1 | Background

The background chapter serves the purpose of providing a theoretical overview of the techniques applied to deal with the proposal of improving users' ability to operate a myoelectric prosthesis by training the user with the novel approach of confidence score feedback. The chapter will cover the usual applied methods of pattern recognition based myoelectric prosthesis control. The idea behind myoelectric prosthetic control is to convert muscles signals (EMG signals), recorded from a user when performing a muscle contraction, into a movement performed by a prosthesis. The EMG signals are used to train a control system in recognizing a pattern in the EMG signals from different muscle contractions. The control system then decides which movement the prosthesis should perform, based on which pattern in the EMG signals that is recognized. The focus of this project is to investigate the user's ability to adapt to how the control system wants the user to perform a muscle contraction for the desired movement to be performed by the prosthesis. The different processes of myoelectric prosthetic control the background chapter will cover are: the mechanics of the movements the control system will be trained to recognize, the generation of EMG, data acquisition, data processing, feature extraction, classification and control output. The pipeline of this process can be seen in figure 1.1. Furthermore, the chapter will cover theory on how confidence scores are calculated, how user training has been used in previous studies and how real-time prosthesis control is evaluated.

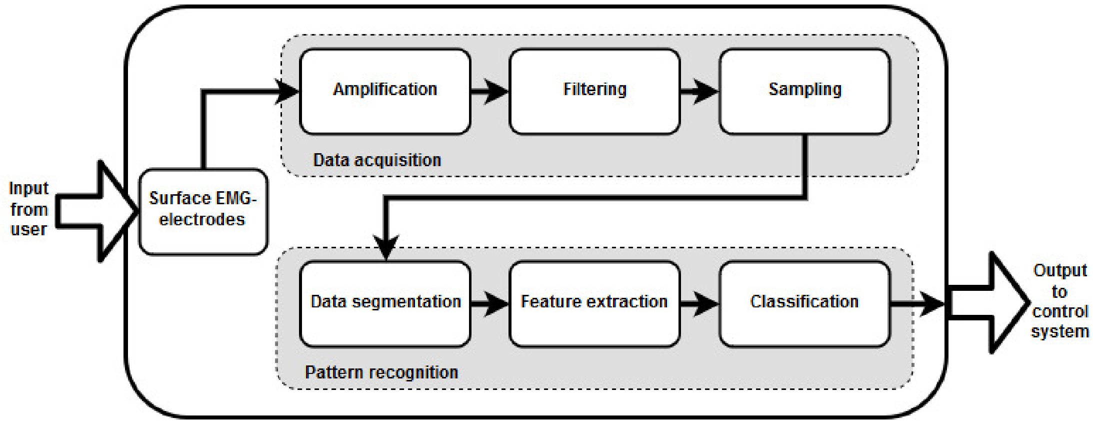


Figure 1.1: The figure shows the pipeline for myoelectric prosthetic control. The EMG signal from the user is first detected by the surface electrodes, after which it is amplified and filtered before it is sampled to process it digitally. To produce a control output the signal is subsequently segmented in windows from which features are extracted that are used to classify which movement has been made, and thus which movement should be performed by the prosthesis. The pipeline is adapted from [1]

1.1 Anatomy of the human lower forearm

This project will use six movements for control of a virtual interface and visual feedback. The movements are extension, flexion, radial and ulnar deviation, closed and opened hand as well as rest. The movements are shown in figure 1.2. These movements are very distinguishable from each other and therefore very useful in classification. Training users to improve performance of these movements for use in a myoelectric prosthesis control scheme, could give a good foundation to build a classification scheme upon.

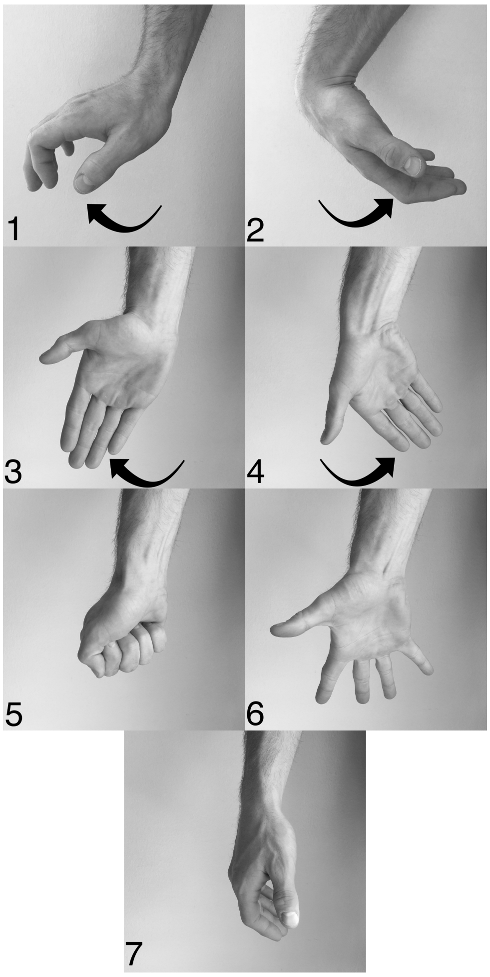


Figure 1.2: The figure shows the six hand movements used in this study as well as rest. The movements are: 1) extension, 2) flexion, 3) radial deviation, 4) ulnar deviation, 5) closed hand, 6) opened hand and 7) rest.

To perform movements of the hand and fingers and at the wrist, many muscles in the lower arm are active. All movements relevant for this study are controlled by muscles in the lower forearm, as well as some in the palm of the hand. However, as EMG recordings will be done from the lower forearm by equipment described in section 1.3.1, it is relevant to gain knowledge on the muscles in the lower forearm. When performing actions at the wrist the following muscles are active: flexor carpi radialis, flexor carpi ulnaris, palmaris longus, extensor carpi radialis longus, extensor carpi radialis brevis, extensor carpi ulnaris. The flexor and extensor muscles are naturally responsible for performing flexion and extension respectively at the wrist. They are however also responsible for performing radial and ulnar deviation, where the flexor carpi radialis and extensor carpi radialis brevis muscles, which are antoganistic muscles when doing flexion and extension, will work together when performing radial deviation. The extensor carpi radialis longus muscle is also responsible for performing radial deviation. The flexor and extensor carpi ulnaris muscles are responsible when performing ulnar deviation. The palmaris longus muscle is only active during flexion at the wrist. [2]

Several more muscles are further specified to perform movements of the fingers but many of these are also active during extension/flexion and radial/ulnar deviations at the wrist. Muscles responsible when opening the hand by extending the fingers are: extensor digitorum, extensor pollicis brevis, extensor pollicis longus, extensor indicis and the extensor digiti minimi muscle. Contrary, the muscles responsible for closing the hand by flexing the fingers are: flexor digitorum superficialis, flexor digitorum profundus and the flextor pollicis longus. [2]

1.2 Electromyography

This project will utilize the method of electromyography to record the muscle activation of the lower arm muscles in relation to the gestures presented in section 1.1 on anatomy. To develop theoretical background knowledge, a short introduction of the essentials of the signal will be presented.

Electromyography is the recording of muscle activity. The amount of activity is found by measuring the electric potential, an action potential triggering a muscle contraction. The process of planning and executing a voluntary movement starts at the motor cortex in the brain, where a nerve impulse is send and travel through the spinal cord to the lower motor neuron. As seen in figure 1.3 the path from alpha motor neuron through the axon to the motor endplates is what makes up a motor unit. The alpha motor neuron originates from the spinal cord along the axon to the muscle it controls. The axon branches out to multiple muscle fibers through motor endplates innervating the muscle fibers.

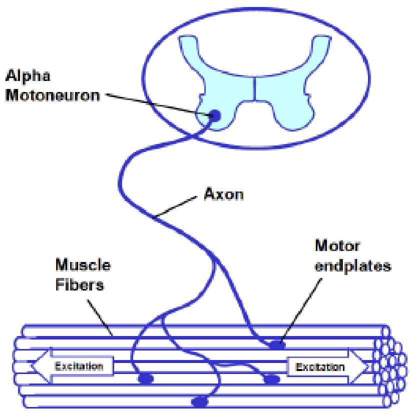


Figure 1.3: The figure illustrates the neural pathway from the alpha motor neuron to the innervated muscle fibers, making up a motor unit.[3]

The essentials of understanding the application EMG is the excitation of muscles. Muscles contract through a series of steps of changing potentials across muscle cell membranes and rapid polarization and repolarization. However, when recordings EMG it is the spread of the motor unit action potential (MUAP) over the muscle membrane that is recorded. [4] Muscles are innervated by a varying number of nerves depending on the individual muscle. The MUAP is conducted to the muscle by nerves from the spinal cord, with the nerve impulses originating from the motor cortex in the brain. Muscles are not activated randomly by individual nerve fibers, but by nerves sorted into motor units. Many motor units are attached to a muscle and consist of a number of the nerves innervating the muscle. An illustration of how one motor unit attach to the muscle fibers of a muscle is illustrated in figure 1.3. When a motor unit activate, all the nerves in the motor unit is activated. This enables a controlled activation of the muscle as well as a activation that reach a higher number of muscle fibers. Motor units are also activated in an asynchronous pattern which enables different muscle fibers to be active at different times, making muscles less prone to fatigue. The force of a muscle contraction can be modulated either by motor unit recruitment or be frequency of activation. In EMG it is the sum of activity of activate motor units that is recorded. [4]

In the scenario of this project multiple EMG electrodes will record signals from many muscles in the lower forearm. This will result in some muscles being very active during some movements as they contract, while other muscles will be inactive, as described in section **??** on antoganistic muscles. On an EMG recording this will be visible as contracting muscles will show high activity, while others will show little to no activity. An illustration of this can be seen on figure 1.4. The figure shows the muscle activity of muscles in the forearm when performing first extension at the wrist followed by flexion. As can be seen the muscle activity is very different between the two muscles. This enables to recognize specific movements based on several EMG recordings with several different electrode placement.

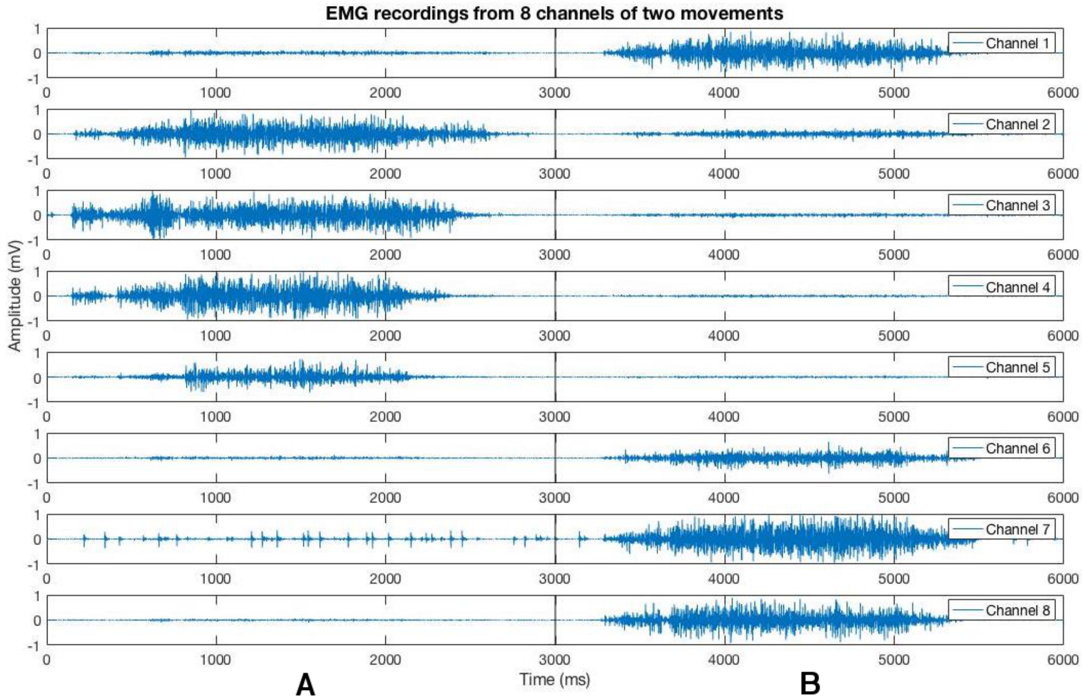


Figure 1.4: Illustration of the activity in an EMG recording of two movements: extension and flexion. Left side A) shows the activity recorded by EMG electrode channels during extension of the hand. Right side B) shown activity during flexion of the hand.

Recording EMG can be done either through the most often used surface EMG (sEMG) or by intramuscular EMG (iEMG). In iEMG a needle is inserted into the muscle measuring the MUAP directly on site. The more often used sEMG uses electrodes to measure the sum of MUAPs on the skin surface will be used to acquire EMG signals in this project. [4]

1.3 Data acquisition

For a myoelectric prosthetic control system to be able to recognize hand movements it needs to be given prior information on how the movements looks like represented as a EMG-signal - this is also called training the control system. Thus, EMG data needs to be acquired from the user and used to train the control system. The following section describes which types of EMG acquisition techniques that are commonly used.

As presented earlier in section 1.2 the source of the EMG signal is motor unit action potentials. The energy generated in action potentials is of a very small size and is measured in microvolts. Very sensitive recording equipments is therefore key in doing electromyography. It is essential to consider the type of electrode intended to use. Electrodes come in various different sizes and shapes and are therefore very depended on the intended measurement site. Typically electrodes made of silver-impregnated plastic are used. They present desired characteristics by being disposable, relatively low price and by having low impedance with the skin. Most electrodes are covered with an adhesive compound in order for them to stick to the skin. These can either be dry or covered with different types of gel, in order to reduce impedance and thereby noise, getting a more accurate EMG recording. Dry electrodes do not use gel, but instead rely on the skin to sweat and thereby decreasing the skin impedance. Dry electrodes should prove better to patients with sensitive skin. Different skin conditions may also effect the electrode-skin impedance. Make-up or hairy skin increase the impedance, thus the recording site should be prepared by removing hair or cleaning the skin with alcohol wipes. [4] The following section will introduce the choice of acquisition device used in this project.

1.3.1 Myo armband

In this project the Myo armband (MYB) from Thalmic Labs will be used for EMG data acquisition. MYB is an electrode armband with eight dry stainless steel electrode-pairs around the inside of the armband, as depicted in figure 1.5. The advantage of dry electrodes is that they do not need to be disposed after usage as conventional gelled EMG-electrodes [4]. In addition, the MYB can communicate wirelessly to a computer via Bluetooth 4.0 [5]. Thus, making it a fast and easy usable device to use both during pilot-testing and for the final experiment. In the following section more information about the MYB will be presented.

MYB records EMG data in a 8-bit resolution. As usual when recording EMG the higher the performed contraction is, the higher the values in the output will be. To avoid interference from power lines a 50 Hz notch filter is built-in in the MYB from the manufacturer. However, the MYB is not able to make any further filtering, therefore this will be implemented later during signal processing described further in section 1.4. The MYB has a 200 Hz sample rate, and thus samples with a lower bandwidth than the EMG spectrum consists of, which is between 10-500 Hz [4]. Using MYB will likely result in an aliased EMG signal and confinement in using features representing the frequency information the signal. Besides having EMG sensors the MYB can provide position and orientation information, using its three inertial measurement units consisting of a three axis gyroscope, a three axis magnetometer and a three axis accelerometer. This inertial information is sampled at 50 Hz. [5]
When initiating the wearing of the armband there is two calibration phases the user must follow before the armband is ready to use - the warm-up phase and the sync phase. During the warm-up phase the armband is ensuring as strong electrical connection with the muscles in the forearm as possible. This is mainly provided by light sweating on the skin under the electrodes, which improve the connection similar to electrode gel [4]. During the sync phase, the armband determines its orientation in space, position and on which arm it is placed. The MYB works most optimal when fitted tightly on the thickest part of the forearm. For users with smaller forearms a set of clips can be added for the armband to get a constrained grip. [5]

Figure 1.5: MYB from Thalmic Labs. Sensor one will always correspond to the first index in the EMG data output array, and sensor two to the second index and so forth.

When using the MYB as method for acquiring data this project will use non-specific electrode placement. The idea of the MYB is to have an easy to use devise, and specific electrode placement is complicated as it requires knowledge on the specific anatomic placement of muscles in the forearm. This project has covered which muscles are active for the chosen movements, in section 1.1, but for the knowledge of the source of the EMG signals. When the MYB is placed correctly on the forearm the need for specific electrode placement is defeated, because of the principle of different muscle activation during different movements, illustrated in figure 1.4.

1.4 Data processing

In order to use the acquired EMG-signal in myoelectric prosthesis control, it first has to be processed, which is referred to as pre-processing, as it is a processing procedure that is done before further processing is performed. Since the acquisition and most processing is done in the MYB before Bluetooth transmission, further processing of the signal is moderate. In myoelectric prosthesis control several features are extracted from each electrode-channel for use in control, instead of using the raw EMG-signal. Thus, the amount of information given to the control system is increased compared to only providing the raw EMG-signal to the control system. The following two sections will briefly describe theory behind filtering and feature extraction in relation to this project.

