



TALLINN UNIVERSITY OF TECHNOLOGY

SCHOOL OF ENGINEERING

Department of Electrical Power Engineering and Mechatronics

# **HUMAN-MACHINE INTERFACE FOR INTUITIVE ROBOT CONTROL**

## **INIMESE JA MASINA LIIDES ROBOTI INTUITIIVSEKS JUHTIMISEKS**

MASTER THESIS

Student: Ivan Kolodko

Student code: 386347MAHM

Supervisors: Ivan Vujaklija, Assistant Professor, EEA Department, Aalto University

Saleh Alsaleh, Engineer, EPEM Department, Taltech

Tallinn, 2020

## AUTHOR'S DECLARATION

Hereby I declare that I have written this thesis independently.

No academic degree has been applied for based on this material. All works, major viewpoints and data of the other authors used in this thesis have been referenced.

"21" May 2020

Author: 

/signature /

This is in accordance with terms and requirements

" 22 " May 2020

Supervisors: Ivan Vujaklija 

/signature/

Saleh Alsaleh .....

/signature/

Accepted for defence

"....." .....2020 .

Chairman of theses defence commission: .....

/name and signature/

# NON-EXCLUSIVE LICENCE FOR PUBLICATION AND REPRODUCTION OF GRADUATION THESIS

I, Ivan Kolodko (date of birth: 24/03/1996)

hereby

1. grant Tallinn University of Technology (TalTech) a non-exclusive license for my thesis

"Human-machine interface for intuitive robot control",

supervised by

Ivan Vujaklija, Saleh Ragheb Saleh Alsaleh,

1.1 reproduced for the purposes of preservation and electronic publication, incl. to be entered in the digital collection of TalTech library until expiry of the term of copyright;

1.2 published via the web of TalTech, incl. to be entered in the digital collection of TalTech library until expiry of the term of copyright.

1.3 I am aware that the author also retains the rights specified in clause 1 of this license.

2. I confirm that granting the non-exclusive license does not infringe third persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.



(signature)

21/05/2020

(date)

# Department of Electrical Power Engineering and Mechatronics

## THESIS TASK

**Student:** Ivan Kolodko 386347MAHM

**Study programme:** MAHM02/18 - Mechatronics

**main speciality:** Mechatronics

**Supervisors:** Ivan Vujaklija, Assistant Professor, EEA Department, Aalto University, +358504707760

Saleh Alsaleh, Engineer, EPEM Department, Taltech, 620 3201

### Thesis topic:

Human-machine interface for intuitive robot control

Inimese ja masina liides roboti intuiitivseks juhtimiseks

### Thesis main objectives:

1. To develop an optimal sEMG non-linear regression method for multiple DoF robotic system motion estimation based on artificial neural network model
2. To identify how different signal processing parameters impact control signal performance

### Thesis tasks and time schedule:

No	Task description	Deadline
1	Background research	01.02.2020
2	Setup development	01.03.2020
3	Data acquisition	01.04.2020
4	Data analysis	25.04.2020
5	Writing	22.05.2020

**Language:** English

**Deadline for submission of thesis:** "22" May 2020

**Student:** Ivan Kolodko



"21" May 2020

**Supervisors:** Ivan Vujaklija

/signature/  


"22" May 2020

Saleh Alsaleh

.....

"....."..... 2020

/signature/

**Head of study programme:**

..... ".....".....2020

/signature/

# TABLE OF CONTENTS

TABLE OF CONTENTS.....	5
LIST OF ABBREVIATIONS .....	7
1. INTRODUCTION .....	10
2. LITERATURE OVERVIEW.....	13
2.1. Elements of electrophysiological HMI .....	13
2.2. EMG generation .....	14
2.3. sEMG acquisition.....	17
2.4. sEMG segmentation and features extraction .....	21
2.5. Classification and regression approaches in myocontrol .....	25
2.6. Multilayer perceptron (MLP) neural networks .....	27
3. MATERIALS AND METHODS .....	31
3.1. Experimental hardware.....	31
3.2. Experimental software .....	32
3.3. Experimental procedure .....	33
3.4. Signal filtering .....	37
3.5. EMG Segmentation and Features Extraction .....	40
3.6. Acquired data sorting and preparation .....	41
3.7. Neural network data management .....	44
3.8. MLP neural network.....	47
3.9. Performance evaluation .....	50
3.9.1 Cross-validation.....	51
3.9.2 Variable feature sets .....	51
3.9.3 Channel reduction.....	52
3.9.4 Variable segmentation window size .....	53
4. RESULTS.....	54

5.	DISCUSSION .....	59
6.	CONCLUSION .....	62
	LIST OF REFERENCES .....	64
	APPENDICES.....	69
	Appendix 1 Data acquisition equipment used .....	69
	Appendix 2 Neural network hyper-parameters.....	71
	Appendix 3 Motion estimation results for cross-validation .....	72
	Appendix 4 Motion estimation results for variable feature sets .....	75
	Appendix 5 Motion estimation results for channel reduction .....	77
	Appendix 6 Motion estimation results for variable segmentation window size .....	80

## **PREFACE**

Using skin surface measurements of skeletal muscles electrical activity (surface electromyography or sEMG) as a human-machine interface (HMI) input, has found its way in a variety of robotic systems. It is also known as myocontrol. The control methods are based on mapping of sEMG input into robot kinematics output. Recent methods based on machine learning regression allow intuitive control of several degrees-of-freedom (DoFs) simultaneously. However, there is still no solution for robust control.

The aim of this thesis is to develop an optimal sEMG non-linear regression method for multiple DoF robotic system motion estimation using multilayer perceptron (MLP) neural network.

sEMG signal was acquired from one subject performing a series of virtual myocontrol tasks with high-density electrode grid. The signal was pre-processed and fed into an MLP model. The model performance was evaluated with cross-validation. The influence of following sEMG processing aspects on motion estimation accuracy were evaluated: 1) extracted features, 2) number of channels, 3) segmentation window size.

This work has designed an sEMG non-linear regression method for human motion intention estimation for multiple degrees-of-freedom robotic control applications. The designed method performance corresponds to state-of-the art control techniques. The highest performance was provided by using a set of root mean square (RMS), waveform length (WL), zero crossings (ZC), slope sign changes (SSC) features; 187 sEMG channels; and 64 samples segmentation window.

These results show that the designed motion estimation model and identified optimal signal processing parameters have potential to be used in HMI systems for intuitive robot control.

**Keywords:** human-machine interface, robotics, surface electromyography, artificial neural networks, master thesis

## **LIST OF ABBREVIATIONS**

ANN	Artificial neural network
BFGS	Broyden-Fletcher-Goldfarb-Shanno
CGD	Competitive gradient descent
DoF	Degree of freedom
EMG	Electromyogram
cEMG	Capacitive electromyography
IIR	Infinite impulse response
ML	Machine learning
MLP	Multilayer perceptron
NN	Neural network
OSS	One-step secant
RMS	Root mean square
sEMG	Surface electromyogram
SSC	Slope sign changes
SOA	State-of-the art

TMG Tactile myography

WL Waveform length

ZC Zero crossings

# 1. INTRODUCTION

Contemporary society relies on various types of machines in every field of human endeavor from household activities to spaceflight. The machines are defined as “any mechanical or electrical device that transmits or modifies energy to perform or assist in the performance of human tasks” [1]. A user operates a machine by means of human-machine interface (HMI).

HMI enables a person to profoundly interact with a device, machine or a system [2]. The basic idea of the interface is to translate a user input into commands for a machine and to present the results to the user. An acknowledged example of an HMI device is a computer mouse [3]. In this system the user input, which is the arm displacement detected with a mouse, is converted to a command for a computer to change the pointers position on a monitor (which is another HMI device). Current HMIs take place in a huge variety of applications from changing a room light intensity to operating industrial processes in smart factories [1].

One trend of modern HMI which aims to allow contactless control, uses human electrophysiological signals as input [1]. These signals can be sensed from different body tissues such as muscles (electromyography), nerves (electroneuronography) or brain (electroencephalography). One advantage is that some of these signals, using corresponding measuring techniques, can be sensed at human skin surface. This provides a measuring device with wearability which makes it suitable for reliable and convenient HMI.

Over the past decades, using skin surface measurements of skeletal muscles signal (surface electromyography or sEMG) as an HMI input has found its way in a vast variety of applications due to its intuitiveness and acquisition simplicity [4]. This approach is widely spread in biomedical engineering, bionics, and robotics. These fields develop HMI applications that may be remarkably valuable for physically challenged persons by providing them with functionality restoration. The applications use sEMG to control assistive devices such as wheelchairs or prostheses. Other use cases where such sEMG input is compelling are controlling of smart glasses or exoskeletons in production or military sites [1], [5]–[7].

In recent years, substantial progress has been made in robotic systems motion control with sEMG signal input known as myocontrol. This control implies mapping muscle signals into robot actuator control signals. In contemporary systems this mapping employs machine learning (ML) approaches. In order to make sEMG-based control intuitive, the mapping should

provide output (i.e. robot kinematics) proportional to the input (i.e. muscles contraction intensity). The output can be position, velocity, or acceleration. It should also provide operation of several degrees-of-freedom (DoFs) simultaneously. Even though the studies in this field are being carried out for several decades, there is still no solution for robust real-time simultaneous proportional control [8], [9].

State-of-the-art (SOA) solutions employ ML algorithms based on pattern classification methods that show performance of 95% and more. However, these systems are not popular with artificial limbs end users due to these robotic devices unnatural behavior. This behavior follows from the contradiction between these control algorithms and limbs natural movements. Natural movements are continuous and engage multiple physiological degrees of freedom (DoFs) at a time. On the other hand, SOA approaches only allow one DoF control at a time. Other approaches use direct control and state machines-based mapping but the number of functions that can be controlled is limited to three in these cases. [8]. These problems have been overcome by using ML regression approaches, which provide direct sEMG mapping into kinematics and hence, allow simultaneous and proportional control [10]. However, there is still no consensus on design factors that should be used in regression-based myocontrol systems to achieve the highest motion accuracy.

To address this drawback, this thesis aims to develop an optimal sEMG non-linear regression method for multiple DoF robotic system motion estimation based on artificial neural network model. In order to accomplish this goal, the thesis will design neural network-based algorithm which uses laboratory-recorded sEMG signal as an input and outputs the controlled device kinematic signals. Additionally, it will identify how different signal processing parameters impact control signal performance.

The scope of this thesis will be restricted to multilayer perceptron neural networks (MLP). Other types of neural networks (e.g. convolutional NNs, recurrent NNs) are not considered since they don't show significant impact in sEMG-driven motion estimation comparing to MLPs [9], [11]. Moreover, this work's scope will be limited to identifying the impact of sensors number on the estimation performance. Furthermore, only the impact of signal segmentation window size and features type on output signal performance will be evaluated. Other signal processing concepts are not considered. For the sake of generality, this work is not linked specifically to any particular kinematic quantity but uses kinematics term in general.

The remainder of this thesis is organized as follows. Chapter 2 presents the literature review of EMG-based HMI systems design. Chapter 3 describes how the system was developed. Chapter 4 presents the results of the system and Chapter 5 discusses them. Chapter 6 concludes the thesis by summarizing the completed work, identifying designed method drawbacks, and describing the system future development. Appendices describe information related to data acquisition hardware, designed neural network hyper-parameters and received motion estimation plots.

## 2. LITERATURE OVERVIEW

In order to achieve the thesis aim, the current situation in myocontrol field was investigated. This chapter presents bio-HMI systems field theoretical overview.

### 2.1. Elements of electrophysiological HMI

As was mentioned in the introduction, the designed HMI uses electrophysiological signal to control a robotic device. In general, electrophysiological HMI systems signal processing can be divided into 4 stages [4]:

1. Data acquisition and data segmentation
2. Feature extraction
3. Classification
4. Controller

This setup is presented at Figure 2.1.

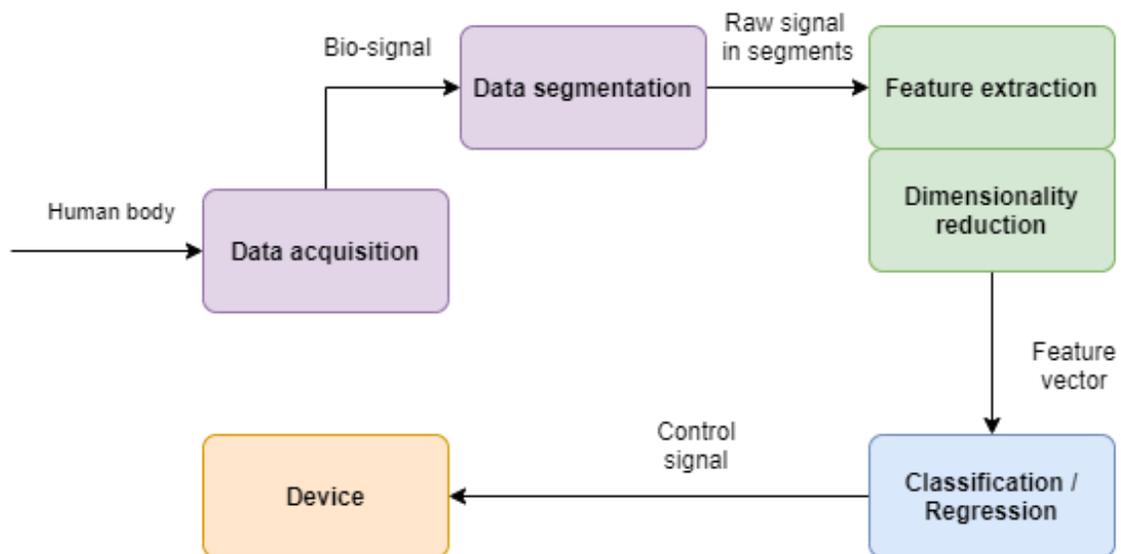


Figure 2.1 bio-HMI main stages. Firstly, the signals are acquired from a human body. After that the raw signal is segmented into separate portions. This is followed by signal features extraction and dimensionality reduction. This feature vector is then proceeded into classification or regression algorithm. This produces the control signal which is fed into controlled device [4]

During the acquisition filtering is induced. This is done to reduce the undesirable effect of noise produced both by human body and acquisition electrical devices [12].

Signal segments, produced after acquisition, are made up of specified number of samples which are used to extract signal features vector. By realizing the dimensionality reduction technique, unnecessary information may be excluded from the feature vector leading to its reduction. This reduced feature vector is then used as an input of a classifier or regressor which maps it into control signal by means of different mathematical algorithms. Lastly, control signals are transferred into controller to provide commands to a machine [4], [13].

## **2.2. EMG generation**

In order to design an HMI device relying on EMG signal as an input, it is necessary to understand the fundamental concepts of EMG generation.

EMG signal generation starts at brain motor cortex where series of neural signals called cortical action potentials are generated [14]. These action potentials are transferred to the motor neurons located in spinal cord by means of nervous system fibres called axons. A motor neuron transfers this stimulus further to muscle fibers by means of motor neuron axon. One motor neuron can be connected to several muscle fibers. The combination of a motor neuron, its axon and muscle fibers to which the axon is connected is called the motor unit, which is the smallest controllable muscular unit (Figure 2.2) [15], [16], [17].

The action potential signal is transferred through the nervous system to the muscle fibers basing on the electro-chemical principles. Outside a neuron and its axon, there is a positively charged electrolyte containing Sodium ions. Inside of them, there is an electrolyte containing positively charged Potassium ions and negatively charged protein cells. These electrolytes are separated with the neuron membrane providing a potential difference between inner and external surfaces of a cell, making it polarized.

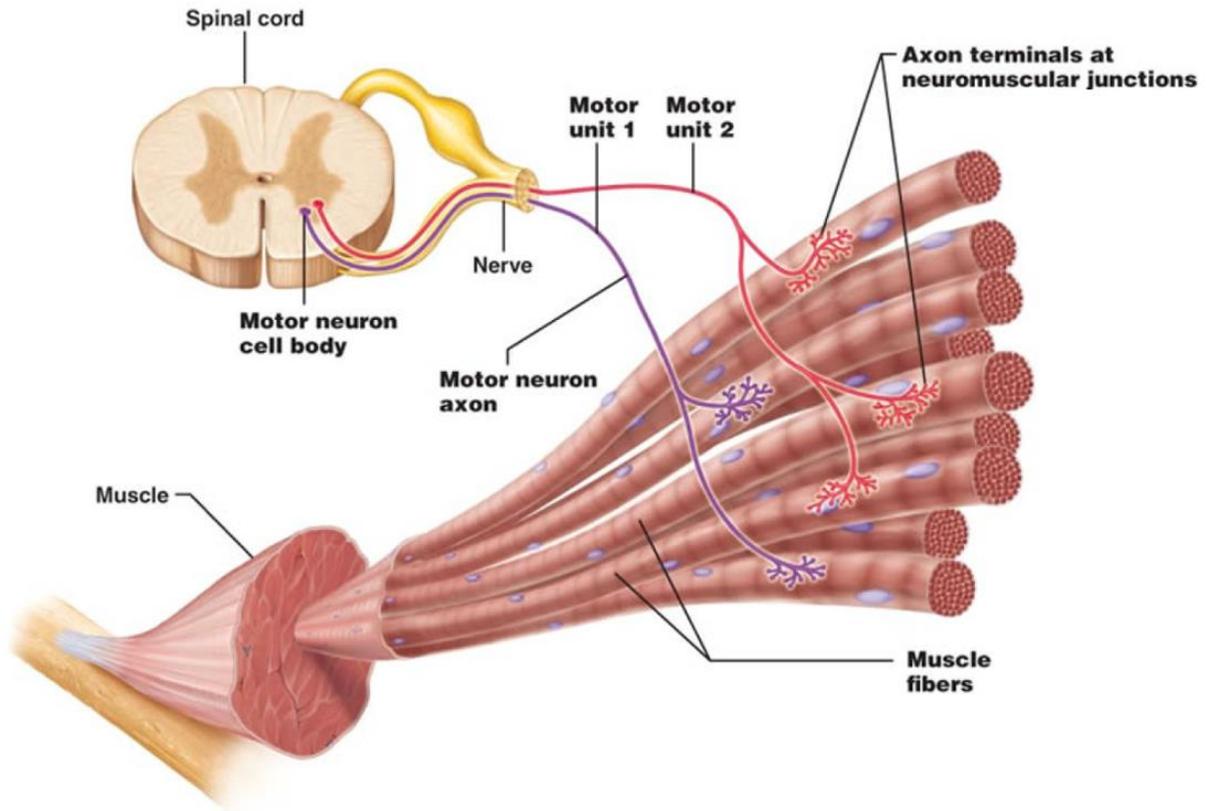


Figure 2.2: Motor unit illustration. Axons of motor neurons extend from the spinal cord to the muscle. There each axon divides into a number of axon terminals that form neuromuscular junctions with muscle fibers scattered throughout the muscle. Adapted from: MARIEB, ELAINE N.; HOEHN, KATJA N., HUMAN ANATOMY & PHYSIOLOGY, 9th, ©2013. Reprinted by permission of Pearson Education, Inc., New York, New York

The cell contains “gates” at membranes which open due to different stimuli (electrical, chemical or mechanical). They make ions move across a membrane changing the voltage across it and making the cell to depolarize. When the cell is depolarization level is above the threshold, it causes a chain reaction: a depolarized region of the a membrane causes “gates” opening in further regions of axons membrane, this, in turn, causes the further gates opening and depolarization, making the signal to transmit through nerves and axons[18], [19], [20]. This process is illustrated at Figure 2.3. The plot of an action potential with its different phases is presented at Figure 2.4. These action potentials are further transmitted to muscle fibers, as was described previously.

Transmitted to muscle fibres, action potential starts an electrochemical reaction there. These reactions provide muscle contractions. The contraction intensity is proportional to a motor neuron fire rate and number of motor units excited [21].

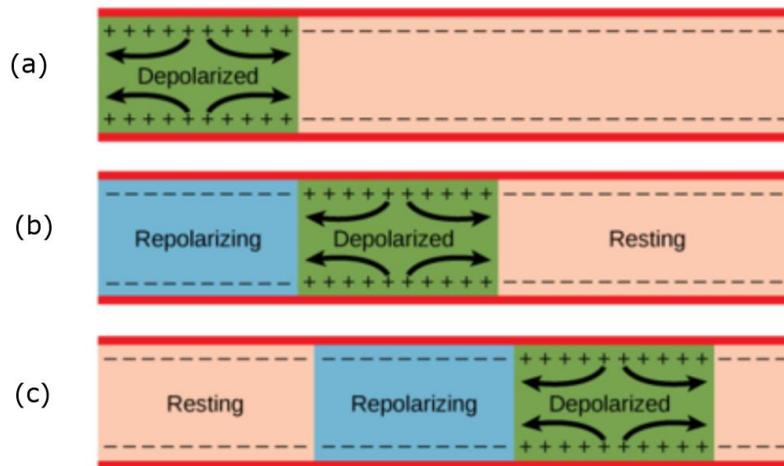


Figure 2.3: Action potential transmission via axon. (a) In response to a signal, the soma end of the axon becomes depolarized. (b) The depolarization spreads down the axon. Meanwhile, the first part of the membrane repolarizes. Because sodium channels are inactivated and additional potassium channels have opened, the membrane cannot depolarize again. (c) The action potential continues to travel down the axon. Adapted from [18]

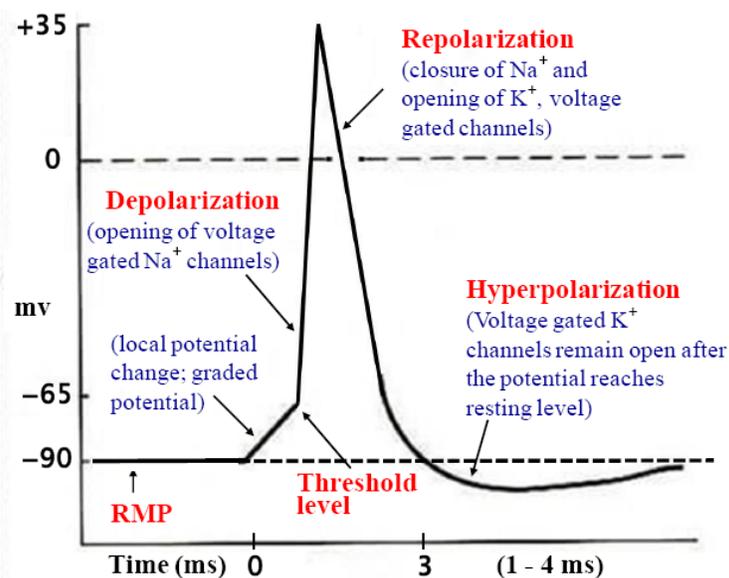


Figure 2.4: Action potential plot with phases. When activation exceeds a threshold level due to local potential change, the depolarization phase starts by opening sodium voltage gates. When they are closed, potassium gates open, starting the repolarization phase and decreasing the voltage. This voltage drop continues until the hyperpolarization phase when voltage gated potassium channels remain open. After that the potential returns to its resting level. Adapted from [18]

Surface EMG (sEMG) technique is used in order to sense the action potentials transmitted into muscle fibers. As far as hundreds of motor neurons are usually connected to a muscle, surface electrodes senses not separate action potentials illustrated at Figure 2.4 but a combination of them. Individual action potentials combined in sEMG signal are attenuated due to volume conduction issues. Body tissues located between signal source and sensor such as bones, muscles, fat, skin have different electrical resistance and capacitance. Original action potentials change their amplitude and frequency during transmission through these tissues [9].

### **2.3. sEMG acquisition**

In order to acquire sEMG, a special equipment, which is described in this section, should be used. This equipment includes sensors and signal amplifier [4]. This section also describes the procedure of sensor placement and experiment setup.

**Sensors** used in sEMG-based applications provide non-invasive technique for detection and measurement of EMG signal. There are also invasive electrodes, but they are not reliable for convenient HMI and are mostly used for clinical studies [22].

sEMG sensors are based on conductive measurement principle which supposes direct physical contact of electrodes and user skin [23]. There are sensors utilizing other measurement principles but they meet serious drawbacks such as low signal-to-noise ratio for capacitive EMG (cEMG) sensors and signal drifting for tactile myography (TMG) sensors [24], [25].

sEMG electrodes can be passive or active. Passive ones should be connected to external circuitry in order to provide muscles signal acquisition. Active electrodes contain in-house pre-amplifier. They provide pre-conditioned signal to the rest of the circuitry [22].

