing should be completed to see if there is enough processing time or power to run a multilayer algorithm that combines both classification of the action and a regression of that chosen movement. In this scenario, a user could begin to lean forward from a stand, and the exoskeleton would classify that the user is initiating gait and then would utilize muscular activation to determine which foot is the intended lead foot. This method would be ideal over a predetermined path after classification.

Future work should also focus on implementing the control algorithms in real time. An exoskeleton knee brace could be built to measure the effects through a serious of stages. First, by simultaneously measuring joint angles of a subject and an exoskeleton through motion capture, testing could be completed such that the exoskeleton could be tested against varying loads without the subject ever donning one. Once a full understanding of the accuracies and delays have been achieved, then metabolic testing should ensue with the subjects operating an exoskeleton.

Finally, future work should involve expanding to more than a single joint. Much of the human body moves synchronously and these outside motions could add increased resilience to a predictive algorithm. Testing should be completed to see if adding the hip, ankle, or contralateral leg joints or muscles into the algorithms would produce greater accuracies in predictions. It could be possible to design a regression algorithm with more than one joint output.

If some of the proposed future work is explored, then this thesis stands to have lain the groundwork for improvements in the understanding of exoskeleton technology and its control.

Bibliography

1. Nussbaum, M.A., et al., *An Introduction to the Special Issue on Occupational Exoskeletons.* IISE Transactions on Occupational Ergonomics and Human Factors, 2019. **7**(3-4): p. 153-162.

2. Gopura, R.A.R.C., et al., *Developments in hardware systems of active upper-limb exoskeleton robots: A review.* Robotics and Autonomous Systems, 2016. **75**: p. 203-220.

3. Lee, H., P.W. Ferguson, and J. Rosen, *Lower Limb Exoskeleton Systems—Overview*, in *Wearable Robotics*. 2020, Elsevier. p. 207-229.

4. Collins, S.H., M.B. Wiggin, and G.S. Sawicki, *Reducing the energy cost of human walking using an unpowered exoskeleton.* Nature, 2015. **522**(7555): p. 212-215.

5. Bi, L., A.G. Feleke, and C. Guan, *A review on EMG-based motor intention prediction of continuous human upper limb motion for human-robot collaboration.* Biomedical Signal Processing and Control, 2019. **51**: p. 113-127.

6. Dollar, A.M. and H. Herr, *Lower Extremity Exoskeletons and Active Orthoses: Challenges and State-of-the-Art.* IEEE Transactions on Robotics, 2008. **24**(1): p. 144-158.

7. Young, A.J. and D.P. Ferris, *State of the Art and Future Directions for Lower Limb Robotic Exoskeletons.* IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2017. **25**(2): p. 171-182.

8. Suzuki, K., et al. *Intention-based walking support for paraplegia patient*. in *2005 IEEE International Conference on Systems, Man and Cybernetics*. 2005. IEEE.

9. Esquenazi, A., et al., *The ReWalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury.* American journal of physical medicine & rehabilitation, 2012. **91**(11): p. 911-921.

10. Baunsgaard, C.B., et al., *Gait training after spinal cord injury: safety, feasibility and gait function following 8 weeks of training with the exoskeletons from Ekso Bionics.* Spinal cord, 2018. **56**(2): p. 106.

11. Zoss, A.B., H. Kazerooni, and A. Chu, *Biomechanical design of the Berkeley lower extremity exoskeleton (BLEEX).* IEEE/ASME Transactions on Mechatronics, 2006. **11**(2): p. 128-138.

12. Jacobsen, S.C., et al., *Research robots for applications in artificial intelligence, teleoperation and entertainment.* The International Journal of Robotics Research, 2004. **23**(4-5): p. 319-330.

13. Bogue, R., *Exoskeletons and robotic prosthetics: a review of recent developments.* Industrial Robot: an international journal, 2009.

14. Kuo, A.D., *A mechanical analysis of force distribution between redundant, multiple degree-of-freedom actuators in the human: Implications for the central nervous system.* Human movement science, 1994. **13**(5): p. 635-663.

15. Zaroug, A., et al., *Overview of Computational Intelligence (CI) Techniques for Powered Exoskeletons*, in *Computational Intelligence in Sensor Networks*, B.B. Mishra, et al., Editors. 2019, Springer Berlin Heidelberg: Berlin, Heidelberg. p. 353-383.

16. Englehart, K. and B. Hudgins, *A robust, real-time control scheme for multifunction myoelectric control.* IEEE Transactions on Biomedical Engineering, 2003. **50**(7): p. 848-854.