1.4.1 Filtering

Filtering is a cornerstone in preparing an EMG-signal for any kind of use. The frequency spectrum of EMG is 10 Hz to 500 Hz. [4]. According to the Nyquist theorem, to achieve a loss-less representation of the signal the sampling frequency must be at least twice the maximum frequency of interest of the original signal [6]. Besides sampling with twice the maximum frequency, EMG is sensitive to artifacts of movement and electrical interference. Due to these circumstances, filters are often implemented to remove these unwanted contributors [7]. General practice in filtering the EMG-signal will include implementing a notch filter with very narrow width and steep slope, at frequencies 49-51 Hz or 59-61 Hz depending on the power supply. The intent is to remove any electrical interference noise. In the low frequency spectrum several recommendations (5 Hz, 10 Hz and 20 Hz) has been made for optimal corner frequency of a high pass filter to remove movement artefacts. A low pass filter is also typically used to remove any noise and unwanted signal above 500 Hz [4].

This project will utilize a MYB for data acquisition and as mentioned in section 1.3.1 the MYB has a sample rate of 200 Hz. In relation to this project a sampling frequency of at least twice the maximum of the recorded signal is not possible, since muscles of the forearm have a maximum frequency of 400-500 Hz [4]. This would require a sample rate of at least 1000 Hz, which cannot be achieved due to limitations in the MYB. Under other circumstances it would be astute to implement an anti-aliasing filter, this however is not possible with the MYB since an anti-aliasing filter should be implemented before the sampling. As this happens inside the MYB as fabricated by Thalmic Labs it is impossible to change for this projects.

1.4.2 Feature extraction

The raw EMG-signal is not itself used for myoelectric prosthesis control, but features that are extracted from it. Thus, increasing the amount of information given to the control system, which facilitates more robust pattern recognition.
There are numerous feature components from an EMG signal which can be extracted either from the time-domain, frequency-domain, or time-frequency domain. Most used are features from the time- and frequency-domain. Time-domain features can be categorized in five different types based on their mathematical properties: energy information, complexity information, frequency information, prediction modelling and time-dependency. Extracting features from the frequency-domain requires a frequency transformation, calculating the spectral properties of the recorded signal, which takes up longer processing time than simply using time-domain features. Time-domain features are often chosen based on their quick and easy implementation as they do not require any transformation before extraction and are calculated based on the raw EMG-signal. In addition, it is important not to choose redundant features for the classifier which would be to chose features providing similar information. [8]

Extracting features for real-time prosthesis control is done by taking segments of the continuous signal, called windows. Calculation on extracting features are done in these discreet windows. This is done instead of using the instantaneous value due to the signals random nature. These windows are often overlapped to create a dense information stream for extraction. The relationship between window and overlap length is significant, when trying to determine the best representation. The window length is a matter of getting enough samples to do the calculation, but too long a window will result in delays slowing the control. Smith et al. [9] found that the optimal window length in a classification control scheme that enables best performance ranges from 150-250 ms. Overlapping the window is a way to faster acquire windows by reusing a determined last segment of the prior window. The amount of overlap is a compromise between classification quality and responsiveness of the prosthesis; a large overlap will provide a shorter output delay, but a worse classification and vice versa [10].

1.5 Classification

For a myoelectric prosthesis to be able to distinguish between movements it needs to perform, a control scheme is needed to categorize the movements. The control scheme is trained by being given information about the EMG signal represented as the features extracted from the raw EMG. If the features between each movement are well separated the control scheme is able to recognize each distinct movement. For this purpose classification control schemes are commonly used. A classifier categorizes each movement as a class, and based on the input features it gives one output for each data window. Using a classifier thus limits the user to only performing movements which have been defined as classes. However, if trained properly a classifier can reach a low error rate for the trained movements [11]. A frequently used classification control scheme for myoelectric prosthetic control is the Linear Discriminant Analysis classifier (LDA) [12, 13, 14, 11]. The advantage of LDA is that whilst having a low computation time it still enables robust control. An assumption about the LDA is that the input needs to be Gaussian distributed, which the EMG probability properties has shown to adhere to [15, 16]. The following section provides further theoretical information about general linear classifiers and LDA.

1.5.1 Linear Classification

A linear classifier is a supervised classification method used to separate classes of data by linear decision boundaries. Each decision boundary is a hyperplane from which the distance to each feature value of a class and the center of the class is maximized. A decision boundary is defined as a linear combination of the feature values x and is given as [15]:

$$g(x) = w^t x + w_0 \tag{1.1}$$

where w is a weight vector deciding the orientation of $g(x)$, and w_0 is a bias deciding the position of the hyperplane in relation to the origin. If $w_0 > 0$ the origin is on the positive side of the decision boundary, and if $w_0 < 0$ the origin is on the negative side. In the case of $w_0 = 0$ the decision boundary passes through origin. The distance from origin to the boundary is given as $\frac{w_0}{||w||}$. The position of the decision boundary is necessary to know to when separating features into regions. [15]

In a two category case the decision rule for deciding classes is to decide class C_1 if $g(x) > 0$ and class C_2 if $g(x) < 0$. $g(x) = 0$ then defines the decision boundary that separates the features into two decision regions R_1 for C_1 and R_2 for C_2. The normal vector v is orthogonal to any vector on the hyperplane, which is used to calculate the distance r from feature values (x) to the decision boundary [15]:

$$r = \frac{g(x)}{||v||} \tag{1.2}$$

The distance from origin and boundary to feature value (x) is needed to decide in which region the feature value belongs. [15] These distances are illustrated in figure 1.6.

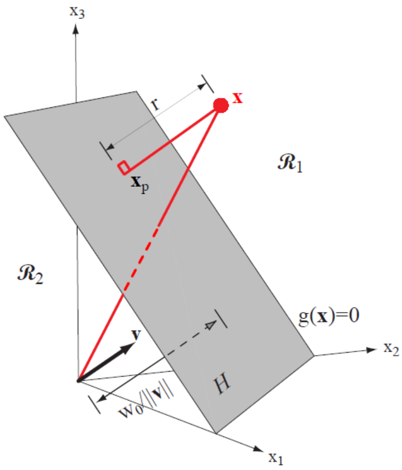


Figure 1.6: A geometric illustration of the linear decision boundary $g(x)$ that separates the feature space into two decision regions R_1 and R_2. x is the feature value, and x_p is the point on the decision boundary in which x is orthogonal projected on vector v. The distances from origin and boundary to feature value x is marked red. [15]

When feature values are to be classified into more than two classes more decision boundaries are needed. This is a multicategory case in which c numbers of boundaries are defined. When defining linear boundaries in this case any number can be chosen, but to minimize ambiguous decision regions the boundaries are defined by [15]:

$$g_i(x) = w_i^t x_i + w_{i0} \qquad i = 1, ..., c, \tag{1.3}$$

This equation follows the notation of the two-category case, with the addition of i numbers of boundaries, feature values and biases. This type of classifier is called a linear machine, dividing the feature space into c regions. A liner machine will be adopted as classification method in this project. Regions R_i and R_j, that are connected is divided by a boundary hyperplane H_{ij} defined by [15]:

$$g_i(x) = g_j(x) \tag{1.4}$$

Often regions are contiguous and will have a single boundary to separate several regions. [15] Illustrations of this case can be seen on figure 1.7.

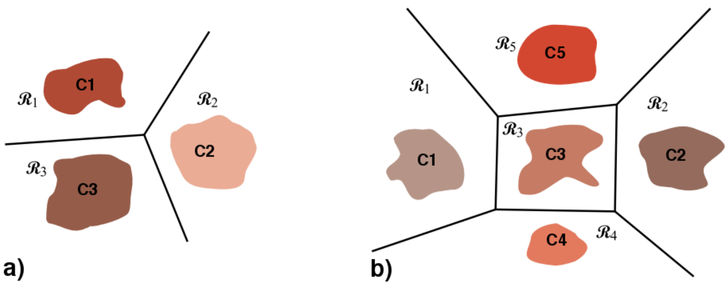


Figure 1.7: A three class (a) and five class (b) case each respectively separated by one decision boundary linear machine. [15]

When the decision boundaries $g_i(w)$ have been calculated as in equation (1.3), the input feature values can be decided upon which class they belong to by calculating the distance to the decision boundary as in equation (1.2).

1.5.2 Linear Discriminant Analysis

A type of linear classifier is the linear discriminant analysis (LDA). Despite the name LDA is a generative model and not a discriminative model. [15] When using LDA the same procedures described in the previous section, section 1.5.1, must be completed. This however can result in classes which can show correlation. When applying LDA, classes cannot have correlation, since the data must be Gaussian distributed. Thus, when applying LDA a constraint of Gaussian distribution is put on the data. This force the class regions to have no correlation and have the same covariance matrix, where only the class means will vary. If class regions are correlated, the decision boundaries made by a discriminative linear classifier are quadratic. Thus, when using LDA and forcing the constraint of Gaussian distribution to the data enabling the decision boundaries to be linear. Additionally, with linear boundaries it is possible to calculate the likelihood of a sample value belonging to a certain class. [17, 15] The proposed method of this study is to calculate confidence scores to use as a means of providing user feedback. This will be made possible by implementing LDA as a linear classifier to enable both a classification based control scheme and to be able to calculate confidence scores to use as feedback in user training.

1.5.3 Classification confidence scores

As described, the additional reason for using LDA as control scheme is because it enables the calculation of confidence scores for the classes, which will be used in the user training, further described in section 1.7 to improve the prosthesis control. The principle of calculating confidence scores for input data is the basis for this project. Based on the classification of feature values by the linear classifier, confidence scores for the classes can be evaluated by computing the posterior probability of each class. Calculating the posterior probability is possible by knowing the likelihood $P(x|C_j)$ and the prior probability $P(C)$. The posterior probability for a class is a value between 0 and 1, and is calculated as follows:

$$P(C_j|x) = \frac{P(x|C_j)P(C)}{P(x)} \tag{1.5}$$

where C_j represents a class and x represents a feature vector. The posterior probability is given as the product of the class conditional probability, $P(x|C_j)$ and the prior probability $P(C)$ divided by a normalization term $P(x)$ that guaranties that the posterior probabilities for all classes sums to one. $P(x|C_j)$ is the probability of obtaining a feature value when selecting samples randomly from a class. $P(C)$ is the probability of a sample from a specific class appears in its correct class, before it have actually appeared. Summation of posterior probabilities for all classes will equal 1.

1.6 Linear regression methods

Classification can be used together with regression methods to provide a combination of the two in a control scheme. The output from the LDA classifier can be set to only decide on which movement is performed, and not at which contraction level the muscles used in the given movement are performing. Thus, the prosthesis can not perform any movement if only using the classification output. In statistics linear regression is often used to determine relations between variables. This notion can also be applied for myoelectric prosthetic control. While classification only provides an output on which class is recognized, a linear regression model provides a continuous output value based on the input value. If the regression model is fitted with information on different contraction levels for a given movement, control proportional to the contraction level will be achieved [18, 19, 20]. In the overall control scheme the classifier can then be used to decide which movement is performed, and a regression model can decide at which contraction level the movement is performed at. Similarly as with the classifier, regressors needs to be trained based on data acquired from the user, where the features extracted from the raw EMG signal is used as input. This procedure is described in the following section.

Different models of linear regression exist to account for different uses. When utilizing regression methods it must be considered what the input and output variables are what type of relation these variables might have. The appropriate regression model must then be applied. Simple linear regression approximate a relation between one dependent variable Y and one independent variable X [21]:

$$Y = \alpha + \beta X + \epsilon \tag{1.6}$$

where Y is the control output for the prosthesis, X is the feature extracted from the EMG signal, β is the regression coefficient in the sampled population, ϵ is the error, and α is the predicted value of Y at $X = 0$. This model can be expanded to estimate relations between one dependent variable and several independent variables. This is called multivariate regression and expands on the equation of simple linear regression, given in equation (1.6) [21]:

$$\hat{Y} = \alpha + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_i X_i + \epsilon_i \tag{1.7}$$

where i in this project would correspond to the number of channels in the MYB [21]. Since this regression model approximates the relation between several independent variables and one dependent variable, this model can be used as a control scheme in myoelectric prosthetics. Here the channel-recordings of muscle activity can be considered independent variables, and used to estimate one control output, which would be the dependent variable. [20]

1.7 User training

As user training in relation to prosthetic control is the main focus of this project an understanding of this concept in relation to receiving a prosthetic device is of great importance. Therefore the following section will cover an introduction to the concept of user training and its importance when preparing a subject to receive a prosthetic device. In addition some of the prior techniques of conducting user training will be presented, facilitating the possibility of assembling a user training protocol based on the most recent and cutting edge results.

When fitting an amputee with a prosthesis, the way the prosthesis is controlled is important. A lot of work lies both ahead and behind fitting a person with a prosthesis. When developing and manufacturing a prosthesis two concepts emerge, one being system training and the other being user training. System training is training the control system to be able to recognize and differ movements based on the EMG-signal being fed to the system. [22] User training on the other hand focuses on training the user in performing distinguishable movements which can be recognized by the control system. Here different types of feedback can be used to inform the user on how well the user performs a movement or how well the system recognizes the users performed movements. [23, 24]

Only few studies have earlier explored the optimal way of giving visual feedback in user training [25]. In a 2014 study Powell et al. [23] provided the user with real-time visual feedback of a virtual prosthetic. This type of feedback is similar to the visual feedback a prosthesis user would receive using a normal prosthesis, although without the sensory feedback of the weight of the prosthesis. An illustration of the setup used in [23] can be seen in figure 1.8.

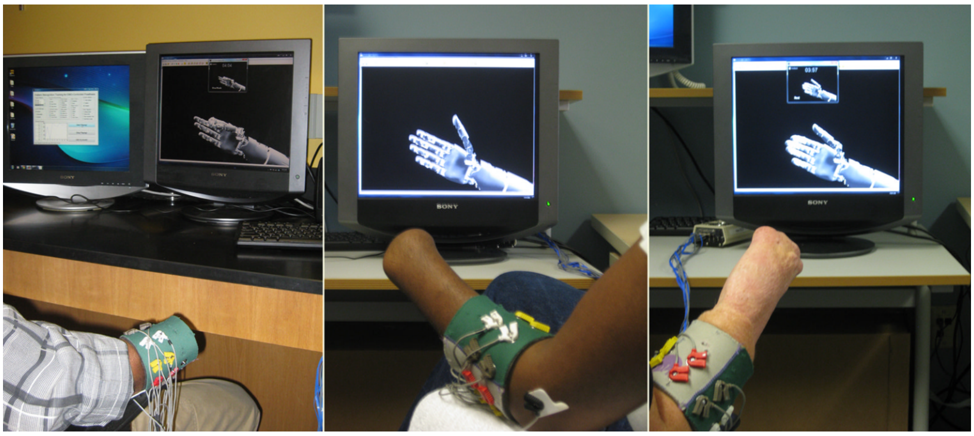


Figure 1.8: Illustration of the experimental setup used in [23]. Initially the user tries to mimic the movement, shown by a prosthesis in the interface, with their phantom limb, thus activating muscles in their residual limb corresponding to the instructed movement. The EMG produced is recorded and used to train a control system. The control system then enables the user to control the prosthesis in the interface, and receive feedback on which movement is performed.

Pan et al. [26] provided a visual feedback of an arrow to be moved on a 2D plane. The arrow was controlled by two DOF's; one controlled the horizontal position of the arrow, while the other could rotate the arrow [26]. Fang et al. [14] provided real-time visual feedback of subjects performed movement in relation to the classes defined in the system. The feedback visualized a map of clusters of different classes which subjects could match the position of a cursor to. When subjects could match the cursor to the centroid of a cluster the performed movement corresponded the best with the class of that movement. [14] An illustration of the experimental setup used in [14] can be seen in figure 1.9.

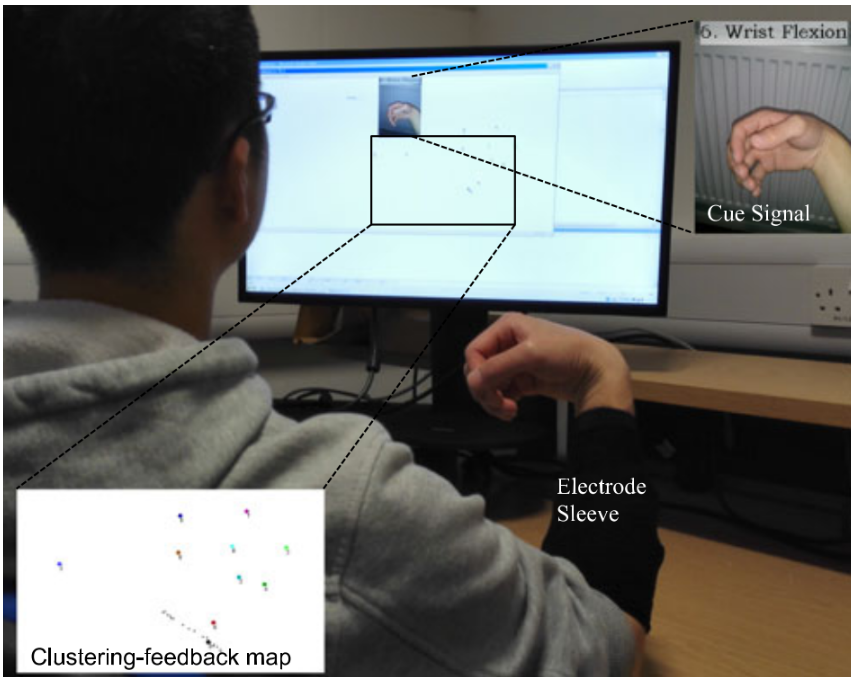


Figure 1.9: An illustration of the experimental setup used in [14]. The user received a cue on which movement should be performed. The user then had to fit a cursor, controlled by the user, to the centroid of the cluster points corresponding to the instructed movement in a clustering-feedback map.