Electrode material should ensure good skin-electrode contact, low impedance and stable behavior in time. The most of commercially available electrodes are manufactured of silver, silver chloride, gold, tin and stainless steel [23].

There are three types of sEMG electrode configurations: monopolar, bipolar, and multipolar. Monopolar implies using single skin electrode with respect to reference electrode placed on electrically neutral tissue such as bone. This arrangement is not recommended because it detects all electrical signals under the electrode surface, including muscle crosstalk, leading to highly noisy recordings [22], [23]. Bipolar configuration uses two EMG detecting surfaces and a reference electrode. It implies a differential amplifier which outputs amplified difference of two recorded signals. This eliminates the common noise components in two signals overpassing the limitations of monopolar arrangement. Multipolar configuration uses more than two skin surfaces to detect the muscle signals. It further increases the acquired signal quality [22]. These configurations are presented at Figure 2.5.

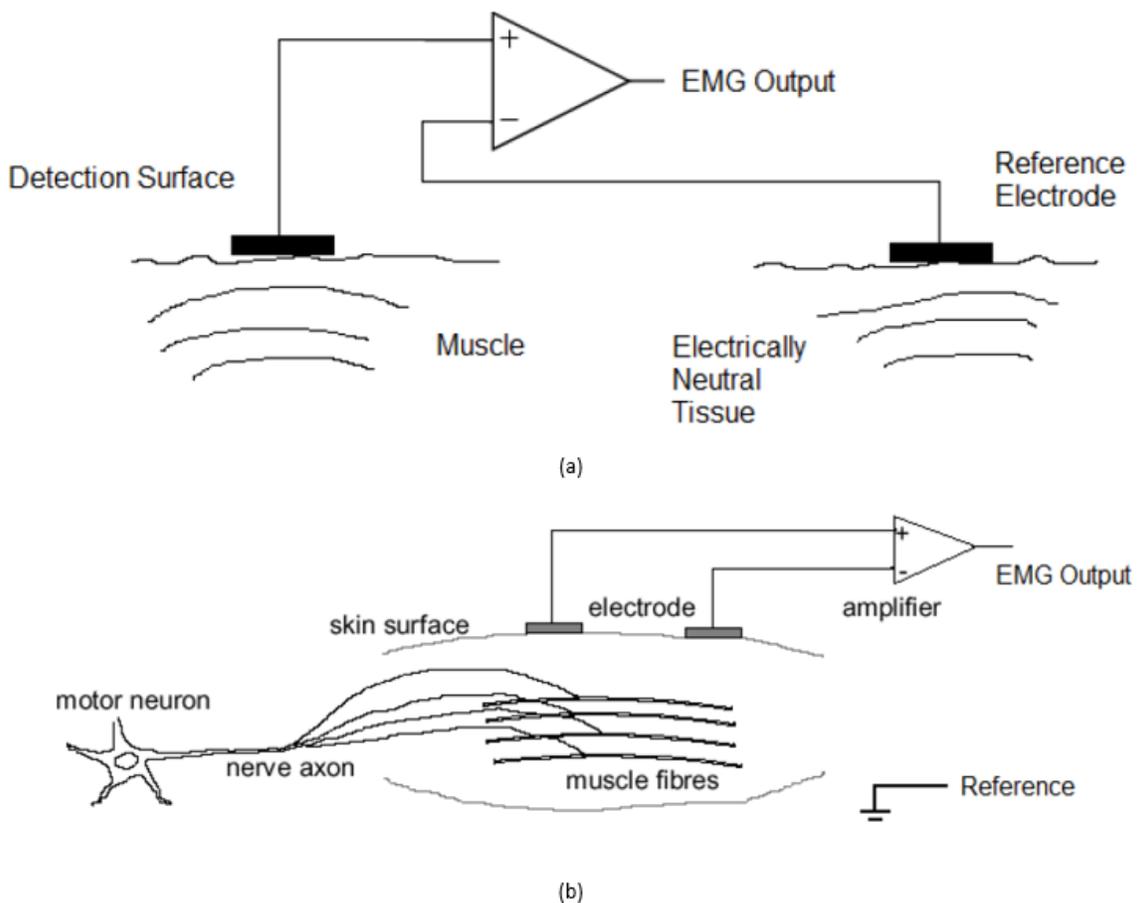


Figure 2.5 sEMG acquisition configurations. (a) Monopolar configuration. Detection surface signal is amplified with respect to reference electrode signal. (b) Bipolar configuration. The difference of two detection surface electrode signals is amplified with respect to a reference. Adapted from [22]

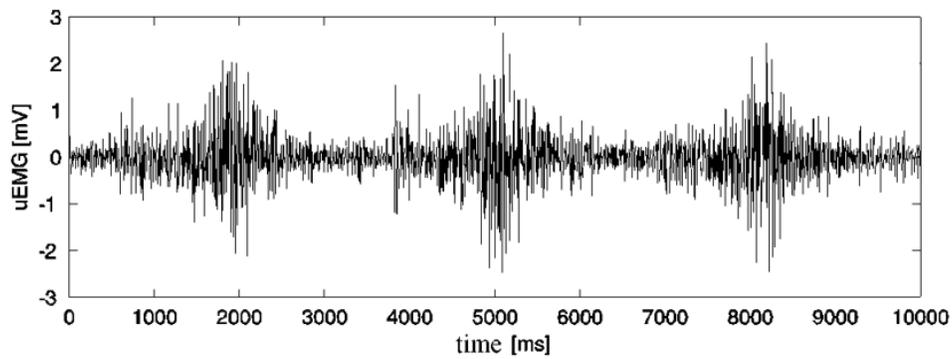
H.J. Hermens et al. [23] have developed a set of recommendations for setting sEMG sensors in related applications basing on different criteria: the electrodes conductive area should not exceed 10 mm in the direction of muscle fibers; the center-to-center distance between two electrodes should be 20 mm or one quarter of a muscle fiber length in case of applying to a small muscle; using pre-gelled electrodes made of silver (Ag) or silver chloride (AgCl) is recommended; the mechanical design of the sensor should allow fixed inter-electrode distance and light weight. All electric cables should be fixed in order to avoid pulling artifacts.

**Signal amplifier** is used in order to provide the signal detection and recording for further manipulations. The signals acquired are filtered, amplified, and digitally converted.

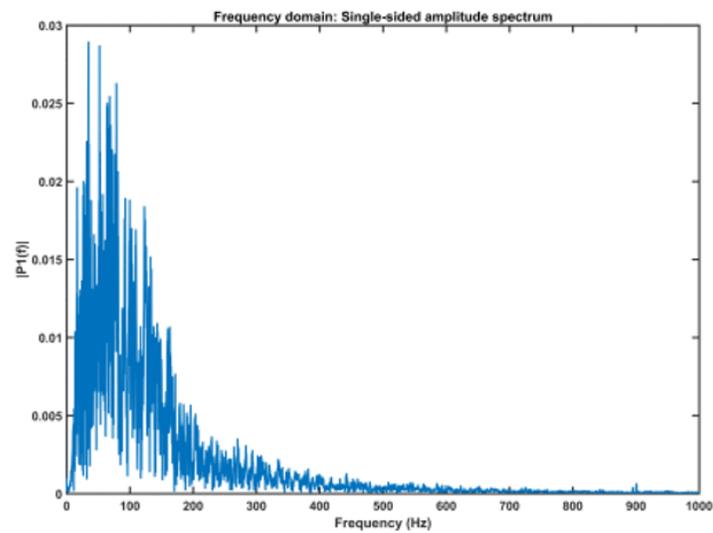
A typical sEMG representation in time and frequency domains is presented at Figure 2.6. The bandpass frequency for the sEMG signal ranges from 10 and 20 Hz (high-pass filtering) to between 500 and 1000 Hz (low-pass filtering) [12], [26]. High-pass filtering is necessary to reduce the effect of motion artifacts which are usually comprised of low-frequency components (less than 10 Hz). Low pass filtering is required to remove high-frequency components to avoid signal aliasing. To avoid the effect of power lines noise (50 or 60 Hz), a sharp notch filter should be used [26]. The described filters are recommended to be of the 2<sup>nd</sup> order to provide adequate data [12], [22].

Amplification is necessary to optimize the resolution of the recording or digitizing equipment. High-quality amplifiers have adjustable gains of magnitudes between 100 and 10000 to maximize sEMG signal-to-noise ratio. This range provides sufficient amplification of the whole sEMG dynamic range which is 50  $\mu$ V – 5 mV [26].

A 16-bit digitizing system is recommended to be used for sEMG signal processing [27]. Additionally, a convention recommends using sampling rate of 800 – 1000 Hz [28], however several researches claim that sampling rates should be at least 3 times higher than the maximum signal component frequency [29], [30].



(a)



(b)

Figure 2.6: sEMG in different domains. (a) Time domain. Raw surface EMG recording for three successive contractions of *m. rectus femoris*. Adapted from [31]. (b) Frequency domain. A typical single-sided spectrum of sEMG signal. Adapted from [32]

**Sensor placement procedure** for experimental data acquisition consists of next steps [23]:

1. Selection of sEMG sensors.
2. Skin preparation.
3. Positioning of a patient.
4. Determination of sensor location.
5. Sensors placement and fixation
6. Connection test.

## 2.4. sEMG segmentation and features extraction

The raw sEMG signal contains valuable information in a particularly useless form. In order to extract this information, the data is divided into representative segments. These segments represent some number of the recorded data samples. The number of data samples in each segment is known as "window length" [33].

There are different techniques of sEMG signal segmentation techniques shown in Table 2.1.

Table 2.1 sEMG Data segmentation techniques [4]

No	Segmentation technique	Short description
1	Disjoint segmentation	Separate data segments with pre-defined window length are used
2	Overlapped segmentation	Segments have a predefined fixed length. A new segment slides over the current segment.
3	Window with constant length and maximum value and mean absolute value of the EMG signal	A certain threshold is calculated based on maximum value and the mean absolute value of the whole EMG signal. Peaks over the defined threshold are considered as the candidate segments
4	Sliding time window and mean slope	If the mean slope of a signal exceeds a certain threshold, the beginning of a segment is detected. The end of the segment is detected when the total variation of a signal falls below another threshold

It was shown that disjoint segmentation and overlapped segmentation allow real-time applications due to reasonable response time and high performance during sEMG classification [4]. The overlapped technique has advantages comparing to disjoint segmentation because it shortens the response time and provides nearly the same accuracy [4].

There are three types of features that are used in EMG control applications [4], [34]: time domain, frequency domain, and time-frequency domain features.

Time-domain features provide a measure of waveform amplitude, frequency and duration. Frequency domain features are based on signal's estimated power spectrum density (PSD). Comparing to time domain feature, they require more computational time. Time-frequency domain features can utilize signal power in both time and frequency providing a more accurate

description on physical phenomena, however, as like as frequency features, they are computationally heavy due to required transformations. The vast of them is presented further [4].

Integrated EMG (IEMG):

$$IEMG_k = \sum_{i=1}^N |x_i|, \quad (2.1)$$

where  $x_i$  - the value of each part of the segment  $k$ ,

$N$  - the length of a segment (i.e. window length).

Mean absolute value (MAV):

$$MAV_k = \frac{1}{N} \sum_{i=1}^N |x_i| \quad (2.2)$$

Modified mean absolute value 1 (MMAV1):

$$MMAV1_k = \frac{1}{N} \sum_{i=1}^N w_i |x_i|, \quad (2.3)$$

$$w_i = \begin{cases} 1, & 0.25 N \leq i \leq 0.75 N \\ 0.5, & \text{otherwise} \end{cases}$$

Modified mean absolute value 2 (MMAV2):

$$MMAV2_k = \frac{1}{N} \sum_{i=1}^N w_i |x_i|, \quad (2.4)$$

$$w_i = \begin{cases} 1, & 0.25 N \leq i \leq 0.75 N \\ \frac{4i}{N}, & 0.25 N > i \\ \frac{4(i-N)}{N}, & 0.75 N < i \end{cases}$$

Mean absolute value slope (MAVS):

$$MAVS_k = MAV_{k+1} - MAV_k \quad (2.5)$$

Variance (VAR):

$$VAR_k = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2 \quad (2.6)$$

where  $\bar{x}$  - mean value of the segment  $k$ .

Waveform length (WL):

$$WL_k = \sum_{i=1}^{N-1} |x_{i+1} - x_i| \quad (2.7)$$

Root mean square (RMS):

$$RMS_k = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \quad (2.8)$$

Zero crossings (ZC):

$$\begin{aligned} ZC &= ZC + 1, & (2.9) \\ &\text{if } (x_i > 0 \text{ and } x_{i+1} < 0) \\ &\text{or } (x_i < 0 \text{ and } x_{i+1} > 0) \\ &\text{and } |x_i - x_{i+1}| \geq \epsilon \\ &\text{or } |x_i - x_{i-1}| \geq \epsilon, \end{aligned}$$

where  $\epsilon$  - a threshold,

Slope sign changes (SSC):

$$\begin{aligned} SSC &= SSC + 1, & (2.10) \\ &\text{if } (x_i > x_{i-1} \text{ and } x_i > x_{i+1}) \\ &\text{or } (x_i < x_{i-1} \text{ and } x_i < x_{i+1}) \\ &\text{and } |x_i - x_{i+1}| \geq \epsilon \\ &\text{or } |x_i - x_{i-1}| \geq \epsilon \end{aligned}$$

Willison amplitude (WAMP):

$$\begin{aligned} WAMP_k &= \sum_{i=1}^{N-1} f(|x_{i+1} - x_i|), & (2.11) \\ f(x) &= \begin{cases} 1, & x > \epsilon \\ 0, & \text{otherwise} \end{cases} \end{aligned}$$

Simple square integral (SSI):

$$SSI_k = \sum_{i=1}^N |x_i^2|, \quad (2.12)$$

Once the feature vector is obtained it is necessary to exclude the redundant data from it and reduce its dimensionality. There are two main ways of dimensionality reduction [4]. First one is feature projection. The idea of this strategy is to specify the best original features combinations to form a new feature set. It can be achieved by using principal component analysis (PCA) which produces an uncorrelated feature set by projecting the data onto the eigenvectors of the covariance matrix. The second is feature selection. This strategy is based on choosing the best subset of the original features vector according to some criteria. Usually, researchers use class separability criteria.

## 2.5. Classification and regression approaches in myocontrol

Myocontrol systems use sEMG features as input to machine learning models [4]. These models map those inputs into robotic system kinematics. The mapping implies individual robot degree-of-freedom activations correspondingly to sensed input [35]. The input varies basing on different human motions as they provide different muscle contractions and hence different sEMG recordings [10], [36].

Currently, the mostly used models involve classification and regression techniques [4], [10], [37]. Classification associates inputs to unique labeled categories (classes) of outputs [38]:

$$y_c = f_c(\mathbf{x}, \boldsymbol{\theta}_c), y_c \in Z, \quad (2.13)$$

where  $\mathbf{x}$  – new observation as a feature vector,

$y_c$  – the category that the new observation belongs to,

$f_c(\cdot)$  – trained classification function,

$\boldsymbol{\theta}_c$  – classification function parameter set,

$Z$  – the set of class labels.

In case of myocontrol, these classes may represent unique kinematic quantities or actions of controlled device degrees-of-freedom (DoFs) [34]. Discrete output makes classification models unsuitable for multiple DoFs robots simultaneous and proportional control [8]. The most commonly used sEMG classifiers are discussed at Table 2.2 [4]

Table 2.2 sEMG classifiers and their description

No	Classifier	Short description
1	Artificial neural networks	ANNs are able to represent both linear and non-linear relationships and learn them directly from data being modeled. This type of classification also meets the real-time constraints which is extremely important in control devices [4].
2	Linear discriminant analysis	It is used when a vast variety of features are used as a classifier input. Its advantage is reduction of computational complexity due to input dimensionality reduction [4]
3	Bayesian classifier	BC is based on applying the Bayes theorem with strong independence assumptions between the features [4].

Table 2.3 sEMG classifiers and their description continued

4	Fuzzy logic	FL classifiers are used for sEMG signals due to these signals unrepeatability. Fuzzy approaches employ tolerance of imprecision, uncertainty and partial truth which provides an opportunity to design a reliable control system [4]
5	Support vector machine	SVM is a kernel-based approach which is used in machine learning tasks demanding classification. It is also used in regression applications [4]
6	Hidden Markov model	HMM is a statistical model in which the system being modeled is assumed to be a Markov process with unobserved states [4]
7	K-nearest neighborhood	KNN is an algorithm that stores all available cases and classifies new cases based on a similarity measure [4]

In contrast to classification, regression models provide continuous outputs [38]:

$$y_r = f_r(\mathbf{x}, \boldsymbol{\theta}_r), \quad y_r \in R, \quad (2.14)$$

where  $\mathbf{x}$  – new observation as a feature vector,

$y_r$  – the output,

$f_r(\cdot)$  – trained regression function,

$\boldsymbol{\theta}_r$  – regression function parameter set.

The major difference to classification is that a regressor does not decide for a certain class but instead estimates a continuous output value for each DoF. This continuity feature of regression models makes them significant in myocontrol applications for multiple DoFs robotic systems [39]. It provides them with proportional and simultaneous control [10], [11], [40].

Several linear and non-linear regression models are usually used in sEMG-based HMI systems [10], [11], [39]. They are described at Table 2.4 sEMG regressors and their description

Table 2.4 sEMG regressors and their description

No	Regressor	Short description
1	Linear regression (LR)	Provides a linear a relationship between dependent variables (kinematics estimation in case of sEMG-based robotics control) and independent variables (sEMG signal) [39].
2	Mixture of linear experts	An extension of LR that provides non-linear relationship between dependent and independent variables. It is more physiologically corresponding to human natural motion [39].
3	Kernel ridge regression	Another extension of LR. Provides solving non-linear dependencies with linear methods by making space transformations [39].
4	Artificial neural networks	ANNs can represent both linear and non-linear relationships. Similar to classification, they are reliable for real-time applications [10], [39].
5	Non-negative matrix factorization	This method decomposes a non-negative input matrix into two low-rank matrices that are also non-negative. This decomposition makes this approach valuable for real-time control applications [35]

**Data** used for ML algorithms implementation is generally divided into two sets – training and test ones. Training dataset is used to fit model parameters during model learning and test set is used to assess the performance of fitted model [41].

Both these datasets consist of inputs and desired outputs, known as targets. As was mentioned earlier, in case of myocontrol the input data is sEMG features vector and the target data contains desired robot kinematic control signals [10], [25].

## 2.6. Multilayer perceptron (MLP) neural networks

A multilayer perceptron is a feedforward artificial neural network model that maps a set of input data into a set of appropriate outputs [42]. An MLP consists of at least three layers with each layer connected to the next one. A layer consists of units (also known as neurons or nodes). Each neuron in one layer connects to every neuron in the following layer with a certain weight  $w$ . Units perform a biased weighted sum of their inputs and pass this activation level through a transfer function (or activation function) to produce their outputs [43], [44].

The number of NN layers and neurons in them defines neural network architecture. The typical architecture of an MLP is presented at Figure 2.7.

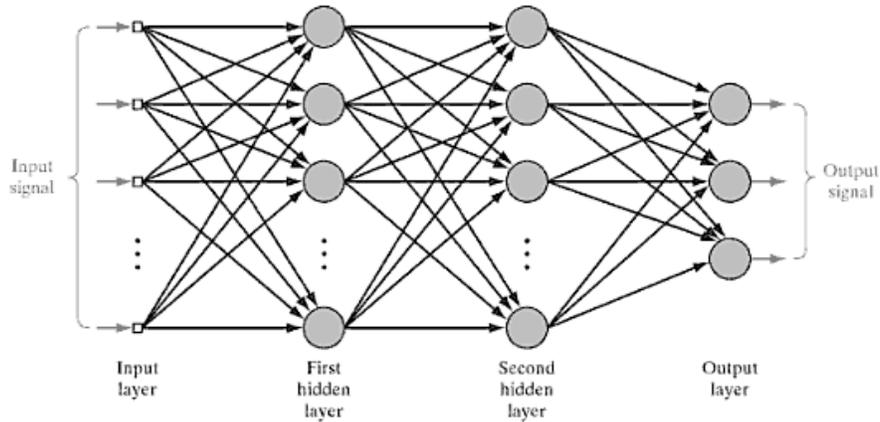


Figure 2.7 General MLP neural network architecture [45]. The network consists of neurons divided into four layers – one input layer, two hidden layer and one output layer

**A neuron activation function** should be differentiable in order to guarantee NN proper learning [43]. An equation of a neuron output is next [44]:

$$a_j^l = f\left(\sum_k w_{jk}^l \cdot a_k^{l-1} + b_j^l\right), \quad (2.15)$$

where  $a_j^l$  – activation of the  $j$ -th neuron in the  $l$ -th layer,

$f(\cdot)$  – neurons activation function,

$w_{jk}^l$  – synaptic weight of connection between  $j$ -th neuron in  $l$ -th layer and  $k$ -th neuron in  $(l-1)$ -th layer,

$a_k^{l-1}$  – activation of the  $k$ -th neuron in the  $(l-1)$ -th layer,

$b_j^l$  – bias of the  $j$ -th neuron in the  $l$ -th layer.

**A supervised training** is the process of providing a network with input data and relating this data with desired network output. Back propagation algorithm is used to train MLPs. The training consists of two phases [45]:

1. Forward phase includes propagation of the input signal in the network layer-by-layer until it reaches the output. During this propagation, the synaptic weights and biases values are fixed.

2. Backward phase produces the error signal by comparing the desired output of the network (target) with the actual output. This error signal propagates in the network layer-by-layer in the backward direction. During this phase, the values of weights and biases (also known as network parameters) are adjusted corresponding to the error signal.

**A cost function** (also known as performance function) is an equation which is used to quantify how close the NN output to its target [44]. It is used in backpropagation to compute the output layer error signal [46]. A cost function is of the form

$$C(W, B, x, t), \quad (2.16)$$

where  $W$  – network’s weights,

$B$  – network’s biases,

$x$  – input of a single training sample,

$t$  – desired output of that training sample.

In order to be used in backpropagation, a cost function must follow 2 requirements [44]:

1. It must be able to be written as an average over cost functions for individual training samples:

$$C = \frac{1}{n} \sum_x C_x, \quad (2.17)$$

where  $C_x$  – cost function for individual training sample,

$n$  – total number of training samples.

2. It must not be dependent on any activation values of a neural network besides the output layer values  $a^L$ .

**Optimization algorithms** (or training functions) are used in order to adjust NN weights and biases during backward phase of back propagation algorithm. These functions are iterative optimization techniques that minimize NN cost function [44]. The optimization problem which is being solved with backpropagation algorithm is formulated next way [47]:

$$\hat{v} = \arg \min_v C(v), \quad (2.18)$$

where  $\hat{v}$  – networks parameters estimated value that minimizes cost function  $C(\cdot)$ ,  
 $v$  – network's parameters (weights and biases)

Gradient descent method is an instance of an optimization technique. Given some cost function and its initial parameters gradient descent calculates the updated values of its parameters with next equation [48]:

$$v' = v - \eta \cdot \nabla C(v), \quad (2.19)$$

where  $v'$  - value of parameters on current iteration,  
 $v$  - value of parameters on previous iteration,  
 $\eta$  - learning rate,  
 $\nabla C(\cdot)$  - cost function gradient.

Generally, backpropagation algorithm iterates until the limit number of iterations (also called epochs) is reached. There are other criteria of algorithm termination which are listed in Section 3.8.

### 3. MATERIALS AND METHODS

This chapter describes the procedure of designing an MLP-based regression method used to map high-density sEMG signal into four robot degrees-of-freedom control signals.

#### 3.1. Experimental hardware

**Sensors.** As far as this work aims to evaluate how the number of data channels influences motion estimation performance, a high-density sEMG sensors grid was used. High-density sEMG sensors also provide flexibility in recording by reducing influence of different factors which can render individual signals unreliable [36], [35].