17. Petrella, R., P. Lattanzio, and M. Nelson, *Effect of age and activity on knee joint proprioception1.* American Journal of Physical Medicine & Rehabilitation, 1997. **76**(3): p. 235-241.

18. Kennedy, S., et al., *Vibratory Response Characteristics of High-Frequency Shape Memory Alloy Actuators.* Journal of Vibration and Acoustics, 2020. **142**(1).

19. Nemat-Nasser, S. and W.-G. Guo, *Superelastic and cyclic response of NiTi SMA at various strain rates and temperatures.* Mechanics of materials, 2006. **38**(5-6): p. 463-474.

20. Huo, W., et al., *Lower limb wearable robots for assistance and rehabilitation: A state of the art.* IEEE systems Journal, 2014. **10**(3): p. 1068-1081.

21. Yamamoto, K., et al., *Development of power assisting suit.* JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing, 2003. **46**(3): p. 923-930.

22. Yamamoto, K., et al. *Stand alone wearable power assisting suit - sensing and control systems*. in *RO-MAN 2004. 13th IEEE International Workshop on Robot and Human Interactive Communication (IEEE Catalog No.04TH8759)*. 2004.

23. Kazerooni, H., A. Chu, and R. Steger, *That which does not stabilize, will only make us stronger.* The International Journal of Robotics Research, 2007. **26**(1): p. 75-89.

24. Novak, D. and R. Riener, *A survey of sensor fusion methods in wearable robotics.* Robotics and Autonomous Systems, 2015. **73**: p. 155-170.

25. Cavanagh, P.R. and P.V. Komi, *Electromechanical delay in human skeletal muscle under concentric and eccentric contractions.* European Journal of Applied Physiology and Occupational Physiology, 1979. **42**(3): p. 159-163.

26. Hermens, H.J., et al., *Development of recommendations for SEMG sensors and sensor placement procedures.* Journal of electromyography and Kinesiology, 2000. **10**(5): p. 361-374.

27. Hermens, H.J., et al., *European recommendations for surface electromyography.* Roessingh research and development, 1999. **8**(2): p. 13-54.

28. Hof, A.L. and J. Van den Berg, *EMG to force processing I: An electrical analogue of the hill muscle model.* Journal of Biomechanics, 1981. **14**(11): p. 747-758.

29. De Luca, C.J., *Myoelectrical manifestations of localized muscular fatigue in humans.* Critical reviews in biomedical engineering, 1984. **11**(4): p. 251-279.

30. Khokhar, Z.O., Z.G. Xiao, and C. Menon, *Surface EMG pattern recognition for real-time control of a wrist exoskeleton.* BioMedical Engineering OnLine, 2010. **9**(1): p. 41.

31. Phinyomark, A., P. Phukpattaranont, and C. Limsakul, *Feature reduction and selection for EMG signal classification.* Expert Systems with Applications, 2012. **39**(8): p. 7420-7431.

32. De Luca, C.J., *The use of surface electromyography in biomechanics.* Journal of applied biomechanics, 1997. **13**(2): p. 135-163.

33. Mathiassen, S., J. Winkel, and G. Hägg, *Normalization of surface EMG amplitude from the upper trapezius muscle in ergonomic studies—a review.* Journal of electromyography and kinesiology, 1995. **5**(4): p. 197-226.

34. Burden, A., M. Trew, and V. Baltzopoulos, *Normalisation of gait EMGs: a re-examination.* Journal of Electromyography and Kinesiology, 2003. **13**(6): p. 519-532.

35. Lehman, G.J. and S.M. McGill, *The importance of normalization in the interpretation of surface electromyography: a proof of principle.* Journal of manipulative and physiological therapeutics, 1999. **22**(7): p. 444-446.

36. Sousa, A.S. and J.M.R. Tavares, *Surface electromyographic amplitude normalization methods: a review.* Electromyography: new developments, procedures and applications, 2012.

37. Allison, G., R. Marshall, and K. Singer, *EMG signal amplitude normalization technique in stretch-shortening cycle movements.* Journal of Electromyography and Kinesiology, 1993. **3**(4): p. 236-244.

38. Knutson, L.M., et al., *A study of various normalization procedures for within day electromyographic data.* Journal of electromyography and kinesiology, 1994. **4**(1): p. 47-59.

39. Tung, W.Y.-W., et al., *Design of a Minimally Actuated Medical Exoskeleton With Mechanical Swing-Phase Gait Generation and Sit-Stand Assistance*. 2013.