All studies observed an improvement in user performance after being exposed to focused user training with visual feedback.
A 2013 study by Scheme et al. [11] proposed a novel approach of utilizing confidence-based rejection to improve system training of myoelectric control. Here a classification control scheme was provided with confidence scores to assist in acceptance or rejection of the class output. The confidence scores were calculated from a modification of Bayes' theorem. Scheme et al. [11] showed a significant improvement in performance with the use of the rejection-capable system when compared to the normal classification scheme. A similar approach could be used in user training by providing the confidence scores of the classification to the user as a form of visual feedback.

1.8 Validating Performance

Measuring the performance of achieved prosthesis control cannot be seen as a trivial task, and different approaches can be used. The achieved performance can be measured by affixing a prosthesis on to the test subject and validate performance hereby. Often, like the current project, the subjects do not consist of actual amputees but instead healthy subject. In these cases the performance validation is done by implementing a virtual test environment where the subjects is to control an object on the computer screen by performing movements. The following section will further elucidate the procedures of such a virtual test for validating prosthetic control.

1.8.1 Modified Fitts' Law

Fitts' law task is a common method of quantifying performance of movements, first proposed by Paul M. Fitts in 1954 [27]. Fitts' law states the that time required to reach a targeted area is function of the width and distance of the target. The output of a Fitts' law task is the throughput, as given by equation (1.8). This measure gives an idea of the trade-off between speed and accuracy. A modified Fitts' law task designed for a virtual 2D and 3D target acquisition task has later been used by [28] and [11] respectively. Here, four additional metrics were added in an online task, where a virtual computer cursor was used to represent the control output [11, 28]. The four additional metrics, path efficiency, overshoot, stopping distance and completion rate, were made by [29] and [30]. While the throughput measure from the conventional Fitts' law task is usable, it does not cover all aspects of the control required to complete a task. The additional four measures were added to quantitatively assess performance of naturalness, spontaneity, and compensatory motions during use. The total five proposed performance measures in assessing myoelectric control are [31]:

Throughput (TP) which represents the trade-off between speed and accuracy. TP uses the relationship of time taken to reach a certain target in seconds (MT) and the index of difficulty (ID). This forms: [11, 27]

$$TP = \frac{1}{N}\sum_{i=1}^{N}\frac{ID_i}{MT_i} \tag{1.8}$$

where i is a specific movement and N is the total number of movements. ID relates to the target distance D and width W. The ID for each task, from the origin to a specific target of a certain size is calculated using [11, 27]:

$$ID = log_2(\frac{D}{W} + 1) \tag{1.9}$$

Path Efficiency (PE) describes the quality of control by making a measure of the straightness of the cursor's path to the target, by making a ratio of the actual path distance versus the optimal path distance. This tests the users ability to continuously control the cursor position. Following the optimal path will result in a PE of 100%. PE is calculated as follows [11, 29]:

$$PE = \frac{Optimal\ Distance}{Actual\ Distance} \tag{1.10}$$

Overshoot (OS) is the number of times the cursor enters and then leaves the target before the dwell time inside the target is reached, across all target in the task, divided by the total number of targets. OS tests the users ability to control the velocity of the cursor accurately. A perfect OS-score of zero is reached if the cursor dwells within the target boundaries on the first try for all targets, and is calculated as the following [11, 29]:

$$OS = \frac{Total\ Number\ of\ Overshoots}{Total\ Number\ of\ Targets} \tag{1.11}$$

Stopping Distance (SD) describes the users ability to rest and thereby perform no movement. The SD measure is the distance moved during the dwell time across all targets, and is given as [11]:

$$SD = \sum_{i=1}^{N} (Distance\ Inside\ Target)_i \tag{1.12}$$

where i is a reached target and N is the total number of reached targets.

Completion Rate (CR) describes the percentage of targets reached within the total allowed time. This gives a general idea of the user's performance, and is calculated as [11, 30]:

$$CR = \frac{Number\ of\ Reached\ Targets}{Total\ Number\ of\ Targets} \tag{1.13}$$

1.9 Data separability

Besides evaluating the user performance in real-time, it would be beneficial to evaluate the clustering of feature values used to fit the classifier as it is this separability of clusters that determines the classifier accuracy. This will provide information on how the clusters between classes separates and how the feature values within clusters bundle. Hereby it can be used to see if the users get better at performing more separable movements. Thus it can be evaluated between sessions if the clusters are more distinguished and thereby easier for the classifier to discriminate between. By doing this between sessions it can be evaluated how distinguishable the clusters are, to judge how well the classifier and regressors will discriminate between the different clusters. For this purpose a Principle Component Analysis (PCA) will be utilized to reduce the dimensionality of the feature space, which subsequently can be used to calculate the distance between clusters and distances from feature values to centroids within clusters. In order to able to visualize the cluster separability a dimensionality reduction to three dimensions is chosen. The following section provides theoretical information on the PCA procedure.

1.9.1 Principal Component Analysis

PCA is used to express a set of possibly correlated variables into uncorrelated components, called principal components (PC). A dataset of many variables can thus be expressed in a reduced dimensionality hyperspace using less variables that are the most defining for the given dataset. Each principal component is orthogonal on the former and are uncorrelated and have zero covariance. They each define the largest variance in an axis, such that the first PC describes the direction of the maximum variance of the dataset. Each following PC describes the next highest variance of the dataset, with the constraint that it is orthogonal and has zero covariance with any of the former PCs. [32] A PC is found by minimizing the variance by projecting the feature values, the red dots in figure 1.10, onto the line describing the highest variance in the data set (purple line) as seen on figure 1.10. The PC (purple line) is found by minimizing the mean square distance between the data points. PCA can be used as the orthogonal projection of data onto a lower dimension linear space, reducing the dimensionality, based on how many PC's is chosen to represent the data after the analysis. [32]

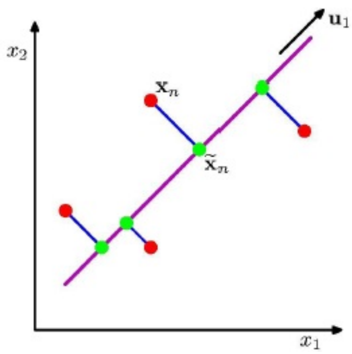


Figure 1.10: Projection of feature values (red dots) onto PC axes (purple).

The algebraic method of calculating the PCs can be done by using Singular Value Decomposition (SVD). The first step is to compute the squared cross product matrix of variances and covariances among every pair of the variables in the data set, where the diagonals are the variances and the off-diagonals are the covariances, as done in the following equation [32]:

$$S = X'X \tag{1.14}$$

Where S is the cross product and X is the feature set matrix. When finding the PCs it includes an eigen-analysis of S. The eigenvalues of are solutions to the following equation [32]:

$$|S - \lambda I| = 0 \tag{1.15}$$

Where λ is the variance of each PC and I is the identity matrix. After solving for λ the eigenvectors can be solved through the following equation [32]:

$$det|S - \lambda I|b_i = 0 \tag{1.16}$$

Where b_i is used to calculate the eigenvectors as in [32]:

$$u_i = \frac{b_i}{\sqrt{b_i' b_i}} \tag{1.17}$$

Where u_i is the i^{th} number of eigenvectors. The number eigenvectors equal the dimension size of the original feature space. The SVD orders the eigenvalues by size, so that $\lambda_1 > \lambda_2 \ldots > \lambda_i$. The scores for each PC is equal to the corresponding eigenvalue for that exact axis. The eigenvalues describe how much of the variance is accounted for by the associated PC. Summation of all eigenvalues accounts for the total variance of the data set; this is called the trace. To find how much the each PC accounts for, the eigenvalue of that PC is divided by the total variance: $\%\ of\ total\ variance\ = \frac{\lambda_i}{Trace}$. Deciding how many PCs the feature space should be reduced to, by setting a threshold of how much of the total variance should be preserved. [32]

1.9.2 Distance measure

After reducing the dimensionality of the original feature set the clusters can be analyzed. For the purpose of measuring distances between and within clusters the centroid of each cluster can be calculated, as in equation (1.18):

$$C = \left[\frac{[x_1 + x_2 + \ ...\ + x_n], [y_1 + y_2 + \ ...\ + y_n],\ ...\ , [k_1 + k_2 + \ ...\ k_n]}{n} \right] \tag{1.18}$$

Where C is the centroid, i is the number of feature point in a dimension and k is the number of dimensions. To calculate the distance between centroids of clusters the Euclidean distance (ED) is computed. The ED is the length of the a line segment connecting points, in this case in form of two centroids p and q. The calculation of ED in a k-dimensional space is as in equation (1.19):

$$ED(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2\ +\ ...\ + (p_k - q_k)^2} \tag{1.19}$$

When calculating the distance from feature values in a cluster to their corresponding centroid the ED is computed likewise. To get a general impression of the distance from the feature values constituting the cluster to the centroid of the cluster the average of the distances is calculated.

2 | Methods

The background chapter covered the basic procedures associated with myoelectric prosthetic control and techniques to deal with the proposal of improving users' ability to operate a myoelectric transradial prosthesis by training the user with confidence score feedback. This involves which hand movements are most commonly used in daily life tasks; how the EMG signal is generated and how is it acquired with surface EMG electrodes using the MYB; how the raw EMG signal is processed before it is segmented in windows; how features are extracted from the segmented signal; how the feature values are used in a classification control scheme to distinguish which movement is performed; how linear regression models are used to obtain proportional control; how confidence scores can be calculated as posterior probabilities from the classification scheme; how user training previously has been used to optimize users' ability to operate a myoelectric transradial prosthesis; and how the user performance is evaluated.
The information acquired in the background section will lay foundation for how the study will be designed, how the procedures will be implemented and which considerations that have been made regarding the implementation. This will be covered in the methods chapter. As the project investigates whether users' ability to operate a myoelectric prosthesis can improve after training with confidence score as visual feedback, a major focus has been put in the implementation of the user training. The chronology of the methods chapter is that the study design first will be presented, after which the implementation of the different procedures will presented, as the procedures are implemented with regards to the how the study design is formed.

2.1 Study Design

This experiment focused on training the user to improve prosthetic control on a fixed pattern recognition-based control system. The novel approach in this study was to provide the user with visual feedback on how well the system recognized the performed movements during user training, by showing the confidence levels of the movements the control system recognized. The following section will lay an overview of the implementation of the different stages of this experiment.

To test if myoelectric prosthetic control could be improved by using visual confidence level feedback the following research hypothesis was made.

Exposing subjects to user training, in which confidence levels of movement recognition is used as feedback, will show statistically significant improvement in performance in a classification-based myoelectric prosthetic control scheme compared to a control group.

To test the hypothesis 16 subjects of mean age 25.3 $\pm$ 1.48 were recruited. 15 subjects were male and 1 female where 14 were right handed and 2 were left handed. Subjects were randomly assigned to either a control group or test group. The subjects enrolled were assessed to meet inclusion criteria presented in the experimental protocol for test subjects in section A.1. The experimental protocol was handed out to possible test participants before enrolment.
The experiment was designed as a three session investigation. In each session both groups had data acquired, received user training and did a performance test through a Graphical User Interface (GUI) developed in MATLAB (2017b). During session one it was chosen to add a performance test before submitting the subjects to user training. This preliminary performance test were set to act as a baseline for each group to highlight any initiating group disparity. A graphical illustration of the stages of the study design can be seen on figure 2.1. Essential for the experiment was the difference in user training highlighted in step 3 where the groups received two different kinds of visual feedback. The test group received a visual feedback of the confidence score the classifier produces, when the subject train the

different kinds of movements. The control group received the same visual training, however this would not inform of the confidence score but instead solely show which movement the classifier thought was being performed. The sections to come will further elaborate on the implementation of each element in the experiment and how the user training differs. During the experiment the subject was seated on a chair, with the dominant arm wearing the MYB hanging relaxed laterally down the torso as seen on figure A.5 in the experimental protocol.

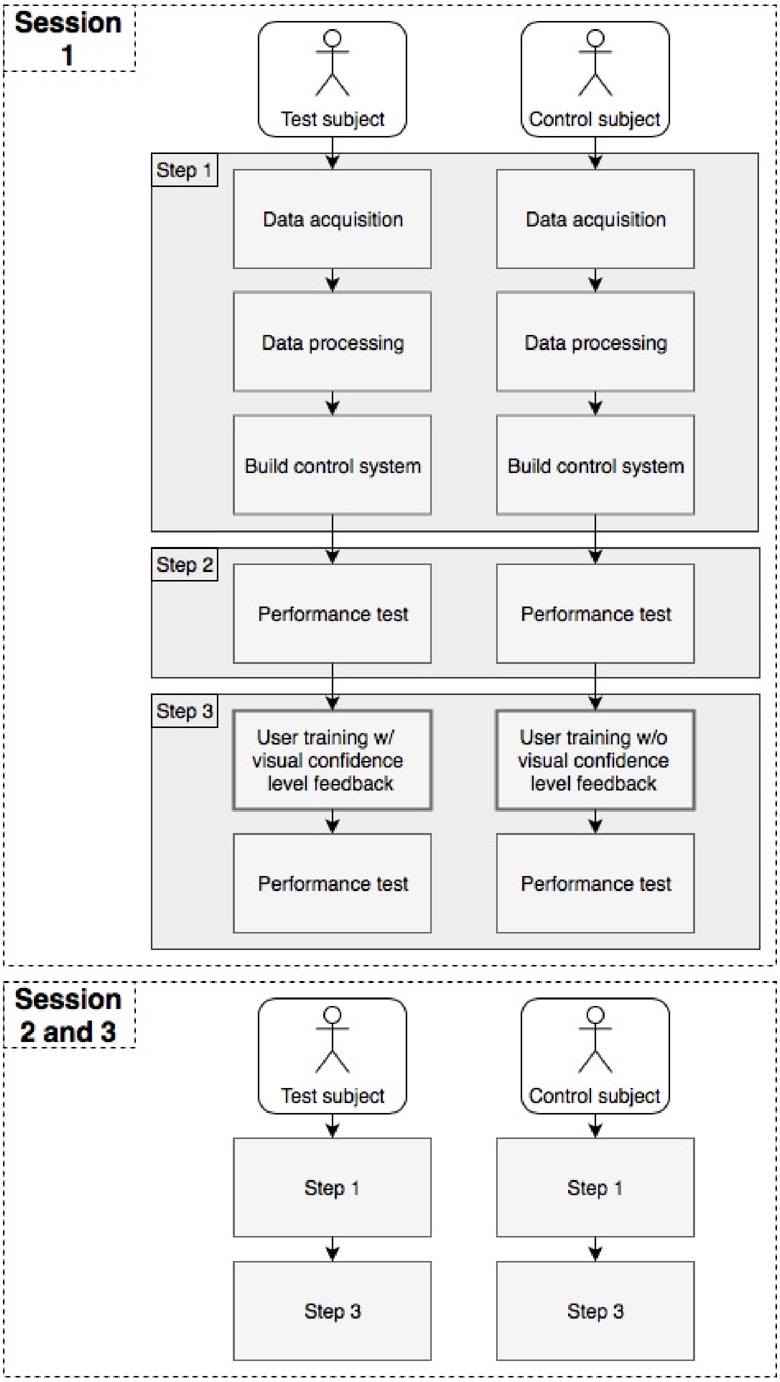


Figure 2.1: Graphical illustration of the experiment showing the steps of each session for the test and control group. Highlighted is user training in step 3 which is the only procedure that varies between the two groups, and thus the area of research interest in the experiment.

2.2 Data Acquisition

This section will clarify the method of acquiring data in this project. For data acquisition the MYB was used to record EMG signals from muscles in the lower forearm described in section 1.1. The recordings was made on test subjects instructed to perform six different hand gestures as introduced in the protocol, section A.1.