In this work three ELSCH064NM3 8-by-8 electrode matrices (OT Bioelettronica, Italy) were used. Each matrix has 64 electrodes (8 columns and 8 rows), ending up with 192 sEMG data channels in total. The inter-electrode distance is 10 mm. An electrode matrix is presented at Figure A1.1 . These electrode matrices were placed around subject's forearm.

**The signal amplifier** used for data acquisition during the experiment is a commercial biosignal amplifier Quattrocento (OT Bioelettronica, Italy). Its layout is presented at Figure A1.2. During the experiment it was connected to a PC. Computer-amplifier setup scheme is presented at Figure A1.3. The amplifier was set to the following properties [49]:

- Fixed gain: 150 V/V;
- Maximum input range: 33 mV<sub>PP</sub>;
- Noise level referred to input: less than 4  $\mu$ V<sub>RMS</sub>;
- Input resistance:  $10^{11}$   $\Omega$ ;
- Common-mode rejection ratio: 95 dB;
- Output range: 0-5 V;
- Insulation voltage: 4000 V<sub>DC</sub>.
- High pass filter frequency: 10 Hz;
- Low pass filter frequency: 900 Hz;
- A/D conversion resolution: 16 bits;
- Sampling frequency: 2048 Hz;

**Channel adapters** AD1x64 (OT Bioelettronica, Italy) were used to connect sEMG electrode grids to signal amplifier inputs. They include instrumental pre-amplifier for each channel. The pre-amplifiers positive inputs receive signal from the electrode and negative signal is received from the reference signal [49].

To provide a signal grounding, two reference electrodes were also attached to subject's wrist by means of attachable strips. One of them grounds channel adapter pre-amplifiers and another was used for the rest of electrical circuitry grounding. The purpose of having a separate reference for adapter pre-amplifiers is to provide them with similar common mode signal of sEMG electrodes [49].

## **3.2. Experimental software**

In order to correlate subject muscle contractions sEMG signals with controlled degrees-of-freedom (DoFs) target kinematics, a special software developed in MATLAB™ (MathWorks, USA, ver. 2019b) was used during sEMG acquisition. It provided the subject with visual cue when it was necessary to articulate any motion. An example of a prompt is presented at Figure 3.1. The software was used to link user wrist motions and control signals of DoFs listed at Table 3.1. This kind of software allowed obtaining of target data without recording of real robot kinematics.

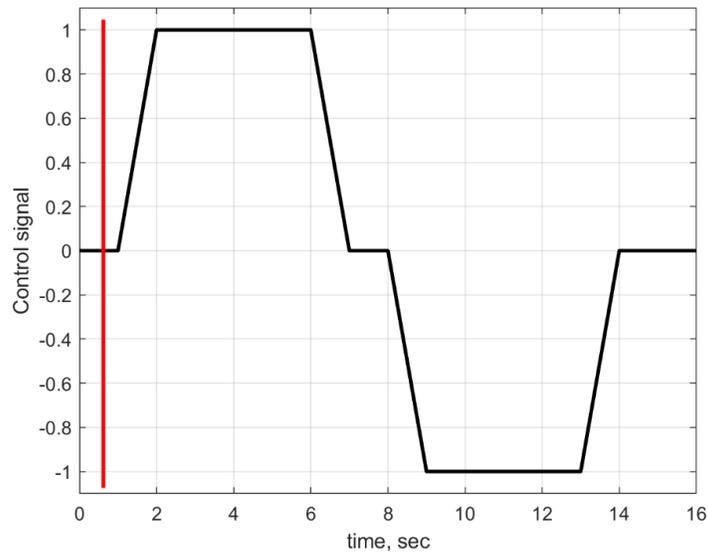


Figure 3.1 Visual prompt used to acquire data. Red line moved along a trajectory with two ramps prompting a subject to increase or decrease their muscles contraction intensity. Black line represented target kinematic control signals for individual degrees of freedom. Control signals varied in direction. Opposite control signal directions corresponded to antagonist user movements (see Table 3.1)

### 3.3. Experimental procedure

One able bodied male participant (23 years old, dominant left side) was seated comfortably in a chair, one meter in front of a PC monitor. Pre-sanitized and pre-gelled electrode matrices described in Section 3.1 were applied to subject's dominant forearm at a distance of one-third of a forearm length from an elbow. The hand was extended and pointed down at a side. Reference straps were also attached to patient forearms closer to a wrist. After that electrodes were connected to the signal amplifier by means of adapters. Then the experimental software described at Section 3.2 was run providing subject with visual cues when perform muscle contractions. One degree of freedom motion was articulated at a time. In total, four degrees of freedom control signals were extracted. The types of motions performed by user, their sequence, relations with controlled DoFs and number of trials are described at Table 3.1. sEMG and targets data related to each DoF control were divided into separate datasets in order to be used later during signal processing.

Table 3.1 User movements and corresponding controlled DoFs

No	User movement	Controlled degree-of-freedom	Number of attempts
1	Wrist flexion	DoF 1	12
	Wrist extension		
2	Radial deviation	DoF 2	13
	Ulnar deviation		
3	Pronation	DoF 3	13
	Supination		
4	Fist closing	DoF 4	12
	Fist opening		

This data acquisition procedure resulted in obtaining four sets of synchronized input and target data. Each set was related to individual DoF. These data sets are described at Figure 3.2.

Each user motion activated forearm muscles differently. Figure 3.3 presents those activation levels. They represent RMS values (2.8) calculated for the ranges of samples related to separate contractions which user executed during the experiment. These values were calculated for each of 192 channels of the electrode grid.

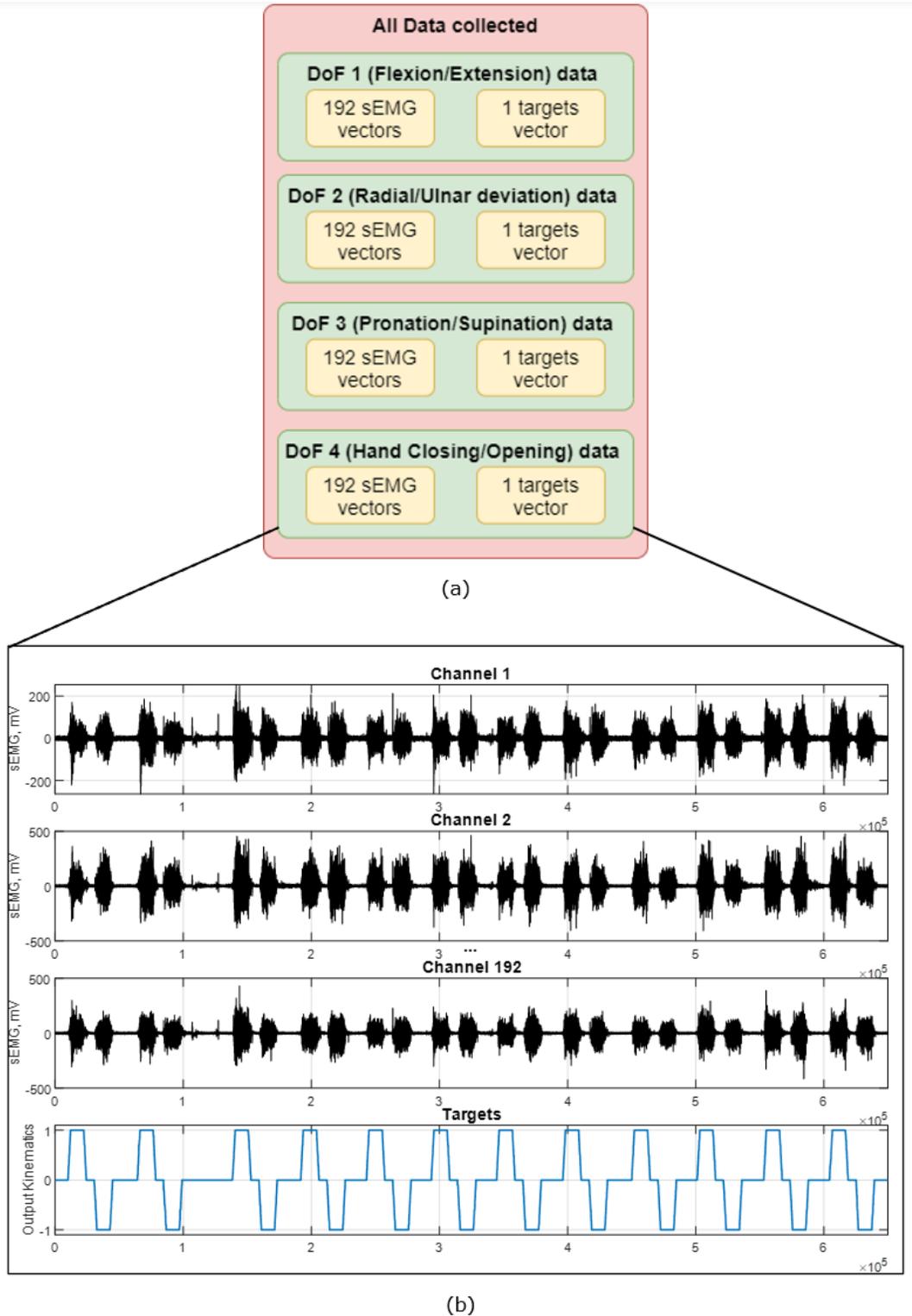


Figure 3.2 Collected data description. (a) General collected data array consists of four DoF-wise separated subsets. Each subset contains 192 sEMG signal vectors and 1 targets vector. (b) Plots of vectors contained in DoF 1 subset. Input vectors of muscle contractions activity are synchronized with target output vector

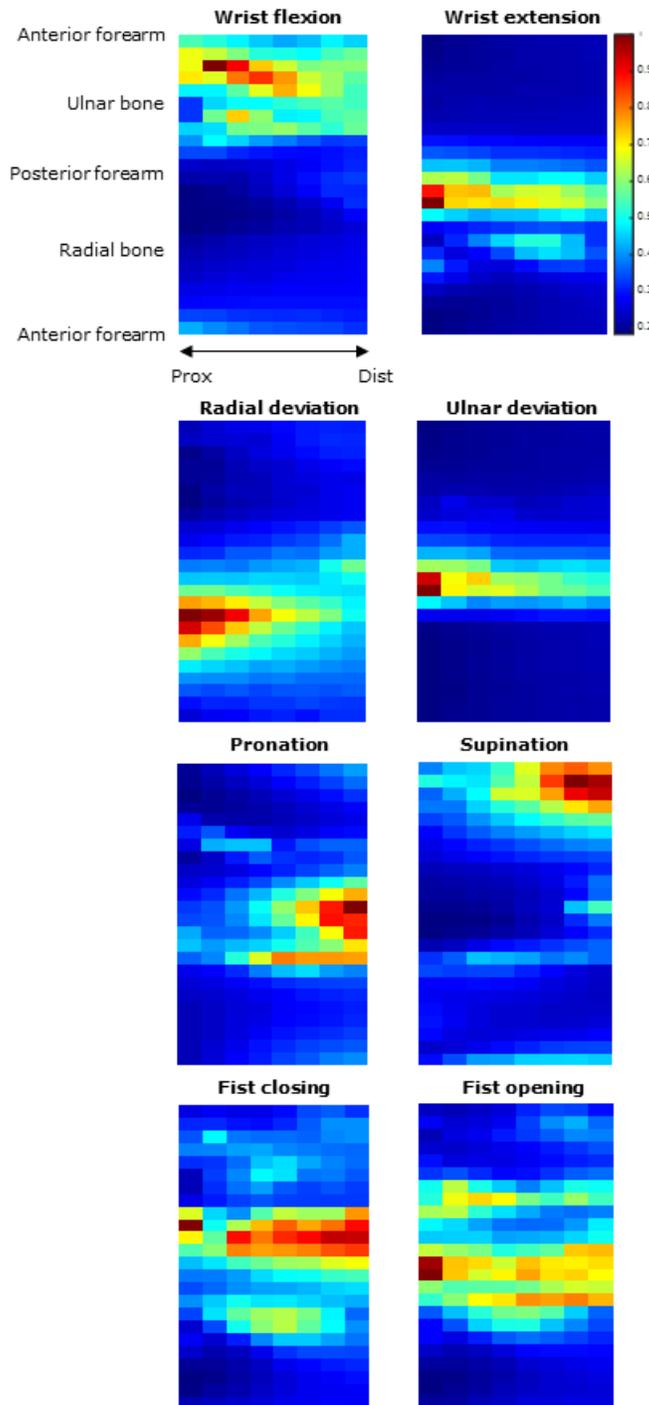
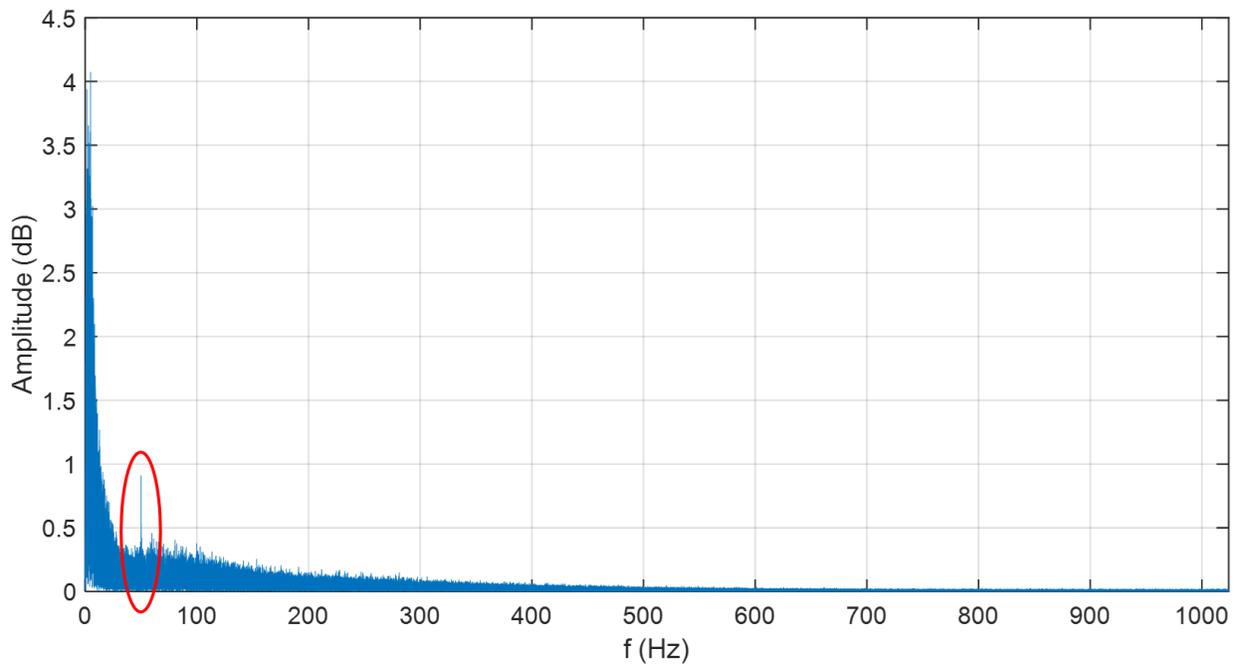


Figure 3.3 DoF-wise activation maps. Each channel activation value was normalized with respect to the most active channel in the array during each contraction, the maximum value in each array was 1. The spatial location of each channel can be specified according to forearm anatomical notations, e.g. "Dist" arrow points to fingers, "Prox" points to torso; tangential channel location can also be estimated by corresponding notations ("ulnar bone", "radial bone" etc.)

### 3.4. Signal filtering

As was discussed in Section 2.3, besides the signal originated in muscles, sEMG recording also contains noise components which should be eliminated. They foul the signal and may lead to its wrong interpretation [12].

**Signal power spectrum** was obtained to estimate the presence of noise components and to design appropriate digital filters. The spectrum was calculated with Fast Fourier Transform using sEMG recording from the first electrode data channel. The spectrum is shown at Figure 3.4.



*Figure 3.4 sEMG signal single-sided power spectrum. The signal has dominant noise components at frequencies below 20 Hz. A highlighted peak in amplitude when frequency is 50 Hz occurs due to line noise component*

Basing on the obtained spectrum and on the fact that the most significant information included into sEMG signal presents in a frequency range of 20 to 350 Hz [12], digital filters were designed. The filters parameters are presented at Table 3.2.

**Band-pass filter** used in this work has a range of frequencies of 25-350 Hz. It is the 6<sup>th</sup> order infinite impulse response (IIR) Butterworth filter. Its magnitude response is presented at Figure 3.5.

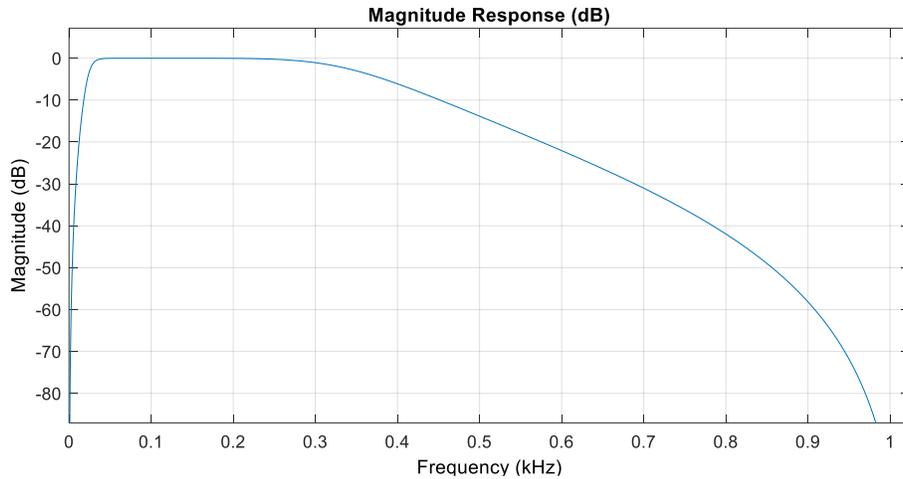


Figure 3.5 Passband filter magnitude response. The signal component amplitudes within the range of frequencies from 25 to 350 Hz are diminished

**Notch filter** was used in this study to eliminate the effect of signal distortion occurring at signal 50 Hz components. It is the 2<sup>nd</sup> order Butterworth IIR filter with cut-off frequencies of 45 and 55 Hz respectively. Notch filter magnitude response is shown at Figure 3.6.

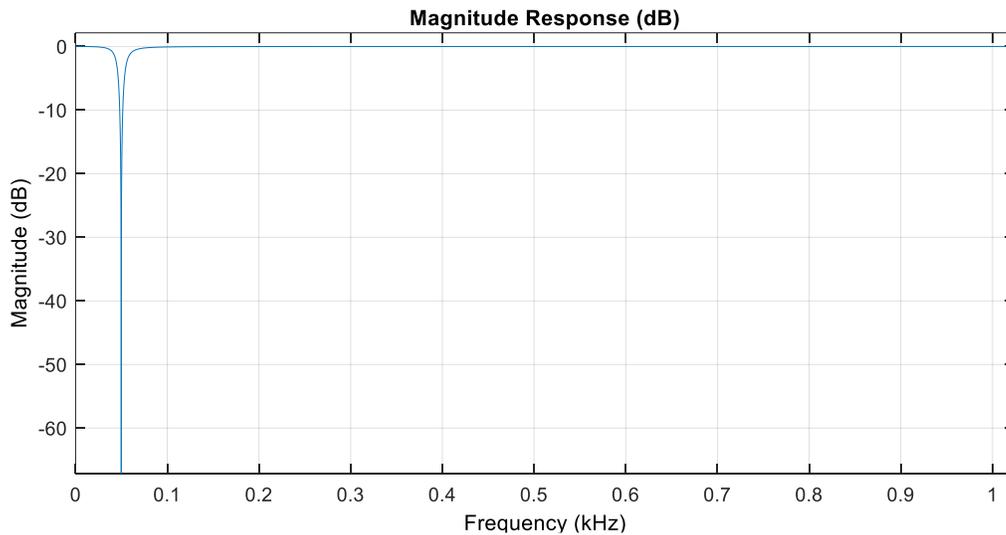


Figure 3.6 Notch filter magnitude response. The signal component amplitudes within the range of frequencies from 45 to 55 Hz are diminished

Table 3.2 Designed filters coefficients and gains

No	Filter type	Section no	Numerator coefficients	Denominator coefficients	Gain
1	Bandpass filter	Section 1	[1,00, 0,00, -1,00]	[1,00, -1,93, 0,93]	0,40
		Section 2	[1,00, 0,00, -1,00]	[1,00, -0,76, 0,44]	0,40
		Section 3	[1,00, 0,00, -1,00]	[1,00, -1,24, 0,25]	0,35
2	Notch filter	Section 1	[1,00, -1,98, 1,00]	[1,00, -1,95, 0,97]	0,95

During the sEMG recording, some signals were distorted due to loosen electrode-skin contact. An example of such a recording is presented at Figure 3.7. It demonstrates how data acquired with loosen channel differs to data collected with properly connected ones.

During the visual analysis of collected data, five channels were identified as spoiled. They were excluded from further signal processing.

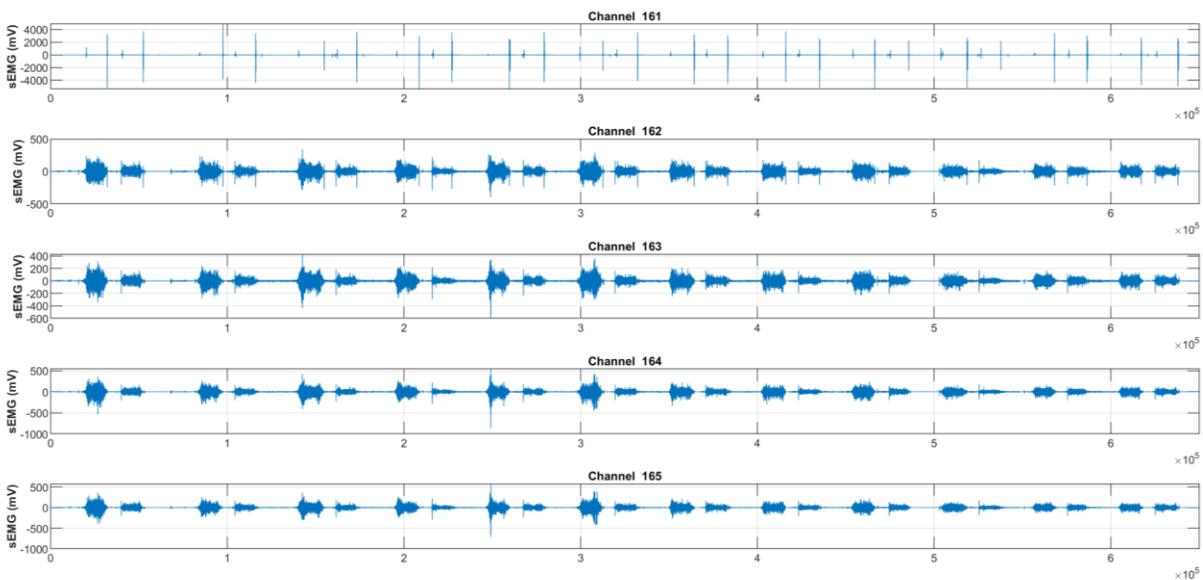


Figure 3.7 sEMG channels recordings. Electrode 161 was detached from patient skin during the experiment which led to inadequate data recordings. It could not be used for further processing

### 3.5. EMG Segmentation and Features Extraction

As Sections 2.4, 2.5 discuss, sEMG signal should be segmented and its features should be extracted from these segments in order to be used in myocontrol applications. This section describes signal segmentation parameters and features used in present work.

Overlapped segmentation was utilized in this study. It relies on two parameters: segmentation windows size  $s$  and windows overlap  $c$ . As a rule of thumb, the overlap  $c$  is admitted being two times less than window size:

$$c(s) = \frac{s}{2}, \quad (3.1)$$

where  $s$  – segmentation window size.

The following sEMG signal features were used throughout this work: Waveform length (2.7), Root mean square (2.8), Zero crossings (2.9), Slope Sign Changes (2.10). Table 3.3 describes them. These features were chosen since they encode different types of information about the signal individually[50].