40. Talaty, M., A. Esquenazi, and J.E. Briceno. *Differentiating ability in users of the ReWalk TM powered exoskeleton: An analysis of walking kinematics*. in *2013 IEEE 13th international conference on rehabilitation robotics (ICORR)*. 2013. IEEE.

41. Neuhaus, P.D., et al. *Design and evaluation of Mina: A robotic orthosis for paraplegics*. in *2011 IEEE International Conference on Rehabilitation Robotics*. 2011. IEEE.

42. Meuleman, J., et al., *LOPES II—design and evaluation of an admittance controlled gait training robot with shadow-leg approach.* IEEE transactions on neural systems and rehabilitation engineering, 2015. **24**(3): p. 352-363.

43. Aoyagi, D., et al. *An assistive robotic device that can synchronize to the pelvic motion during human gait training*. in *9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005.* 2005. IEEE.

44. Grosu, V., et al., *Design of smart modular variable stiffness actuators for robotic-assistive devices.* IEEE/ASME Transactions on Mechatronics, 2017. **22**(4): p. 1777-1785.

45. Koller, J.R., C.D. Remy, and D.P. Ferris. *Comparing neural control and mechanically intrinsic control of powered ankle exoskeletons*. in *2017 International Conference on Rehabilitation Robotics (ICORR)*. 2017.

46. Liu, D.-X., et al., *Gait phase recognition for lower-limb exoskeleton with only joint angular sensors.* Sensors, 2016. **16**(10): p. 1579.

47. Mattioli, F.E.R., et al. *Classification of EMG signals using artificial neural networks for virtual hand prosthesis control*. in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. 2011.

48. Joshi, C.D., U. Lahiri, and N.V. Thakor. *Classification of gait phases from lower limb EMG: Application to exoskeleton orthosis*. in *2013 IEEE Point-of-Care Healthcare Technologies (PHT)*. 2013. IEEE.

49. Borbély, B.J. and P. Szolgay, *Real-time inverse kinematics for the upper limb: a model-based algorithm using segment orientations.* Biomedical engineering online, 2017. **16**(1): p. 21.

50. Koike, Y. and M. Kawato, *Estimation of dynamic joint torques and trajectory formation from surface electromyography signals using a neural network model.* Biological cybernetics, 1995. **73**(4): p. 291-300.

51. Ding, Q.C., et al. *A novel EMG-driven state space model for the estimation of continuous joint movements*. in *2011 IEEE International Conference on Systems, Man, and Cybernetics*. 2011.

52. Cavallaro, E.E., et al., *Real-Time Myoprocessors for a Neural Controlled Powered Exoskeleton Arm.* IEEE Transactions on Biomedical Engineering, 2006. **53**(11): p. 2387-2396.

53. Chan, A.D. and K.B. Englehart, *Continuous myoelectric control for powered prostheses using hidden Markov models.* IEEE Transactions on Biomedical Engineering, 2004. **52**(1): p. 121-124.

54. Kilicarslan, A., et al. *High accuracy decoding of user intentions using EEG to control a lower-body exoskeleton*. in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2013.

55. Kim, P., *Matlab deep learning.* With Machine Learning, Neural Networks and Artificial Intelligence, 2017. **130**: p. 21.

56. Zhang, Q., et al., *Simultaneous and continuous estimation of shoulder and elbow kinematics from surface emg signals.* Frontiers in neuroscience, 2017. **11**: p. 280.

57. Aung, Y.M. and A. Al-Jumaily, *Estimation of upper limb joint angle using surface EMG signal.* International Journal of Advanced Robotic Systems, 2013. **10**(10): p. 369.

58. Liu, J., et al., *EMG-based continuous and simultaneous estimation of arm kinematics in able-bodied individuals and stroke survivors.* Frontiers in neuroscience, 2017. **11**: p. 480.

59. Baum, E.B. and F. Wilczek. *Supervised learning of probability distributions by neural networks*. in *Neural information processing systems*. 1988.

60. Andriacchi, T., et al., *A point cluster method for in vivo motion analysis: applied to a study of knee kinematics.* 1998.

61. Wu, G. and P.R. Cavanagh, *ISB recommendations for standardization in the reporting of kinematic data.* Journal of biomechanics, 1995. **28**(10): p. 1257-1261.

62. Grood, E.S. and W.J. Suntay, *A joint coordinate system for the clinical description of three-dimensional motions: application to the knee.* 1983.

63. Fleischer, C. and G. Hommel, *A Human--Exoskeleton Interface Utilizing Electromyography.* IEEE Transactions on Robotics, 2008. **24**(4): p. 872-882.