For acquiring data a Graphical User Interface (GUI) was designed using MATLAB. In the GUI it was possible to change settings for different types of recordings. The first type of recording was a baseline measurement. This recording was made in order to be able to reduce the baseline noise. This was done by subtracting the baseline from the EMG signal when the the EMG signal reached higher than the baseline. When the EMG signal was below the baseline, it was set as 0.
The second recording type was a Maximum Voluntary Contraction (MVC) which was a 15 second recording of the subject's maximum contraction of one movement that could be kept constant for 15 seconds without developing muscle fatigue. The mean MVC across all channels was set as a reference value for the following recordings.
The third type of recording was of EMG signals used to train the control system. The recordings of EMG signals were based on fractions of the MVC, which could be set using a menu in the GUI. As stated in the protocol, section A.1, three contraction levels was used: 40%, 50% and 70%. The level of contraction defined the height of the plateau of a trapezoid trajectory which would be plotted in a window in the GUI. When doing EMG recordings the subjects must perform the instructed movement to control the height of a cursor in the trapezoid plot to best match the trajectory of the trapezoid. The cursor height was calculated as the mean EMG signal across channels normalized based on the MVC. The subject only controlled the height (EMG intensity) of the cursor as the cursor would automatically move forward along the x-axis in relation with time. The recording time was 15 seconds: 2.5 seconds rest at the initiation and ending, 2.5 seconds on the trajectory incline and decline and 5 seconds on the plateau. This approach provided data from a performed movement in both the transition and steady state phase. This data acquisition method was applied since the use of dynamically changing force data in training a classification-based control scheme has shown to improve performance and tolerance to proportional control [13]. During recordings the investigators evaluated whether the subject followed the trajectory well enough. Furthermore to evaluate the training data, the investigators observed a spider-plot during the acquisition. The spider-plot showed the amplitude output for each channel in the MYB. If the activation pattern of the channels changed dramatically, it was a sign of fluctuations in muscle activation, and thus the subject did not perform the instructed movement. If this was observed the recording was discarded and a new was be acquired. An illustration of the data acquisition GUI is shown in figure 2.2.

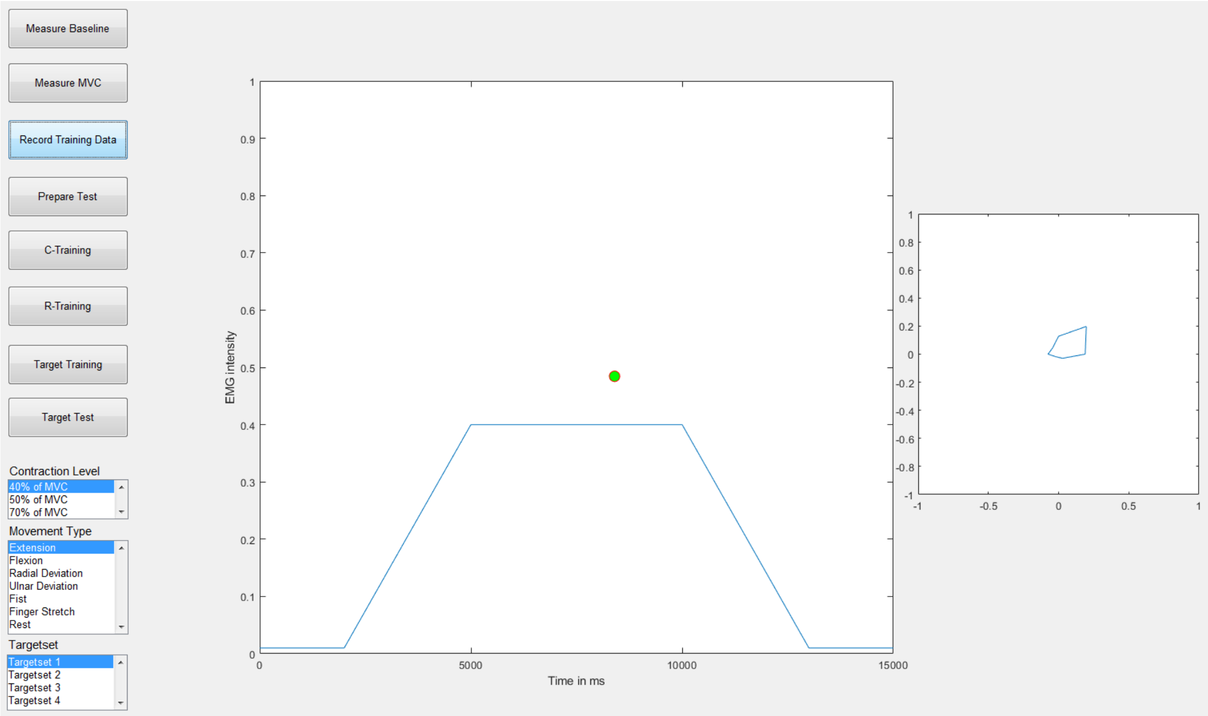


Figure 2.2: The implemented data acquisition interface. On the left is different buttons shown, where only "Measure Baseline", "Measure MVC" and "Record Training Data" is used in the data acquisition. The "Contraction Level" menu forms the trapezoidal trajectory and "Movement Type" saves the performed movement the correct label. In the center is the trapezoidal trajectory and the cursor representing the EMG signal. On the right is the spider-plot used to evaluate the quality of the performed movement.

2.3 Data processing

The following two sections will cover the implementation of the filter used to prepare the EMG-signal and the extraction of features to represent the signal. Choices behind implemented methods builds on background knowledge acquired in section 1.4.

2.3.1 Filtering of signal

As earlier mentioned in section 1.4.1, due to the MYB specifications limiting the sample rate to 200 Hz and due to movement artefacts in the low-frequency spectrum, it would be resourceful to implement a bandpass filter to avoid a biased signal. In the interest of representing the signal with its true properties a second order Butterworth bandpass filter has been implemented with cut-off frequencies at 10 Hz and 90 Hz. A filter steeper than second order was deselected due to a chosen trade-off between filter performance and computational performance, which is of great importance when doing real-time control. In figure 2.3 is the result of the bandpass filter implementation shown. The unfiltered signal shows frequency components in low-frequency spectrum around 0-10 Hz and indicating frequency components at 100 Hz. Both ends of the spectrum have been attenuated limiting impact of artefacts. Furthermore is the presence of the build-in 50 Hz notch filter seen as explained in section 1.3.1.

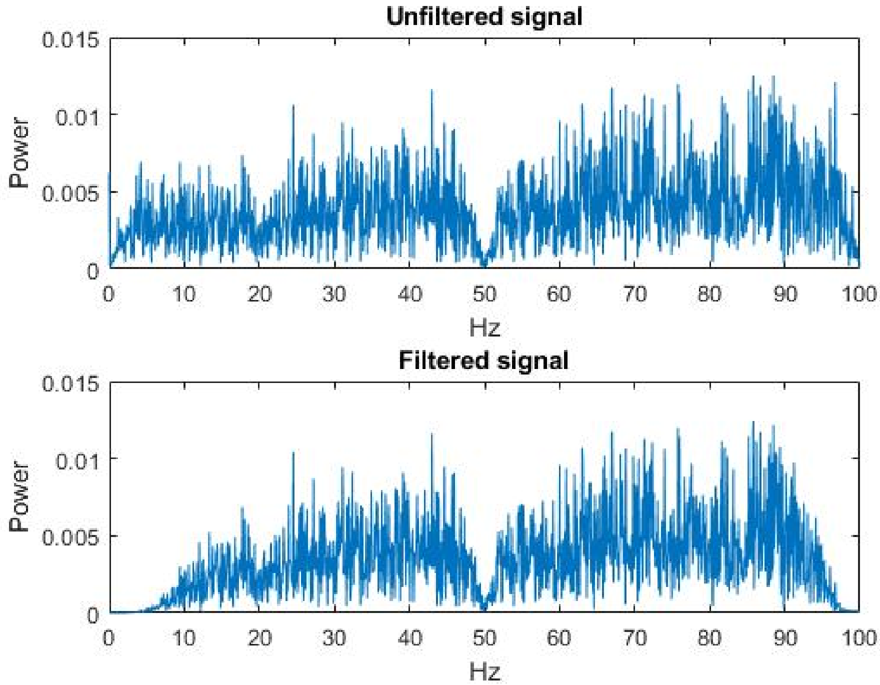


Figure 2.3: Frequency spectrum of a randomly selected EMG-recording showing the difference before an after implementing the bandpass filter. The unfiltered signal shows frequency components in low-frequency spectrum around 0-10 Hz and indicating frequency components at 100 Hz. The filtered signal shows reduction in the signal outside the cut-off frequencies.

2.3.2 Feature extraction

In section 1.4.2 it was stated that when extracting features for real-time prosthesis control, features are extracted from segments of the signal called windows. This projects will for the feature extraction utilize a window size of 200 ms and a 50 % overlap for all channels. This thereby gives the possibility of calculating and updating the feature values ten times a second, thus minimizing the update delay in the real-time user training and performance test.

The features chosen to represent the information of movements contained in the signal is primarily based on recommendations from [33] where they found the optimal features for a real-time classification control scheme using the MYB. Donovan et al. [33] used so called space domain features along with the MYB and got a five percent higher accuracy than by using the well known Hudgins time domain features. A total of seven features, Mean Absolute Value (MAV), Mean Mean Absolute Value (MMAV), Scaled Mean Absolute Value (SMAV), Correlation Coefficient (CC), Mean Absolute Difference Normalized (MADN), Mean Absolute Difference Raw (MADR) and Scaled Mean Absolute Difference Raw (SMADR) were derived and the following section will explain the extraction of each. This project will use SMAV, CC, MADN and SMADR for the final classification to reduce redundancy, but all seven will be explained because some features are a combination of others. [33] Furthermore it has been chosen to extract the time domain feature of waveform length (WL) to represent frequency related information of the signal. The extraction of this feature will lastly be explained as well.

MAV is a feature that primarily is affected by the force produces when making a contraction. MAV is extracted for each window and calculated for each of the i^{th} channel. The extraction is expressed as:

$$MAV_i = \frac{\sum_{n=1}^{ws} |x_i[n]|}{ws} \tag{2.1}$$

where ws is the window size, the number of raw data points in that exact window. $x_i[n]$ is the n^{th} raw data points from the i^{th} channel.

The mean MAV across all channels, MMAV, is used to remove dependency of movement intensity. MMAV is calculated by using the MAV of all channels for the current window, and is done as following:

$$MMAV = \frac{\sum_{i=1}^{8} MAV_i}{8} \tag{2.2}$$

MMAV can be used to scale the MAV feature creating the SMAV feature. This feature should represent a non-dimensional relationship between channels. SMAV is simply calculated as:

$$SMAV_i = \frac{MAV_i}{MMAV} \tag{2.3}$$

As each of the eight EMG sensors in the MYB are located around the arm, they acquire signals from a mixture of sources. Also individual sources may affect multiple sensors depending on their size. Due to this a source measured by multiple sensors will effect their acquired signal correlation. An idea is therefore to calculate the correlation coefficient between each channel and its neighboring channel.

$$CC_i = \frac{\sum_{n=1}^{ws} X_i[n]X_{i+1}[n]}{ws} \tag{2.4}$$

$X_i[n]$ is the n^{th} normalized data point from channel i. When calculating CC the data from each window is normalized by subtracting its mean value from each raw data point, and afterwards divided by their standard deviation.

Calculating CC can prove rather demanding in computational power due to the series og multiplication operations. Therefore Donovan et al. [33] proposed introducing a mean absolute difference-based feature of lower computational complexity which still characterizes the spatial relationship between channels. The MAD feature is normalized in the same way as CC, making up the MADN feature calculated as:

$$MADN_i = \frac{\sum_{n=1}^{ws} |X_i[n] - X_{i+1}[n]|}{ws} \tag{2.5}$$

If the normalization of the signal proves too demanding the feature can be calculated on the raw EMG-signal without the normalization. This makes up the MADR feature, calculated as:

$$MADR_i = \frac{\sum_{n=1}^{ws} |x_i[n] - x_{i+1}[n]|}{ws} \tag{2.6}$$

As the SMAV feature the MAD feature can be scaled by MMAV to remove movement intensity dependency. SMADR is calculated for each channel by:

$$SMADR_i = \frac{MADR_i}{MMAV} \tag{2.7}$$

As stated in the beginning some of these features introduce redundancy, subsequently the features of SMAV, CC, MADN and SMADR are the ones used for classification. [33]

To further improve the decision foundation of the classifier it was proposed to include the time domain feature of WL calculated by:

$$WL_i = \sum_{n=1}^{N-1} |x_{i+1}[n] - x_i[n]| \tag{2.8}$$

WL is a measure of the signal complexity by calculating the cumulative length for each channel [8].

2.4 Building the Control System

Following the data acquisition and processing, the training data was used for movement classification. The features extracted for each of the seven movements were used for building the classifier. In section 2.4.1 the implementation of the classification and an explanation of its output will be covered. Furthermore to be able to obtain proportional control a regression based models were made. The implementation of proportional control will be explained in section 2.4.2. An explanation of how the classifier and regression models were used in the user training and in the performance test can be found in section 2.5 and section 2.6 respectively.

2.4.1 Movement Classification

For classifying movements Linear Discriminant Analysis was used as presented in section 1.5. The classifier was fitted with the previously acquired training data in order to build the control system. The acquired training data was assembled into matrices for each of the seven movements with one of the five features, containing the feature values for each of the eight channels. An example of one of these matrices can be seen in equation (2.9). This matrix contains n feature values for the feature CC for all three intensities of extension across all eight channels.

$$AllIntCC\_Ex = \begin{bmatrix} \begin{bmatrix} CCExtension40_{1,1}, CCExtension40_{1,2} \cdots CCExtension40_{1,8} \\ \vdots \qquad \ddots \qquad \vdots \\ CCExtension40_{n,1}, CCExtension40_{n,2} \cdots CCExtension40_{n,8} \end{bmatrix} \\ \begin{bmatrix} CCExtension50_{1,1}, CCExtension50_{1,2} \cdots CCExtension50_{1,8} \\ \vdots \qquad \ddots \qquad \vdots \\ CCExtension50_{n,1}, CCExtension50_{n,2} \cdots CCExtension50_{n,8} \end{bmatrix} \\ \begin{bmatrix} CCExtension70_{1,1}, CCExtension70_{1,2} \cdots CCExtension70_{1,8} \\ \vdots \qquad \ddots \qquad \vdots \\ CCExtension70_{n,1}, CCExtension70_{n,2} \cdots CCExtension70_{n,8} \end{bmatrix} \end{bmatrix} \tag{2.9}$$

The matrix consists of three sub-matrices: one for each of the intensities acquired as explained in section 2.2. The naming of the matrix is explained as that $AllInt$ denotes all intensities, CC denotes the CC feature and Ex denotes the extension movement. Similar matrices were constructed for all other features for all movements named in the same fashion as the AllIntCC_Ex matrix. All these matrices were assembled into one large training matrix, TM, in a five-dimensional feature space as seen below in equation (2.10).

$$TM = \begin{bmatrix} AllIntCC\_Ex, AllIntSMAV\_Ex, AllIntSMADR\_Ex, AllIntMADN\_Ex, AllIntWL\_Ex \\ AllIntCC\_Fl, AllIntSMAV\_Fl, AllIntSMADR\_Fl, AllIntMADN\_Fl, AllIntWL\_Fl \\ AllIntCC\_Rd, AllIntSMAV\_Rd, AllIntSMADR\_Rd, AllIntMADN\_Rd, AllIntWL\_Rd \\ AllIntCC\_Ud, AllIntSMAV\_Ud, AllIntSMADR\_Ud, AllIntMADN\_Ud, AllIntWL\_Ud \\ AllIntCC\_Ch, AllIntSMAV\_Ch, AllIntSMADR\_Ch, AllIntMADN\_Ch, AllIntWL\_Ch \\ AllIntCC\_Oh, AllIntSMAV\_Oh, AllIntSMADR\_Oh, AllIntMADN\_Oh, AllIntWL\_Oh \\ AllIntCC\_Re, AllIntSMAV\_Re, AllIntSMADR\_Re, AllIntMADN\_Re, AllIntWL\_Re \end{bmatrix} \tag{2.10}$$

The classifier was trained by fitting the matrix presented in equation (2.10) with labels for each of the movements, by using the fitcdiscr function in MATLAB. The fitcdiscr function makes a LDA classifier model as described in section 1.5.2. The classifier thereby formed seven classes, one for each of the movements, with linear decision boundaries separating them. For calculating the real-time use of classification outcome and confidence scores in user training and performance test as intended, the predict function in MATLAB was used. The function was continuously evaluating each feature value to the different movement classes in the five dimensional feature space. Thus, the feature values were assigned to the movement class they were most likely to belong to based on the training data. The predict function also calculated the probability membership for the feature values to all classes giving an idea of how confident the classifier was on deciding a certain movement class and thereby indicating the correctness of the movement performed.
The classifier was only used to decide upon which movement was performed, thus not used in performing proportional control. For this purpose linear regression models were used.