Because they describe all the signal time-domain properties such as amplitude, power and frequency

Features extraction decreases the size of initial sEMG vectors proportionally to segmentation windows size and their overlap, so the resulting feature vectors length  $l_f$  is:

$$l_f = \frac{l_{EMG}}{d}, \quad (3.2)$$

where  $l_{EMG}$  – length of each sEMG data vector,

$$d = s - c.$$

Table 3.3 Description of sEMG features used in present work [34], [51]

No	Feature	Description
1	Waveform length (WL)	This feature provides a measure of signal amplitude, frequency, and duration.
2	Root mean square (RMS)	This feature represents the square root of sEMG signal average power over segmentation window.
3	Zero crossings (ZC)	This feature is the number of times a signal changes its sign within a segmentation window. The resultant value of this feature provides a measure of signal frequency.
4	Slope Sign Changes (SSC)	This feature is the number of times a signal changes its slope within a segmentation window. The resultant value of this feature also provides a measure of signal frequency.

Features extraction provides construction of MLP input data array. This array consists of  $I$  vectors:

$$I = C \cdot F, \quad (3.3)$$

where  $C$  – number of sEMG data channels,

$F$  – number of features that are extracted from the signal.

### 3.6. Acquired data sorting and preparation

**Targets shape** was changed to step-like form. Figure 3.9 illustrates this modification for a DoF 2 movement. This reshaping was motivated by fact that such kind of targets provides less computational complexity to a neural network algorithm [44]. Reshaped targets are only used for NN training. Original targets were used for testing purpose. This modification is illustrated at Figure 3.9.

**Resizing** of target arrays was done to make their length equal to input data vectors. According to (3.2), each feature vector has length of  $I_f$ . In order to equalize input and target sizes, only each  $d$ -th (3.2) target data sample should be considered. This modification is also illustrated at Figure 3.9. Original targets were not resized in order to be used for NN test.

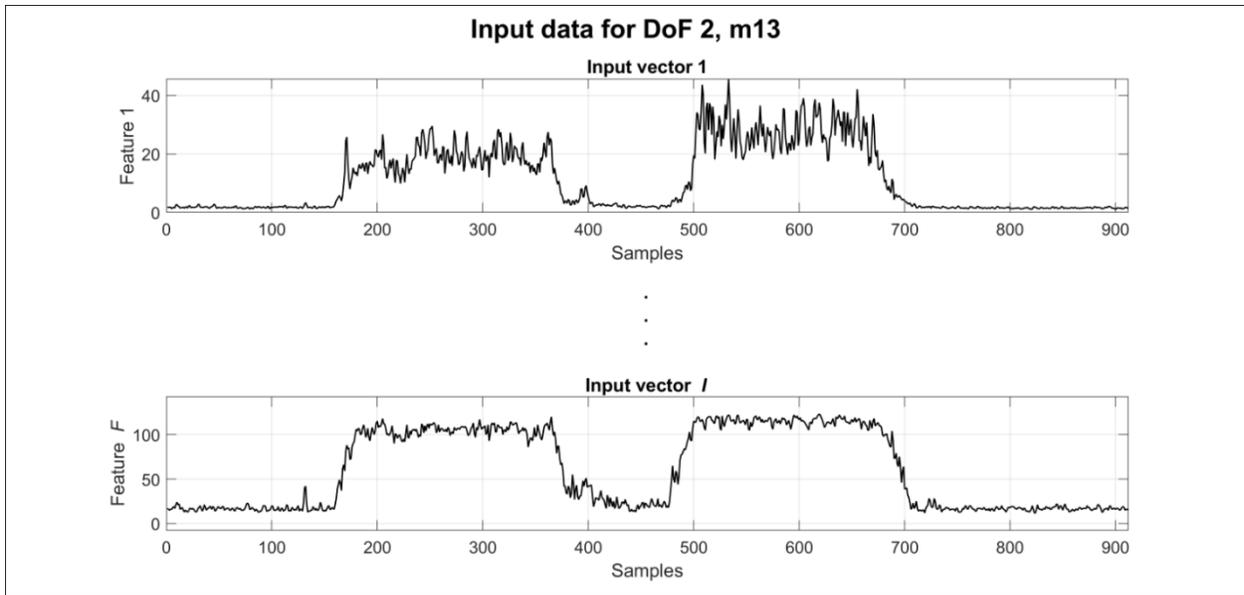
**Movement-wise segmentation** was done to provide flexibility in training and testing data sets construction. As described at Table 2.1, a subject performed 12 full hand movements for

DoF 1 control, 13 for DoF 2, 13 for DoF 3, and 12 for DoF 4. Each full movement consisted of two contractions implying individual DoF control in opposite directions.

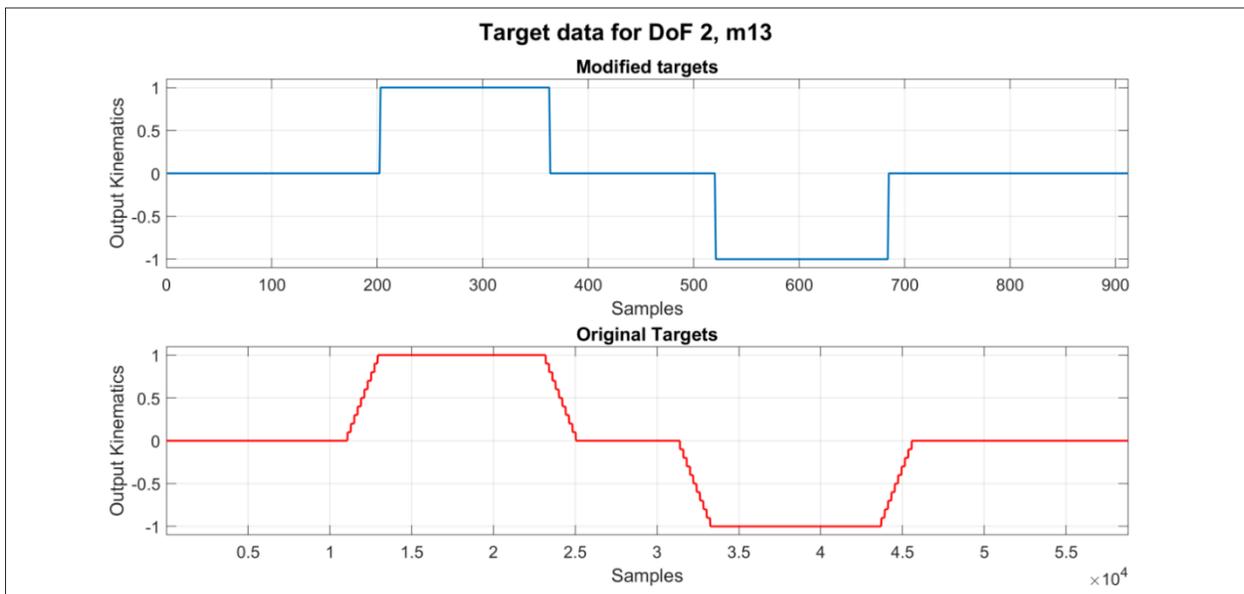
This segmentation was also applied to signal feature vectors providing motion-wise split of input and output data for each DoF. Separated targets datasets are illustrated at Figure 3.8. A motion-related inputs and modified targets arrays have length  $I_{Dxm}$ , where  $D$  relates to controlled degree-of-freedom ( $D=1...4$ ) and  $m$  relates to motion repetition.



Figure 3.8 Movement-wise divided data sets. For every DoF, each motion data were segmented. The data consists of input and target sub-sets. Input data is compiled of extracted sEMG features. Target data consists of robot DoFs kinematics. Notations in parentheses represent user control motions responsible for corresponding DoF-s control: Flex – flexion, Ext – extension; Rad – radial deviation, UI – ulnar deviation; Pro – pronation, Sup – supination; Clo – fist closing, Ope - fist opening



(a)



(b)

Figure 3.9 Segmented data instances. (a) Instance of 13<sup>th</sup> motion related input data for DoF 2 control. Input array consists of  $I$  feature vectors which is related to number of extracted features  $F$  and sEMG channels considered. (b) Instance of 13<sup>th</sup> motion related target data for DoF 2 control. Modified targets size differs to original ones. Their shape is also different

### 3.7. Neural network data management

To achieve proportional simultaneous control over multiple degrees-of-freedom, previously segmented data provided at Figure 3.8 should be arranged into training and test sets. This section discusses these sets construction, which is illustrated at Figure 3.11.

**Training data set** is used during an ANN learning. It is composed of sEMG feature vectors as inputs and different robotic DoFs modified kinematics as targets. Inputs data array has dimensions of  $I$  rows (3.3) and  $L_{tr}$  (3.4) columns. Targets array consists of 4 rows and  $L_{tr}$  columns. The number of each DoF motion data involved into this array is presented at Table 3.4.

$$L_{tr} = \sum_{D=1}^4 \sum_{i=1}^{M_{Dtr}} l_{Dmi} \quad (3.4)$$

where  $D$  – number of controlled DoF,

$M_{Dtr}$  – number of motions used for training,

$l_{Dmi}$  – individual motions length.

**Test data set** is composed of the same types of data. Its dimensions are  $I$  rows and  $L_{test}$  (3.5) columns for inputs and 4 rows and  $L_{test}$  columns for targets. The number of each DoF motions data involved into this array is also presented at Table 3.4.

$$L_{test} = \sum_{D=1}^4 \sum_{i=1}^{M_{Dtest}} l_{Dmi} \quad (3.5)$$

where  $D$  – number of controlled DoF,

$M_{Dtest}$  – number of motions used for test,

$l_{Dmi}$  – individual motions length.

Table 3.4 Number of DoF motions data used for training and testing

Degree-of-freedom number	Number of training data $M_{D tr}$	Number of test motions data $M_{D test}$
1	10	2
2	11	2
3	11	2
4	10	2

**Estimation results** should be resized and smoothed. Resizing is necessary as far as original targets were reduced correspondingly to extracted features as was described in Section 3.6. To return motion estimates to original targets size, each instance of NN output should be sequentially copied  $d$  times. It is illustrated at Figure 3.10. This procedure is necessary to recreate the original sensors sampling rate as far as the DoFs control is based on it.

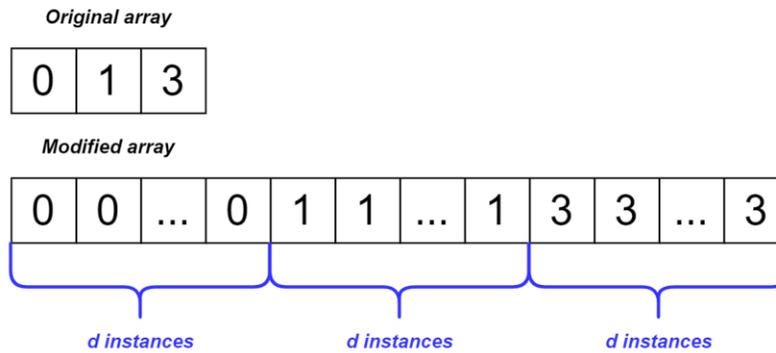


Figure 3.10 Data array resizing. Each instance of original array is reproduced  $d$  times in modified array. Instances original order is preserved

After size modification, the NN outputs were also smoothed using moving average technique. It is described with (3.6).

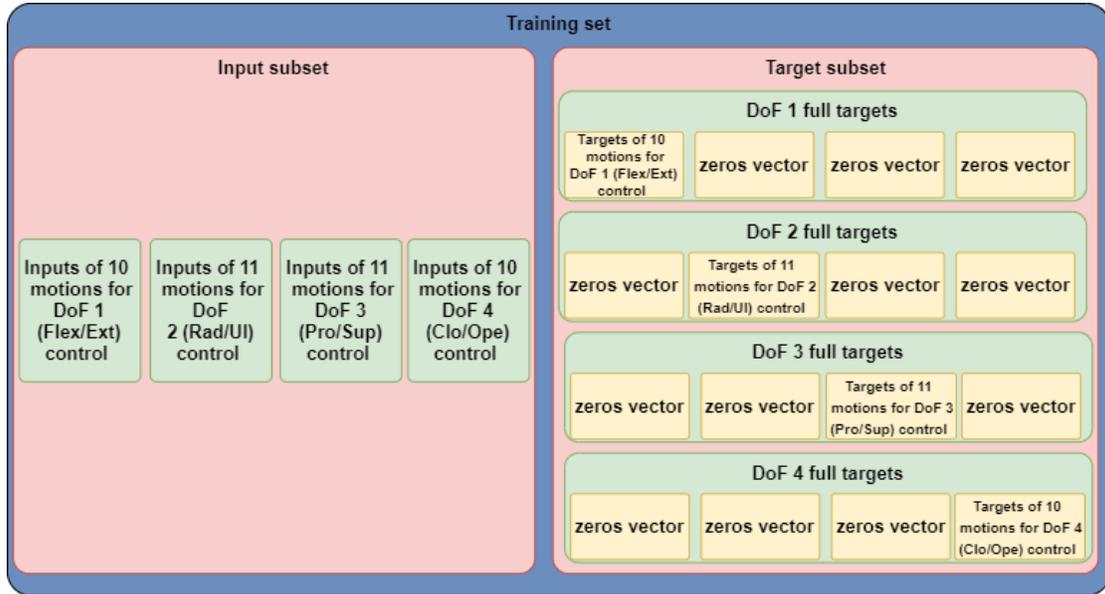
$$y_m[n] = \frac{1}{N} \sum_{i=0}^{N-1} y[n+i], \quad (3.6)$$

where  $y_m[n]$  – averaged output instance result at instance  $n$ ,

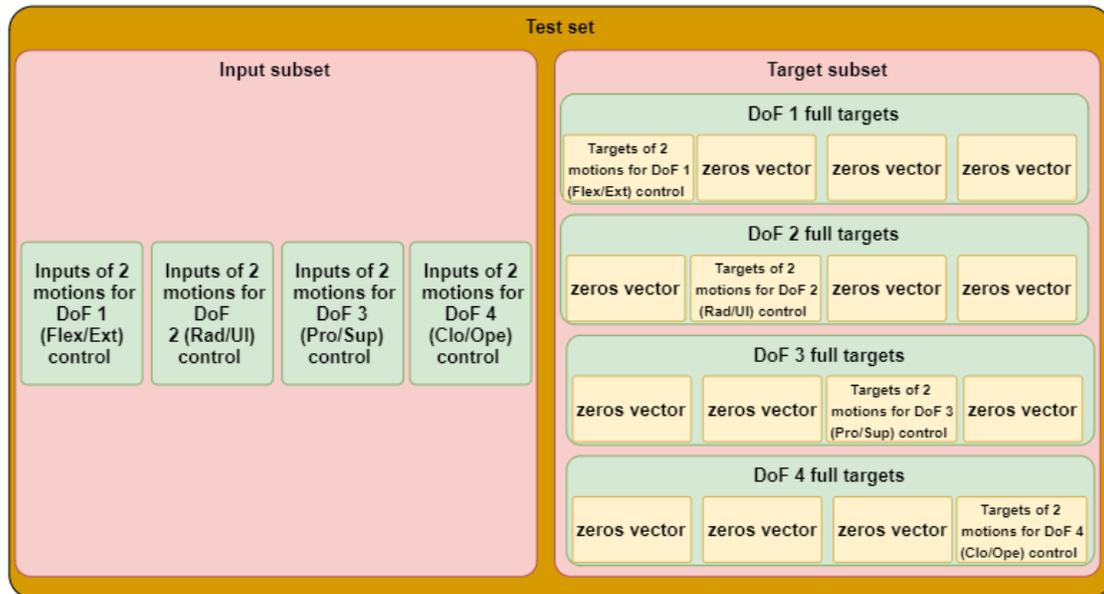
$N$  – averaging window length,

$y[n+i]$  – original output value at instance  $n+i$ .

Besides smoother output, this kind of filtering provides control delay. In order to achieve a trade-off between control signal smoothness and speed, window length was set to be 300 samples (around 150 milliseconds) which is suitable for myocontrol [10], [35].



(a)



(b)

Figure 3.11 Training and test NN data sets structure. Every consists of input and target subsets. Input subsets contain pre-defined number of motion-related input arrays. Corresponding modified DoF kinematics are contained into target subset. They should be sustained with zero target vectors to represent the absence of any motion during other DoF-s control.

(a) Training set containment. (b) Test set contains

### 3.8. MLP neural network

**Architecture** of the used network is presented at Figure 3.12. The input layer is consisted of  $I$  neurons, calculated with equation (3.3). Each input neuron is provided with an sEMG signal feature.

Hidden layer contains 10 neurons. This value is assigned by a heuristic that 2 neurons estimate control signal for each DoF in positive and negative directions respectively, summing up as 8 neurons. Other two neurons are meant to handle the noise in the input signal.

Output layer consists of four neurons. Each of them provides corresponding control signals ( $a_1...a_4$ ) to each degree of freedom (DoF 1...DoF 4) in opposite directions.

**Cost function** used for the network in this work is the mean squared normalized error (MSE) function (3.7). It measures the network's performance according to the mean of squared errors.

$$C(v) = \frac{1}{m} \sum_{i=1}^m (h_v(x^{(i)}) - y^{(i)})^2, \quad (3.7)$$

where  $m$  – the number of training examples,

$x^{(i)}$  – the input vector for the  $i$ -th training example,

$y^{(i)}$  – the targets vector for the  $i$ -th training example,

$v$  – network parameters (weights and biases)

$h_v(x^{(i)})$  – the algorithm prediction for the  $i$ -th training example using parameters  $v$ .

**Optimization algorithm** (or training function) used in this work is the One Step Secant (OSS) backpropagation algorithm. OSS algorithm is a trade-off between Broyden–Fletcher–Goldfarb–Shanno (BFGS) and conjugate gradient descent (CGD) algorithms. It provides more accuracy than CGD and consumes less computation power than BFGS. It also shows better performance with regression problems [52]. This algorithm computes a current iteration minimum search direction in the field of NN parameters according to (3.8) [53], [54].

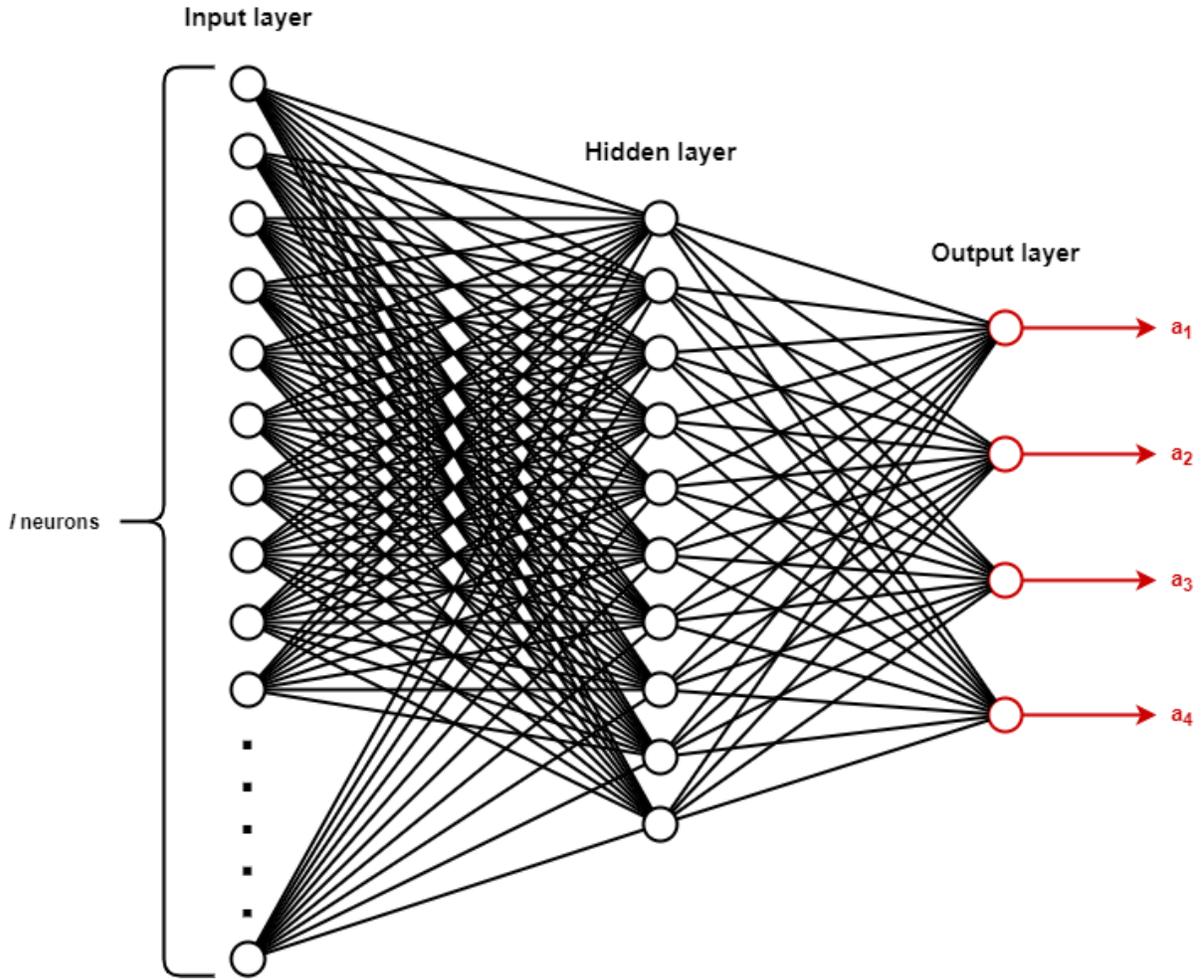


Figure 3.12 Architecture of the neural network for extracting control signals for 4 degrees-of-freedom robotic system. Input layer consists of  $I$  neurons, each of them represents sEMG signal features. Hidden layer consists of 10 neurons. The output is composed of four neurons which provide control signals in two opposite directions to individual DoFs

$$dX = -gX + A_c \cdot X_{step} + B_c \cdot dgX, \quad (3.8)$$

where  $gX$  – the value of gradient,

$X_{step}$  – the change in the weights on the previous iteration,

$dgX$  – the change in the gradient from the last iteration,

$A_c$  and  $B_c$  – combinations of scalar products appearing during computations.

Training is stopped when any of the following conditions occur:

- The maximum number of epochs (repetitions) is reached.
- The maximum amount of time is exceeded.
- Performance (cost function) is minimized to the goal.
- The performance gradient is less than its limit value.
- Validation performance has increased more than a limit of times since it lastly decreased – validation checks number.

**Activation functions** used in this work are discussed further. As was described in Section 2.7, network nodes process their inputs using activation function. In the designed MLP, hidden layer neurons transferred their inputs with tan-sigmoid activation function (3.9):

$$a_j^l = \frac{2}{1 + e^{-2 \cdot z_j^l}} - 1 \quad (3.9)$$

where  $z_j^l = w_{jk}^l \cdot a_k^{l-1} + b_j^l$ .

Output neurons used pure linear activation function (3.10):

$$a_j^l = z_j^l \quad (3.10)$$

These functions are graphical represented at Figure 3.13.

**Other hyper-parameters** used in NN design and which differ from standard ones defined in “feedforwardnet” function of MATLAB Deep Learning Toolbox are listed below:

- Number of epochs:  $N_e = 10000$ .
- Number of validation checks:  $N_c = 1000$ .
- Goal performance value:  $C_{goal} = 0$ .
- Minimum gradient value:  $g_{min} = 1^{-10}$ .

These and standard hyper-parameters are presented at Appendix 2.

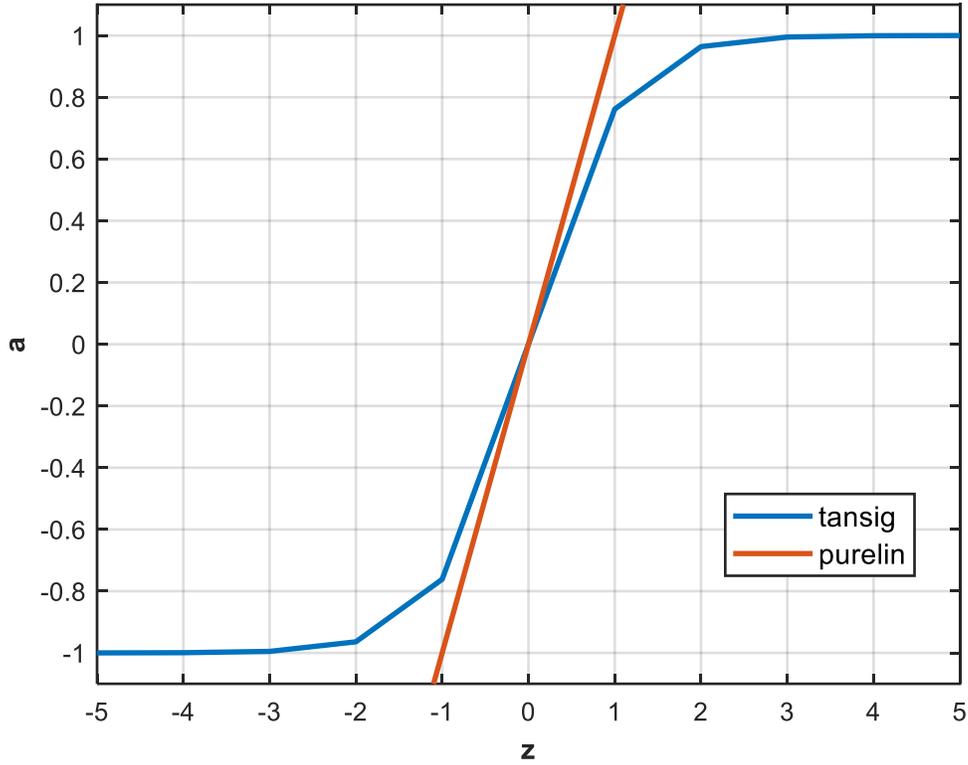


Figure 3.13 Activation functions plots.  $a$  – neuron activation output,  $z$  – neuron input. Tansig is used for hidden layer neurons, Purelin is used for output layer neurons. Sub- and superscripts used in the function equations are diminished

### 3.9. Performance evaluation

To evaluate the performance of MLP estimator numerically,  $R^2$  coefficient of determination was calculated for every DoF motion estimation.  $R^2$  is a common tool in sEMG control applications to measure the proportion of data variation between targets and outputs [10], [40], [55].

$$R^2 = 1 - \frac{\sum_{i=1}^N (x_i - \hat{x}_i)^2}{\sum_{i=1}^N (x_i - \bar{x})^2}, \quad (3.11)$$

where  $x_i$  – a DoF activation estimation provided by NN,

$\hat{x}_i$  – the value of corresponding target activation,

$\bar{x}$  – mean value activation estimations,

$N$  – the total number of output values.

As can be seen from (3.11), the closer the  $R^2$  value to unity, the better the estimator performance.

### 3.9.1 Cross-validation

In order to evaluate the designed machine learning algorithm, a cross-validation technique should be performed. It also allows to ensure that the NN model is able to generalize to an independent input data and estimate desired output properly [56]. For these purposes, a repeated random sub-sampling validation was implemented.

The randomization was completed motions-wise across each DoF. For every DoF, two random motions inputs and targets were put into test set. Remaining motions data were put into training set. This procedure was done five times providing five random NN sets such as those as presented at Figure 3.11. However, data sets constructed for cross-validation were different because DoF 4 control inputs were placed before DoF3. The same was done with corresponding targets.

These data sets provided different values of motion estimation performance. These values were evaluated with  $R^2$  coefficient (3.11). Mean values and standard deviation of this coefficient for each DoF control were calculated. The resultant motion estimation plots are presented at Appendix 3.

### 3.9.2 Variable feature sets

As was described in Section 3.1, sEMG features vector is calculated after the signal extraction. In order to estimate which composition of features vector provides enhanced MLP regression-based motion estimation, three sets of features, described at Table 3.3, were compiled. These sets containment is presented at Table 3.5.

*Table 3.5 Feature sets used to estimate how different features impact motion estimation performance*

Set No	Set features
1	RMS, SSC, WL, ZC
2	RMS
3	SSC

Signal processing described at Sections 3.4 – 3.10 involving these feature sets provided three motion estimation results for four simultaneously controlled DoFs. These results were evaluated using  $R^2$  coefficient (3.11). Their mean and standard deviation were also calculated. The resultant motion estimation plots are presented at Appendix 4.

### 3.9.3 Channel reduction

To evaluate the effect of sEMG electrodes quantity on performance of kinematics estimation, the electrodes were divided into five groups. These groups were composed of 187, 95, 48, 24 and 16 sEMG data channels. Some electrodes were excluded from consideration due to loose skin contact during data acquisition. Channel groups layout is presented at Figure 3.14.

Data acquired with each set were processed according to the procedure described in Sections 3.4 – 3.10. This processing provided five motion estimations of four DoFs controlled simultaneously. Estimation performance was evaluated using  $R^2$  coefficient (3.11). These values were averaged over controlled DoFs to provide the performance mean and standard deviation values. The resultant motion estimation plots are presented at Appendix 5.

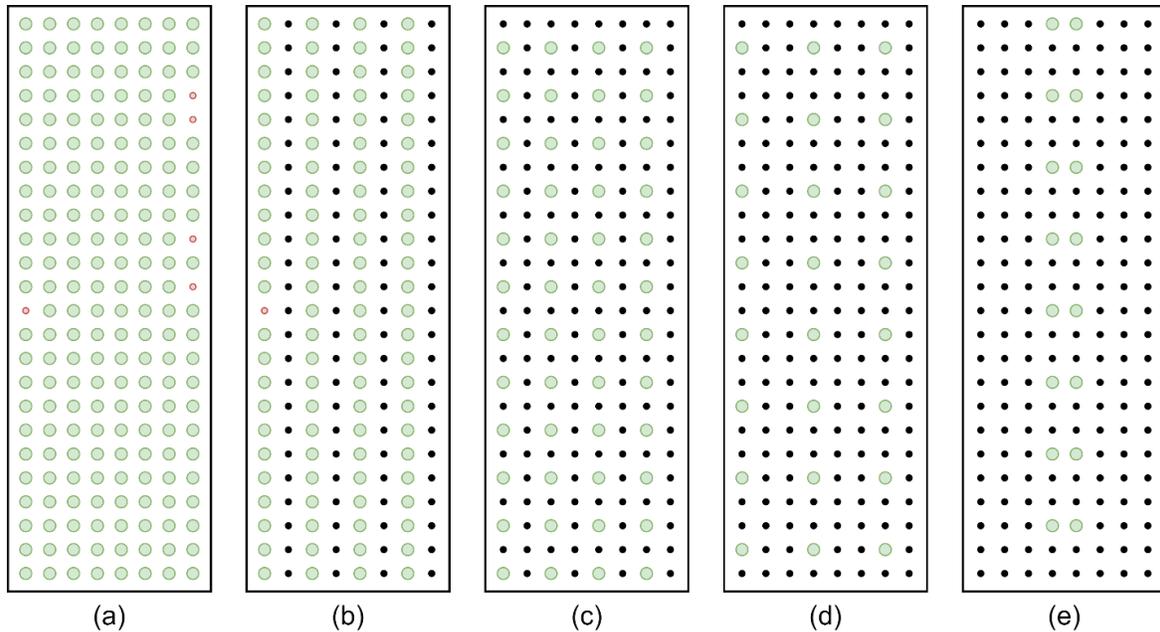


Figure 3.14 Electrode groups. Bigger green circles are the electrodes used for sEMG data extraction. Smaller circles data are not considered. Red circle electrodes are nonrelevant due to skin loose contact during data acquisition. Black circle electrodes are not considered to facilitate evaluation method. (a) group of 187 channels, (b) group of 95 channels, (c) group of 48 channels, (d) group of 24 channels, (e) group of 16 channels

### **3.9.4 Variable segmentation window size**

In order to evaluate how kinematics estimation performance correlates with overlapped segmentation parameters, seven values of window size were introduced to process sEMG signal as Sections 3.4 – 3.10 describe. These values were 16, 32, 64, 128, 256, 512 and 1024 samples. According to (3.3), windows overlaps were 8, 16, 32, 64, 128, 256 and 512 correspondingly. This signal processing resulted in seven MLP regression-based motions estimations for four simultaneously controlled DoFs.  $R^2$  coefficient of determination (3.11) evaluated these results performance. These values were averaged over controlled DoFs to provide the performance mean and standard deviation values. The resultant motion estimation plots are presented at Appendix 6.

## 4. RESULTS

An example of motion estimation results with corresponding targets and  $R^2$  coefficients are shown at Figure 4.1. It presents kinematics estimation for each of 4 controlled DoF.

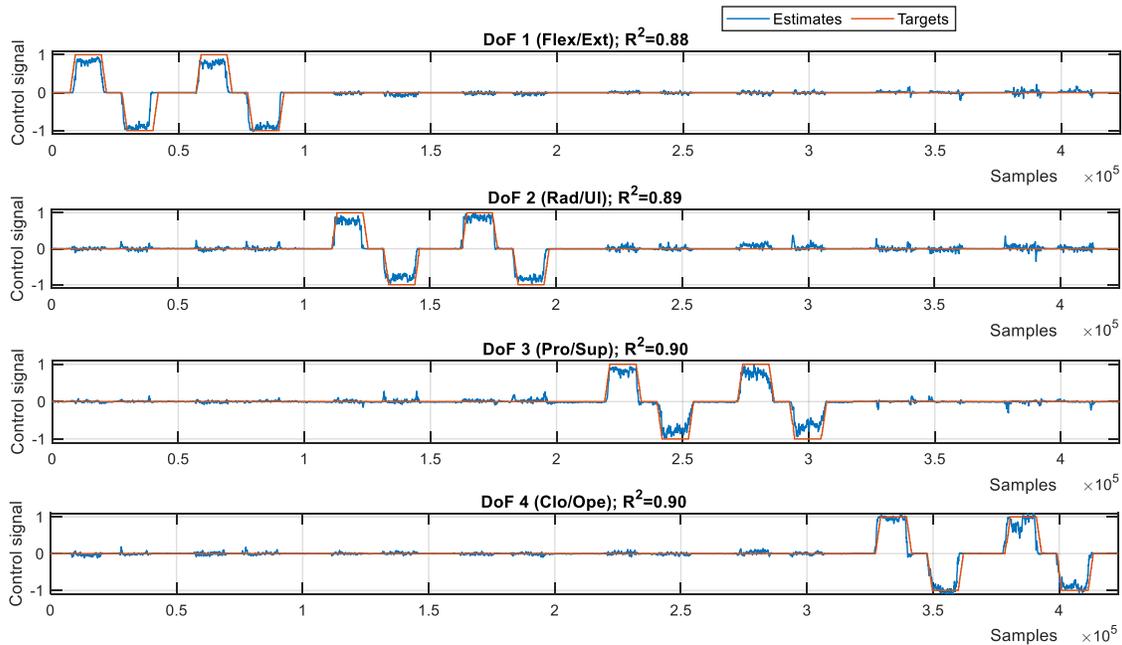


Figure 4.1 Kinematics estimation results for each DoF with its  $R^2$  performance coefficient. Red line demonstrates target kinematic control signals, blue line is the NN estimation results. Notations in parentheses represent user control motions responsible for corresponding DoF-s control: Flex – flexion, Ext – extension; Rad – radial deviation, UI – ulnar deviation; Pro – pronation, Sup – supination; Clo – fist closing, Ope – fist opening

**Cross-validation results** are presented at Figure 4.2. It demonstrates the mean performance coefficient values sustained with their standard deviations calculated for each DoF. As can be seen from the figure, the best control is achieved with wrist flexion/extension (DoF 1). The mean  $R^2$  value for it over 5 randomly composed test subsets is 0,92 with standard deviation of 0,02. Pronation/supination (DoF 3) provides close value of performance – 0,90, but its standard deviation is greater (0,04). DoF 2 mean control performance with radial and ulnar deviation value is 0.87 with standard deviation of 0,03, which is smaller than one for DoF 3. The lowest motion estimation of around 0,75  $R^2$  value is shown when DoF 4 is controlled with fist closing and opening. The standard deviation of this DoF control is also the highest among all the other (0,08). Plots of estimation results used in cross-validation are presented at figures A3.1 – A3.5

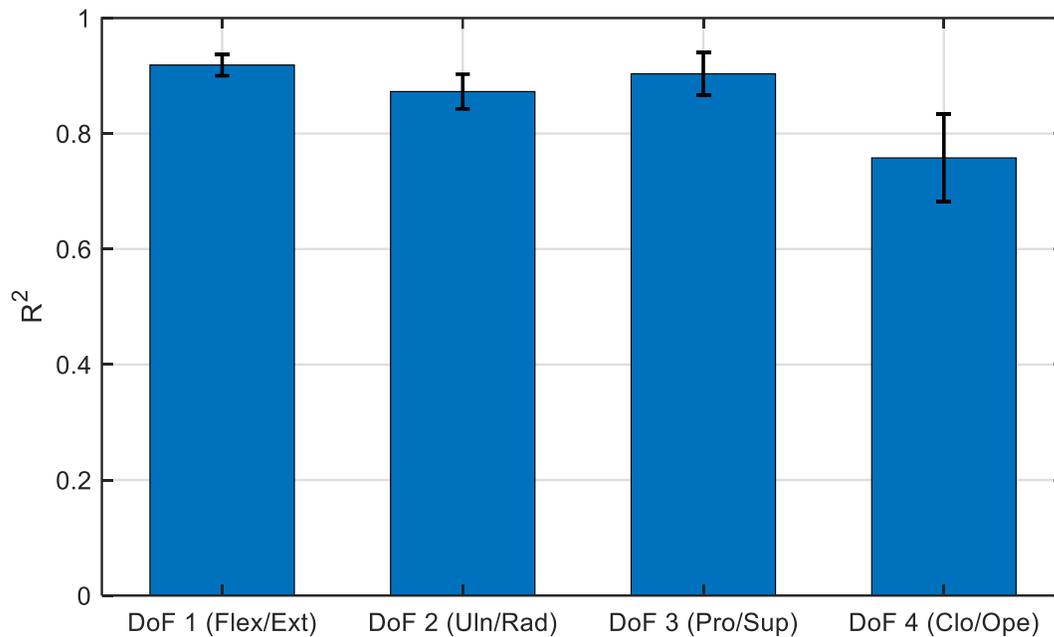


Figure 4.2 Performance plot for random validation setup. Mean values and standard deviations of each DoF control performance coefficient  $R^2$ . Text in the parentheses describes which user motions controlled either DoF

**Motion estimation with variable feature sets** is shown at Figure 4.3. It presents values of  $R^2$  coefficient between MLP test targets and its motion estimations for four controlled degrees of freedom when signal feature sets 1 (RMS, WL, ZC, SSC), 2 (RMS) and 3 (SSC) described in Section 3.9.2 were used. Figures A4.1, A4.2, A4.3 give specific plots of NN-based motion estimation together with NN targets used to calculate the coefficient.

It can be seen from Figure 4.3 that in case of controlling the DoF 1 with wrist flexion/extension, using features set 1 leads to performance coefficient value of 0,85, set 2 provides value of 0,87 and set 3 leads to  $R^2$  value of 0,72.

As for DoF 2, controlled with ulnar and radial deviations, it can be observed that using feature sets 1 and 3 provides performance values similar to ones obtained for DoF 1 which are 0,85 and 0,72 correspondingly. However, in case of using set 2, the performance of DoF 2 control decreases down to 0,82.

In case of controlling DoF 3 with pronation and supination, usage both of sEMG features, corresponding to previously defined set 1 and set 2, to create the NN estimator input data, leads to  $R^2$  coefficient of determination value of 0,85. Set 3 provides performance weight value of 0,69.

DoF 4 control using features set 1 shows the performance coefficient  $R^2$  value of 0,79 and set 2 provides this coefficient value of 0,78. Using set 3 drops the performance down to the value of 0,61.

The figure also demonstrates that the mean value of the coefficient of determination over all DOFs between MLP targets and MLP outputs in case of using signal feature sets 1 and 2 equals 0.84. However, the standard deviation of the results when using set 1 (0,03) is smaller comparing to using set 2 (0,04), as clearly illustrated in Figure 4.3. From it, it can also be observed that the mean performance coefficient value of kinematics estimation over all degrees of freedom when features set 3 is used, equals 0,69 with standard deviation of 0,05.

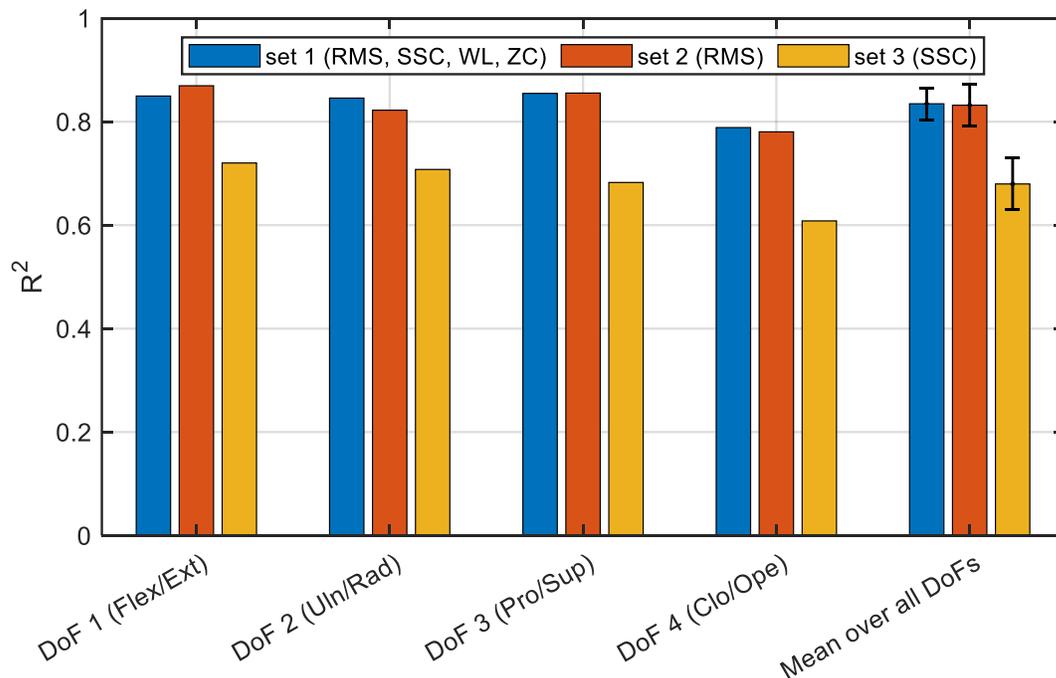


Figure 4.3 Performance plot for variable features setup. Performance coefficient  $R^2$  of motion estimation is evaluated for 4 controlled degrees-of-freedom when 3 different sets of input signal features were used to calculate the estimator input. Mean performance values over those DoFs with standard deviations are also presented

**Channel reduction** effect on motion estimation is shown at Figure 4.4. It presents the dependency of motion estimation performance on the number of sEMG signal channels. It shows that for every DoF, the average performance of kinematics estimation has the lowest value of 0,77 with standard deviation of 0,02 when only 16 channels are used. The performance mean value increased to 0,82 when 24 electrodes data are considered, but corresponding standard deviation increases to 0,05 likewise. This augmentation of mean continues with increasing the channels number until 95 channels are involved into measurement. With this number of electrodes, the performance trend reaches its plateau of 0,83 while standard deviation value decreases (0,04 for 48 and 95 channels). It can also be observed there are some disagreements of individual DoF-s performance plots with the common trend, especially in cases of DoF 1 and DoF 4. DoF 1 decreases its performance from 0,86 to 0,85 and DoF 4 increases it from 0,77 to 0,79 when 187 channels are used providing standard deviation of 0,03. Specific plots of motion estimation results obtained using different channels number are shown at Figures A5.1 - A5.5.

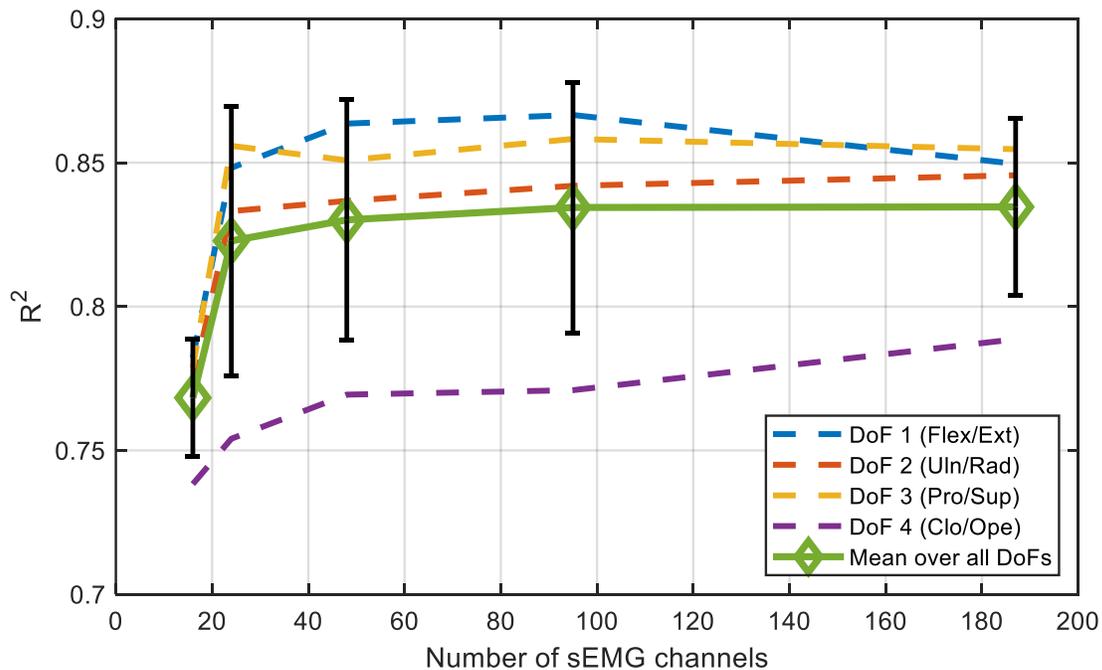


Figure 4.4 Performance plot for channels reduction setup. Each DoF motion estimation performance coefficient  $R^2$  depended on number of input data channels and mean of these values are presented. Mean performance standard deviations for 16, 24, 48, 95 and 187 channels are also shown

**Segmentation window size** impact on motion estimation is presented Figure 4.5. It presents the dependency of kinematics performance on segmentation window size. The figure demonstrates that, generally, the performance value increases significantly from  $0,22 \pm 0,07$  to  $0,90 \pm 0,01$  when segmentation window size increase from 16 to 64 samples. For window size values in range from 64 to 256 the performance value doesn't change notably, besides a small drop to  $0,85$  with standard deviation of  $0,03$  for 128 samples window length. When more than 256 samples are used, the performance decreases. It is also important to mention that DoF 1 disagrees with this trend providing increasing to deviation of motion estimation model. Specific plots of motion estimation results obtained using different channels number are shown at Figures A6.1 – A6.7.