2.4.2 Proportional Control

To obtain proportional control linear regression models, regressors, for each movement were made. For this purpose multivariate linear regression was used, as explained in section 1.6. To fit a regressor dependent and independent variables were needed. The independent variables were set as MAV features extracted from the raw EMG data acquired from the data acquisition as explained in section 2.2. To ensure that the regressor did not produce an output during rest, the MAV features extracted from the raw EMG from a resting position were included as independent variables in each regressor. An example of the independent variables used to fit the regressor for the extension movement is seen in equation (2.11).

$$AllIntMAV\_Ex = \begin{bmatrix} \begin{bmatrix} MAVRest_{1,1}, MAVRest_{1,2} \cdots MAVRest_{1,8} \\ \vdots \quad \ddots \quad \vdots \\ MAVRest_{n,1}, MAVRest_{n,2} \cdots MAVRest_{n,8} \end{bmatrix} \\ \begin{bmatrix} MAVExtension40_{1,1}, MAVExtension40_{1,2} \cdots MAVExtension40_{1,8} \\ \vdots \quad \ddots \quad \vdots \\ MAVExtension40_{n,1}, MAVExtension40_{n,2} \cdots MAVExtension40_{n,8} \end{bmatrix} \\ \begin{bmatrix} MAVExtension50_{1,1}, MAVExtension50_{1,2} \cdots MAVExtension50_{1,8} \\ \vdots \quad \ddots \quad \vdots \\ MAVExtension50_{n,1}, MAVExtension50_{n,2} \cdots MAVExtension50_{n,8} \end{bmatrix} \\ \begin{bmatrix} MAVExtension70_{1,1}, MAVExtension70_{1,2} \cdots MAVExtension70_{1,8} \\ \vdots \quad \ddots \quad \vdots \\ MAVExtension70_{n,1}, MAVExtension70_{n,2} \cdots MAVExtension70_{n,8} \end{bmatrix} \end{bmatrix} \tag{2.11}$$

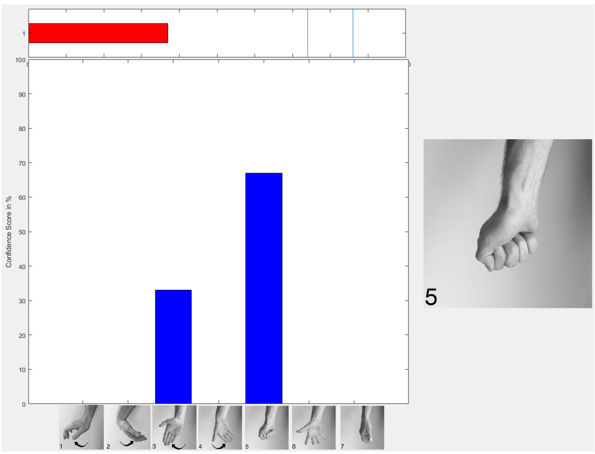
The matrix contains four sub-matrices consisting of n feature values across all eight channels for rest and the three contraction levels: 40 %, 50 % and 70 % respectively. The naming of the matrix is explained as that $AllInt$ denotes all intensities, MAV denotes the MAV feature and Ex denotes the extension movement. The independent variables matrices used to fit the regressors for the other five movements were named in the same fashion. The purpose of using data from several contraction levels was to ensure proportional control.
The dependent variables were similar to the independent variables. However, only a single output per window was desired. Therefore the mean of the feature values extracted from a window was calculated and scaled according to the MVC. The MVC was set as reference value of 1. The dependent variables corresponding to rest were set as 0. Thus, the dependent variables used to fit the regressor was a vector. An example of the dependent variables used to fit the regressor for the extension movement is seen in equation (2.12)

$$AllIntMAV\_ExScaled = \begin{bmatrix} \begin{bmatrix} 0_{1,1} \\ \vdots \\ 0_{n,1} \end{bmatrix} \\ \begin{bmatrix} MAVExtension40Scaled_{1,1} \\ \vdots \\ MAVExtension40Scaled_{n,1} \end{bmatrix} \\ \begin{bmatrix} MAVExtension50Scaled_{1,1} \\ \vdots \\ MAVExtension50Scaled_{n,1} \end{bmatrix} \\ \begin{bmatrix} MAVExtension70Scaled_{1,1} \\ \vdots \\ MAVExtension70Scaled_{n,1} \end{bmatrix} \end{bmatrix} \tag{2.12}$$

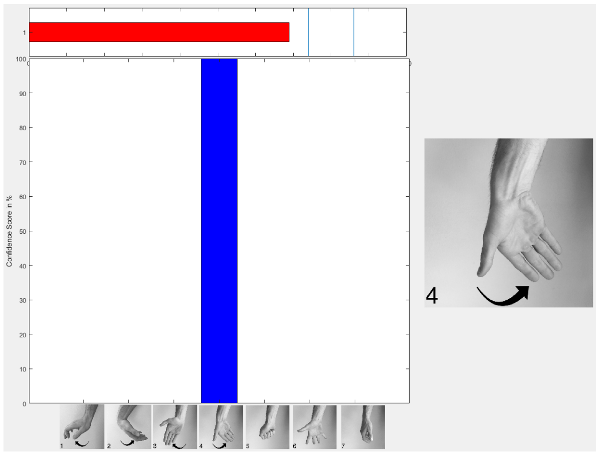
The output of the regressor would then be larger the more forceful a contraction the subject produced and vice versa. The *fitlm* function in MATLAB was used to fit a regressor with the independent variables from equation (**??**) and dependent variables from equation (2.12) as input. A total of six regressors were made; one for each movement used in the experiment.
The regressors use for proportional velocity control in the control system was controlled by the classifier decided upon which movement was performed. The output movement were then fitted by the regressor and used for proportional control.

2.5 User training

User training was the essential element of investigation in this experiment, with the movement confidence feedback as the area of interest. This section provides information on how the visual feedback was presented to the subjects in the two experiment groups during the user training, and what the objective for the subjects was.
The user training interface contained the following feedback: an illustration of the movement needed to be performed, a horizontal bar visualizing the contraction level and a vertical bar plot visualizing which movement is being recognized by the control system, as shown in figure 2.4. The difference in feedback given between subject group, lied in the information given in the vertical recognition bar plot.



(a) Test group user training interface.



(b) Control group user training interface.

Figure 2.4: Illustration of the user training interface for the test group (a) and the control group (b). The vertical bar plot indicates which movement is being recognized indicated by the images of each movement and the horizontal bar plot indicates contraction level. The two vertical lines in the contraction level bar plot illustrates the contraction level interval the subject must reach. The large picture of a movement on the right of the bar plot indicates which movement needs to be performed. The difference between the feedback the two subject groups receive is the information given in the vertical recognition bar plot. The control group only sees a full bar of the movement the control system recognizes the most, whereas the test groups receives the exact recognition probabilities of all movements.

The illustration of the movement needed to be performed was shown for 30 seconds, after which an illustration indicating rest was shown for 7 seconds followed by a countdown from 3 to 1 seconds indicating the time left of the resting period. Thus, the subject needed to perform a movement for 30 seconds and rest for 10 seconds before another movement needs to be performed. The subjects needed to perform all movements in four different contraction level intervals of their maximal intensity, starting with the highest interval: 75-85 %, 55-65 %, 35-45 % and 15-25 %, visualized by the two vertical lines in the horizontal contraction level bar plot. The subjects needed to perform all movements in the same contraction level interval before moving to a new interval. The instructed movements were trained in a random order.
The horizontal bar showed the contraction level. This was calculated as the mean of the latest three in-

tensity outputs as computed in section 2.4.2, regardless of the movement being recognized. This resulted in a 400 ms delay in the visualization of the horizontal bar at the initiation of the training of a movement, due to the windowing used in feature extraction as mentioned in section 2.3.2. However, the delay was not noticeable and the averaging of the intensity output resulted in a smooth visualization of the vertical bar.
The vertical bar plot showed which movement(s) the control system recognized. For a movement to be recognized as an active movement, the subjects had to perform the movement with more than 15 % contraction intensity. The test group received information on the exact probabilities for the movements that were recognized. Thus, more bars could appear at the same time as seen in figure 2.4 (a). The purpose of this feedback was for the subjects to adapt to how the control system recognized the instructed movement. It gave the subjects the possibility of noticing which movements that also were recognized by the control system when performing the instructed movement. When the instructed movement was not recognized with a 100 % certainty the subject could use the information to slightly correct the performed movement until the control system recognized the instructed movement with a 100 % certainty. This bar plot was calculated as the mean of the recognition certainties calculated from the latest three feature inputs, which resulted in a smooth visualization of certainties for the movements in the bar plot.
The control group only received information on which movement was recognized the most, represented as a single full bar at the recognized movement as seen in figure 2.4 (b). Thus, the control group was not informed on the exact probabilities of which movements the control system recognized. This bar plot was calculated by finding the highest mean certainty output among all movement classes. This was done taking the mean of the last three feature inputs in each movement class..
To motivate the subjects and to train the transition to and from resting position a task was included in the user training. The subjects were instructed in performing the instructed movement with a 100 % certainty inside the instructed contraction level interval. When this was reached the horizontal bar would turn from red to green. The subjects were instructed in withholding the green color for one second, after which the horizontal bar would turn blue and a light sound was played. After this task was completed the subjects was instructed in returning to rest and perform the task again. The objective for the subject was then to make the horizontal bar blue as many times as possible during an instructed movement of an instructed contraction level. The number of times the horizontal bar got blue during an instructed movement in an instructed contraction level was saved for later data analysis.

To summarize, the overall objective for the subjects during the user training was to adapt to how the control system recognized the performed movement. The user training was implemented for the subjects to possibly improve their ability to use the control system. Their ability to use the control system was then evaluated in the modified Fitts' Law task.

2.6 Performance Test

As stated in section 1.8 the theory behind Fitts Law, this project will utilize a modified Fitts' Law test, incorporating addition performance metrics, consisting of a virtual target reaching test to evaluate the progress of subjects going through the different types of user training. The proposed modified Fitts' law test utilizes the performance metrics of, throughput, path efficiency, overshoot, stopping distance and completion rate. The following section describes how the Fitts' Law task has been implemented in this project.

2.6.1 Virtual target reaching test

The virtual targets reaching test was implemented into the same GUI used for data acquisition and user training, first mentioned in section 2.2. When enabling the target reaching test in the GUI the subject was met with the interface shown in figure 2.5. Here the subjects controlled the position of the cursor by performing movements shown above, below, and on the sides of the borders of the grid area.

Thus, extension of the hand would move the cursor to the right of the grid, and flexion would move the cursor to the left. Similarly, radial and ulnar deviation moved the cursor up and down respectively. This approach was used to improve the intuitiveness of the control where the direction of the cursor relate to the directions subjects will perform hand movements. Subjects controlled the size of the cursors red area by opening and closing the hand, where an open hand increases the area and a closed hand decreases the area.
Each target was presented by an area with a center and an outer circle. Targets existed in three sizes in order to test opened and closed hand degree of freedom. The target reaching test consisted of reaching a total of 16 targets which each appeared for 15 seconds at different distances from the center of the grid area. The order of appearance was fixed but different for each of the four sessions but the same across subjects, thus individual subjects will experience the targets as appearing in a new order with each session.

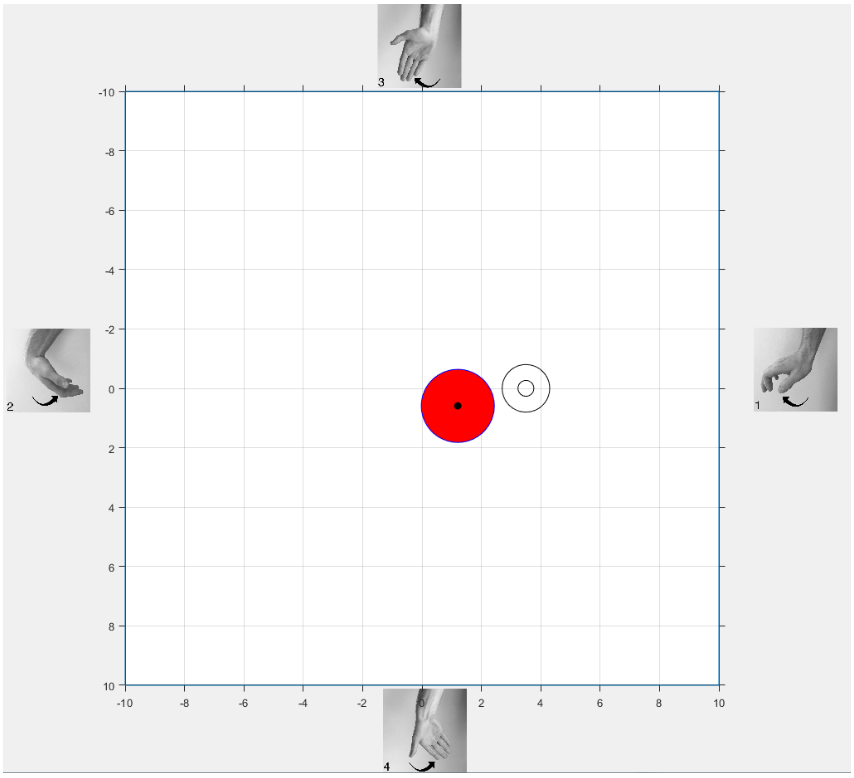


Figure 2.5: The implemented interface for the modified Fitts' Law task. The grid was the area in which the subject must reach targets with the controlled cursor. The cursor is the red circle area with the black dot in the center. Targets were shown as a circle area with an bigger outer circle. Performing the movement corresponding to the axis image would move the cursor in the direction of the image.

Subjects had to reach the targets inner circle with the cursor dot and expand or decrease the red area of the cursor to reach a size close to that of the target. A moderate size threshold was implemented to make it possible to reach targets, without a 100% accuracy of control. If a subject reached a target, the cursor would change color from red to green, and if the subject withholds this position for a 1 second dwell the cursor changed to blue, and a bell chime will sound to indicate that the target was reached. The cursor position was reset to the center of the grid area and the color of the cursor would revert back to red. If a target was not reached within 15 seconds the current target would disappear, a new target would be shown and the cursor position would be set to the center of the grid area. The approach of resetting the cursor position after each target was to equalize the path for every subject.

As mentioned earlier multiple performance metrics were recorded during the target reaching test. One of these measures was throughput which depend on the index of difficulty for the targets, as stated in section 1.8.1. The ID was calculated for all targets in the 3-dimensional target space shown on figure 2.6.

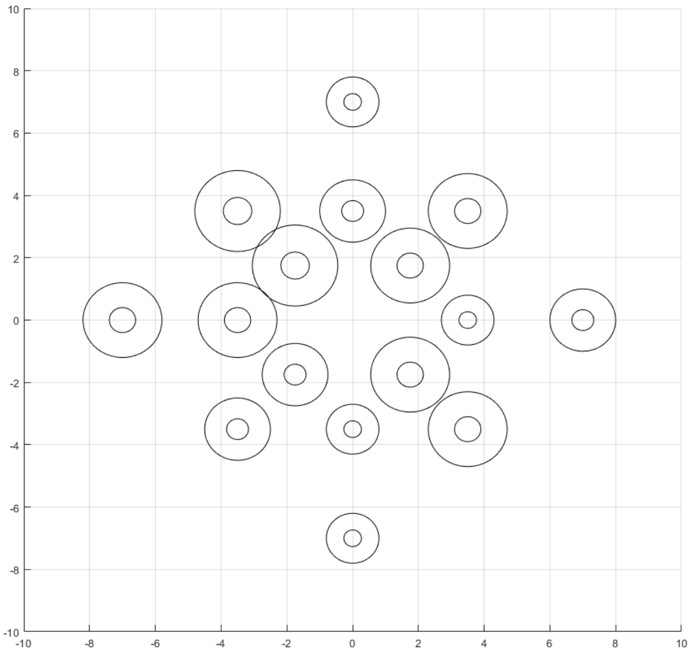


Figure 2.6: The implemented interface for the modified Fitts' Law task were all 16 targets to be reached are shown. Here it is seen that the smaller center circle has the same width in all targets. Furthermore the three different target sizes are shown.

The configuration of targets in the test resulted in 8 combinations of width and distance. The smaller circle has the same width in each target thereby possessing the same area in which the cursor center has to be within. The index of difficulty for the targets used in the modified Fitts' law test can be seen in table 2.1

Table 2.1: The index of difficulty used in the modified Fitts' law test.

Distance	Width	Index of Difficulty
28.0	$\frac{1}{3}$	6.41
24.5	$\frac{1}{3}$	6.22
22.0	$\frac{1}{3}$	6.01
18.5	$\frac{1}{3}$	5.82
16.0	$\frac{1}{3}$	5.61
13.0	$\frac{1}{3}$	5.32
12.5	$\frac{1}{3}$	5.27
9.5	$\frac{1}{3}$	4.88

A trace of the cursor movement throughout the whole test is recorded to decide the subjects path deviation from the optimal path to calculate the path efficiency and distances to targets. An example of a cursor trace is shown in figure 2.7.