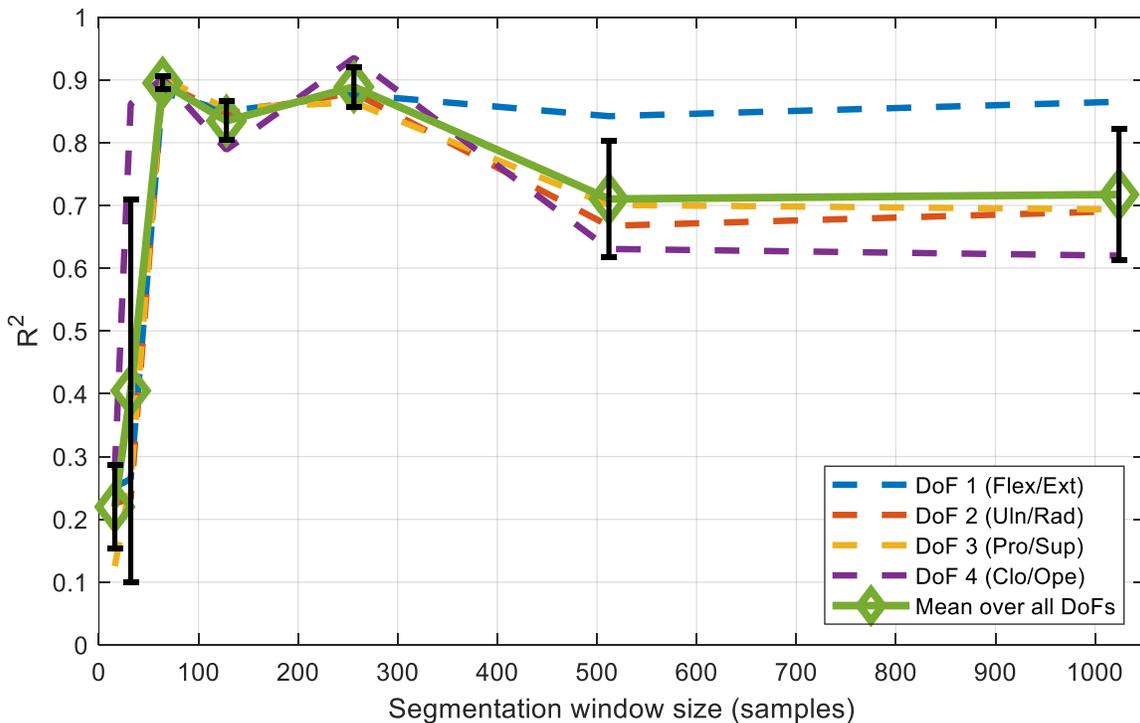


Figure 4.5 Performance plot for variable segmentation window size setup. Each DoF control performance parameter  $R^2$  changing with the segmentation window size is shown. The performance mean over all DoFs and standard deviations for window sizes of 16, 32, 64, 128, 256, 512 and 1024 are also presented

## 5. DISCUSSION

**Cross-validation.** As was stated in Chapter 4, DoF 1, DoF 2 and DoF 3, motion estimation mean performance values resulted 0,92, 0,87, 0,90 correspondingly during cross-validation.

Figure 3.3 shows that this is likely due to different channel coactivations during contractions. Groups of channels activated when the user executes movements responsible for these DoFs (flexion/extension, ulnar/radial deviation, pronation/supination), differ significantly. This input data variation provides a good performance of MLP-based estimation, which is common for neural networks [44].

On the contrary, DoF 4 control performance mean value is low (0.75) in comparison to other degrees-of-freedom. Figure 3.3 also shows that this is likely due to channel coactivations. As can be seen from the figure, fist closing and opening activates a lot of forearm muscles under the sEMG sensors. Plenty of them are activated during other contractions. These coactivations may lead to low mean and high standard deviation values of DoF 4 kinematics estimation performance.

**Variable feature sets.** As was stated in Chapter 4, sets 1 (RMS, WL, ZC, SSC) and 2 (RMS) provide similar performance values of estimation. This can be explained by the fact that the RMS feature includes information of the signal's amplitude and frequency in it. However, even though using those sets provides close mean performance results, set 1 is more stable comparing to set 2 due to its smaller standard deviation (0,03 vs 0,04). This could be due to increased rigidity of input-output relations in the designed MLP when more input signal features are used whereas more features provide more input layer neurons as described by (3.3).

On contrast, using set 3 results in a lower performance. This may be due to the fact that SSC lacks the information needed for high-accuracy motion estimation. According to a study [2], SSC contains only signal frequency-related information but not amplitude-related.

The best value of motion estimation performance was achieved by using sEMG features set 1 consisting of RMS, SSC, ZC and WL. The averaged value of it over all DoFs is 0,85 with standard deviation of 0,03.

**Channel reduction.** As was also stated at Chapter 4, on average, the estimation performance value is smallest when 16 channels are used. This is due to the lack of input data for MLP estimator. The performance increases significantly when 24 channels are involved into estimation. This may be the result of choosing more spatially separated electrodes that provide more variable and hence valuable data to MLP. Moreover, 24 channels provided 32 more input neurons to the network which also could increase its performance [44]. Further increasing the number of input channels to 48 slightly increased the average performance which was the same as in case of using maximum number of channels available. This means that closely-spaced sensors may provide the MLP with identic data which doesn't provide estimation accuracy.

However, increasing of data channels number decreases the deviation of movement estimation among all DoF-s. It is due to the fact that DoF 1 and DoF 4 do not follow the average trend, that clearly demonstrated at Figure 4.4. This may be due to more channel co-activations during forearm muscles contractions. These co-activations could lead to DoF 1 performance drop. On the other hand, they could provide more distinguished NN input for better DoF 4 control performance.

The best average estimation performance was achieved with 187 channels. It's value is 0,84 with standard deviation of 0,03.

**Variable segmentation window size.** As was documented in Chapter 4, the average performance value increases significantly when segmentation window size increases from 16 to 64 samples. This demonstrates that small sized segmentation window does not provide enough data samples to extract important sEMG features information.

There is no change in average performance value until window size is greater than 256. Starting from 512 samples size of segmentation window, the performance value decreases. This is due to the fact that windows overlapping parameter approaches and excesses the value of 300 samples which is the moving average filter window. This reduces the output smoothness, hence reduces the output performance.

Figure 4.5 also presents a small drop in performance when segmentation window of 128 samples is used for every DoF. This might be due to ineffectual initial combination of neural network weights and biases provided by backpropagation algorithm [44].

DoF 1 control performance disagrees with the average trend. When segmentation window size is greater than 256 it still increases. Its activation channels singularity may be the reason of this behavior.

The best estimation performance was achieved with segmentation window of 64 samples. It provides  $R^2$  value of 0,90 on average with standard deviation of 0,01.

## 6. CONCLUSION

This thesis has successfully designed an sEMG non-linear regression method for human motion intention estimation for multiple degrees-of-freedom robotic control applications. Moreover, it evaluated, in an offline way, the influence of signal processing parameters on the estimation performance.

The method was realized with MLP artificial neural network model. The model accepted processed sEMG signal and produced signals for four DoFs simultaneous proportional control.

The model performance was evaluated with cross-validation procedure. The performance was measured with  $R^2$  coefficient of determination. The performance is in  $R^2$  range from 0,75 to 0,92 for different DoFs control which corresponds to state-of-the art sEMG-based control techniques reported in the literature [4], [57]. Additionally, three sEMG signal processing parameters influence on motion estimation were evaluated: extracted feature types, channels number, segmentation windows size. The highest performance was provided by using a set of RMS, WL, ZC, SSC features; 187 sEMG channels, however 24 channels are enough to obtain prominent performance; and 64 samples segmentation window.

The designed motion estimation model has potential to be used in HMI system for intuitive robot control. However, the sensors used in it do not provide it with convenient using due to demanding skin preparation [58], [59]. Furthermore, the experimental procedure was completed with only one subject. Thus, it is hard to provide definitive statement on method's generalization ability.

Future work should be concentrated on experiments with actual robotic systems to proof the designed concept with online validation. Additionally, extending of controlled degrees-of-freedom number should be studied. Furthermore, research should be dedicated to involving different types of sensors to increase the system accessibility [25], [60], [61]. Besides that, work should be devoted to extending motion dataset variability. The data should be collected from several subjects. Together with that, the data acquisition should be elaborated with different testing conditions which may influence myocontrol performance [62]. These include changing arm positions [63], electrode shifts [64], verbal encouragement [65], multiple experimental sessions [35].

## KOKKUVÕTE

See lõputöö on edukalt kavandanud sEMG mittelineaarse regressioonimeetodi inimese liikumise tahte hindamiseks mitme vabadusastme robotjuhtimisrakenduse jaoks. Veelgi enam, see hindas võrguühenduseta viisil signaalitöötlemise parameetrite mõju hinnangute jõudlusele.

Meetod realiseeriti MLP tehisnärvivõrgu mudeli abil. Mudel aktsepteeris töödeldud sEMG-signaali ja andis signaale nelja DoF samaaegse proportsionaalse juhtimise jaoks.

Mudeli jõudlust hinnati ristvalideerimise protseduuriga. Toimivust mõõdeti  $R^2$  määramiskoeffitsiendiga. Erinevate DoF juhtimiste korral on jõudlus  $R^2$  vahemikus 0,75–0,92, mis vastab kõige tiptasemel SEMG-põhiste juhtimismeetoditele, mida kirjeldatakse kirjanduses. Lisaks hinnati kolme SEMG-signaali töötlemise parameetrit, mis mõjutavad liikumise prognoosimist: ekstraheeritud funktsiooni tüübid, kanalite arv, segmenteerimisakna suurus. Suurima jõudluse andis RMS, WL, ZC, SSC funktsioonide komplekt; 187 sEMG-kanalit, kuid silmapaistva jõudluse saavutamiseks piisab 24 kanalist; ja 64 proovi segmenteerimise aken.

Projekteeritud liikumise prognoosimise mudelit saab potentsiaalselt kasutada HMI süsteemis roboti intuiitivseks juhtimiseks. Siiski ei võimalda selles kasutatavad andurid seda nõudliku naha ettevalmistamise tõttu mugavaks kasutamiseks. Lisaks viidi eksperimentaalne protseduur läbi ainult ühe katsealusega. Seega on meetodi üldistamisvõime kohta raske täpset väidet anda.

Edasine töö peaks keskenduma eksperimentidele tegelike robotisüsteemidega, et tõestada kavandatud kontseptsiooni veebipõhise valideerimisega. Lisaks tuleks uurida kontrollitud vabadusastmete arvu laiendamist. Lisaks sellele tuleks uurimistöole pühendada eri tüüpi andurite kaasamine, et parandada süsteemi juurdepääsetavust. Lisaks tuleks teha tööd liikumisandmete varieeruvuse laiendamiseks. Andmeid tuleks koguda mitmelt isikult. Koos sellega tuleks andmete kogumist töötada välja erinevate katsetingimustega, mis võivad mõjutada müokontrolli jõudlust. Nende hulka kuuluvad käeasendite muutmine, elektroodide nihutamine, verbaalne julgustamine, mitmed katsesessioonid.

## LIST OF REFERENCES

- [1] J. Cannan and H. Hu, "Human-Machine Interaction ( HMI ): A Survey," *Tech. Rep. CES-508*, 2011.
- [2] "Human-Machine Interface (HMI) - Unieke." <https://www.unieke.com/us/hmi/> (accessed Apr. 18, 2020).
- [3] "The History of the Computer Mouse - Article - Computing History." <https://www.computinghistory.org.uk/det/613/the-history-of-the-computer-mouse/> (accessed Apr. 18, 2020).
- [4] E. J. Rechy-Ramirez and H. Hu, "Bio-signal based control in assistive robots: a survey," *Digit. Commun. Networks*, vol. 1, no. 2, pp. 85–101, 2015, doi: 10.1016/j.dcan.2015.02.004.
- [5] L. Peternel, T. Noda, T. Petrič, A. Ude, J. Morimoto, and J. Babič, "Adaptive control of exoskeleton robots for periodic assistive behaviours based on EMG feedback minimisation," *PLoS One*, vol. 11, no. 2, pp. 1–26, 2016, doi: 10.1371/journal.pone.0148942.
- [6] T. Kawase, T. Sakurada, Y. Koike, and K. Kansaku, "A hybrid BMI-based exoskeleton for paresis: EMG control for assisting arm movements," *J. Neural Eng.*, vol. 14, no. 1, 2017, doi: 10.1088/1741-2552/aa525f.
- [7] N. Kosmyna, C. Morris, T. Nguyen, S. Zepf, J. Hernandez, and P. Maes, "Attentivu: Designing EEG and EOG compatible glasses for physiological sensing and feedback in the car," *Proc. - 11th Int. ACM Conf. Automot. User Interfaces Interact. Veh. Appl. AutomotiveUI 2019*, pp. 355–368, 2019, doi: 10.1145/3342197.3344516.
- [8] J. Ning, S. Dosen, K.-R. Müller, and D. Farina, "Myoelectric Control of Artificial Limbs— Is There a Need to Change Focus?," *IEEE Signal Process. Mag.*, no. 3, pp. 148–152, 2012, doi: 10.1016/j.anplas.2017.08.003.
- [9] R. H. Chowdhury, M. B. I. Reaz, M. A. Bin Mohd Ali, A. A. A. Bakar, K. Chellappan, and T. G. Chang, "Surface electromyography signal processing and classification techniques," *Sensors (Switzerland)*, vol. 13, no. 9, pp. 12431–12466, 2013, doi: 10.3390/s130912431.
- [10] I. Vujaklija, V. Shalchyan, E. N. Kamavuako, N. Jiang, H. R. Marateb, and D. Farina, "Online mapping of EMG signals into kinematics by autoencoding," *J. Neuroeng. Rehabil.*, vol. 15, no. 1, pp. 1–9, 2018, doi: 10.1186/s12984-018-0363-1.
- [11] L. H. Smith, T. A. Kuiken, and L. J. Hargrove, "Evaluation of linear regression simultaneous myoelectric control using intramuscular EMG," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 4, pp. 737–746, 2016, doi: 10.1109/TBME.2015.2469741.
- [12] C. J. De Luca, L. Donald Gilmore, M. Kuznetsov, and S. H. Roy, "Filtering the surface EMG signal: Movement artifact and baseline noise contamination," *J. Biomech.*, vol. 43, no. 8, pp. 1573–1579, 2010, doi: 10.1016/j.jbiomech.2010.01.027.
- [13] A. L. De Sousa Freitas, V. P. De Camargo, H. M. P. Teixeira, R. Balancieri, and T. E. Colanzi, "Gesture and voice-based natural user interface for electronic whiteboard system in a medical emergency department," *ACM Int. Conf. Proceeding Ser.*, 2017, doi: 10.1145/3160504.3160534.
- [14] M. N. Economo *et al.*, "Distinct descending motor cortex pathways and their roles in

- movement," *Nature*, vol. 563, no. 7729, pp. 79–84, 2018, doi: 10.1038/s41586-018-0642-9.
- [15] R. Merletti and P. Parker, *Electromyography*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2004.
- [16] A. Del Vecchio, F. Negro, F. Felici, and D. Farina, "Associations between motor unit action potential parameters and surface EMG features," *J. Appl. Physiol.*, vol. 123, no. 4, pp. 835–843, 2017, doi: 10.1152/jappphysiol.00482.2017.
- [17] D. Winter, "EMG interpretation," *Electromyogr. Ergon.*, pp. 109–125, 2017, doi: 10.1201/9780203758670.
- [18] "35.2B: Nerve Impulse Transmission within a Neuron: Action Potential - Biology LibreTexts."  
[https://bio.libretexts.org/Bookshelves/Introductory\\_and\\_General\\_Biology/Book%3A\\_General\\_Biology\\_\(Boundless\)/35%3A\\_The\\_Nervous\\_System/35.2%3A\\_How\\_Neurons\\_Communicate/35.2B%3A\\_Nerve\\_Impulse\\_Transmission\\_within\\_a\\_Neuron%3A\\_Action\\_Potential](https://bio.libretexts.org/Bookshelves/Introductory_and_General_Biology/Book%3A_General_Biology_(Boundless)/35%3A_The_Nervous_System/35.2%3A_How_Neurons_Communicate/35.2B%3A_Nerve_Impulse_Transmission_within_a_Neuron%3A_Action_Potential) (accessed Apr. 20, 2020).
- [19] "How Neurons Communicate | Boundless Biology."  
<https://courses.lumenlearning.com/boundless-biology/chapter/how-neurons-communicate/> (accessed Apr. 20, 2020).
- [20] E. Hill, T. K. Karikari, K. G. Moffat, M. J. E. Richardson, and M. J. Wall, "Introduction of Tau oligomers into cortical neurons alters action potential dynamics and disrupts synaptic transmission and plasticity," *eNeuro*, vol. 6, no. 5, pp. 1–20, 2019, doi: 10.1523/ENEURO.0166-19.2019.
- [21] M. Al Harrach, "Modeling of the sEMG/Force relationship by data analysis of high resolution sensor network," 2016.
- [22] M. Zahak, "Signal Acquisition Using Surface EMG and Circuit Design Considerations for Robotic Prosthesis," *Comput. Intell. Electromyogr. Anal. - A Perspect. Curr. Appl. Futur. Challenges*, 2012, doi: 10.5772/52556.
- [23] H. J. Hermens, B. Freriks, C. Disselhorst-Klug, and G. Rau, "Development of recommendations for SEMG sensors and sensorplacement procedures," *J. Electromyogr. Kinesiol.*, vol. 10, pp. 361–374, 2000, doi: 10.1200/JCO.2017.76.5966.
- [24] G. Borelli, J. Jovic Bonnet, Y. Rosales Hernandez, K. Matsuda, and J. Damerau, "Spectral-Distance-Based Detection of EMG Activity from Capacitive Measurements," *IEEE Sens. J.*, vol. 18, no. 20, pp. 8502–8509, 2018, doi: 10.1109/JSEN.2018.2865580.
- [25] C. Castellini, R. Kõiva, C. Pasluosta, C. Viegas, and B. Eskofier, "Tactile Myography: An Off-Line Assessment of Able-Bodied Subjects and One Upper-Limb Amputee," *Technologies*, vol. 6, no. 2, p. 38, 2018, doi: 10.3390/technologies6020038.
- [26] B. S. Day, "Important Factors in Surface EMG Measurement," *Measurement*, pp. 1–17, 2002, [Online]. Available: [http://edge.rit.edu/content/P08027/public/IRB/Papers/intro\\_EMG.pdf](http://edge.rit.edu/content/P08027/public/IRB/Papers/intro_EMG.pdf).
- [27] G. De Luca, "Fundamental Concepts in EMG Signal Acquisition," *Distribution*, no. March, pp. 1–31, 2003.
- [28] H. J. Hermens *et al.*, "European Recommendations for Surface ElectroMyoGraphy," *Roessingh Res. Dev.*, pp. 8–11, 1999, doi: 10.1016/S1050-6411(00)00027-4.

- [29] P. G. O. E, M. DR, and B. N, "Oversampling data acquisition to improve resolution of digitized signals.," *Biomed. Sci. Instrum.*, vol. 34, pp. 137–142, Jan. 1997.
- [30] J. C. Ives and J. K. Wigglesworth, "Sampling rate effects on surface EMG timing and amplitude measures," *Clin. Biomech.*, vol. 18, no. 6, pp. 543–552, 2003, doi: 10.1016/S0268-0033(03)00089-5.
- [31] V. Medvedev and M. Cifrek, "Kinesiological Electromyography," *Intech*, p. 13, 2012, doi: 10.1016/j.colsurfa.2011.12.014.
- [32] "Frequency-domain analysis in Matlab | Motor behaviour." <https://motorbehaviour.wordpress.com/2017/09/18/frequency-domain-analysis-in-matlab/> (accessed Apr. 21, 2020).
- [33] Triwiyanto, O. Wahyunggoro, H. A. Nugroho, and Herianto, "Effect of window length on performance of the elbow-joint angle prediction based on electromyography," *J. Phys. Conf. Ser.*, vol. 853, no. 1, 2017, doi: 10.1088/1742-6596/853/1/012014.
- [34] E. J. Rechy-Ramirez and H. Hu, "Stages for Developing Control Systems using EMG and EEG Signals: A survey," 2011.
- [35] M. Ison, I. Vujaklija, B. Whitsell, D. Farina, and P. Artemiadis, "High-Density Electromyography and Motor Skill Learning for Robust Long-Term Control of a 7-DoF Robot Arm," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 4, pp. 424–433, 2016, doi: 10.1109/TNSRE.2015.2417775.
- [36] D. Farina *et al.*, "The extraction of neural information from the surface EMG for the control of upper-limb prostheses: Emerging avenues and challenges," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 4, pp. 797–809, 2014, doi: 10.1109/TNSRE.2014.2305111.
- [37] N. Jiang, H. Rehbaum, I. Vujaklija, B. Graitmann, and D. Farina, "Intuitive, online, simultaneous, and proportional myoelectric control over two degrees-of-freedom in upper limb amputees," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 3, pp. 501–510, 2014, doi: 10.1109/TNSRE.2013.2278411.
- [38] Y. Ren, L. Zhang, and P. N. Suganthan, "Ensemble Classification and Regression-Recent Developments, Applications and Future Directions [Review Article]," *IEEE Comput. Intell. Mag.*, vol. 11, no. 1, pp. 41–53, 2016, doi: 10.1109/MCI.2015.2471235.
- [39] J. M. Hahne *et al.*, "Linear and nonlinear regression techniques for simultaneous and proportional myoelectric control," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 2, pp. 269–279, 2014, doi: 10.1109/TNSRE.2014.2305520.
- [40] A. Ameri, E. N. Kamavuako, E. J. Scheme, K. B. Englehart, and P. A. Parker, "Support vector regression for improved real-time, simultaneous myoelectric control," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 6, pp. 1198–1209, 2014, doi: 10.1109/TNSRE.2014.2323576.
- [41] E. Alpaydin, *Introduction to machine learning*. .
- [42] S. K. Pal and S. Mitra, "Multilayer Perceptron, Fuzzy Sets, and Classification," *IEEE Transactions on Neural Networks*, vol. 3, no. 5. pp. 683–696, 1992.
- [43] K. W. Yusof and M. Hussain, "Engineering Challenges for Sustainable Future," in *Engineering Challenges for Sustainable Future*, 3rd ed., N. A. Wan Abdullah Zawawi, Ed. London, UK: Taylor & Francir group, 2016, pp. 304–307.

- [44] M. A. Nielsen, "Neural Networks and Deep Learning." Determination Press, 2015.
- [45] "Multilayer Perceptron | Abdelrahman Elogeel's Blog." <https://elogeel.wordpress.com/2010/05/10/multilayer-perceptron-2/> (accessed Apr. 25, 2020).
- [46] "machine learning - A list of cost functions used in neural networks, alongside applications - Cross Validated." <https://stats.stackexchange.com/questions/154879/a-list-of-cost-functions-used-in-neural-networks-alongside-applications> (accessed May 08, 2020).
- [47] R. Hostettler and S. Särkkä, "Basics of Sensor Fusion," 2019.
- [48] S. Ruder, "An overview of gradient descent optimization algorithms," pp. 1–14, 2016, [Online]. Available: <http://arxiv.org/abs/1609.04747>.
- [49] U. Manual, *Quattrocento*, no. March. Bioelettronica, 2020.
- [50] B. Hudgins, P. Parker, and R. N. Scott, "A New Strategy for Multifunction Myoelectric Control," *IEEE Trans. Biomed. Eng.*, vol. 40, no. 1, pp. 82–94, 1993, doi: 10.1109/10.204774.
- [51] L. J. Hargrove, A. M. Simon, R. Lipschutz, S. B. Finucane, and T. A. Kuiken, "Non-weight-bearing neural control of a powered transfemoral prosthesis," *J. Neuroeng. Rehabil.*, vol. 10, no. 1, pp. 1–11, 2013, doi: 10.1186/1743-0003-10-62.
- [52] "Choose a Multilayer Neural Network Training Function - MATLAB & Simulink - MathWorks Nordic." <https://se.mathworks.com/help/deeplearning/ug/choose-a-multilayer-neural-network-training-function.html;jsessionid=476cd28524c346971caf2ec639bc> (accessed May 05, 2020).
- [53] "One-step secant backpropagation - MATLAB trainoss - MathWorks Nordic." <https://se.mathworks.com/help/deeplearning/ref/trainoss.html> (accessed May 05, 2020).
- [54] R. Battiti, "First- and Second-Order Methods," vol. 166, pp. 141–166, 1992.
- [55] A. D'Avella, A. Portone, L. Fernandez, and F. Lacquaniti, "Control of fast-reaching movements by muscle synergy combinations," *J. Neurosci.*, vol. 26, no. 30, pp. 7791–7810, 2006, doi: 10.1523/JNEUROSCI.0830-06.2006.
- [56] "A Gentle Introduction to k-fold Cross-Validation." <https://machinelearningmastery.com/k-fold-cross-validation/> (accessed May 10, 2020).
- [57] K. Englehart and B. Hudgins, "A Robust, Real-Time Control Scheme for Multifunction Myoelectric Control," *IEEE Trans. Biomed. Eng.*, vol. 50, no. 7, pp. 848–854, 2003, doi: 10.1109/TBME.2003.813539.
- [58] M. Ortiz-Catalan, F. Rouhani, R. Branemark, and B. Hakansson, "Offline accuracy: A potentially misleading metric in myoelectric pattern recognition for prosthetic control," *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, vol. 2015-November, pp. 1140–1143, 2015, doi: 10.1109/EMBC.2015.7318567.
- [59] I. Vujaklija *et al.*, "Translating research on myoelectric control into clinics-are the performance assessment methods adequate?," *Front. Neurobot.*, vol. 11, no. FEB, pp. 1–7, 2017, doi: 10.3389/fnbot.2017.00007.