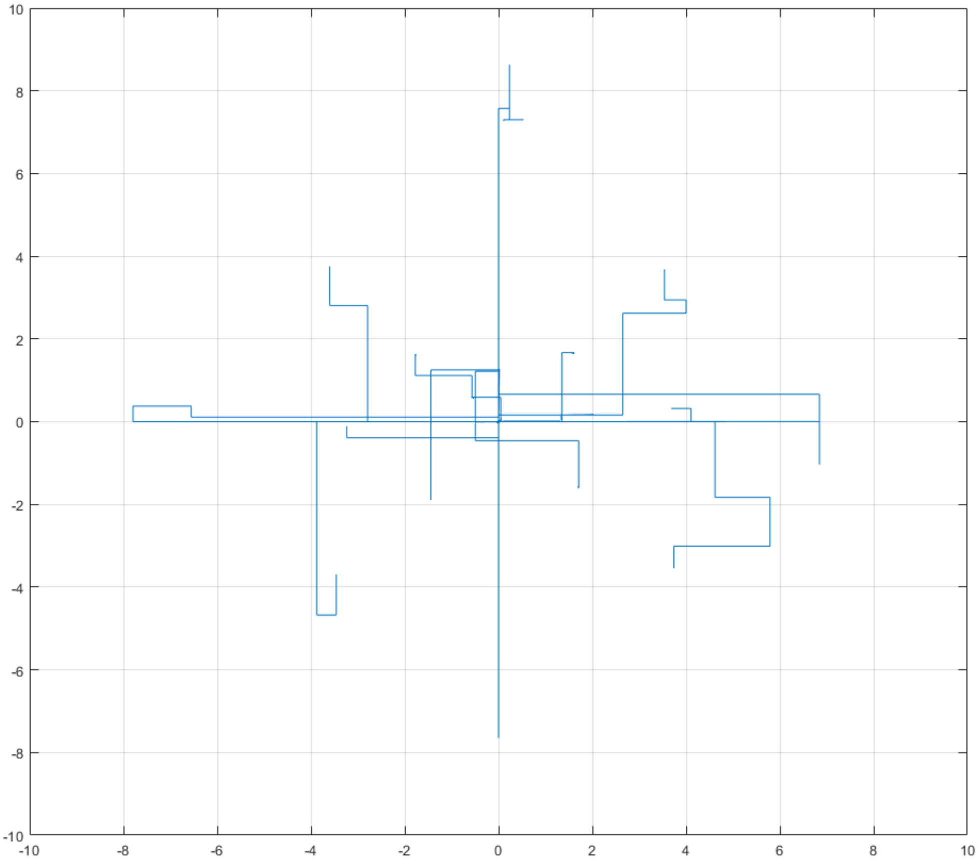


Figure 2.7: The trace of the cursor movement after a target reaching test. This information is not visible or shown to subjects.

The number of times a target is reached and exited without completing the dwell time, is recorded and used to calculate subjects overshoot. Similarly to tracking the travelled distance of the cursor inside the grid area, the travelled distance inside of each target is also recorded to calculate the stopping distance. The number or reached targets is recorded. The total amount of time possible to use in completing the target reaching test is: $16\ targets\ *\ 15s = 240s = 4\ min$.

Following the completion of recording target reaching test data from all subjects the performance metrics introduced in section 1.8.1 are calculated and presented in the Results: chapter 3.

3 | Results

In this chapter the results processed from collected data will be presented. All data processing have been done in accordance to earlier introduced theory and implementations of methods described in chapter 1 and chapter 2 respectively. The statistic results have been computed through a Friedman test, since the data proved to be non-parametric. Then a Tukey-Kramer test were done to correct for the problem of multiple comparison. All data processing have been compiled using MATLAB.

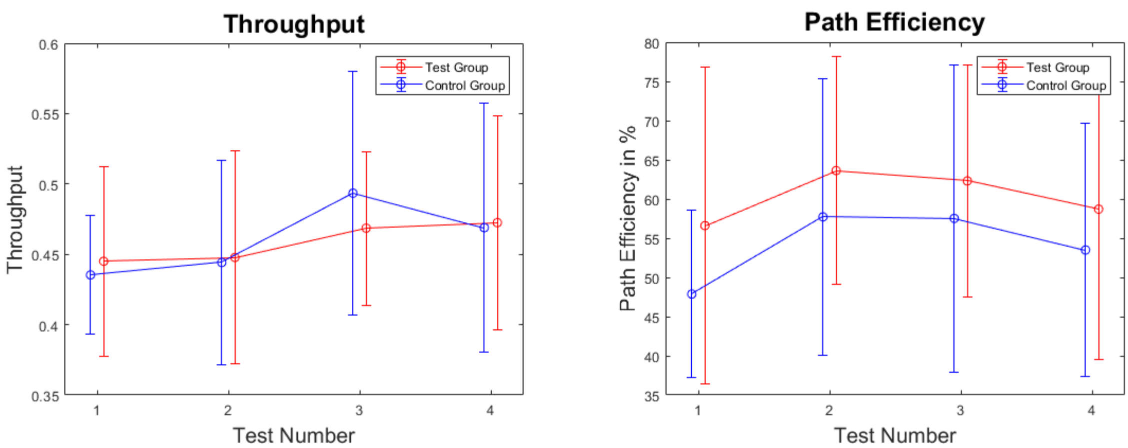
3.1 Fitts' Law results

This section will present the results acquired from the Fitts' Law target reaching test described in section 2.6. The test had five metrics which each express a parameter of subjects performance. Subjects were divided into two groups, one test group which received continuous classifications scores during user training, and a control group which received binary classification scores during user training. The results have been plotted for each metric over all four sessions, with mean and standard deviations.

3.1.1 Between test and control group

Here the results from the Fitts' Law targets reaching test between the two groups will be presented.

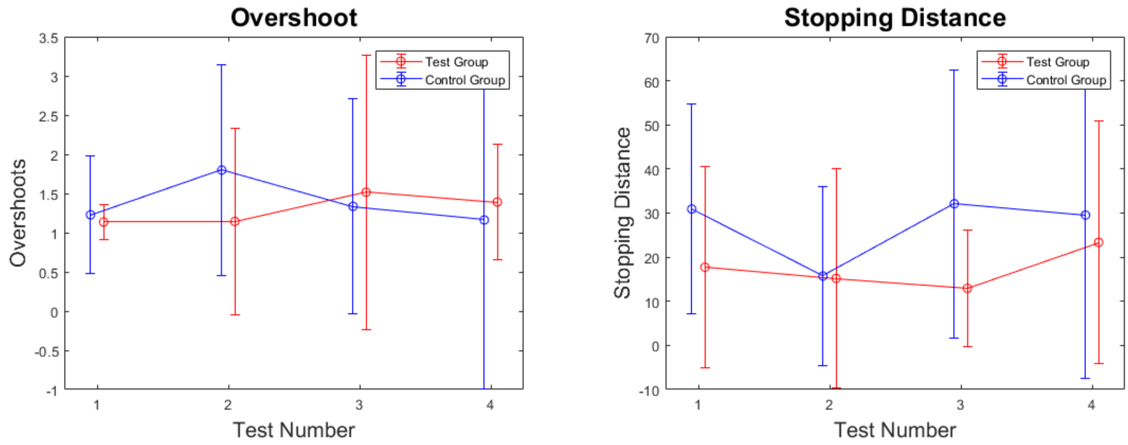
The aim of using the Fitts' Law target reaching test were to apply a method to qualitatively evaluate the performance of subjects before, during and after completing three sessions of user training. The information drawn from the metrics are described in section 1.8.1. No significant difference were found between any groups performance of any of the performance metrics ($p > 0.05$).



(a) Throughput metric for the Fitts' Law test between the test and control group. There is no significant difference between the groups ($p > 0.05$).

(b) Path efficiency metric for the Fitts' Law test between the test and control group. There is no significant difference between the groups ($p > 0.05$).

Figure 3.1: Presentation of the result metrics throughput and path efficiency.



(a) Overshoot metric for the Fitts' Law test between the test and control group. There is no significant difference between the groups ($p > 0.05$).

(b) Stopping distance metric for the Fitts' Law test between the test and control group. There is no significant difference between the groups ($p > 0.05$).

Figure 3.2: Presentation of the result metrics overshoot and stopping distance.

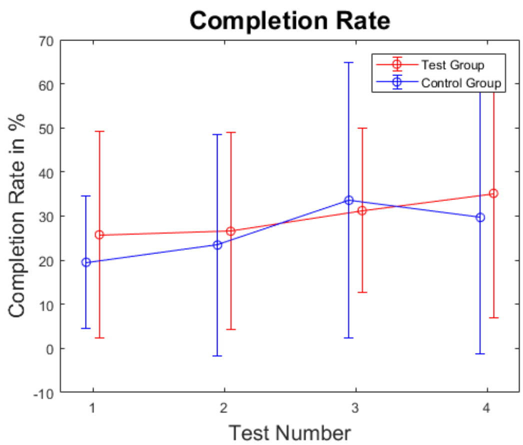


Figure 3.3: Completion rate metric for the Fitts' Law test between the test and control group. There is no significant difference between the groups ($p > 0.05$).

3.1.2 Across sessions

No significant difference in performance was found between the four tests with $p > 0.05$ for both the overall Friedmans test and the Tukey-Kramer correction to test difference between each session. This was the case for both the test and control group, which means there was no significant development of performance between either of the sessions for any group.

3.2 User training results

This section will present the results acquired from measurements taken during subjects user training sessions. During user training subjects were instructed to train the performance of the six chosen movements. During this training it were recorded the number of times subjects correctly performed a instructed movement to the contraction interval shown in the training interface. Statistical test were run on this data in a similar fashion as with the Fitts' Law results, described in section 3.1.

3.2.1 Total completion rate

During user training the total completion rate is defined by the number of times a subject correctly performed a movement and held the contraction bar at the given interval until completion. The user training interface is described in section 2.5.

A p-value of $p > 0.05$ was found for both groups in the Friedmans test comparing the performance over the three sessions. A significant difference was not found in any of the cases when performing Tukey-Kramer correction on the between-session Friedmans tests $p > 0.05$. This means that there was no significant development of performance in the training for any of the two groups.

3.2.2 Results for contraction levels

Friedmans test was applied to examine if there was a development in the ability to reach the different contraction levels within the three training sessions. A p-value of $p > 0.05$ was found for both groups, with the Tukey-Kramer correction yielding $p > 0.05$ for the comparison of the three sessions. This means there was no significant development in the ability to reach different intensities within the user training. No difference was found between the two groups ability to reach different intensities during training ($p > 0.05$).

3.2.3 Results for movement classification

Comparing the ability to reach different positions within the training showed no significant difference between the three sessions ($p > 0.05$), with the Tukey-Kramer correction resulting in $p > 0.05$ between all sessions for both the test and control group. This shows that there was no significant development in the ability to reach different positions during training.

A significant difference ($p < 0.05$) was found between the test and control groups ability to reach the closed hand movement, with a mean of 26.8 ± 13.5 for the test group and 38 ± 12.2 for the control group. No significant difference was found for any of the other movements when comparing the two groups ($p > 0.05$).

3.3 Data acquisition/separability results

In this section results from the data acquisition will be presented. The data from the data acquisition is used for training of the system to build the classification control for each individual subject. This data were classified into regions decided by the LDA classifier used as control scheme in this project. For each region there is a class consisting of a cluster of data points. In this section the results of the analysis of the clusters are presented.

3.3.1 Between cluster distances

For both groups the mean distance between the cluster centroids were calculated. There was found no significant difference in the development of cluster distances between the groups ($p > 0.05$). Likewise, the between cluster distances were tested between sessions, where no significance were found with a p-value of $p > 0.05$.

3.3.2 Within cluster distances

The mean distance from data points of a cluster to the cluster centroid were also calculated. Here there was found no significant difference for the test group ($p > 0.05$), but for the control group a significant difference with a p-value of $p < 0.05$ were found. Here the Tukey-Kramer correction showed that the significant difference were found between the control group's session one and three ($p < 0.05$), where the mean distance within clusters for session one were 502.02 ± 274.88, and for session three were 323.43 ± 171.13.

Results also show that the control group achieved a significant improvement when compared to the test group, in the mean distances within clusters ($p < 0.05$). In the third session the test group had a mean distance within clusters of 584.34 ± 250.02, while the control group had 323.43 ± 171.13.

4 | Discussion

5 | Conclusion

Bibliography

[1] Bart Peerdeman et al. "Myoelectric forearm prostheses: State of the art from a user-centered perspective". In: *The Journal of Rehabilitation Research and Development* 48.6 (2011), p. 719. ISSN: 0748-7711. DOI: 10.1682/JRRD.2010.08.0161. URL: http://www.rehab.research.va.gov/jour/11/486/pdf/peerdeman486.pdf.

[2] Frederic H. Martini, Judi L. Nath, and Edwin F. Bartholomew. *Fundametals of Anatomy&Physiology*. 2012. ISBN: 9788578110796. DOI: 10.1017/CBO9781107415324.004. arXiv: arXiv:1011.1669v3.

[3] Peter Konrad. "The abc of emg". In: *A practical introduction to kinesiological ...* April (2005), pp. 1–60. ISSN: 15583597. DOI: 10.1016/j.jacc.2008.05.066. URL: http://demotu.org/aulas/controle/ABCofEMG.pdf.

[4] Jeffrey R. Cram. *Cram's Introduction to Surface EMG*. Eleanor Cr. 2012. ISBN: 9780801026935. DOI: 10.1016/S0167-8922(09)70001-X. arXiv: arXiv:1011.1669v3.

[5] *Thalmic Labs Myo Armband*. 2013. URL: https://developer.thalmic.com/forums/.

[6] Marco Pozzo. "Biomedical Technology and Devices Handbook". In: 1949 (2004).

[7] Carlo J. De Luca et al. "Filtering the surface EMG signal: Movement artifact and baseline noise contamination". In: *Journal of Biomechanics* 43.8 (2010), pp. 1573–1579. ISSN: 00219290. DOI: 10.1016/j.jbiomech.2010.01.027. URL: http://dx.doi.org/10.1016/j.jbiomech.2010.01.027.

[8] Angkoon Phinyomark, Pornchai Phukpattaranont, and Chusak Limsakul. "Feature reduction and selection for EMG signal classification". In: *Expert Systems with Applications* 39 (2012). ISSN: 09574174. DOI: 10.1016/j.eswa.2012.01.102. URL: http://dx.doi.org/10.1016/j.eswa.2012.01.102.

[9] Lauren H Smith et al. "Determining the optimal window length for pattern recognition-based myoelectric control: balancing the competing effects of classification error and controller dela". In: 19.2 (2014), pp. 186–192. DOI: 10.1109/TNSRE.2010.2100828. Determining.

[10] Todd R. Farrell and Richard F. Weir. "The optimal controller delay for myoelectric prostheses". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 15.1 (2007), pp. 111–118. ISSN: 15344320. DOI: 10.1109/TNSRE.2007.891391.

[11] Erik J. Scheme, Bernard S. Hudgins, and Kevin B. Englehart. "Confidence-based rejection for improved pattern recognition myoelectric control". In: *IEEE Transactions on Biomedical Engineering* 60.6 (2013), pp. 1563–1570. ISSN: 00189294. DOI: 10.1109/TBME.2013.2238939.

[12] K Englehart and B Hudgins. "A robust, real-time control scheme for multifunction myoelectric control". In: *IEEE Trans Biomed Eng* 50.7 (2003), pp. 848–854. ISSN: 0018-9294. DOI: 10.1109/TBME.2003.813539. URL: http://www.ncbi.nlm.nih.gov/pubmed/12848352{\%}0Ahttp://ieeexplore.ieee.org/ielx5/10/27145/01206493.pdf?tp={\&}arnumber=1206493{\&}isnumber=27145.

[13] E Scheme and K Englehart. "A comparison of classification based confidence metrics for use in the design of myoelectric control systems". In: *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (2015), pp. 7278–7283. DOI: 10.1109/EMBC.2015.7320072.

[14] Yinfeng Fang et al. "Classifier-Feedback-Based User Training". In: 64.11 (2017), pp. 2575–2583.

[15] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. 2nd Editio. 2001, p. 680. ISBN: 0471056693. DOI: 10.1007/BF01237942. arXiv: 0-387-31073-8.

[16] Kianoush Nazarpour et al. "A note on the probability distribution function of the surface electromyogram signal". In: *Brain Research Bulletin* 90.1 (2013), pp. 88–91. ISSN: 03619230. DOI: 10.1016/j.brainresbull.2012.09.012. URL: http://dx.doi.org/10.1016/j.brainresbull.2012.09.012.

[17] R. A. Fisher. *The Use of Multiple Measurements in Taxonomic Problems*. 1936. URL: http://dx.doi.org/10.1111/j.1469-1809.1936.tb02137.x.

[18] Han Jeong Hwang, Janne Mathias Hahne, and Klaus Robert Müller. "Real-time robustness evaluation of regression based myoelectric control against arm position change and donning/doffing". In: *PLoS ONE* 12.11 (2017), pp. 1–22. ISSN: 19326203. DOI: 10.1371/journal.pone.0186318.

[19] J. M. Hahne et al. "Linear and Nonlinear Regression Techniques for Simultaneous and Proportional Myoelectric Control". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 22.2 (2014). ISSN: 1534-4320. DOI: 10.1109/TNSRE.2014.2305520. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6742730.

[20] Simon Bruun et al. "The effect of limb position on myoelectric prosthetic control using linear regression". In: (2017). URL: http://projekter.aau.dk/projekter/da/studentthesis/the-effect-of-limb-position-on-myoelectric-prosthetic-control-using-linear-regression(233b63c3-a4b0-4a8d-9978-17ae6df6c662).html.

[21] Zar Jerrold H. *Biostastical Analysis*. 5th ed. Pearson, 2009. ISBN: 978-0321656865.

[22] Anders Fougner et al. "Control of upper limb prostheses: Terminology and proportional myoelectric controla review". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 20.5 (2012). ISSN: 15344320. DOI: 10.1109/TNSRE.2012.2196711.

[23] Michael A. Powell, Rahul R. Kaliki, and Nitish V. Thakor. "User training for pattern recognition-based myoelectric prostheses: Improving phantom limb movement consistency and distinguishability". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 22.3 (2014), pp. 522–532. ISSN: 15344320. DOI: 10.1109/TNSRE.2013.2279737.