- [60] T. Roland, S. Amsüss, M. F. Russold, C. Wolf, and W. Baumgartner, "Capacitive Sensing of Surface EMG for Upper Limb Prostheses Control," *Procedia Eng.*, vol. 168, no. September, pp. 155–158, 2016, doi: 10.1016/j.proeng.2016.11.190.
- [61] T. Roland, K. Wimberger, S. Amsuess, M. F. Russold, and W. Baumgartner, "An insulated flexible sensor for stable electromyography detection: Application to prosthesis control," *Sensors (Switzerland)*, vol. 19, no. 4, 2019, doi: 10.3390/s19040961.
- [62] R. Merletti and D. Farina, *Surface electromyography: physiology, engineering, and applications*. .
- [63] N. Jiang, S. Muceli, B. Graimann, and D. Farina, "Effect of arm position on the prediction of kinematics from EMG in amputees," *Med. Biol. Eng. Comput.*, vol. 51, no. 1–2, pp. 143–151, 2013, doi: 10.1007/s11517-012-0979-4.
- [64] A. J. Young, L. J. Hargrove, and T. A. Kuiken, "The effects of electrode size and orientation on the sensitivity of myoelectric pattern recognition systems to electrode shift," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 9, pp. 2537–2544, 2011, doi: 10.1109/TBME.2011.2159216.
- [65] Y. Chae, C. Choi, J. Kim, and S. Jo, "Noninvasive sEMG-based control for humanoid robot teleoperated navigation," *Int. J. Precis. Eng. Manuf.*, vol. 12, no. 6, pp. 1105–1110, 2011, doi: 10.1007/s12541-011-0147-z.

# APPENDICES

## Appendix 1 Data acquisition equipment used

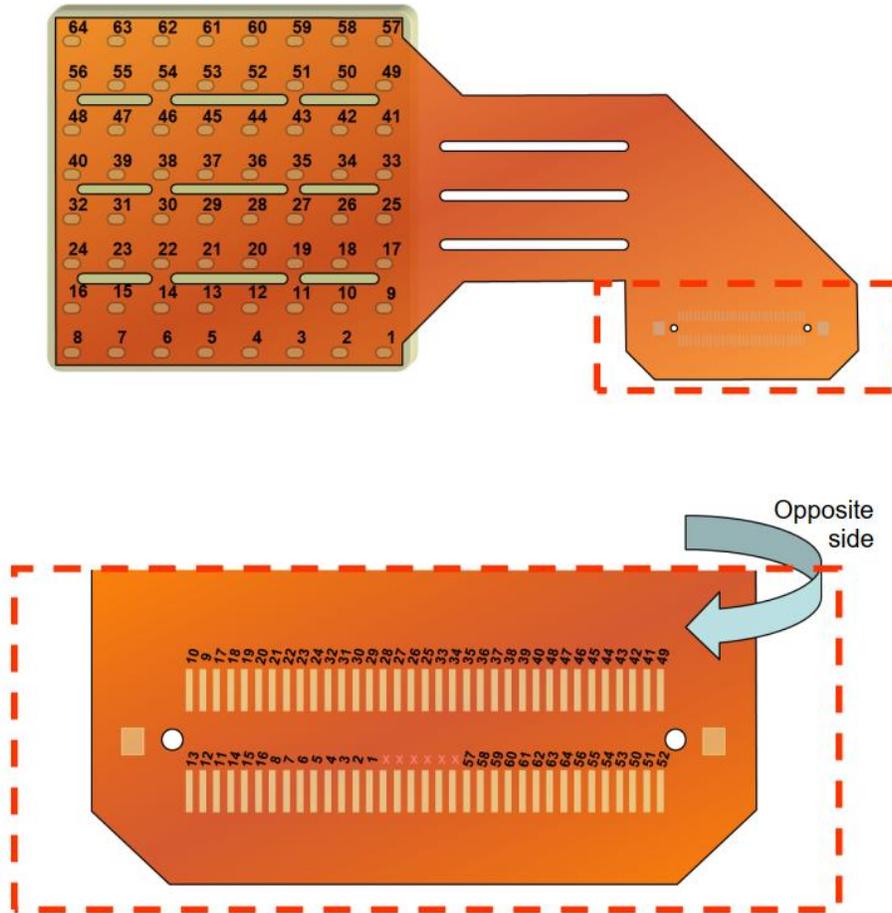


Figure A1.1 ELSCH064NM3 matrix of 64 sEMG channels. Inner-electrode distance is 10 mm. 3 of them were attached to subject forearm during data acquisition. Adapted from [49]



Figure A1.2 Signal amplifier used during data acquisition. Adapted from [49]

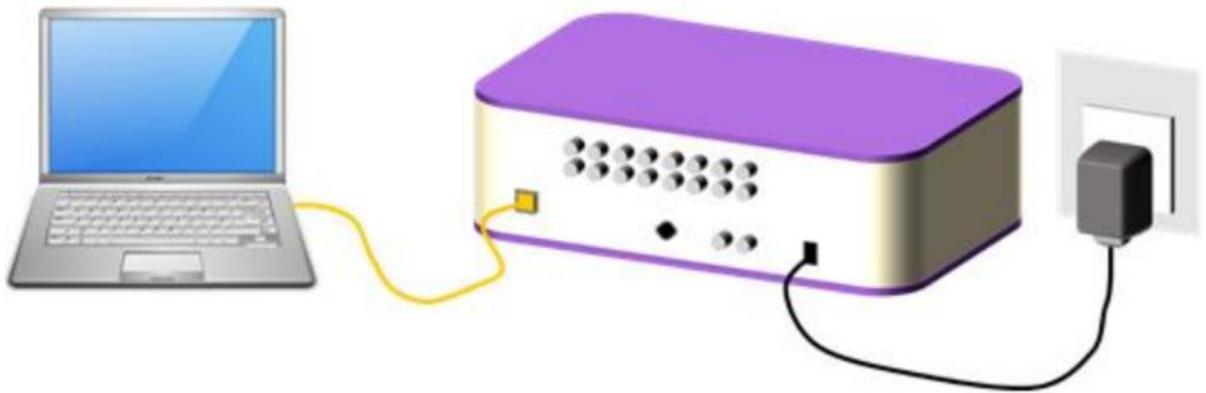


Figure A1.3 Amplifier-computer interface during data acquisition. Adapted from [49]

## Appendix 2 Neural network hyper-parameters

Table A2.1 Neural network hyper-parameters

No	Parameter	Value
1	Maximum Epochs	10000
2	Maximum Training Time	$\infty$
3	Performance Goal	0
4	Minimum Gradient	$1^{-10}$
5	Maximum Validation Checks	1000
6	Line search function	'srchbac'
7	Scale Tolerance	20
8	Alpha	0.001
9	Beta	0.1
10	Delta	0.01
11	Gamma	0.1
12	Lower Limit	0.1
13	Upper Limit	0.5
14	Maximum Step	100
15	Minimum Step	$1^{-6}$
16	$B_{\text{Max}}$	26

### Appendix 3 Motion estimation results for cross-validation

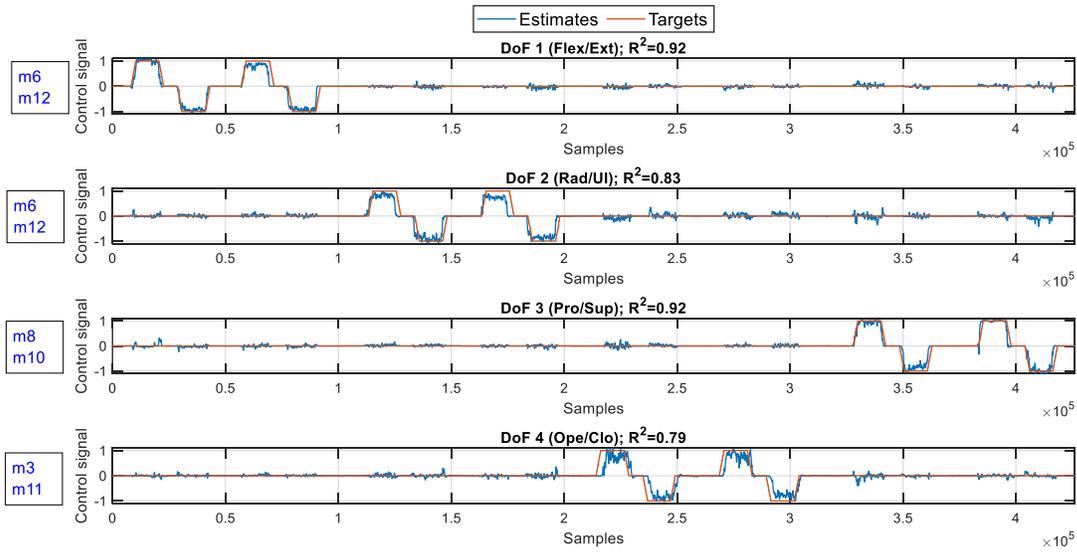


Figure A3.1 Motions estimation results and its performance values for each DoF for the first randomly compiled NN dataset. Text in the boxes represents which movement data were chosen for testing of DoFs kinematics estimation. This network used next separate movements data: DoF 1 – movement 6 (m6), movement 12 (m12); DoF 2 – movement 6 (m6), movement 12 (m12); DoF 3 – movement 8 (m8), movement 10 (m10); DoF 4 – movement 3 (m3), movement 11 (m11)

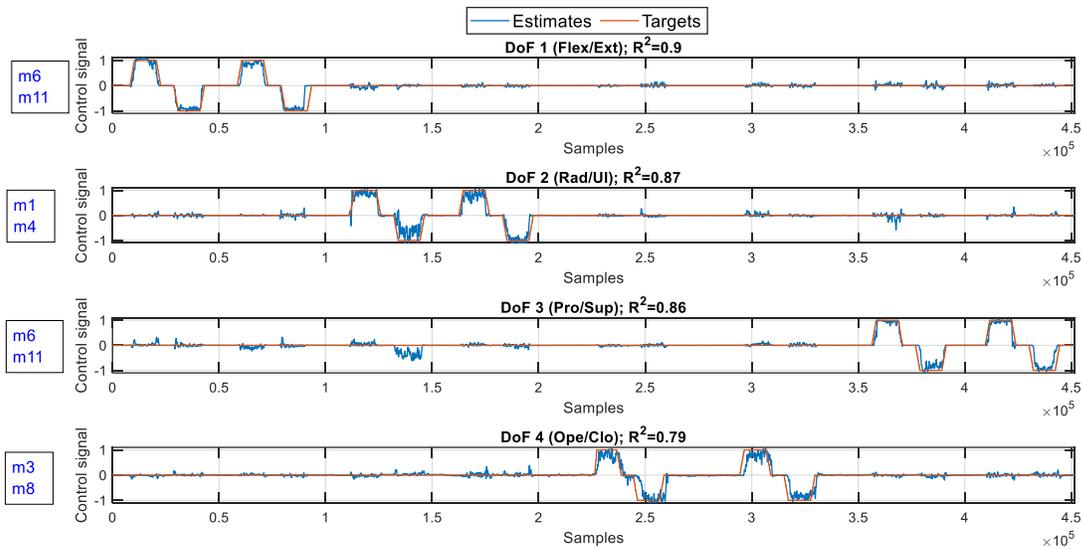


Figure A3.2 Motion estimation results and its performance values for each DoF for the second randomly compiled NN dataset. Text in the boxes represents which movement data were chosen for testing of DoFs kinematics estimation. This network used next separate movements data: DoF 1 – movement 6 (m6), movement 11 (m11); DoF 2 – movement 1 (m1), movement 4 (m4); DoF 3 – movement 6 (m6), movement 11 (m11); DoF 4 – movement 3 (m3), movement 8 (m8)

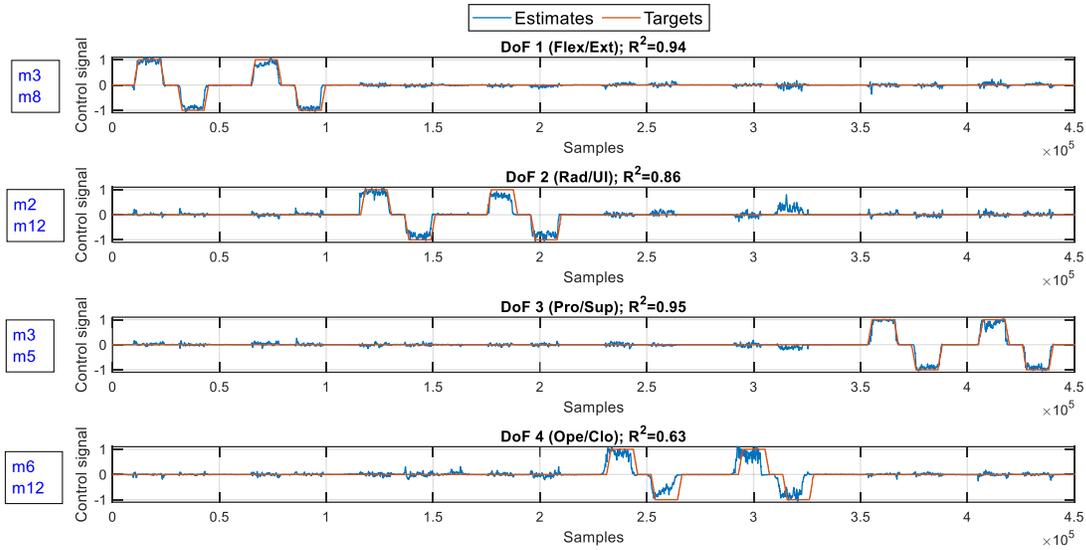


Figure A3.3 Motion estimation results and its performance values for each DoF for the third randomly compiled NN dataset. Text in the boxes represents which movement data were chosen for testing of DoFs kinematics estimation. This network used next separate movements data: DoF 1 – movement 3 (m3), movement 8 (m8); DoF 2 – movement 2 (m2), movement 12 (m12); DoF 3 – movement 3 (m3), movement 5 (m5); DoF 4 – movement 6 (m6), movement 12 (m12)

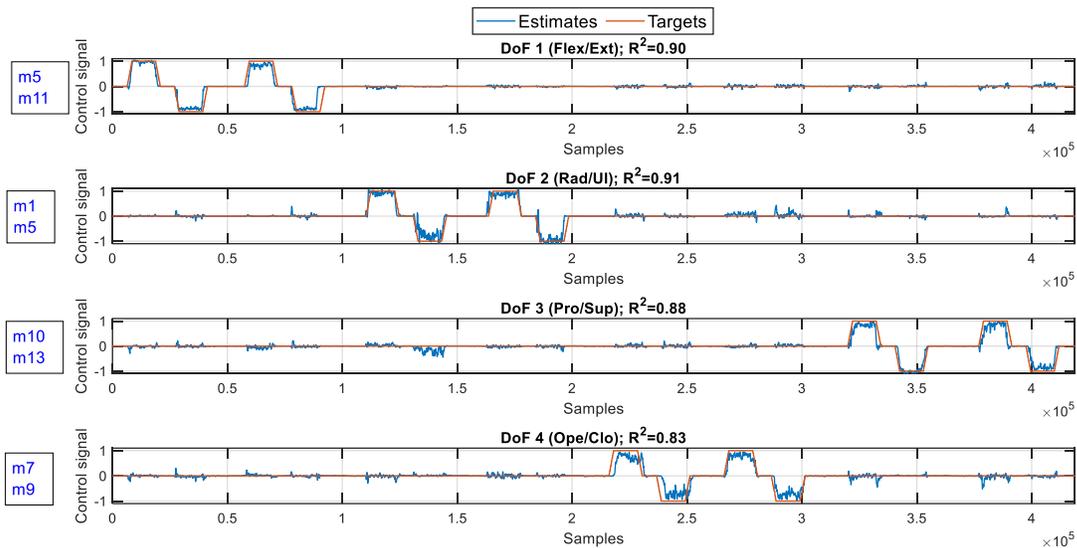


Figure A3.4 Motion estimation results and its performance values for each DoF for the fourth randomly compiled NN dataset. Text in the boxes represents which movement data were chosen for testing of DoFs kinematics estimation. This network used next separate movements data: DoF 1 – movement 5 (m5), movement 11 (m11); DoF 2 – movement 1 (m1), movement 5 (m5); DoF 3 – movement 10 (m10), movement 13 (m13); DoF 4 – movement 7 (m7), movement 9 (m9)

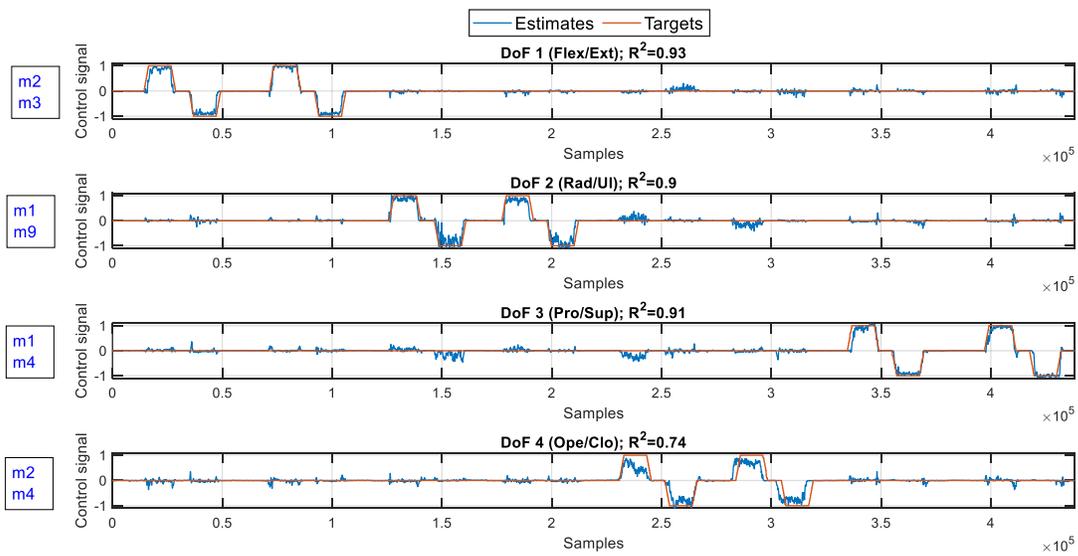


Figure A3.5 Motion estimation results and its performance values for each DoF for the fifth randomly compiled NN dataset. Text in the boxes represents which movement data were chosen for testing of DoFs kinematics estimation. This network used next separate movements data: DoF 1 – movement 2 (m2), movement 3 (m3); DoF 2 – movement 1 (m1), movement 9 (m9); DoF 3 – movement 1 (m1), movement 4 (m4); DoF 4 – movement 2 (m2), movement 4 (m4)

## Appendix 4 Motion estimation results for variable feature sets

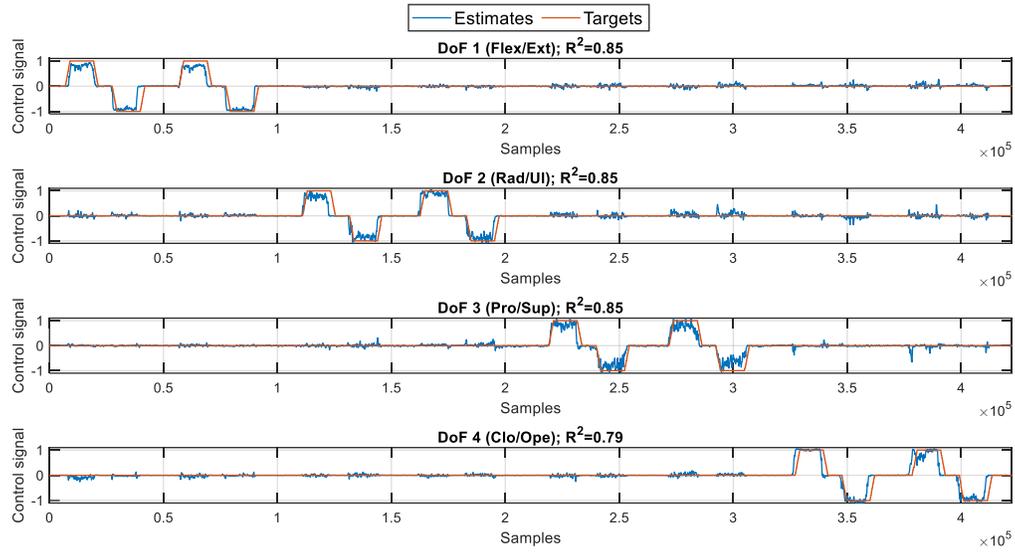


Figure A4.1 Estimation results for using features set 1. Each DoF motion estimation is presented together with its performance coefficient

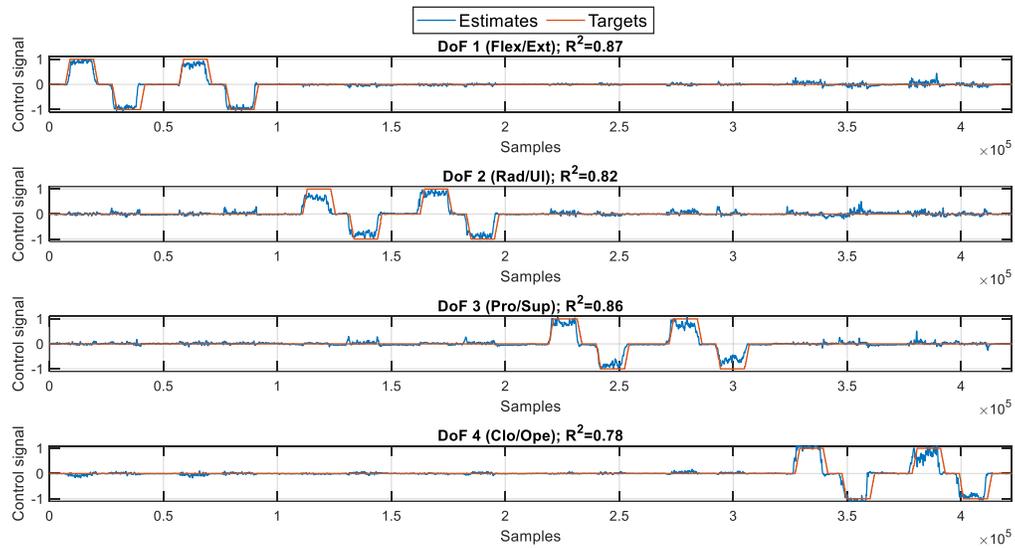


Figure A4.2 Estimation results for using features set 2. Each DoF motion estimation is presented together with its performance coefficient

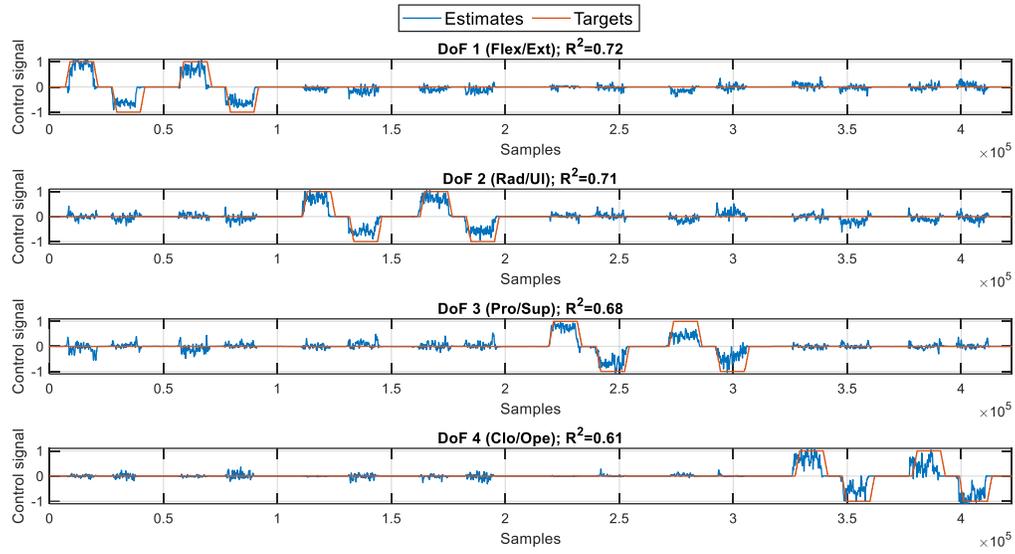


Figure A4.3 Estimation results for using features set 3. Each DoF motion estimation is presented together with its performance coefficient

## Appendix 5 Motion estimation results for channel reduction

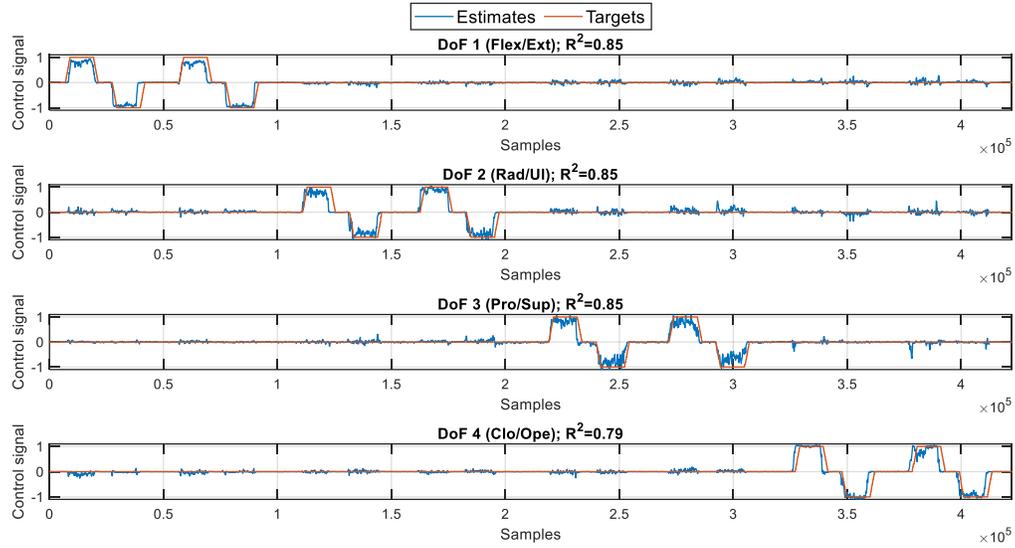


Figure A5.1 Estimation results for using features 187 sEMG channels. Each DoF motion estimation is presented together with its performance coefficient

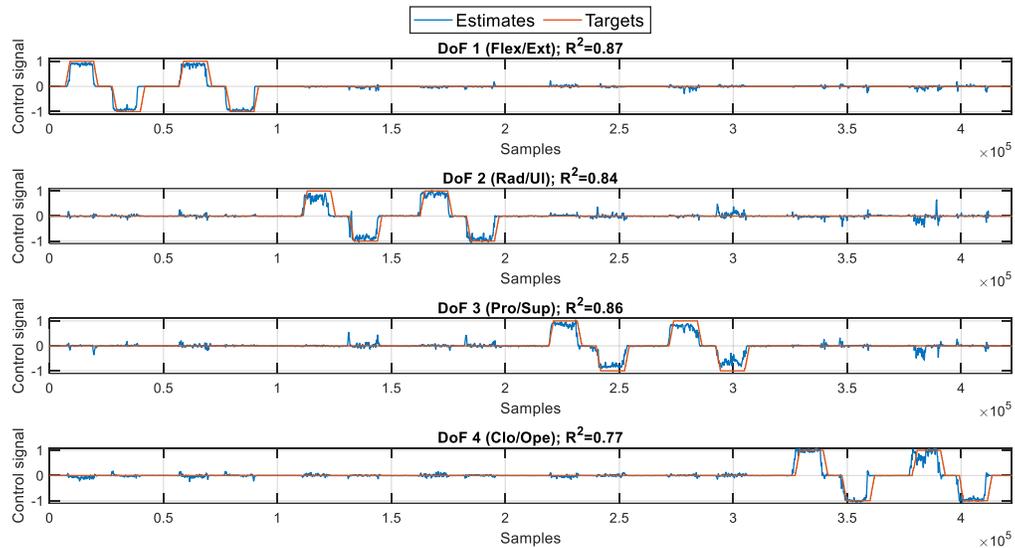


Figure A5.2 Estimation results when using 95 sEMG channels. Each DoF motion estimation is presented together with its performance coefficient

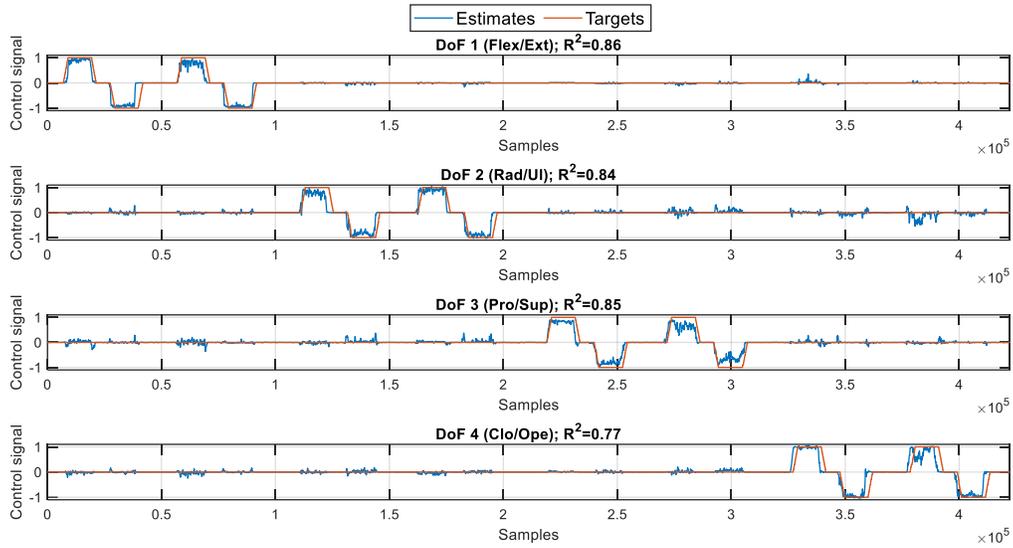


Figure A5.3 Estimation results when using 48 sEMG channels. Each DoF motion estimation is presented together with its performance coefficient

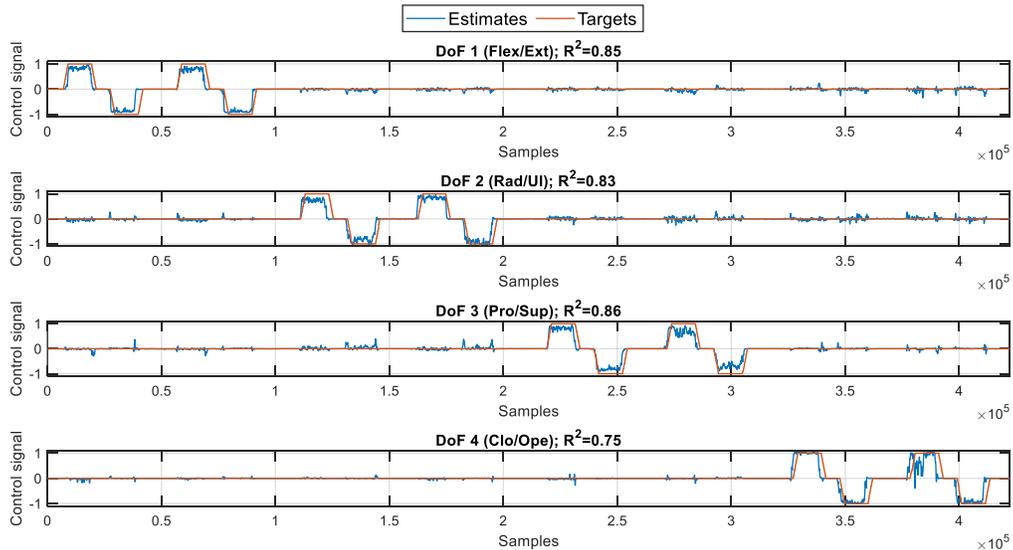


Figure A5.4 Estimation results when using 24 sEMG channels. Each DoF motion estimation is presented together with its performance coefficient

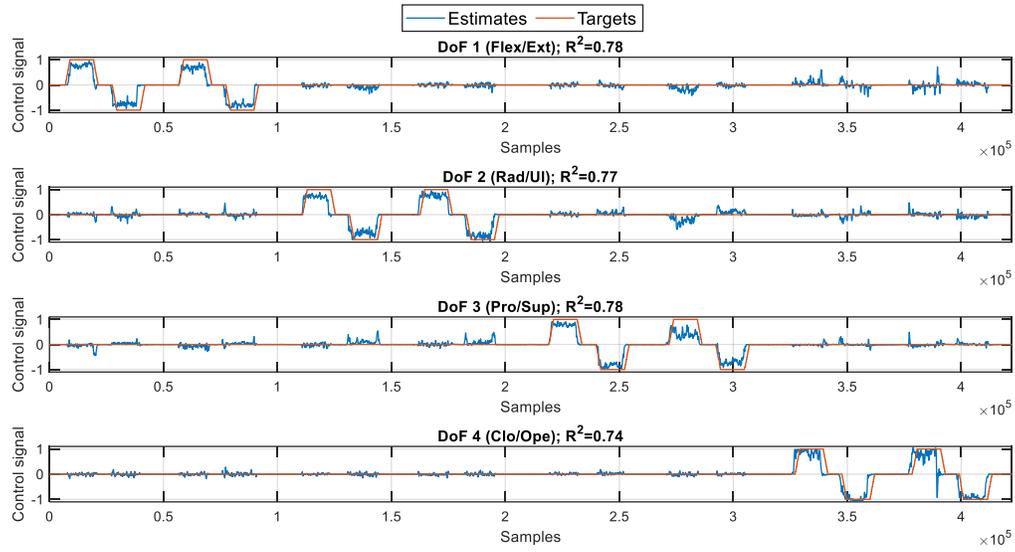


Figure A5.5 Estimation results when using 16 sEMG channels. Each DoF motion estimation is presented together with its performance coefficient

## Appendix 6 Motion estimation results for variable segmentation window size

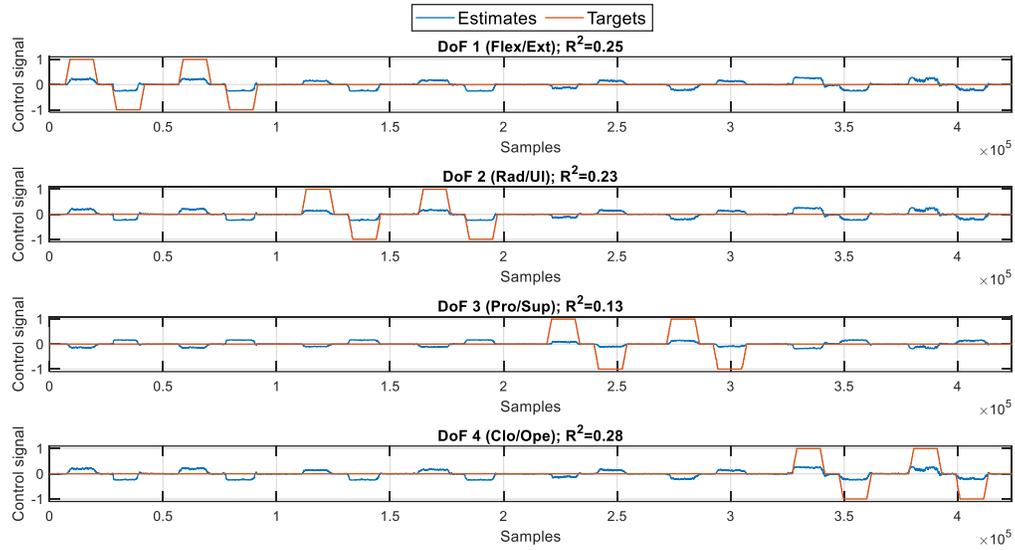


Figure A6.1 Estimation results when using 16 samples segmentation windows. Each DoF motion estimation is presented together with its performance coefficient

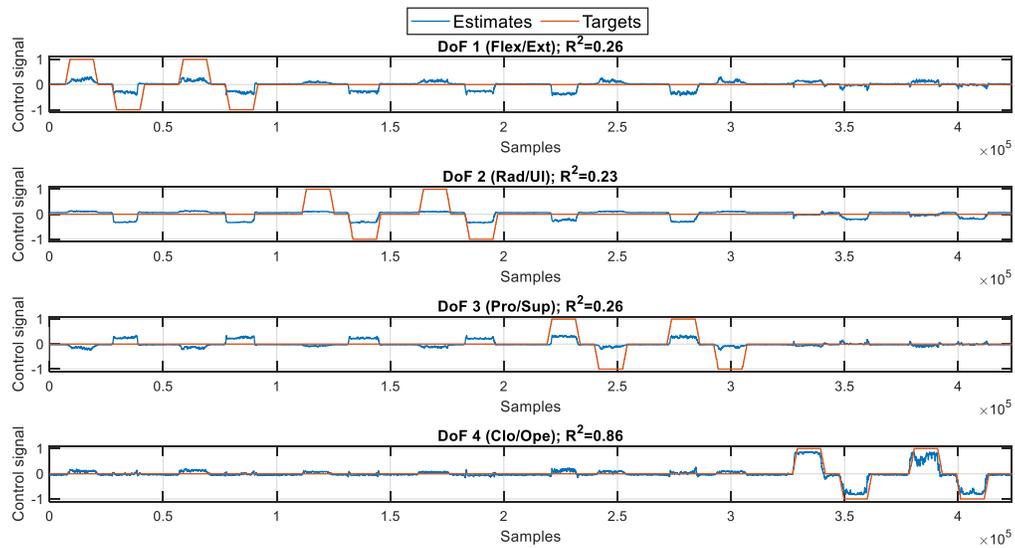


Figure A6.2 Estimation results when using 32 samples segmentation windows. Each DoF motion estimation is presented together with its performance coefficient

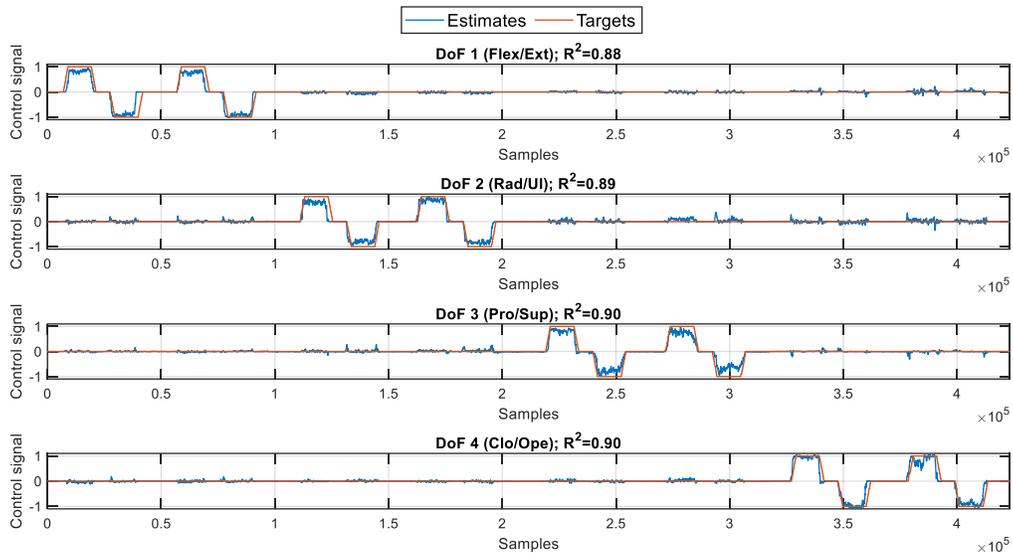


Figure A6.3 Estimation results when using 64 samples segmentation windows. Each DoF motion estimation is presented together with its performance coefficient

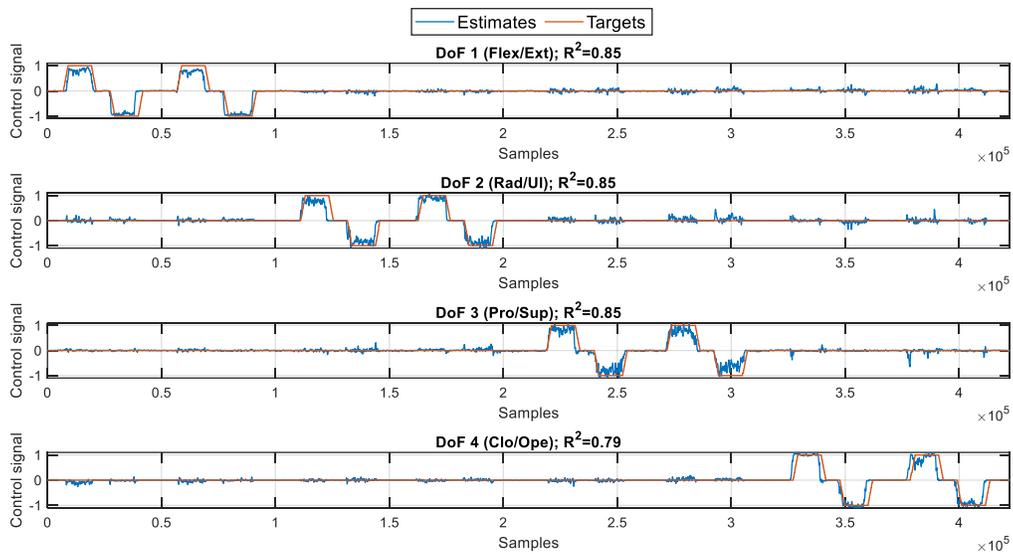


Figure A6.4 Estimation results when using 128 samples segmentation windows. Each DoF motion estimation is presented together with its performance coefficient

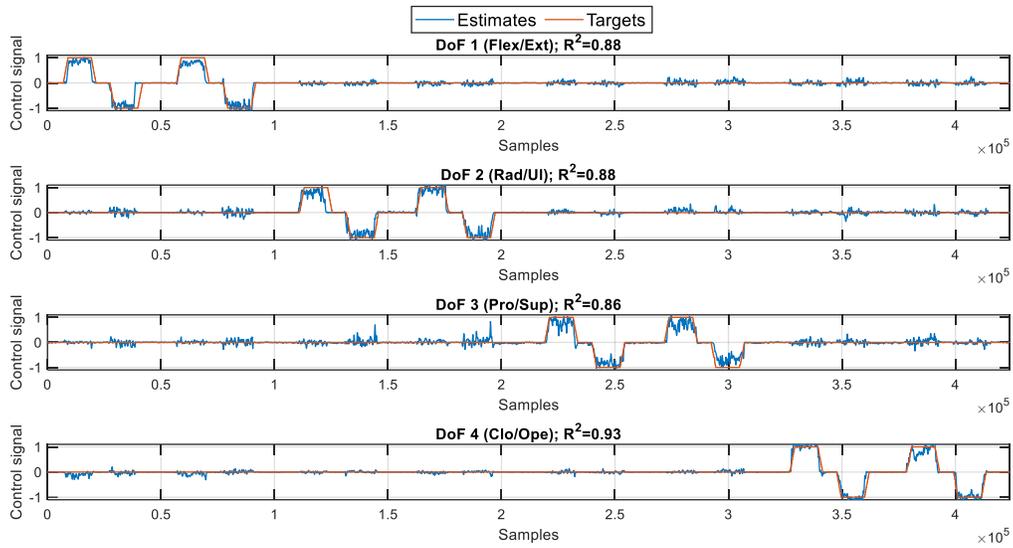


Figure A6.5 Estimation results when using 256 samples segmentation windows. Each DoF motion estimation is presented together with its performance coefficient

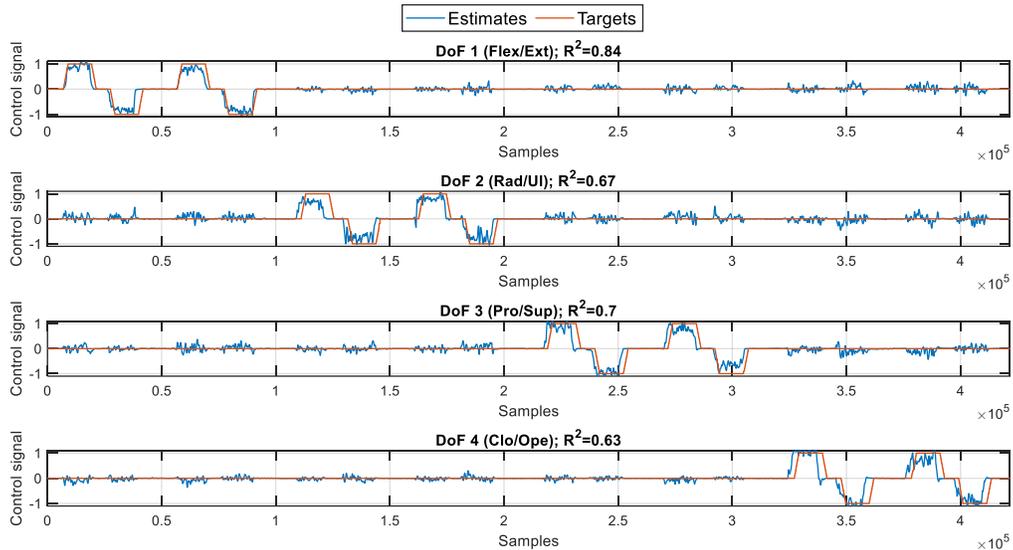


Figure A6.6 Estimation results when using 512 samples segmentation windows. Each DoF motion estimation is presented together with its performance coefficient

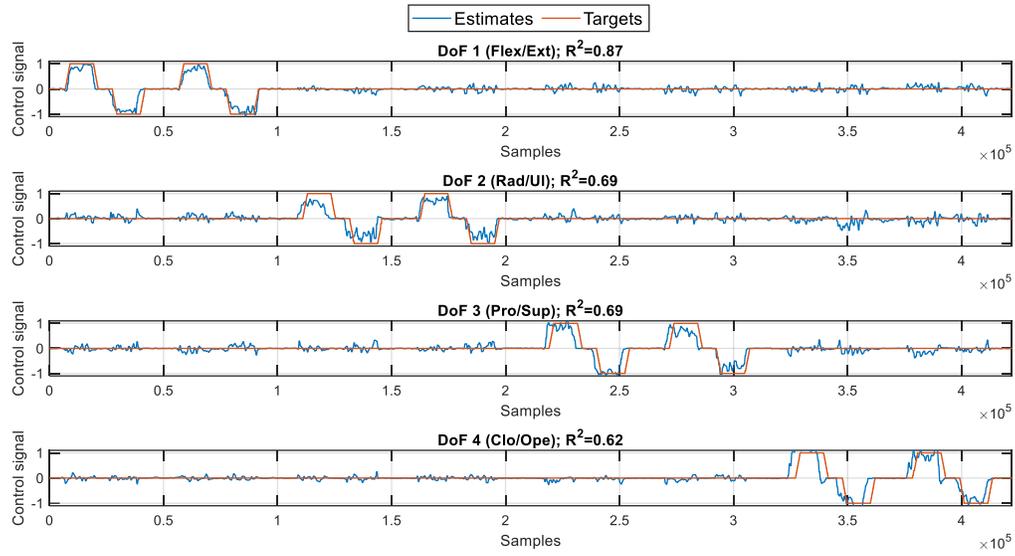


Figure A6.7 Estimation results when using 1024 samples segmentation windows. Each DoF motion estimation is presented together with its performance coefficient