[24] Ann M. Simon, Blair A. Lock, and Kathy A. Stubblefield. "Patient training for functional use of pattern recognition controlled prostheses". In: 24.2 (2013), pp. 56–64. DOI: 10.1097/JPO.0b013e3182515437.Patient.

[25] Ning Jiang et al. "Myoelectric Control of Artificial Limbs: Is There a Need to Change Focus?" In: *IEEE Signal Processing Magazine* 29.5 (2012), pp. 150–152. ISSN: 1053-5888. DOI: 10.1109/msp.2012.2203480.

[26] L. Pan et al. "Transcranial direct current stimulation versus user training on improving online myoelectric control for amputees". In: *Journal of Neural Engineering* 14.4 (2017). ISSN: 17412552. DOI: 10.1088/1741-2552/aa758e.

[27] Paul M. Fitts. "The Information Capacity of the Human Motor System in Controlling the Amplitude of Movements". In: *Journal of Experimental Psychology* 47.3 (1954), pp. 381–391.

[28] Ernest N. Kamavuako, Erik J. Scheme, and Kevin B. Englehart. "On the usability of intramuscular EMG for prosthetic control: A Fitts' Law approach". In: *Journal of Electromyography and Kinesiology* 24.5 (2014), pp. 770–777. ISSN: 18735711. DOI: 10.1016/j.jelekin.2014.06.009. URL: http://dx.doi.org/10.1016/j.jelekin.2014.06.009.

[29] Richie Poulton et al. "Evaluation of Head Orientation and Neck Muscle EMG Signals as Command Inputs to a Human-Computer Interface for Individuals with High Tetreplegia". In: 360.9346 (2013), pp. 1640–1645. ISSN: 08966273. DOI: 10.1016/S0140-6736(02)11602-3.Association. arXiv: NIHMS150003.

[30] Ann M. Simon et al. "The Target Achievement Control Test: Evaluating real-time myoelectric pattern recognition control of a multifunctional upper-limb prosthesis". In: (2011). ISSN: 09652140. DOI: 10.1002/ana.22528.Toll-like. arXiv: NIHMS150003.

[31] E Scheme and K Englehart. "Validation of a selective ensemble-based classification scheme for myoelectric control using a three dimensional Fitts' law test". In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 21.4 (2013), pp. 616–623. ISSN: 1558-0210. DOI: 10.1109/tnsre.2012.2226189.

[32] John L. Semmlow and Benjamin Griffel. *Biosignal and Medical Image Processing*. 3rd ed. Taylor & Francis Group, 2014, p. 608. ISBN: 978-1-4665-6736-8.

[33] Ian M. Donovan et al. "Simple space-domain features for low-resolution sEMG pattern recognition". In: *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* (2017), pp. 62–65. ISSN: 1557170X. DOI: 10.1109/EMBC.2017.8036763.

A | Appendices

A.1 Experiment protocol for test subjects

Title of the project
Using confidence levels of movement recognition in user training to improve prosthesis control

Details on investigators
All investigators are 2nd semester biomedical engineering master students at Aalborg University.

Purpose
The purpose of this experiment is to train the subject in getting better at controlling a functional prosthesis. The subject will be training seven hand movements that is used in activating the prosthesis. In the experiment the prosthesis will be represented on a computer screen, where the subject will receive visual feedback on how the prosthesis interpret the hand movement performed by the subject. By receiving this visual feedback it is hypothesized that the user will get better at controlling the prosthesis over time.

Background
Electromyography (EMG), or muscle signals, is widely used for controlling functional lower arm prosthetics for transradial amputees. The ideal purpose of a functional prosthesis is to behave as functional as possible compared to a biological arm. Functional prosthetics that rely on pattern recognition-based control are becoming exceedingly good in performance in a clinical environment, due to highly optimized system control. However, still only one commercially available pattern recognition-based prosthesis exist. Users reject these functional prosthetics usually due to functionality issues when utilizing them in daily life tasks outside the clinical environment. Many improvements have been made in the area of system control, but another approach of improving the prosthetic control is by training the user. User training has only been explored scarcely in the research literature, thus, new techniques to improve the user's ability to control a prosthesis are yet untouched. This experiment will focus on training the user to improve prosthetic control on a fixed pattern recognition-based control system. The novel approach in this study is to provide the user with information on how well the system recognizes the performed movement during user training.

Research hypothesis
Exposing subjects to user training, in which confidence levels of movement recognition is used as feedback, will show statistically significant improvement in performance in a classification-based myoelectric prosthetic control scheme, when compared to subjects who have not had the same feedback during user training.

Session time
The experiment consist of three sessions, which are spread over three consecutive days; one session per day. Each session is estimated to have a total duration of 30-60 minutes.

Inclusion criteria
The subject needs to be:

- able bodied.
- above 18 years old.
- able to read, understand and speak Danish and/or English.

- assessed by the investigators to understand and perform the instructions given during the experiment.

Exclusion criteria
The subject must not have:

- diseases that might influence subject performance.

Experiment procedure

The experiment consists of three sessions containing different procedures as illustrated on figure A.1. The concept and chronology of each procedure is described below the illustration. During the experiment it is important that the subject is placed sitting on a chair, with the arm wearing the Myo armband (MYB) hanging relaxed down by the side of the body, as shown in figure A.7 on page 47 illustrating the experiment setup.

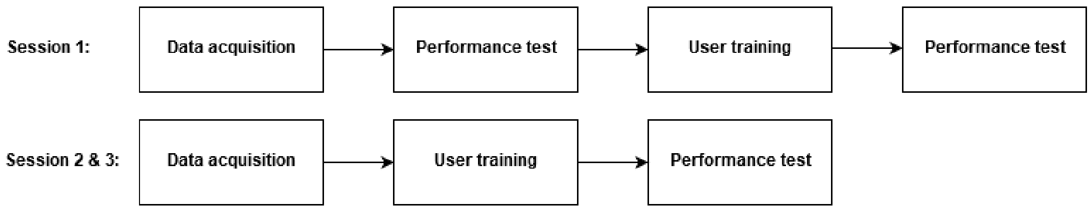


Figure A.1: Pipeline for the three sessions in the experiment and what procedures each session contains.

Data acquisition
For the myoelectric control system to be able to identify a performed movement as the movement that is actually performed, it needs information about how the movement looks when represented as a EMG signal. Thus, EMG data needs to be acquired from the forearm of the subject while the subject performs the movements that is used in the experiment, see figure A.6 on page 46. This data is fed to the control system to train the system to recognize each movement. In this experiment EMG data will be acquired from the subject with an EMG-electrode armband: MYB from Thalmic Labs. The chronology of this procedure is as follows:

1. Apply MYB on dominant forearm at the thickest part.
2. Synchronize MYB by performing wrist extension until three distinct vibrations are felt from the MYB.
3. Perform 15 seconds of maximum voluntary contraction (MVC) of instructed movement. The MVC is a contraction the subject is able to withhold in a constant intensity for the 15 seconds. Following the MVC the subject will be given a 30 seconds resting period to avoid muscle fatigue.
4. Perform three 15 seconds contraction trails of respectively 40%, 50% and 70% of MVC. During these contractions the subject will control a green marker representing the EMG signal and try to follow a trapezoidal trajectory as precise as possible. The trapezoidal trajectory consists of two 2.5 second transition phases and one 5 second plateau phase. Between each trial the subject will be given a 10 seconds resting period to avoid muscle fatigue.
5. Repeat step 3-4 until training data from all four wrist movements has been recorded.

An illustration of the interface used for data acquisition is shown in figure A.2.

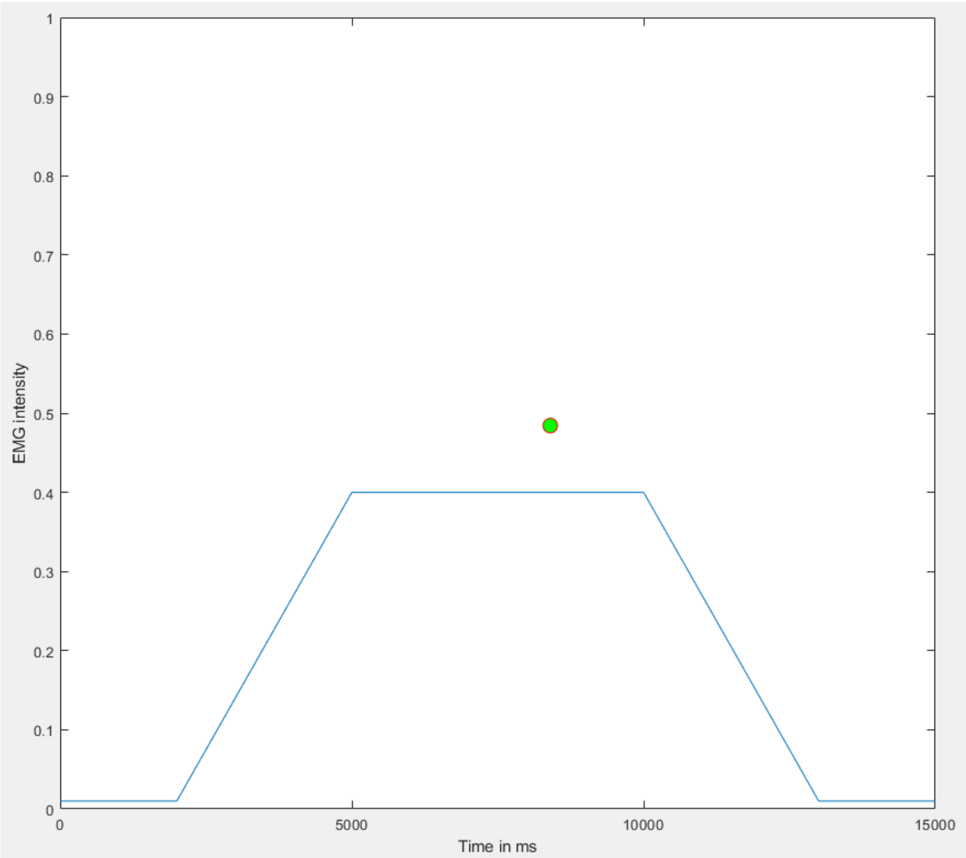


Figure A.2: Illustration of the data acquisition interface showing the trapezoidal trajectory and the green marker representing the EMG signal.

User training

The purpose of user training is for the subject to train the movements used in the performance test. During the user training the subject will train one movement at a time at different contraction levels. When training a movement, visual feedback in form of confidence levels on how well the control system recognizes movements, is shown in percentage in a bar plot. In addition, the level of contraction is shown in a horizontal bar above the other bar plot. When performing the instructed movement at the instructed level of contraction the horizontal bar plot will appear green; if it is outside the instructed level or if the system does not recognize the performed movement, it appears red. The aim for the subject is to reach and withhold the instructed contraction level with 100 % confidence for each movement. When the subject withholds the contraction level inside the instructed contraction level for 1 seconds with a 100 % confidence the colour of the horizontal bar will appear blue. This indicates that the subject is performing well. After it has appeared blue, the subject must return to rest and perform the movement again and try to reach the instructed contraction level with a 100 % confidence. An additional aim for the subject is to make the horizontal bar plot appear blue as many times as possible. The chronology of this procedure is as follows:

1. Perform instructed movement at 75-85 % contraction level for 30 seconds followed by 10 seconds rest.
2. Perform step 1 for the remaining movements.

3. Repeat step 1-2 at 55-65 % contraction level.
4. Repeat step 1-6 at 35-45 % contraction level.
5. Repeat step 1-6 at 15-25 % contraction level.

An illustration of the interface used for user training is shown in figure A.3.

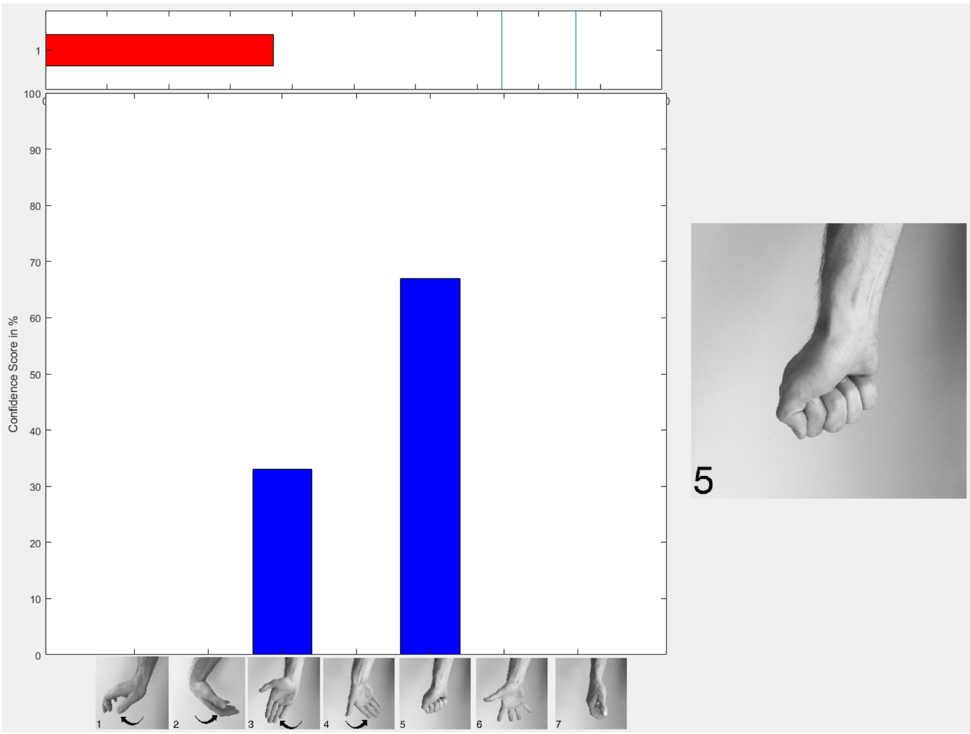


Figure A.3: Illustration of the user training interface showing the bar plot indicating the confidence level of movement recognition and horizontal bar plot indicating contraction level. The picture on the right side of the bar plot indicates which movement needs to be performed.

User Training

The purpose of user training is for the subject to train the movements used in the performance test. During the user training the subject will train one movement at a time at different contraction levels. When training a movement, visual feedback in form of which movement the control system recognizes, is shown in a bar plot. In addition, the level of contraction is shown in a horizontal bar above the other bar plot. When performing the instructed movement at the instructed level of contraction the horizontal bar plot will appear green; if it is outside the instructed level or the control system does not recognize the performed movement, it appears red. The aim for the subject is to reach and withhold the instructed contraction level for the instructed movement while the control system recognizes it. When the subject withholds the contraction level inside the instructed contraction level for 1 seconds while the control system recognizes it the colour of the horizontal bar will appear blue. This indicates that the subject is performing well. After it has appeared blue, the subject must return to rest and perform the movement again and try to reach the instructed contraction level while the recognition of the control system matches the performed movement. An additional aim for the subject is to make the horizontal bar plot appear blue as many times as possible. The chronology of this procedure is as follows:

1. Perform instructed movement at 75-85 % contraction level for 30 seconds followed by 10 seconds

rest.

2. Perform step 1 for the remaining movements.
3. Repeat step 1-2 at 55-65 % contraction level.
4. Repeat step 1-6 at 35-45 % contraction level.
5. Repeat step 1-6 at 15-25 % contraction level.

An illustration of the interface used for user training is shown in figure A.4.

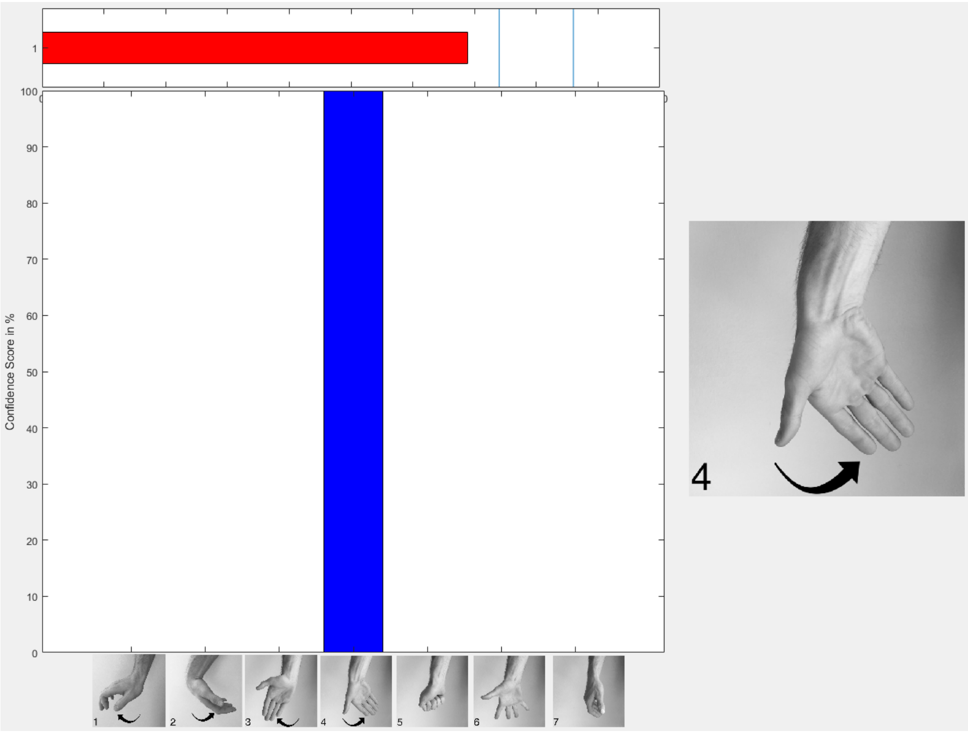


Figure A.4: Illustration of the user training interface showing the bar plot indicating which movement is being recognized and the horizontal bar plot indicating contraction level. The picture on the right side of the bar plot indicates which movement needs to be performed.

Performance test

The purpose of the performance test is to assess the subject's ability to control a prosthesis. Instead of doing a test with a real prosthesis a virtual alternative has been developed for this experiment. The prosthesis is represented as a red circular cursor with a black dot inside in a Cartesian coordinate system, which the subject can move as well as expand and shrink in size by performing the trained movements. The following bullets describe which movement corresponds to which action in the coordinate system:

- Extension moves the cursor right.
- Flexion moves the cursor left.
- Radial deviation moves the cursor up.
- Ulnar deviation moves the cursor down.

- Closed hand shrinks the cursor.
- Opened hand expands the cursor.
- Rest keeps the cursor still.

The performance test consists of a target reaching test, where the subject must reach 16 targets of different sizes and locations. A target consists of a circle with a smaller circle inside. Only one target will be visible at a time. For the subject to reach a target and make a new appear, the subject must center the black dot of the cursor in the small circle of the target and expand/shrink the cursor to fit the size of the outer circle of the target. The cursor will appear green, when located at the correct position. The subject must dwell the cursor in a target for 1 seconds for it to be reached. When the cursor has dwelled for 1 second, it will appear blue for 1 second to indicate that the target has been reached. If a target is not reached within 15 seconds a new target will appear. When a new target appears the cursor will reset its position the origin. The aim for the subject is to reach as many target as possible as quickly as possible. The subject is only able to perform one movement at a time, as trained in the user training. Thus, no simultaneously performed movements will be recognized by the control system. The chronology of this procedure is as follows:

1. Use 2 minutes to get acquainted with the test.
2. Reach the visible target.

An illustration of the interface used for the performance test is shown in figure A.5.

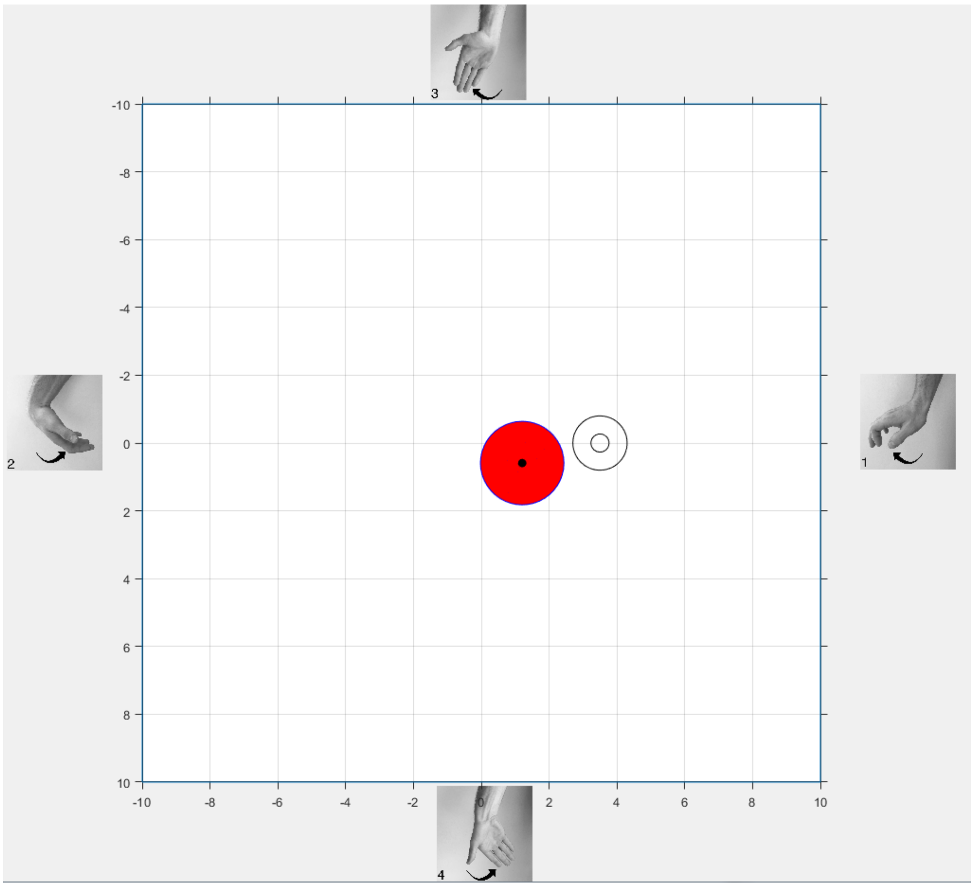


Figure A.5: Illustration of the performance test interface showing a target and the cursor representing the prosthesis output. The pictures on the axes indicate which movement must be performed to move the cursor in a certain direction.

Movements used in the experiment

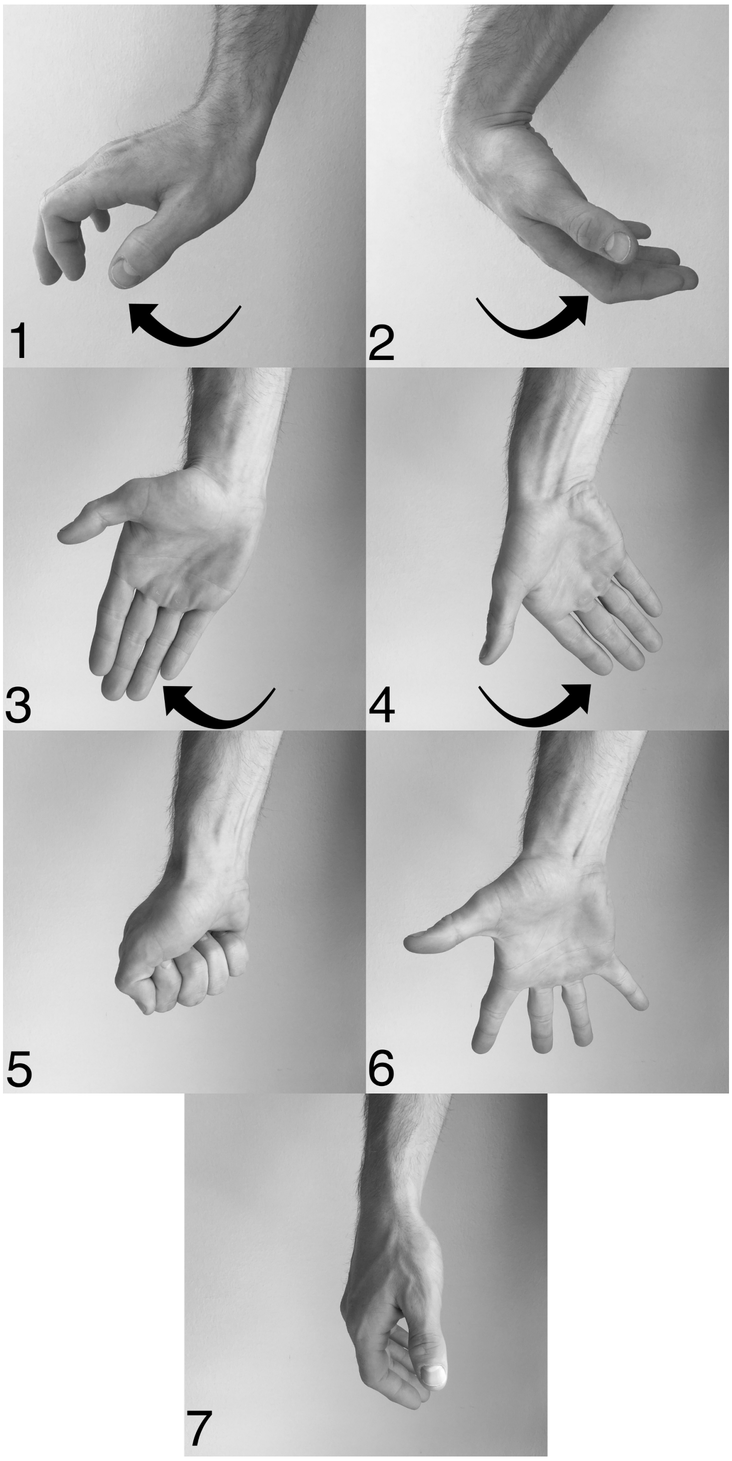


Figure A.6: Illustration of the movements used in the experiment. 1: extension, 2: flexion, 3: radial deviation, 4: ulnar deviation, 5: closed hand, 6: opened hand and 7: rest.

Experiment setup

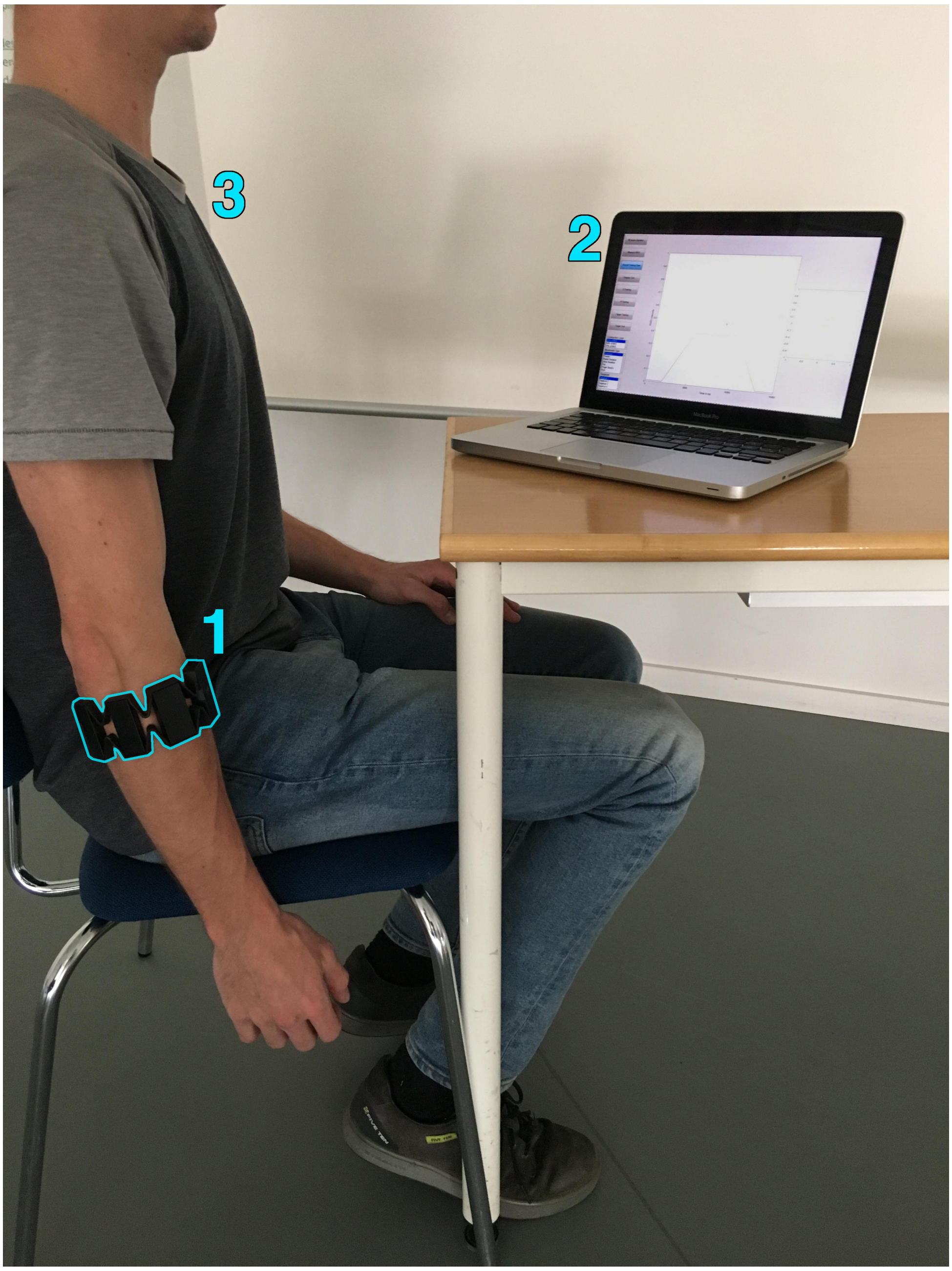


Figure A.7: Illustration of the experimental setup; 1: MYB, 2: computer with interface and 3: subject. The subject is seated facing the computer screen with the arm wearing the MYB hanging relaxed down the side of the torso.

A.2 Experiment protocol for investigators

Subject name:

Session number:

This protocol functions as a checklist for the investigators in the experiment "Using confidence levels of movement recognition in user training to improve prosthesis control". The checklist is used to ensure all steps in the experiment is performed correctly and that no steps will be neglected. The experiment consists of 3 session of 3-4 procedures in each session, as shown in figure A.8. The same procedures (data acquisition, user training and performance test) occur in all sessions and needs to be performed similarly each session. A checklist for each procedure is described in the sections below figure A.8.

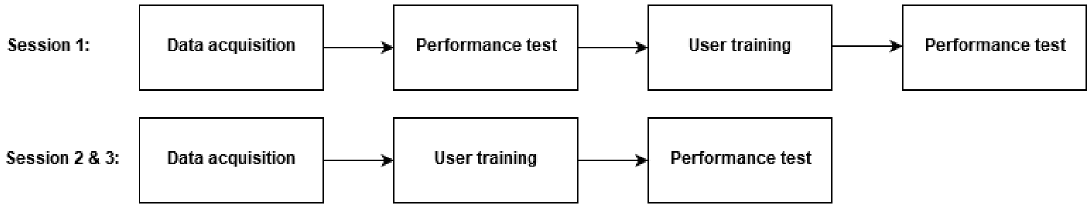


Figure A.8: Pipeline for the three sessions in the experiment and what procedures each session contains.

The instruction of the of aim the respective procedures and content and functions in the interfaces is based on the information written in the experiment protocol for test subjects. It is expected that the subject has read the experiment protocol handed out prior the experiment, but the information regarding the respective procedures is retold to verify that the subject has understood the following procedure.

Data acquisition

☐ Disinfect MYB with alco-swabs.

☐ Disinfect MYB application area of subject's dominant forearm with alco-swabs.

☐ Instruct subject to stand in anatomical standard position.

☐ Mark with a permanent marker the size of the main channel (channel with LED) of the MYB on the most lateral position of the thickest circumference of the subject's dominant forearm.

☐ Instruct subject in applying MYB with the main channel (channel with LED) located on the marked position. The MYB must be worn so that the LED is located as distally as possible. Add clips to tighten the MYB if necessary.

☐ Ensure that the main electrode-channel is placed correctly.

☐ Instruct subject to sit on a chair facing the screen showing the interface, with the arm wearing the MYB hanging relaxed lateral to the torso.

☐ Connect MYB in armband manager.

☐ Instruct subject in synchronizing MYB by performing extension until three distinct vibrations are felt from the MYB.

☐ Instruct subject in the movements about to be performed in the data acquisition.

- ☐ Instruct subject in performing an MVC; that the contraction must be steady during the 15 seconds.
- ☐ Record MVC for one movement. Observe spider plot meanwhile. If the activation pattern for the channels alters too much during the recording is to be discarded and a new must be acquired.
  - ☐ Extension
  - ☐ Flexion
  - ☐ Radial deviation
  - ☐ Ulnar deviation
  - ☐ Closed hand
  - ☐ Opened hand
- ☐ Instruct the subject in tracing the trapezoidal trajectory with the green cursor in different contraction levels of the MVC.
- ☐ Record contraction levels of MVC for one movement. Observe spider plot meanwhile. If the activation pattern for the channels alters too much during the recording is to be discarded a new must be acquired.
  - ☐ Extension: ☐ 40 %, ☐ 50 %, ☐ 70 %
  - ☐ Flexion: ☐ 40 %, ☐ 50 %, ☐ 70 %
  - ☐ Radial deviation: ☐ 40 %, ☐ 50 %, ☐ 70 %
  - ☐ Ulnar deviation: ☐ 40 %, ☐ 50 %, ☐ 70 %
  - ☐ Closed hand: ☐ 40 %, ☐ 50 %, ☐ 70 %
  - ☐ Opened hand: ☐ 40 %, ☐ 50 %, ☐ 70 %
- ☐ Build regressors for each movement and build classifier trained with all movements.

User training

- ☐ Instruct subject in aim of the user training, and explain the content and functions of the interface.
- ☐ Initiate user training.

Performance test

- ☐ Instruct subject in aim of the performance test, and explain the content and functions of the interface.
- ☐ Initiate performance test.
- ☐ Save all training data and performance measures in folder named after name of subject, session number and which experiment group the subject belongs to.

Comments: