

Pattern Recognition Systems for Myoelectric Control of Biomechatronic Prosthetic Hands

A thesis presented by

Alberto Dellacasa Bellingegni

in partial fulfilments of the requirements of the degree of

Philosophiae Doctor

in Biomedical Engineering

Coordinator

Prof. Giulio Iannello

Supervisor

Prof. Loredana Zollo

Co-supervisor

Prof. Eugenio Guglielmelli

November 2017

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

Alberto Dellacasa Bellingegni

Contents

ABSTRACT	7
CHAPTER 1 INTRODUCTION	11
1.1 Objectives.....	12
CHAPTER 2 OVERVIEW ON UPPER LIMB PROSTHETICS	17
2.1 Introduction.....	17
2.2 Protheses for Trans-Radial Amputation	18
2.3 Clinically Available Myoelectric Controls	22
2.4 Advanced Myoelectric Controls	24
2.5 Conclusion	28
CHAPTER 3 COMPARATIVE ANALYSIS OF PATTERN RECOGNITION ALGORITHMS FOR HAND MYOELECTRIC CONTROL	31
3.1 Introduction.....	31
3.2 Material and Methods	32
3.2.1 <i>sEMG Data Acquisition Protocol</i>	32
3.2.2 <i>NLR, MLP, and SVM Classification Algorithms</i>	35
3.2.3 <i>NLR, MLP, SVM Classifiers and Optimization Algorithm Implementation</i>	37
3.2.4 <i>NLR, MLP, SVM Downsampling and Creation of Generalization Set</i>	40
3.2.5 <i>LDA Classifier</i>	41
3.2.6 <i>Data Analysis</i>	42
3.3 Results.....	44
3.3.1 <i>Max Degree of Polynomial Features for NLR</i>	44

3.3.2	<i>Max Number of Hidden Layers for MLP</i>	46
3.3.3	<i>Max Number of Neurons for MLP</i>	47
3.3.4	<i>NLR, MLP, SVM Comparison Based on TR Sampling Rate</i>	48
3.3.5	<i>NLR, MLP, SVM Comparison Based on Computational Burden</i>	50
3.3.6	<i>NLR, MLP, SVM Comparison Based on EOF</i>	52
3.3.7	<i>NLR, MLP, SVM and LDA Comparison</i>	54
3.4	Discussion	57
3.5	Conclusions	60
 CHAPTER 4 EMBEDDED CONTROL OF PROSTHETIC HANDS VIA PATTERN RECOGNITION		63
4.1	Introduction	63
4.2	Design and Development	65
4.2.1	<i>The Proposed Myoelectric Control Strategy for Hands Prostheses</i>	65
4.2.2	<i>Control Strategy Variant for Force Management</i>	68
4.2.3	<i>Hardware</i>	71
4.2.4	<i>Software</i>	73
4.2.5	<i>Firmware</i>	75
4.3	Results and Conclusion.....	78
 CHAPTER 5 SEMG AND EENG PATTERN RECOGNITION FOR PROSTHETIC HAND CONTROL		83
5.1	Introduction.....	83
5.2	Methods.....	85
5.2.1	<i>Simultaneous ENG and sEMG recordings from Amputee Subject</i>	85
5.2.2	<i>Synthetic neural Data</i>	86
5.2.3	<i>The proposed ENG classification Method</i>	87

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni, discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018. La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca, a condizione che ne venga citata la fonte.

5.2.4	<i>Spike Sorting</i>	88
5.2.5	<i>Data Analysis</i>	90
5.3	Results	91
5.3.1	<i>Algorithm performance for simultaneous ENG and sEMG recordings</i>	91
5.3.2	<i>Performance decay with the number of classes</i>	93
5.4	Discussion	95
5.5	Conclusion	97
CHAPTER 6 CONCLUSIONS		101
LIST OF PUBLICATIONS		105
APPENDIX		107
BIBLIOGRAPHY		113
FIGURE CAPTIONS		123
TABLE CAPTIONS		129



Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Dellacasa Bellingegni Alberto

Abstract

The human hand is considered as the highest example of dexterous system capable of interacting with different objects adapting its manipulation abilities to them. Therefore, the hand loss causes severe impairment for the amputee and can significantly reduce quality of life. In the last 70 years there have been significant improvements in the upper limb prosthetic hand field thanks to the advancements in the technological field and in the surgical procedure leading to prosthetic devices that are more conceived to reproduce aesthetical as well as functional features of the lost limb. However, the currently adopted hand prosthesis surface electromyography (sEMG) control strategies, representing the clinical state of art, do not provide the users with a natural control feeling and do not exploit all the potential of commercially available multi-fingered hand prostheses. Pattern recognition (PR) and machine learning techniques applied to sEMG can be effective for a natural control based on the residual muscles contraction of amputated people corresponding to phantom limb movements. As the researches has reached an advanced grade accuracy, these algorithms are mature and widely validated, and the embedding is necessary for the realization of prosthetic devices.

This thesis wants to address the specific issue of enhancing both the performance and the control feeling of existing multi-grasp prosthetic hands, by designing a new embedded control based on pattern recognition algorithms applied to sEMG signals. To this purpose, firstly a comparison among different supervised machine learning techniques on data collected from 30 people with trans-radial amputation is carried out in order to provide innovative engineering tools and indications on how to choose the most suitable classification algorithm based on the application and the desired results for prostheses control. Then, the obtained result has been used for the design and evaluation of an embedded control system (hardware-firmware-software) for hand prostheses capable to handle up to five different grasping movements, successfully tested on amputee subjects.

As complementary activity, this thesis proposes a new approach for neural control of hand prostheses, grounded on pattern recognition applied to the envelope of

neural signals. It was demonstrated that it is possible to apply the well-known techniques of EMG pattern recognition to a conveniently processed neural signal and can pave the way to the application of neural gesture decoding in upper limb prosthetics. This intends to overcome limitations of traditionally adopted techniques of sEMG and ENG processing allowing both to control a prosthetic device and to stimulate the PNS.

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Chapter 1

Introduction

The hand loss causes severe impairment for the amputee and can significantly reduce quality of life. The relevance of upper limb loss in the international scenario motivates the flourishing research in the field of upper-limb prosthetic [1].

In clinics the state-of-the-art technology for people with trans-radial amputation is commonly, a dual-site controlled myoelectric hand prosthesis. The available single degree of freedom (DoF) is actuated by applying a simple threshold or a proportional amplitude method on surface electromyography (sEMG) signals recorded from antagonistic muscles (e.g., wrist flexor and wrist extensor) that can be easily contracted in a separate way. In the case of multi-fingered hand prosthesis with several degrees of freedom (DoFs), but still having two control signals, the switching between DoFs or predefined grasps is normally made by co-contraction, as in a finite state machine (control strategy). This serial operation, despite being extremely robust, is slow and unnatural; in addition, it requires considerable training and cognitive effort [2]. On the other hand, Targeted Muscler Re-innervation (TMR) [3], via surgical operation, allows replacing nerves from the stump of persons with amputation to different anatomical muscles (e.g., chest muscles) in order to obtain independent signals. The risk associated to the surgical re-innervating operation is the main drawback that limits the applicability of this technique to all the kinds of amputations [4]- [1]. Hence, PR techniques based on sEMG currently represent the best compromise between invasiveness and prosthesis controllability and thanks to the notable scientific progress, allows increasing the number of controllable DoFs by keeping low the number of utilized electrodes [5]. Recognizing the user's will, control strategy resorting to PR techniques could improve performance by mapping the actuation of the prostheses

on sEMG signals produced as result of phantom limb gestures [6]. The system becomes more user-friendly and makes easier complex tasks that may include the sequential actuation of different DoFs.

1.1 Objectives

Myoelectric control systems based on PR techniques (Fig. 1) rely on supervised machine learning classification algorithms.

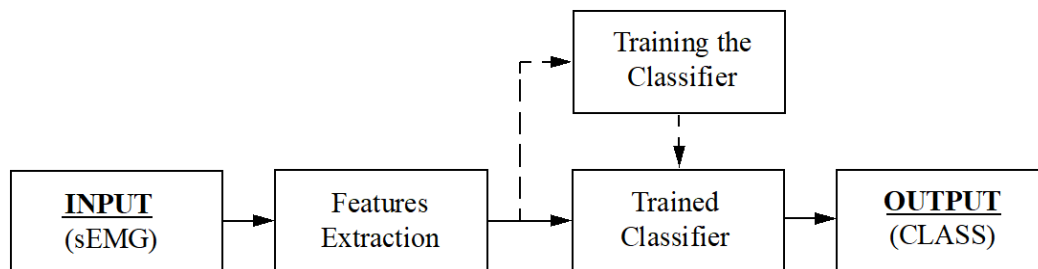


Figure 1. Block diagram of a generic pattern recognition system based on sEMG signals.

An initial training phase is needed, during which the system learns the way of linking the gestures to specific myoelectric patterns. Subsequently, the trained system is able to associate the desired task to the recorded pattern. Usually the feature extraction step precedes classification of sEMG signals; hence the most important components of the recorded myoelectric signal on a chosen time window are identified and selected [7] in order to improve the stability of the features (reducing variance and increasing classification performance). Previous studies suggest that the optimum window length for PR controls ranges from 150 to 250 ms depending on the skill of the subject [8]. For real-time applications it is conventionally accepted that the actuation delay must be less than 300 ms. Therefore here it is proposed to use “raw” filtered sEMG signals as input features; this enables an extreme reduction of the classification time and response time of the system without significant loss of system performance [9]- [10]- [11]. The saved time is used to improve the stability of the classification by means of post processing techniques as voting and/or threshold policies [12]- [13].

Linear classifiers, such as Linear Discriminant Analysis (LDA), Logistic Regression (LR) or Support Vector Machine (SVM) with linear kernel, and

nonlinear classifiers, such as Non-linear Logistic Regression (NLR), Support Vector Machine (SVM) with nonlinear kernels and Multi-Layer Perceptron (MLP), represent the state-of-the-art about PR classifiers [14]- [15]. The main difference between linear and nonlinear classifiers consists in the shape of the decision boundary: straight line, or plane in the first case and curved line, or surface, in the second. Performance, complexity and computational time usually increase together. Hence, the choice of a classification algorithm should not be entirely relied upon performance, but rather on a trade-off between computational burden and performance, especially in embedded systems. The main issue related to these algorithms is the robustness and reliability of the classification and it is unlikely to use directly the classification outputs for the online control of the prosthetic device. Hence, a control strategy able to combine robustness, reliability and less cognitive effort has to be included between the classification algorithm and the internal control of the prosthetic hand. Therefore, aim of this thesis is to provide useful insights into the choice of the suitable classifier (and its specific internal settings) for the embedded control of multi-fingered hand prostheses and to design a fully embedded control unit able to recognize the user will, via PR algorithms, and to control a multi-fingered prosthetic hand exploiting a new control strategy.

However, although solutions based on patten recognition on EMG signals are promising, they suffer from the limitation that the subject cannot be provided with a natural sensory feedback. On the other hand, it has been shown that invasive solutions based on neural electrodes allow directly stimulating the Peripheral Nervous Systems (PNS) and eliciting in the patients close-to-natural tactile feedback [16]- [17]- [18]- [19]. This kind of interfaces can be used with the twofold purpose of stimulating the PNS and recording the neural information coming from the brain to drive the residual muscles. Hence, assuming that it is possible to decode human motion intention from the neural signals, i.e. the electroneurogram (ENG), neural interfaces offer the huge advantage that they can both record from and stimulate the PNS in a more natural way than the other interfacing systems. Hence, as complementary activity, a new method for processing ENG signals specifically aimed at controlling hand prostheses has been proposed. This intends to overcome limitations of traditionally adopted techniques of ENG processing by (i) computing the ENG envelope (eENG), starting from the recorded neural signals; (ii) resorting to sEMG PR techniques applied to the eENG. The main advantages of

the proposed approach are related to the reduced computational burden and the on-line processing (which are paramount for closed loop interfaces).

This thesis is structured as follows:

- Chapter 2 presents an overview of the upper limb prostheses at the moment clinically available and the state-of-art about research myoelectric controls for prosthetic hands;
- In Chapter 3 is reported a comparative analysis of PR algorithms on sEMG data collected from 30 people with trans-radial amputation in order to identify the most suited one for the use in the control of multi-fingered prosthetic hands. The following contents are taken from the paper that the candidate has published on *Journal of NeuroEngineering and Rehabilitation* [20];
- Chapter 4 presents the innovative control strategy and the detailed description of the steps taken for the design and evaluation of an embedded control unit for the control of multi-fingered prosthetic hands;
- Chapter 5 presents a new approach for neural control of hand prostheses, grounded on pattern recognition applied to the envelope of neural signals. The following contents are taken from the paper that the candidate has submitted on *Journal of Neuroscience Methods* [21];
- In Chapter 6 are finally reported a discussion and conclusive remarks.

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Chapter 2

Overview on Upper Limb Prosthetics

2.1 Introduction

Limb prostheses are tailor-made medical devices that try to replace the functionality of lost joints and to date they often draw on the engineering field of biomechanics in their technological solutions. In particular, upper limb myoelectric prostheses require a high-tech contribution because they try to replace many complex functions while respecting rigid constraints in terms of weight, size and aesthetics.

The European Standard BS EN ISO 9999:2016 establishes a classification and terminology of assistive products, especially produced or generally available, for persons with disability. In these terms, prostheses are defined as “externally applied devices used to replace wholly or in part an absent or deficient body part. [...] An upper limb prosthetic system is a collection of compatible components, usually produced by a single manufacturer and commercially available; the components may be integrated with any individually manufactured component to produce a range of different upper limb prostheses”. Despite the advances in prosthetics, not all people with amputation choose to use a prosthesis to compensate the lack of the limb. Indeed, subjects that have suffered from a complete limb loss, or who were born with this absence, in some cases feel the prosthetic device as an obstacle. Inexperience and annoyance in the use of a foreign body represent the main reported issue. For this reason, it is possible to find in literature several types of prosthetic control exploiting machine-learning techniques in order to increase the acceptability of the prosthesis.

This chapter presents the state-of-art trans-radial prosthetic hands and a detailed overview of four of the most common pattern recognition algorithms used in scientific literature for the control of prosthetic hands.

2.2 Prostheses for Trans-Radial Amputation

It is generally possible to pick out four fundamental elements in the mechanical architecture of a generic upper limb prosthesis: the terminal device, artificial limb segments, a certain number of functional components (active and passive articulation) driving the artificial segments and the socket [22]. The socket is the interface between the patient by means of which the prosthesis is suspended to the subject's stump and has a fundamental role for satisfactory use of the device. Made with plastic resins or carbon fiber, each socket is custom hand-made from a plaster cast of the stump in order to ensure the maximum comfort. There are three main types of prostheses in the upper limb scenario: (i) Cosmetic, (ii) Kinematic, and (iii) Myoelectric. Cosmetic restoration is a popular prosthetic option that is similar in appearance to the non-affected arm or hand (Fig. 2) and provides just a simple aid in balancing and carrying (passive prostheses). This type of prostheses is used by people for which the outward appearance plays a very important role or by people that cannot successfully used active prostheses. Cosmetic Prostheses are very realistic and assure a psychological support to permit the amputees to live in the society avoiding to be shy for their difference with people having sound limbs.



Figure 2. Cosmetic Prosthesis.

A kinematic prosthesis is powered and controlled by gross body movements. These movements (usually of the shoulder, upper arm or chest) are captured by a harness system, which is attached to cables controlling the active joints (Fig. 3).



Figure 3. Kinematic elbow Prosthesis.

Despite excellent construction characteristics, lightness, robustness and reliability such prostheses provide a low level of comfort, due to the presence of the support straps and traction cables.

Myoelectric prostheses the energy required to operate the active joints is supplied by external sources. The widespread kind of this category uses small electrical motors to provide power to the articulations. The motors driving the active joints are directly activated by the amputee by means of sEMG input commands that are collected by proper sensors placed in appropriate location provided in the socket of the prosthesis (Fig. 4). These signals are processed by a programmable electronic circuit, which carries out the control strategy to operate the device.



Figure 4. Myoelectric Prosthesis.

Unlike a body-powered prosthesis that requires gross body movement to operate it, a myoelectric prosthesis only requires the wearer to generate simple control signals. However, the battery system requires maintenance, (which includes charging, discharging, eventual disposal and replacement) and, because of the electromechanical components, this type of prosthesis tends to be heavier than other prosthetic options, although advanced suspension techniques can minimize this sensation.

For this last type of prostheses are commercially available two types of terminal device: sDoF and DOFs hands. sDoF hands, or manipulators, are prosthetic devices the manipulator is a prosthesis optimized for manual work and, in its realization, usually no particular attention is paid to the aesthetics in lieu of gripping force, robustness and reliability (Fig. 5).



Figure 5. Single degree of freedom hands. a) Ottobock myoelectric speed hand without external cover. b) Ottobock myoelectric speed hand with external cover. c) Ottobock electric Greifer terminal device.

DoFs hand prostheses are instead very similar to the human hand with five fingers and on the market, there are three main devices: “*Michelangelo*” (Ottobock), “*Ultra-Limb*” (Touch Bionics) and “*Bebionic*” (Steeper). The Michelangelo hand [23] (Fig. 6) is an anthropomorphic prosthesis with three degrees of actuation: one to open-close the hand, one to adduct-abduct the thumb and the last to rotate the wrist.



Figure 6. Michelangelo hand prosthesis by Ottobock.

Bebionic3 hand [24] (Fig. 7a) is a five-fingered prosthetic device with one actuator for each finger. It can achieve 14 different grip patterns and is designed to handle almost anything that the amputee needs to do in an average day, from eating meals and carrying bags, to opening doors switching on lights and typing. The sound hand of the user moves Thumb and wrist.

At last, the Ultra-Limb prosthesis [25] (Fig. 7b) features powered rotating thumb and individually active articulating fingers offering unparalleled dexterity and reliable access to precision grip patterns. With 24 grip options can also be controlled via a mobile application.



Figure 7. a) Bebionic3S prosthetic hand by RLS Steeper. b) Ultra-Limb prosthetic hand by Touch Bionics

2.3 Clinically Available Myoelectric Controls

As mentioned above, myoelectric prostheses use small electrical motors, which are directly driven by the amputee by means of sEMG input resulting from the contraction of residual muscles of the stump. In clinics, the state of art allows the user to control just one degree of freedom (DoF) at a time even in the case of DOFs hands. The available DoF can be actuated by different amplitude methods on sEMG signals recorded from antagonistic muscles (e.g., wrist flexor and wrist extensor) that can be easily contracted in a separate way. Three are the main myoelectric controls to regulate the speed of the prosthesis:

1. *Single Threshold;*
2. *Double Threshold;*
3. *Proportional.*

The single threshold (Fig. 8a) is the simplest type of motion control allowing the constant speed actuation of the motor when the sEMG signal exceeds a pre-set threshold. Both the speed and the threshold are customizable according to the characteristic of the person with amputation.

Having very few requirements, this technique has the advantage to be applicable to almost all subjects. On the other hand, not being able to adjust the speed in real time is undoubtedly an unnatural solution, which has great limitations both in the taking of objects and in human interactions. A direct evolution from this method is represented by the double threshold (Fig. 8b) in which the movement speed can take two values discriminated by two different sEMG threshold.

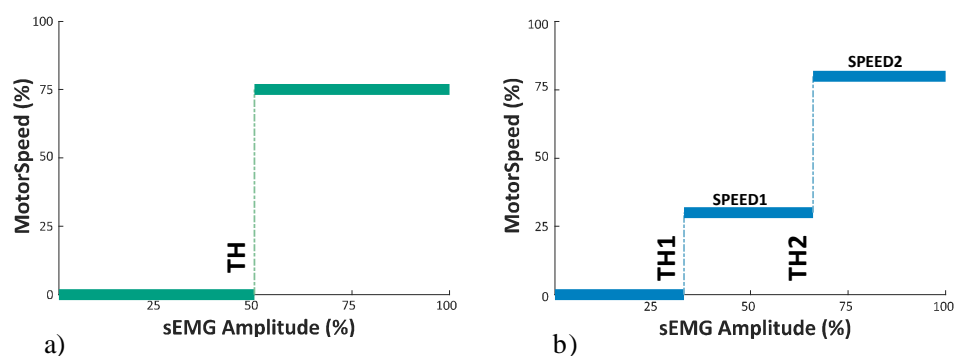


Figure 8. a) Single Threshold control trend. b) Double Threshold control trend.

Exceeding the first threshold the prosthesis moves with a minimum speed, while surpassing the second one results in a greater speed. Again, it is possible to customize both the thresholds and the rotation speeds of the actuators. This model of control, although still rudimentary, represent an improvement in the usability of the prosthetic device, however it still suffers issues due to real-time uncontrollability of the movement speed. Proportional sEMG signal amplitude control (Fig. 9), which is the most commonly used, overcomes these issues by regulating the speed of the actuation imposing a proportional linearity with the muscular contraction.

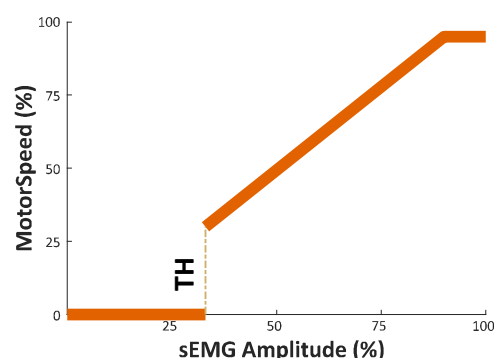


Figure 9. Proportional control trend.

In the case of multi-fingered hand prosthesis with DoFs as well as for multi-limb prostheses, but still having two control signals, the switching between DoFs, or different active joint, or predefined grasps is obtained by mean of a control strategy usually representable as a finite state machine. Certainly, the most used control strategy relies on a cyclic selection (Fig. 12). The switch among the states of the cycle is achievable by mean of different trigger command. Commonly:

1. *Electro-Mechanical Switch*: the transition to the next state can takes place pressing an electro-mechanical switch with the sound limb.
2. *Third Sensor*: the progress in the cycle is made by contracting a muscle that has exclusively this function (e.g. anteposition of the shoulder) on which is placed a dedicated sEMG sensor.
3. *Co-contraction*: The selection signal relies on a combination of signals from the motor control sensors by contracting the two command muscles.

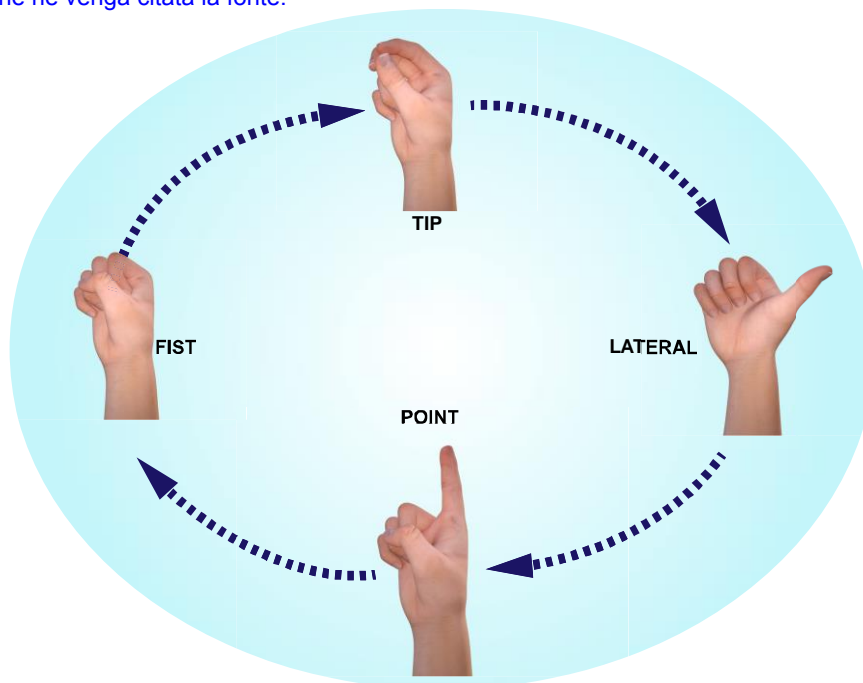


Figure 10. Cyclic Selection Control Strategy representation for the selection of the desired grasp from a predefined set.

This serial operation, despite being extremely robust, is slow and unnatural; in addition, it requires considerable training and cognitive effort [2]. Indeed, in order to avoid the cyclic selection different strategies has been proposed, exploiting multi-position slide switches or even through a mobile app, but they are far from a physiological motion control. Hence, thanks to its simplicity and robustness, agonist/antagonist proportional control is still the most adopted option for myoelectric prostheses in commercially available systems as well as in clinical applications [26].

2.4 Advanced Myoelectric Controls

The technological advancement and the stimulus towards overcoming the cyclic selection strategy has led to the development of control systems that combine technological advances and those in the surgical field. In these terms two additional techniques have been developed, i.e. the Targeted Muscle Reinnervation (TMR) [27] and the recent nerve transfer in brachial plexus injury [28]. In the TMR [27], the remaining arm nerves are reallocated to residual chest or upper-arm muscles

that are no longer biomechanically functional due to the amputation. Once re-innervated, these muscles serve as biological amplifiers of motor commands from the transferred arm nerves and provide physiologically appropriate sEMG signals for the arm control. This procedure is especially applied to subjects with very proximal amputation, which usually control the motors of the prosthetic arm through switches actuated with residual shoulder movement or myoelectric signals acquired from muscles of the chest and back. With respect to these control techniques, TMR (Figure 11) presents several advantages, such as improvements in function (measured both objectively and subjectively), ease of use, simultaneous control of more than one DoF, fast and seamless motion [29].



Figure 11. TMR prosthetic setup.

In [28] nerve transfer in brachial plexus injury is presented. It is defined “bionic reconstruction.” After free functioning muscle transfer procedure for the restoration of shoulder and elbow functions, the hand muscle activity has been restored selectively transferring the nerves in order to optimize the number of electromyographic sites. The surgical procedure and the rehabilitation program allowed the improvement of sEMG activity and the maximization of the prosthetic hand functions. However, the high cost and risk associated to the surgical re-innervating operation are the main drawbacks that limits the applicability of these techniques to all the kinds of amputations [4]- [1].

On the other hand, the development of PR techniques on sEMG signals aims to increase the number of controllable DoFs (and consequently the number of feasible functions) through a physiological control without a significant increase in cost and risk. Myoelectric control based on PR techniques resorts to supervised machine learning algorithms [7]: in an initial training phase the system learns to associate different hand gestures to different myoelectric patterns based on the phantom limb effect (sensation that an amputated or missing limb is still attached). This association is then adopted in the daily use of the prosthesis. Since approximately 60 to 80% of individuals with amputation experience phantom sensations in their amputated limb, this technique could be applied for a large number of people with upper limb amputation. The first step of PR consists of the feature extraction: the main components of the recorded myoelectric signal are identified and selected in a time window between 150 and 250ms, depending on the skill of the subject [8]. The main purpose of this step is to enhance the information content, retaining information about contraction discrimination while discarding the irrelevant ones. Feature extraction techniques are typically in the time domain and in the frequency domain [14]. Commonly used time domain techniques are: mean absolute value (MAV) [30]- [31]- [32], zero crossing (ZC) [30], waveform length (WL) [33], root mean square (RMS) [30], slope sign change (SSC) [33], and AR model [34]. Techniques in the frequency domain are more accurate, but also computationally more demanding than time-domain techniques. They include: Short-Time Fourier Transform (STFT), wavelet transform (WT) [35]- [36]- [37]- [38]- [39]- [40] and wavelet packet transform (WPT) [41]. In order to reduce the computational complexity and, at the same time, increase the performance of the subsequent classification [42], the dimensionality reduction through the Principal Component

Analysis (PCA) can be applied to EMG signals [43]. Classification follows feature extraction and dimensionality reduction. It is responsible for the decoding of the patient motor intention. PR classifiers [44]- [45]- [14] can be grouped in the following main categories, sorted by increasing complexity: linear classifiers, such as Linear Discriminant Analysis (LDA) or Perceptron or Support Vector Machine (SVM), non-linear classifiers, such as Non Linear Logistic Regression or SVM with non-linear kernels, and Multilayer Perceptron or Multilayer SVM. The main difference between linear and non-linear classifiers is the shape of the decision boundaries that divide the features space in classes: straight line (or plane or hyper-plane) in the linear case and curved in the non-linear one. An extensive inspection of classifiers can be found in [15] and [7]. In the literature, a large number of feature sets and classification algorithms employed in myoelectric control have been investigated and compared in detail. Notwithstanding, there is no clear evidence of the superiority of one classifier over the other ones; it is shown that classifiers can reach similar performance in terms of offline accuracy, provided that an appropriate feature set and an adequate number of sampling sites of the EMG signal are used [15]. In [46] the effects of the choice of feature sets over classifier performance are in-depth investigated. Moreover, methods based on “raw” filtered EMG signals have been recently proposed; they allow considerably decreasing the time for feature extraction and skipping the feature reduction step without significant loss of system performance [10]- [11].

The main issue related to control based on PR algorithms on sEMG signals is the robustness and reliability of the classification and it is unlikely to use directly the classification outputs for the online control of the prosthetic device. It is worth observing that the viability of PR in a clinical setting should consider that off-line accuracy could not correspond to real-time performance [47]. Actually, despite the first proposed control scheme based on PR dates back to the late sixties [48]- [49], only recently its clinical viability appears to be closer [15], especially thanks to the improvements achieved in signal processing, multichannel instrumentation and microprocessor technology. Indeed, the first prostheses control device based on PR and surface electrodes (COAPT [50]) is commercially available since January 2015 (Fig. 12). Developed in cooperation with Dr. Todd Kuiken and NECAL laboratory at the Rehab Institute of Chicago, it is undergoing clinical trials in several US rehabilitation centres, also in conjunction with TMR. Nevertheless, long is the way

for a, physiological, robust and reliable control system and the developing PR based control is still an open challenge.

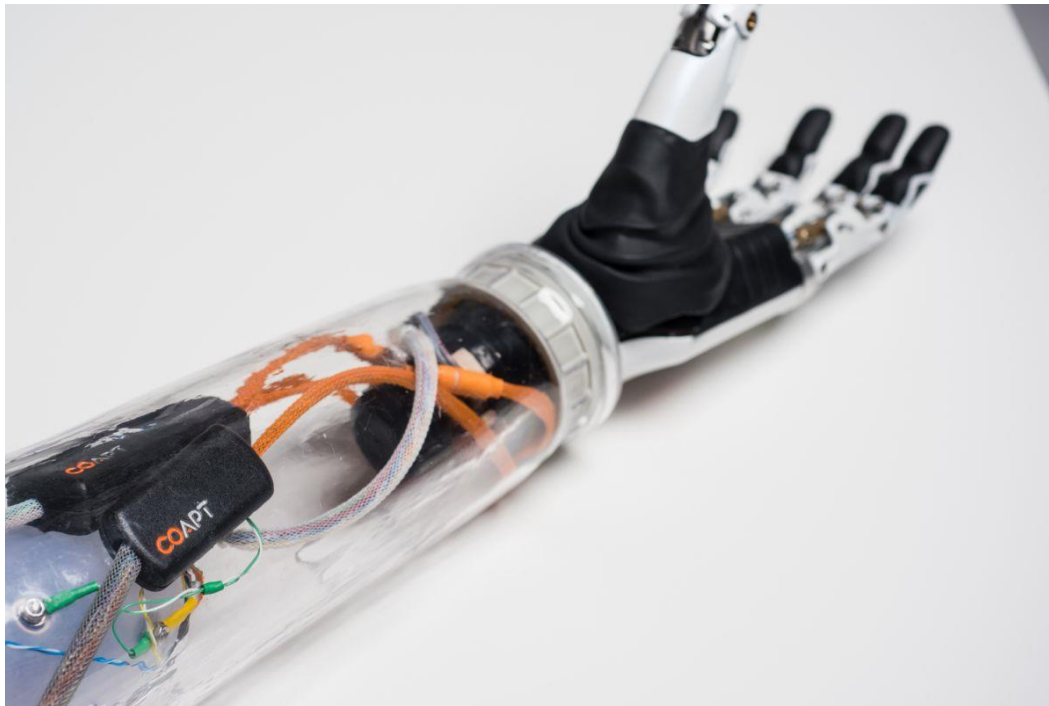


Figure 12. COAPT-Engineering Prosthetic Setup.

2.5 Conclusion

In this chapter, a review about myoelectric control for upper limb prosthesis has been presented. The main limitation of agonistic/antagonistic control consists of the limited number of controllable DoFs. However, thanks to its simplicity and robustness, results to be the most adopted control option for myoelectric prostheses in commercially available systems as well as in clinical application. In order to overcome these issues invasive solutions has been investigated (e.g. TMR) but the high cost and risk associated to the surgical re-innervating limit the applicability of these techniques to all the kinds of amputations. On the other hand, the development of PR techniques on sEMG signals aims to increase the number of controllable DoFs (and consequently the number of feasible functions) through a physiological control without a significant increase in cost and risk. Indeed, the first prostheses

control device based on PR and surface electrodes (COAPT) is commercially available since January 2015. The main issue related to control based on PR algorithms on sEMG signals is the robustness and reliability hence, the way for a physiological, robust and reliable PR based controls is still an open challenge.

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Chapter 3

Comparative Analysis of Pattern Recognition Algorithms for Hand Myoelectric Control

3.1 Introduction

For the purposes of this thesis, after a careful study of the literature concerning pattern recognition algorithms applied to prosthetics, the search for the most suitable algorithm for the application in the clinical practise on amputated patients is followed. Furthermore, in the recent work [2] it has been shown that every classifier can be potentially employed in the control of multi- DoFs if properly optimized. Hence, the choice of the most suitable classifier for the embedded use in prosthetic control has to be taken on a statistical basis. Performance, complexity and computational time usually increase together. Indeed, the choice of a classification algorithm should not be entirely relied upon performance, but rather on a trade-off between computational burden and performance, especially in embedded systems. To this purpose, a comparative analysis among NLR, MLP, SVM with Radial Basis Function (RBF) kernel, and LDA with time domain feature extraction, considered as benchmark classifier, on sEMG data from 30 people with trans-radial amputation is carried out, in terms of performance and computational burden. The use of LDA with time domain feature extraction in on-line control of prosthetic devices has been demonstrated by several studies [51]- [52]; this method is now commercially available in the US by COAPT [50].

The following contents are taken from the paper that the candidate has published on *Journal of NeuroEngineering and Rehabilitation* [20].

3.2 Material and Methods

3.2.1 *sEMG Data Acquisition Protocol*

The same acquisition protocol as in [53] was used to collect the sEMG data from the subjects participating in the experiments. Thirty people with trans-radial amputation, aged between 18 and 65, free of known muscular and/or neurological diseases, participated in the experiments. Each subject gave informed consent before performing the experiments, which were approved by local scientific and ethical committees, and were already experienced in myoelectric control of prosthetic hands. Six commercial active sEMG sensors (Ottobock 13E200=50, 27 mm x 18 mm x 9.5 mm), representing the state of art about myoelectric sensor, were equidistantly placed on a silicone adjustable bracelet (Fig. 13a) and were fastened on subject's stump (Fig. 13b). The number of the sensors to be used was chosen because it was considered the highest number, which on average is possible to place into the socket without compromising the structural integrity of the prosthesis. These sensors operate in the range 0-5 V with a bandwidth of 90-450 Hz and a common rejection ratio higher than 100 dB. The first sensor was located on the flexor carpi-radialis muscle, while the sixth sensor on the brachio-radialis muscle. These two muscles were identified by manual inspection of the stump; then, sEMG sensors were equally spaced each other on the silicone bracelet. The bracelet was located about 5cm below the subject's elbow, in line with the positioning of the electrodes, commonly used to control the myoelectric prosthesis. The data was collected using a purpose-built software on LabView platform by means of a NI DAQ USB 6002 device in order to sample the six sEMG signals at 1 kHz frequency and with 12 bits resolution. Each subject was sitting in a comfortable chair in front of a PC monitor (Fig. 13b), where one of five hand gestures was randomly shown. The subjects were instructed to reproduce steady state the displayed gesture with their phantom limb. Once the signals became stable the sampling session started and continued for 2 s obtaining for each sensor 2000 samples. The gestures to reproduce were selected among the eight canonical hand postures [7] and were "Rest" (relaxed hand), "Spherical" (hand with all fingers closed), "Tip" (hand with thumb and finger touching to pick up a small object), "Platform" (hand completely

open and stretched), and “Point” (hand with all fingers closed, except for the index finger that is pointing).

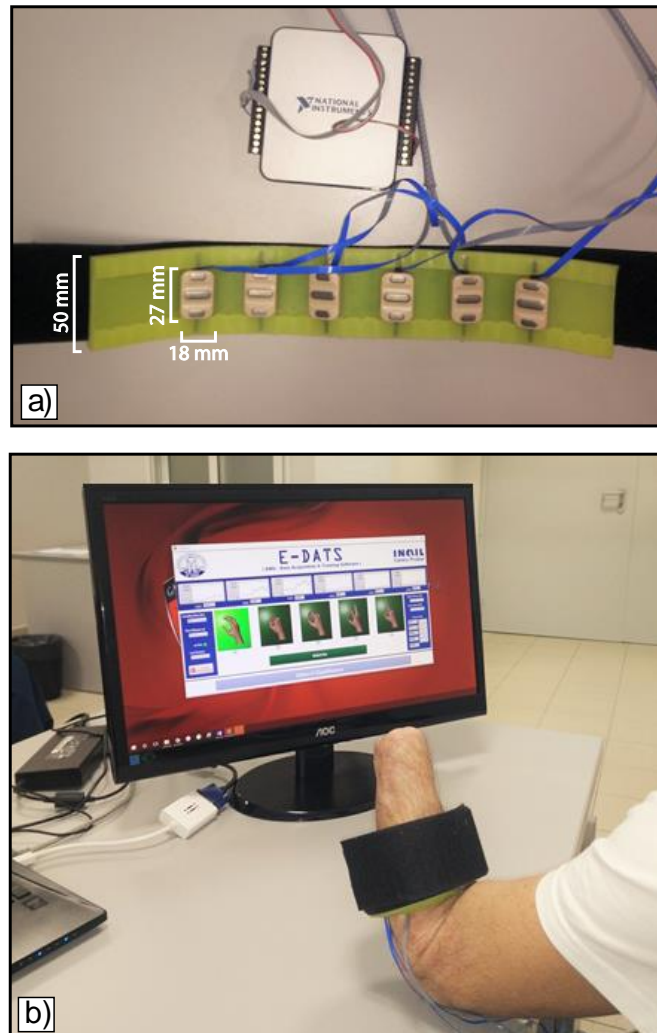


Figure 13. Experimental Setup a) sEMG bracelet and NI DAQ USB 6002; b) Subject positioning and acquisition Software.

Each acquisition started from “Rest” position; after two seconds of acquisition, the subjects were asked to return to the Rest posture. Moreover, the subjects were instructed to accomplish the task with the minimum muscular contraction and focus on the main phantom fingers related to the gesture. The selected gesture was shown as in Fig. 14. Ten repetitions of each gesture were accomplished in a single acquisition session with an inter-stimulus interval of about 5 s. Figure 14 also shows a case of the raw recording from the six sEMG sensors for all the imagined movements. The plot is related to a single acquisition session from one of the subjects who took part to the experiment.

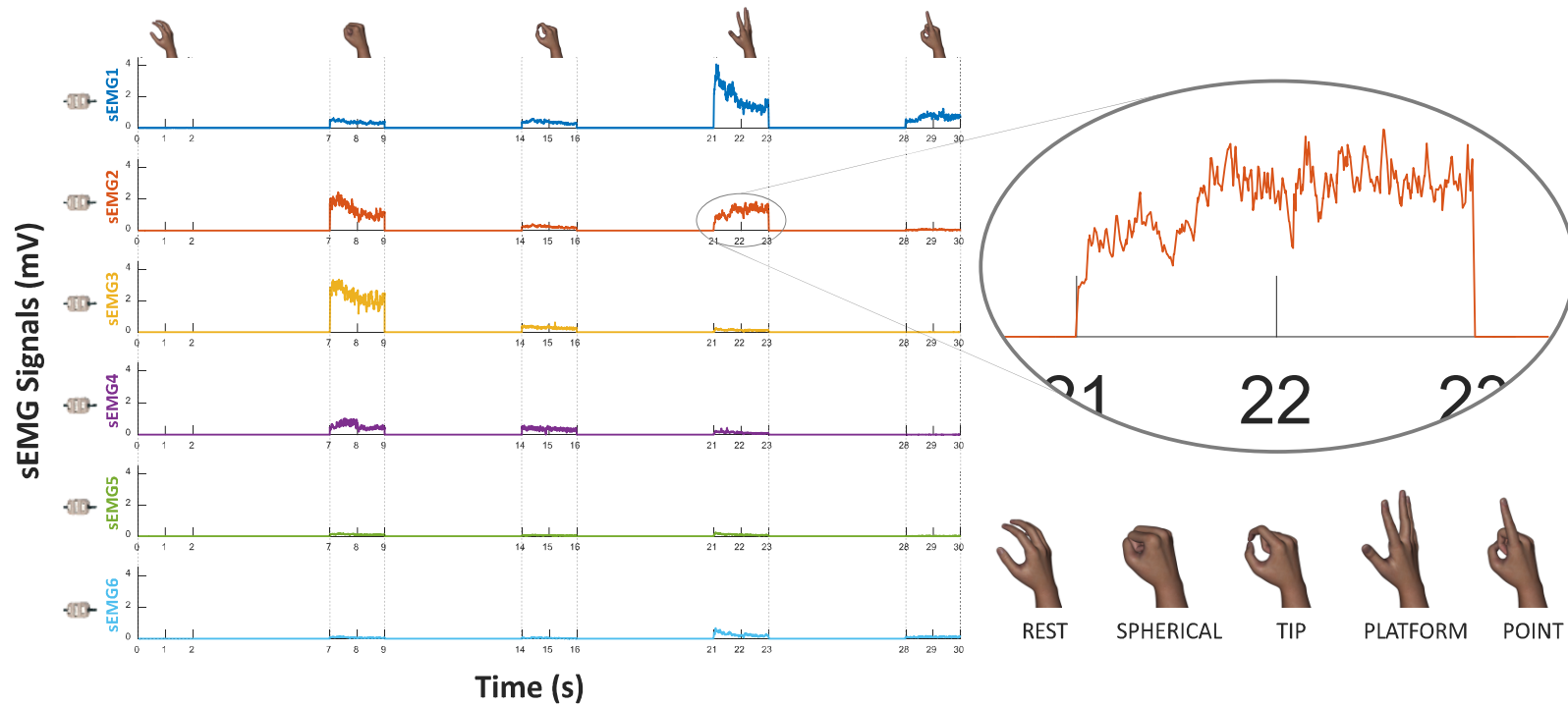


Figure 14. Graphic display of the selected gestures and of the raw recording for the six different channels at the same time for all the imagined movements of a single acquisition session from one of the subjects who took part to the experiment.

3.2.2 NLR, MLP, and SVM Classification Algorithms

In order to obtain a fast response real-time classification no feature extraction was performed from the recorded signals, hence the sEMG signal are used directly as input for the classification algorithms. The unique operation done on sEMG signals is the scaling. It consists of subtracting the mean value to each signal and dividing the result by the range. Hence, for each time step (i) we obtain a six-element vector $x^{(i)}$ of scaled sEMG signals, which is used as input for the classifiers to compare, i.e.: NLR, MLP, LDA, and SVM with RBF kernel. Supervised machine learning techniques are commonly adopted in problems where there is no functional relationship $y = f(x)$ that binds the inputs $x^{(i)}$ with the corresponding class (y). There are two different approaches to classification: the first one returns a distribution $P(y|x)$; the second one returns a result without any probability of class membership [54].

LR [55], or Perceptron, is a linear and binary supervised classification algorithm that calculates the class membership, probability using the following logistic function

$$P(1|x, \theta) = \begin{cases} g(\theta^T \cdot x) = \frac{1}{1 + e^{-(\theta^T \cdot x + \theta_0)}} \\ 1 - P(y = 0|x, \theta) \end{cases}, \quad (1)$$

where θ and θ_0 are the classification parameters vector and the bias term, respectively, and $g(\cdot)$ is the logistic, or sigmoid, function. In order to achieve a NLR the creation of additional input features (*interaction terms*) is needed. For this study, additional polynomial features were used, which were obtained as a combination product of the starting input features (e.g. $x_1; x_2; x_1 \cdot x_2; x_1^2; x_2^2; \dots$). The prediction of class labels (h_θ) for LR or NLR algorithm is then achieved by comparing the distribution $P(y|x)$ with a *decision threshold* (TH) as

$$h_\theta(x) = \begin{cases} P(1|x, \theta) \geq TH \rightarrow 1 \\ P(1|x, \theta) < TH \rightarrow 0. \end{cases} \quad (2)$$

MLP [54]- [55] is a particular case of supervised Artificial Neural Network (ANN) where each node, or neuron, of the architecture implements a logistic function. The network architecture has an input layer, one or more hidden layers (with the same number of neurons), and an output layer with one neuron for each class to be classified. The output vector of the l -th layer ($a^{(l)}$) of this particular classifier is obtained through *forward propagation* as

$$a^{(l)} = \begin{cases} x, & l = 1. \\ g\left(\Theta^{(l-1)} \cdot a^{(l-1)} + \Theta_0^{(l-1)}\right), & l = 2, 3, \dots, L. \end{cases} \quad (3)$$

Where $\Theta^{(l)}$, $\Theta_0^{(l)}$ are the classification parameters matrix and the bias vector associated with the l -th layer, respectively, and L indicates the output layer. Hence, the output of the network is a vector $Pv(y|x)$ whose elements represent the class membership probability expressed as

$$Pv\left(y|x, \Theta^{(l)}, \Theta_0^{(l)}\right) = a^{(L)}. \quad l = 1, 2, \dots, L. \quad (4)$$

Also, for MLP it is possible to achieve the prediction of class labels (h_θ) by comparing each value of the distribution vector $Pv(y|x)$ with TH and assigning to h_θ the index of the element of $Pv(y|x)$ that represents the maximum among all those resulted above the decision threshold.

SVM [54]- [55] is a linear and binary supervised classification algorithm that considers only dichotomous distinction between two classes, and assigns class label 0 or 1 to unknown data item [21] as follows

$$h_\theta(x) = \begin{cases} \left(\theta^T \cdot x + \theta_0\right) \geq +1 \rightarrow 1 \\ \left(\theta^T \cdot x + \theta_0\right) \leq -1 \rightarrow 0. \end{cases} \quad (5)$$

In order to obtain a nonlinear classifier, a *kernel function* needs to be included into the model. A kernel function is a similarity function (f), satisfying the Mercer's Theorem, that expresses the similarity between the generic input vector x and a

landmark (s), representing the two classes. Typically, a selection of all the x vectors recorded for training the SVM algorithm are set as landmarks and the j -th element of f for a RBF kernel becomes

$$f_j = \exp\left[-\frac{|x - s^{(j)}|^2}{2\gamma}\right], \quad j = 1, 2, \dots, n. \quad (6)$$

where n is the number of landmarks chosen as representative vector of classes 0 and 1, and γ is the internal RBF parameter. Then the input features vector becomes f and the class labels for a SVM with RBF kernel ¹ are assigned as

$$h_\theta(f) = \begin{cases} (\theta^T \cdot f + \theta_0) \geq +1 \rightarrow 1 \\ (\theta^T \cdot f + \theta_0) \leq -1 \rightarrow 0. \end{cases} \quad (7)$$

Classification parameters θ , θ_0 , $\Theta^{(l)}$, and $\Theta_0^{(l)}$ are obtained from the minimization of a particular cost function $J(\cdot)$ associated with each classifier.

3.2.3 NLR, MLP, SVM Classifiers and Optimization Algorithm Implementation

NLR, MLP and SVM classification algorithms were implemented in MATLAB. For NLR and MLP the code was ad-hoc developed, while for SVM the open source library libsvm3.20 [23] was used. The developed function that implements NLR allows the user to choose the maximum value of the variable D , which encodes a structure of polynomial features as reported in Table 1.

As polynomial features are intended the starting features high till the indicated degree and all the multiplications that arise from the possible permutations without repetitions of a maximum number of elements corresponding to the indicated degree.

¹ note that in the following SVM will be used to indicate SVM with RBF kernel

ENCODING THE VARIABLE D

Max D Value	Description	Example
1	Linear case (LR)	$x_1, x_2, x_3, x_4, x_5, x_6$
2	max 2 nd degree	$x_1, \dots, x_6, \dots, x_1x_2, \dots, x_5x_6, \dots, x_1^2, \dots, x_6^2$
3	max 3 rd degree	$x_1, \dots, x_6^2, \dots, x_4x_5x_6, \dots, x_6^3$
4	max 4 th degree	$x_1, \dots, x_6^3, \dots, x_3x_4x_5x_6, \dots, x_6^4$
5	max 5 th degree	$x_1, \dots, x_6^4, \dots, x_2x_3x_4x_5x_6, \dots, x_6^5$
6	max 6 th degree	$x_1, \dots, x_6^5, \dots, x_1x_2x_3x_4x_5x_6, \dots, x_6^6$
7	max 7 th degree	$x_1, \dots, x_6^6, x_1^7, x_2^7, x_3^7, x_4^7, x_5^7, x_6^7$

Table 1. Encoding the variable D.

A *cross-entropy error* cost function has been associated to the NLR algorithm and is expressed a

$$J(\theta, \theta_0) = -\frac{1}{m} \left[\sum_{i=1}^m y^{(i)} \cdot \ln g(\theta^T \cdot x^{(i)} + \theta_0) \right] + \left[\sum_{i=1}^m (1 - y^{(i)}) \cdot \ln (1 - g(\theta^T \cdot x^{(i)} + \theta_0)) \right], \quad (8)$$

where m is the number of samples used to train the algorithm and $y^{(i)}$ is the known class membership of the i -th sample. Being NLR a binary classification algorithm, a *one vs. all* approach was implemented to address the multi-class classification problem.

The developed function that implements MLP allows the user to decide the maximum number of hidden layers and the maximum number of neurons for each of them. A *mean square error* cost function has been associated to the MLP algorithm, as

$$J(\Theta, \Theta_0) = \frac{1}{m} \sum_{i=1}^m \sum_{k=1}^K \left[y_k^{(i)} - (a_k^{(L)})^{(i)} \right]^2, \quad (9)$$

where K is the number of classes to be recognized, $y_k^{(i)}$ is the known k -th element of the class membership vector of the i -th sample, and $a_k^{(L)}$ is the k -th element of the evaluated membership probability vector of the i -th sample.

As previously mentioned, the SVM classifier with RBF kernel has been developed exploiting the open source library *libsvm3.20* that is widely used for multiclass machine learning problems. More detailed information can be found in [23]-[24]-

[25]. Anyway, the cost function $J(\cdot)$ associated to the SVM algorithm can be expressed as

$$J(\theta, \theta_0) = -C \left[\sum_{i=1}^m y^{(i)} \cdot \ln g(\theta^T \cdot f + \theta_0) \right] + \\ -C \left[\sum_{i=1}^m y^{(i)} \cdot \ln(1 - g(\theta^T \cdot x^{(i)} - \theta_0)) \right] + \frac{1}{2} [\theta^T \cdot \theta + (\theta_0)^2], \quad (10)$$

The developed function allows the user to set the value regularization parameters C that appear into the cost function implemented in libsvm3.20 and the value of the internal RBF parameter γ . In this case, to address the multiclass classification problem it has been chosen to rely on a *one vs. one* method as recommended by the developers for practical usage of the library [56]- [57].

Since each of the aforementioned classifiers requires to set internal parameters, in addition to classification parameters θ , θ_0 , $\Theta^{(l)}$, and $\Theta_0^{(l)}$, it is coupled with an iterative optimization algorithm. The optimization strategy relies on a *three ways data split approach* [58]. Hence, the initial data set is divided into three subsets: “*Training Set*” (TR) containing 60% of the data, “*Cross Validation Set*” (CV) containing 20% of the data, and “*Test Set*” (TS) containing the remaining 20% of the data. These subsets are iteratively filled through a *random shuffle* until a configuration with a proportionated class number is reached. The TR is used to train the supervised classification algorithms by minimizing the specific cost function. As minimization algorithm, Resilient Backpropagation (RProp) [59] has been chosen for NLR and MLP and Limited memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) [60] for SVM. Each single classifier is iteratively trained with all the possible configurations of its internal parameters, varying each of these within an appropriate range of values. The CV is then used to evaluate performance of each configuration (i.e. model), in order to avoid overfitting and find out the best model.

The F1Score [61] was used in this study to assess performance, in lieu of accuracy, being more robust also for classes that do not have a perfect symmetrical cardinality. Considering this simple confusion matrix

	(y = 1)	(y = 0)
($h_o = 1$)	nP	nFP
($h_o = 0$)	nFN	nN

(11)

where nP is the number of true positive, nN the number of true negative, nFP the number of false positive and nFN the number of false negative, F1Score can be evaluated as

$$\begin{cases} PR = \frac{nP}{(nP + nFN)} \\ RE = \frac{nP}{nP + nFP} \\ F1Score = 2 \cdot \frac{PR \cdot RE}{PR + RE} \cdot 100, \end{cases} \quad (12)$$

where PR is called *Precision* and RE is called *Recall*.

After determining the optimal classifier model, the TS is used to achieve an estimation of the performance that the classifier is expected to show when new features are provided as input.

3.2.4 NLR, MLP, SVM Downsampling and Creation of Generalization Set

For each subject involved in the experiment, the data sampled at 1 kHz were organized in a matrix; each column of the matrix was coupled with an EMG sensor. Hence, the choice of avoiding features extraction based on time windowing of sEMG generated $10^5 \times 6$ data (large-scale datasets) and, consequently, a very long time (more than 4 hours per subject) is required to complete training and optimization for each classification algorithm. Therefore, downsampling has been applied to speed up the whole process. The discarded data were used to compose a new set of data called “*Generalization Set*” (GS) which has been used as second test set in order to obtain an estimation of the generalization ability of each classification algorithm. In particular, for a downsampling step equals to 10 (one in ten), the GS will contain 90% of the data, the TR 6%, the CV 2%, and the TS the remaining 2% of the data. In other terms, the results evaluated on TS represent an

estimation of the classification ability when the signal to classify is sampled at the same frequency of the training data (a downsampling step equals to 10 produce a 100 Hz dataset) while results evaluated on GS represents an estimation of the classification ability when classifying a signal sampled up to 1 kHz.

3.2.5 LDA Classifier

LDA is a linear and binary supervised classification algorithm that considers a dichotomous distinction between two classes, and assigns class label 1 or 2 to unknown data item relying on the following decision function

$$h_{\beta}(x) = \begin{cases} (\beta^T \cdot x + \beta_0) \geq 0 \rightarrow 1 \\ (\beta^T \cdot x + \beta_0) < 0 \rightarrow 2 \end{cases}, \quad (13)$$

where β and β_0 are the classification parameters vector and the bias term, respectively. Classification parameters can be evaluated as

$$\begin{cases} \beta = \Sigma^{-1} \cdot (\mu_1 - \mu_2) \\ \beta_0 = -\beta^T \cdot \left(\frac{\mu_1 + \mu_2}{2} \right) + \ln \left(\frac{\Pi_1}{\Pi_2} \right), \end{cases} \quad (14)$$

where Σ is the pooled covariance matrix, μ_1 , μ_2 and Π_1 , Π_2 are the mean vectors and the prior probabilities of class 1 and class 2, respectively. Since this classifier does not require setting internal parameters, training and test rely on a *two ways data split approach* [58]. Hence, the initial dataset is divided into training set and test set. The training set contains 70% of the data (TR_{70%}), and the test set contains the remaining 30% of the data (TS_{30%}). The subsets are iteratively filled through a *random shuffle* until a configuration with proportionated class number is reached. The TR_{70%} is used to train the classifier evaluating classification parameters β and β_0 ; on the other hand, the TS_{30%} is used to estimate the classifier performance when new features are provided as input. Being LDA a binary classification algorithm, a *one vs. all* approach was implemented to address the multi-class classification problem. The class label (c) is predicted as

$$h_{\beta}(x) = \max_c \left({}_c\beta^T \cdot x + {}_c\beta_0 \right) \quad \text{and} \quad \begin{cases} {}_c\beta = \Sigma^{-1} \cdot \mu_c \\ {}_c\beta_0 = -{}_c\beta^T \cdot \left(\frac{\mu_c}{2} \right) + \ln(\Pi_c), \end{cases} \quad (15)$$

where ${}_c\beta$ and ${}_c\beta_0$ are the classification parameters vector and the bias term of c class, respectively. For building our LDA benchmark classifier five commonly used time domain features were considered²: Mean Absolute Value (MAV), Root Mean Square (RMS), Slope Sign Change (SSC), Waveform Length (WL) and Variance (σ^2). They were extracted in windows of 250 ms with an overlap of 200 ms [52]. Since the training of the LDA classifier is performed by means of Eq. (9) and the feature extraction avoids the generation of large-scale-dataset, a short time is required to complete the training of the classifier and there is no need to perform down sampling. The classification algorithm was implemented in MATLAB with an ad hoc developed software code.

3.2.6 Data Analysis

The study was divided into three parts: the first one investigated the optimal range of D (initial guess 1-7) for NLR, and the range of maximum number of layers (initial guess 1-10) and neurons (initial guess 1-30) for MLP, while the second part is focused on the comparison among the NLR, MLP and SVM classification algorithms. The third part is focused on the comparison with our ground truth, the LDA classifier. The first part can be seen as a preliminary investigation in order to reduce the evaluation time of the comparison among the three classifiers. A downsampling step equal to 10 (and corresponding to a 100Hz sampling frequency) has been applied to data collected from 30 people with trans-radial amputation. Performance of each algorithm has been measured by means of the F1Score (12) value and a statistical analysis has been based on the Wilcoxon Signed-Rank test, which has been shown to be appropriate for comparing different classifiers in common datasets [2]- [62]. Statistical significance was considered at $p < 0.05$. The maximum value of D, of the number of layers, and of the number of neurons have

² note that in the following LDA will be used to indicate LDA with 5-time domain features

been obtained by means of a sequential statistical analysis, starting from the simplest case and then sequentially comparing all the others until a high significant difference of performance is found. This is taken as the new benchmark for all the subsequent comparisons. The process ends when it is found the last case in which the differences are not statistically significant compared to all subsequent cases.

The second part, the core of our work, resorted to the results obtained in the first part to compare NLR, MLP, and SVM considering both performance and run-time computational burden on EMG data collected from 30 people with trans-radial amputation. As regards the SVM, the range of variation of the regularization parameter C belongs to $0-10^4$, with variable steps starting from 0.01 and doubling each time, while γ belongs to $0-50$ (with a pitch equals to 0.1); both have been empirically determined in previous tests. The computational burden was evaluated through the number of parameters ($n\theta$), expressing the cardinality of classification vector θ (1) (7) or matrices Θ (4) that identify the particular classification algorithm. In detail, the number of matrix elements created by the libsvm training function, which are necessary to run the evaluated SVM model, were used for evaluating the cardinality of SVM parameters. Particularly they were: *rho*, *sv_coef*, and *SVs* [57]. The values of sample rate were: 5 Hz, 10 Hz, 20 Hz, 40 Hz, and 100 Hz (corresponding to 200, 100, 50, 25 and 10 downsampling step). Again, the statistical analysis has been performed through a Wilcoxon Signed-Rank test with significance threshold set to 0.05. Lastly a combined index, called EOF (Embedding Optimization Factor), that takes into account both performance and computational burden has been calculated. It is defined as

$$\left\{ \begin{array}{l} \text{if } (N\Theta > n\theta) \rightarrow P = \frac{(N\Theta - n\theta)}{N\Theta} \cdot 100 \\ \text{if } (N\Theta \leq n\theta) \rightarrow P = 0 \\ EOF = \frac{2 \cdot (F1Score \cdot P)}{(F1Score + P)}, \end{array} \right. \quad (16)$$

where $N\Theta$ is the maximum acceptable number of parameters. This index plays a paramount role in the implementation of these algorithms in embedded systems, where memory storage and program memory are limited. To this purpose, as

representative example, $N\Theta$ has been chosen as equal to the maximum number of parameters storable into a 256 KB memory, which is typically used for high performance embedded microcontrollers applied to prosthetic hands (e.g. Touch Bionics I-Limb, Ultra and Robo-Limb). As each parameter is coded as a float which 4 memory bytes are needed to just store one of them, hence, for our example, the maximum number of storable parameters is $64 \cdot 10^3$ classification parameters. This is an application example of how that index and $N\Theta$ can be evaluated, but the same method can be applied taking into account different size of memory and/or other constraints, such as the available RAM memory or the evaluation time for a single classification (which is related to the microcontroller clock frequency).

In the third part a comparative analysis among the three non-linear classifiers and the LDA was carried out. Since LDA was trained and tested with data sampled at 1 kHz (without downsampling), NLR, MLP and SVM models with the highest EOF values on GS were taken for the comparison. Again, the analysis was performed taking into account classification performance, computational burden and EOF index. The statistical analysis was performed through a Wilcoxon Signed-Rank test with significance threshold set to 0.05.

3.3 Results

The results are presented in boxplots where the central line represents the median value; the edges of the box are the 25th and the 75th percentiles; the whiskers give the range of the data without outliers; solid markers represent the mean value.

3.3.1 Max Degree of Polynomial Features for NLR

Figure 15 shows the values of F1Score of TS and GS over the max degree of polynomial features (indicated with D) applied as input to NLR. Table 2 summarizes the F1Scores averaged over 30 people with trans-radial amputation and the corresponding obtained standard deviation (s).

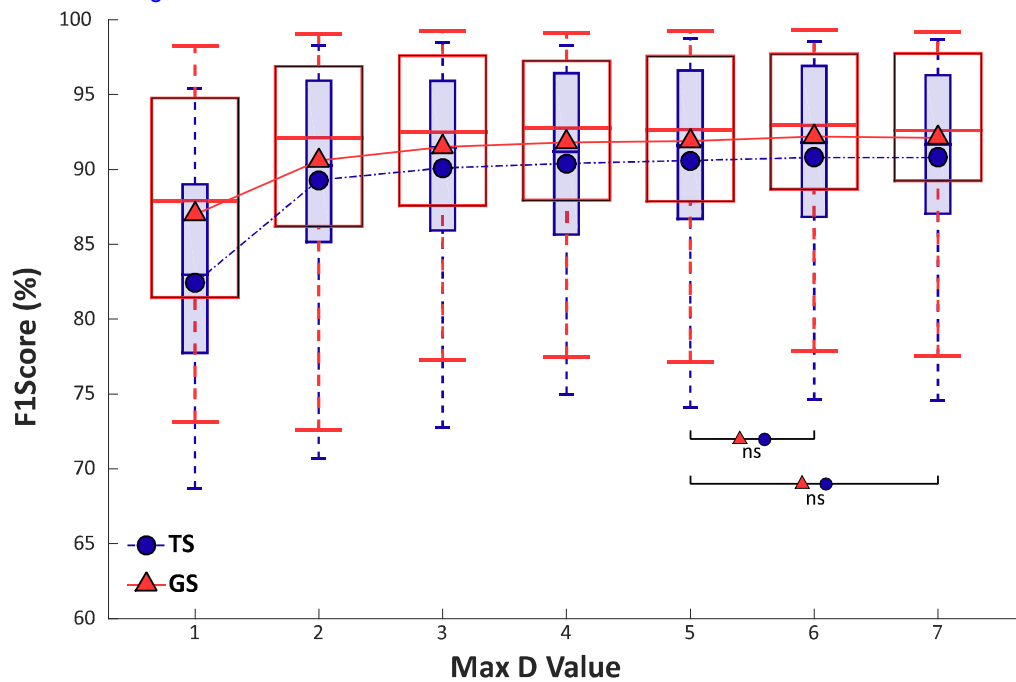


Figure 15. F1Score of Test Set (smaller boxes) and Generalization Set (bigger boxes) of 5 classes over the maximum value of variable D calculated from 30 people with trans-radial amputation. The figure also shows the trend of the mean value for both Sets. Statistical non-significance over value 5 is shown by “ns”.

In both cases, the maximum is reached by setting 7 as maximum D value, but the Wilcoxon Signed-Rank test applied to the F1Score values points out no statistically significant difference for polynomial features over the value 5 for both GS and TS. The result seems to indicate that, for people with trans-radial amputation, the system performance saturates setting the maximum D value of the polynomial features over value 5 as showed in Fig. 15 by the trend lines of the mean values.

CLASSIFICATION PERFORMANCE FOR NLR

Max D value	F1Score (%)	
	TS	GS
1	82.4 (8.4 s)	87.0 (9.4s)
2	89.3 (7.1s)	90.6 (7.1s)
3	90.1 (6.9s)	91.5 (6.4s)
4	90.4 (6.7s)	91.8 (6.2s)
5	90.6 (6.6s)	91.9 (6.2s)
6	90.8 (6.4s)	92.2 (5.9s)
7	90.8 (6.6s)	92.1 (6.2s)

Table 2. Mean values and standard deviation of F1Score of Test Set and Generalization Set of 5 classes over the maximum value of variable D calculated from 30 people with trans-radial amputation. The highest values per Set are highlighted in bold. See Fig. 15 for a graphic display and statistical significance.

3.3.2 Max Number of Hidden Layers for MLP

Figure 16 shows the values of F1Score of TS and GS over the max number of hidden layers. Each hidden layer has maximum 30 neurons for MLP. Table 3 summarizes the F1scores averaged over 30 people with trans-radial amputation and the corresponding standard deviation (s).

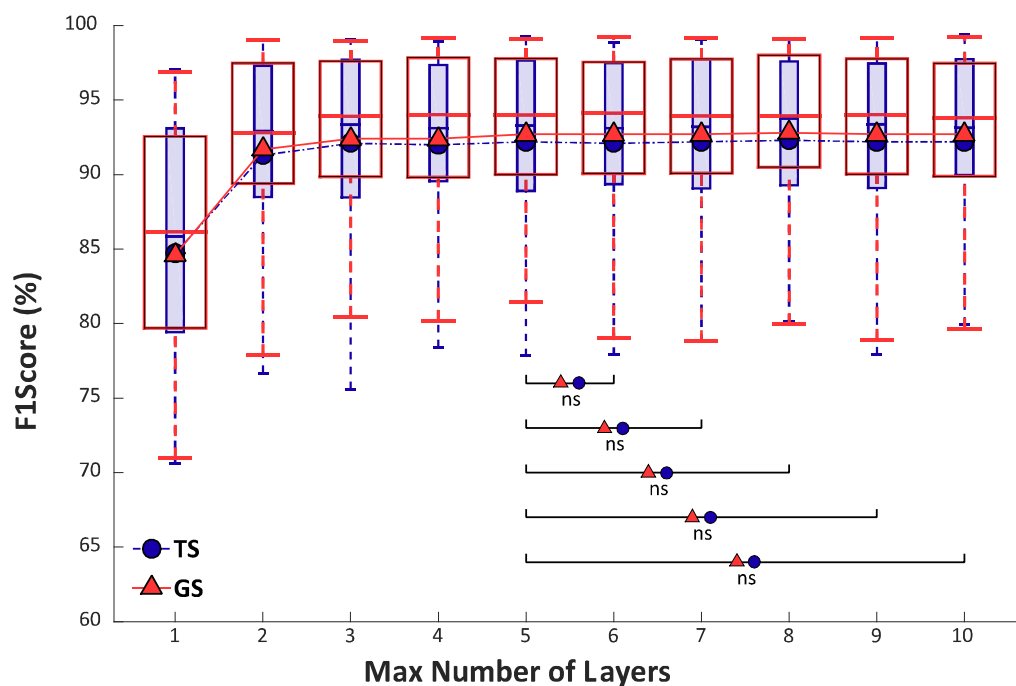


Figure 16. F1Score of Test Set (smaller boxes) and Generalization Set (bigger boxes) of 5 classes over the maximum number of layers having fixed at 30 the maximum number of neurons for each hidden layer calculated from 30 people with trans-radial amputation. The figure also shows the trend of the mean value for both Sets. Statistical non-significance over value 5 is shown by “ns”.

In both cases, the best performance is obtained for a maximum number of layers equal to 8, but the Wilcoxon Signed-Rank test applied to the values of achieved F1Score values points out no statistically significant difference over 5 hidden layers for both GS and TS. This probably means that for people with trans-radial amputation the system performance saturates for a maximum number of hidden layers over the value 5.

Max Number of Layers	F1Score (%)	
	TS	GS
1	84.7 (10.1s)	84.6 (9.9s)
2	91.3 (6.8s)	91.7 (6.5s)
3	92.1 (6.4s)	92.4 (6.1s)
4	92.0 (6.1s)	92.4 (6.1s)
5	92.2 (6.2s)	92.7 (5.8s)
6	92.1 (6.0s)	92.7 (5.7s)
7	92. (5.9s)	92.7 (5.8s)
8	92.3 (5.7s)	92.8 (5.6s)
9	92.2 (6.0s)	92.7 (5.8s)
10	92.2 (5.9s)	92.7 (5.6s)

Table 3. Mean values and standard deviation of F1Score of Test Set and Generalization Set of 5 classes over the maximum number of layers having fixed at 30 the maximum number of neurons for each hidden layer calculated from 30 people with trans-radial amputation. The highest values per Set are highlighted in bold. See Fig. 16 for a graphic display and statistical significance.

3.3.3 Max Number of Neurons for MLP

Figure 17 summarizes the values of F1Score of TS and GS with respect to the max number of neurons for a MLP with maximum 5 hidden layers varying by 5 the number of neurons until the value 23, for compactness.

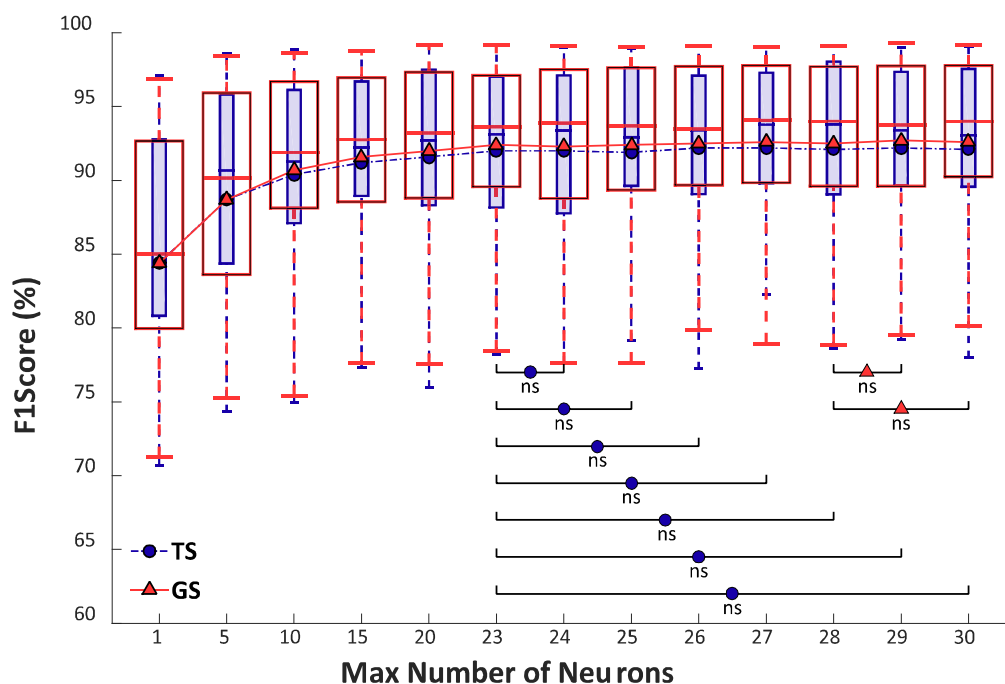


Figure 17. F1Score of Test Set (smaller boxes) and Generalization Set (bigger boxes) of 5 classes over the maximum number of neurons for each layer. The maximum number of hidden layers calculated from 30 people with trans-radial amputation has been fixed at 5. The figure also shows the trend of the mean value for both Sets. Statistical non-significances over value 23 for and overvalue 28 for GS are shown by "ns".

Table 4 shows all the F1Scores averaged over 30 people with trans-radial amputation and the corresponding standard deviation (s). The Wilcoxon Signed-Rank test applied to the achieved values of F1Score points out no highly statistically significant difference over 23 for TS and over 28 for GS. This probably means that for people with trans-radial amputation the system performance saturates for a maximum number of neurons between 23 and 28 depending on the frequency of the signals to classify.

CLASSIFICATION PERFORMANCE VARYING THE MAXIMUM NUMBER OF NEURONS FOR A MAXIMUM 3 HIDDEN LAYERS MLP

Max Number of Neurons	F1Score (%)		Max Number of Neurons	F1Score (%)	
	TS	GS		TS	GS
1	84.4 (9.9s)	84.4 (9.9s)	16	91.3 (6.6s)	91.6 (6.5s)
2	84.0 (10.7s)	84.1 (10.7s)	17	91.4 (6.4s)	91.9 (6.2s)
3	86.5 (9.9s)	86.6 (9.8s)	18	91.5 (6.4s)	92.0 (6.0s)
4	88.1 (9.0s)	88.3 (8.9s)	19	91.6 (6.3s)	92.0 (6.0s)
5	88.7 (9.0s)	88.7 (8.8s)	20	91.6 (6.6s)	92.0 (6.2s)
6	89.2 (8.1s)	89.4 (8.1s)	21	91.8 (6.5s)	92.2 (6.2s)
7	89.7 (7.0s)	90.0 (7.5s)	22	91.6 (6.4s)	92.1 (6.1s)
8	89.9 (7.5)	90.2 (7.2s)	23	92.0 (5.9s)	92.4 (5.9s)
9	90.1 (7.4s)	90.4 (7.3s)	24	92.0 (6.1s)	92.3 (6.1s)
10	90.4 (7.0s)	90.7 (7.1s)	25	91.9 (6.3s)	92.4 (6.1s)
11	90.8 (6.9s)	91.1 (6.7s)	26	92.2 (6.2s)	92.5 (6.0s)
12	90.8 (7.2s)	91.0 (7.0s)	27	92.2 (6.1s)	92.6 (5.8s)
13	90.8 (6.8s)	91.4 (6.5s)	28	92.1 (6.0s)	92.5 (6.0s)
14	91.1 (6.7s)	91.4 (6.6s)	29	92.2 (6.0s)	92.7 (5.7s)
15	91.2 (6.5s)	91.6 (6.2s)	30	92.1 (5.9s)	92.6 (5.9s)

Table 4. Mean values and standard deviation of F1Score of Test Set and Generalization Set of 5 classes over the maximum number of neuron having fixed at 5 the maximum number of layers calculated from 30 people with trans-radial amputation. The highest values per Set are highlighted in bold. See Fig. 17 for a graphic display and statistical significance.

3.3.4 NLR, MLP, SVM Comparison Based on TR Sampling Rate

Figure 18 shows the values of F1Score of TS and GS, obtained training the classifiers on TR sampled at increasing sampling rate (or at decreasing downsampling step) for NLR, MLP, and SVM. As mentioned in Sect. II, NLR and MLP has been optimized by using the results previously obtained by limiting to 5 the maximum D value, for NLR and to 5 and 28 the maximum number of layers and neurons, respectively, for MLP.

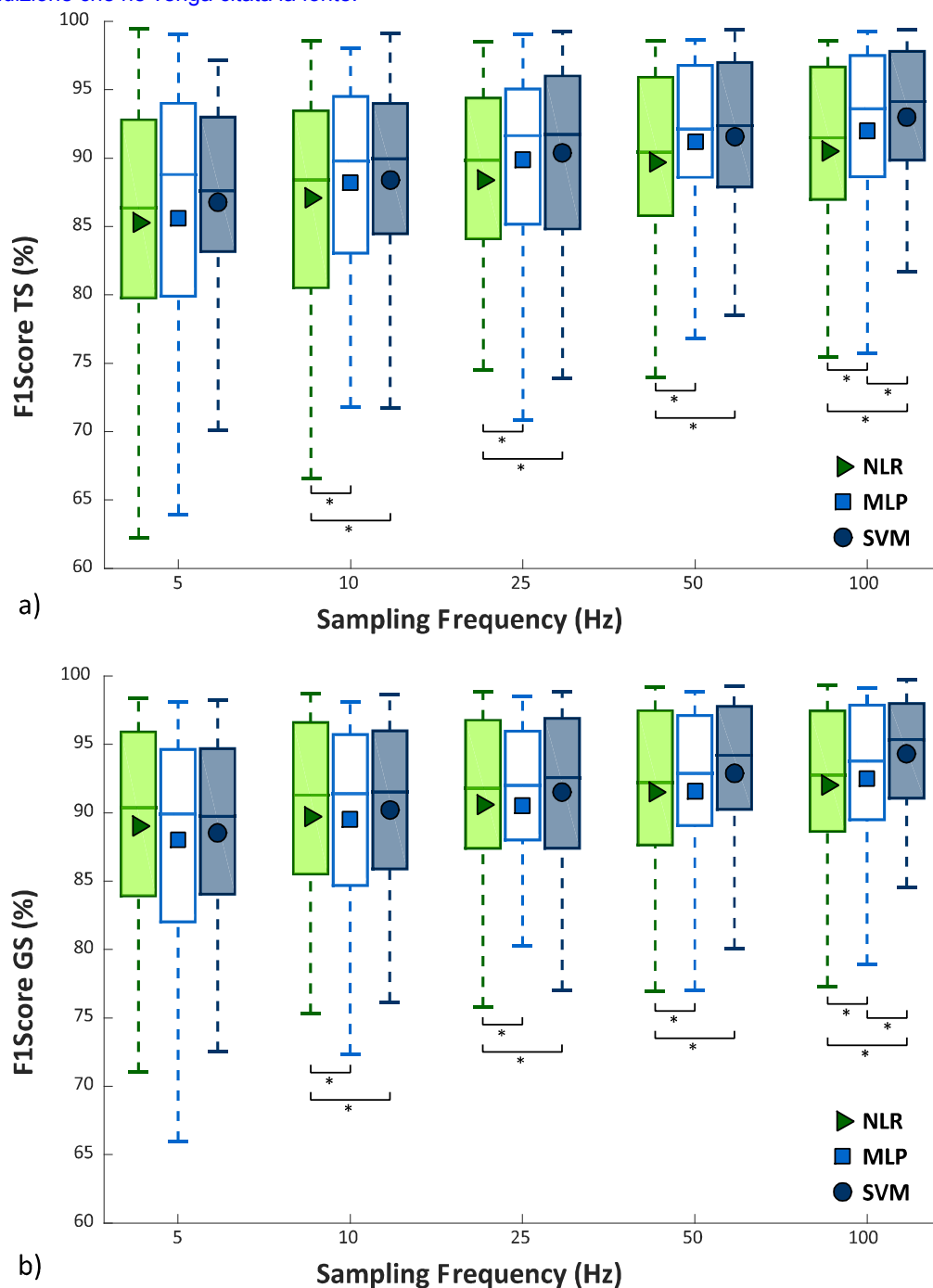


Figure 18 F1Score values from 30 people with trans-radial amputation increasing the sampling frequency of the dataset used to train and cross validate the NLR, MLP, and SVM algorithms and 5 classes. Statistical significance is shown by “*”. a) F1Score values for Test Set; b) F1Score values for Generalization Set.

Table 5 shows the F1Scores averaged over 30 people with trans-radial amputation and the corresponding standard deviation (s) for the three algorithms and both datasets TS and GS. Afterwards, performance of NLR, MLP, and SVM were compared, at different sampling frequencies of the dataset used to train the

algorithms, through a Wilcoxon Signed-Rank test. For both TS and GS, the analysis reports no statistically difference between the three classifiers when training the algorithms with a 5 Hz sampled dataset, and that NLR achieved significant lower value than MLP and SVM with the others sampling frequencies. Conversely, MLP achieved statistically significant lower performance than SVM only using a 100Hz frequency.

F1SCORE FOR EACH ALGORITHM OVER THE INCREASING THE SAMPLING RATE OF THE TRAINING SET

F1Score (%) TS			
Sampling Frequency	NLR	MLP	SVM
5 Hz	85.3 (9.2s)	85.6 (11.0s)	86.8 (7.5s)
10 Hz	87.1 (8.1s)	88.2 (8.0s)	88.4 (8.1s)
20 Hz	88.4 (7.8s)	89.9 (7.2s)	90.4 (6.8s)
40 Hz	89.7 (6.8s)	91.2 (6.5s)	91.6 (6.2s)
100 Hz	90.5 (6.6s)	92.0 (6.2s)	93.0 (5.4s)
F1Score (%) GS			
5 Hz	89.0 (8.0s)	88.0 (8.2s)	88.5 (7.4s)
10 Hz	89.7 (7.7s)	89.5 (7.3s)	90.2 (6.7s)
20 Hz	90.6 (7.1s)	90.5 (6.8s)	91.5 (6.1s)
40 Hz	91.5 (6.4s)	91.6 (6.3s)	92.9 (5.5s)
100 Hz	92.0 (6.1s)	92.5 (5.9s)	94.3 (4.4s)

Table 5. Mean values and standard deviation of F1Score of Test Set and Generalization Set of 5 classes from 30 people with trans-radial amputation varying the frequency of the dataset used to train and cross validate the NLR, MLP and SVM classifier. The highest values per classifier are highlighted in bold. See Fig. 18 for a graphic display and statistical significance.

3.3.5 NLR, MLP, SVM Comparison Based on Computational Burden

Figure 19 shows the number of classification parameters ($n\theta$), obtained training the classifiers on datasets sampled at increasing sampling rate (or at decreasing downsampling step) for NLR, MLP, and SVM. Variable $n\theta$ is regarded as an index quantifying the algorithm computational burden. Again NLR and MLP has been optimized thanks to the previously obtained results. As the model of the classifier adopted for TS and GS is the same, also the complexity in the two cases is the same. Table 6 shows $n\theta$ averaged over 30 people with trans-radial amputation and the corresponding standard deviation (s) for the three algorithms.

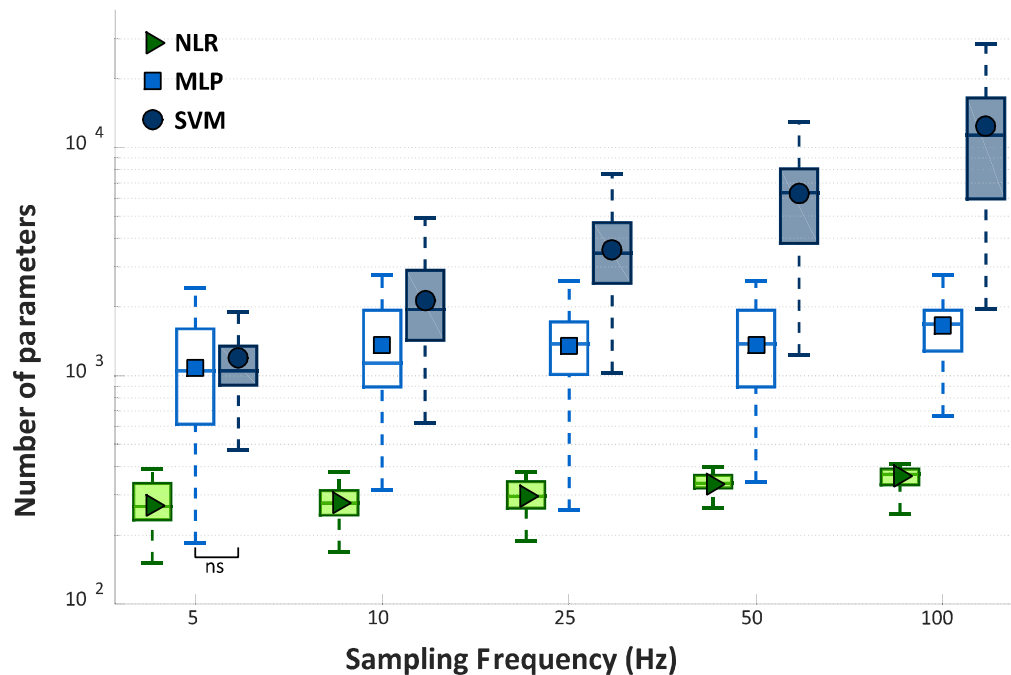


Figure 19. Number of classification parameters from 30 people with trans-radial amputation increasing the sampling frequency of the dataset used to train and cross validate the NLR, MLP, and SVM algorithms and 5 classes. The y axis in logarithmic scale. Statistical non-significance is shown by “ns”.

By comparing the algorithms at different sampling rates for the dataset used to train the three algorithms, it can be observed that SVM is always characterized by the highest computational cost, while NLR by the lowest one. While NLR and MLP remain statistically different they retained values of $n\theta$ that always belong to the same order of magnitude (10^2 for NLR and 10^3 for MLP), SVM initially scores values statistically equals to MLP (5 Hz) and then diverged with respect to the sampling rate.

COMPUTATIONAL BURDEN INCREASING THE SAMPLING RATE

Sampling Frequency	Number of Parameters (θ)		
	NLR	MLP	SVM
5 Hz	269 (65s)	1082 (662s)	1196 (515s)
10 Hz	279 (53s)	1361 (648s)	2127 (1086s)
20 Hz	296 (52s)	1356 (629s)	3576 (1699s)
40 Hz	334 (40s)	1362 (627s)	6304 (3378s)
100 Hz	362 (41s)	1654 (605s)	12393 (7760s)

Table 6. Mean values and standard deviation of classification parameters from 30 people with trans-radial amputation varying the frequency of the dataset used to train and cross validate the NLR, MLP and SVM classifier and 5 classes. The highest values per classifier are highlighted in bold. See Fig. 19 for a graphic display and statistical significance.

This difference in behavior of the SVM classifier is due to its unique architecture that generates a number of landmarks (6), which are strictly related to the number of the classification parameters, depending on the numerosity of the dataset used to train the algorithm. Therefore, the higher the sampling frequency the more numerous the TR will be and, consequently, a high number of landmarks to represent the data is needed. All the others comparisons proved to be statistically different among them.

3.3.6 NLR, MLP, SVM Comparison Based on EOF

As previously mentioned in this section it was reported a result of an applicative example comparing NLR, MLP and SVM classifiers using EOF as comparison index. The only constraint adopted in this analysis is the burden on a 256 KB memory that the classification parameters to be stored produce. Figure 20 shows values of EOF for TS and GS, obtained training the classifiers on datasets sampled at increasing sampling rate (or at decreasing downsampling step) for NLR, MLP, and SVM. Again, NLR and MLP were optimized using the results previously obtained.

CLASSIFICATION EOF PERFORMANCE INCREASING THE SAMPLING RATE			
Sampling Frequency	EOF (%) TS		
	NLR	MLP	SVM
5 Hz	91.6 (5.5s)	91.1 (6.7s)	92.0 (4.6s)
10 Hz	92.7 (4.8s)	92.6 (4.8s)	92.2 (5.2s)
20 Hz	93.4 (4.6s)	93.6 (4.2s)	92.3 (4.8s)
40 Hz	94.2 (3.8s)	94.3 (3.7s)	90.8 (5.6s)
100 Hz	94.6 (3.7s)	94.5 (3.4s)	86.2 (9.4s)
EOF (%) G s			
5 Hz	93.8 (4.6s)	92.7 (4.8s)	92.9 (4.5s)
10 Hz	94.2 (4.5s)	93.4 (4.2s)	93.3 (4.4s)
20 Hz	94.7 (4.1s)	93.9 (4.0s)	92.9 (4.4s)
40 Hz	95.2 (3.5s)	94.5 (3.5s)	91.5 (5.2s)
100 Hz	95.5 (3.4s)	94.8 (3.2s)	86.7 (9.1s)

Table 7. Mean values and standard deviation of EOF of Test Set and Generalization Set of 5 classes from 30 people with trans-radial amputation varying the frequency of the dataset used to train and cross validate the NLR, MLP and SVM classifier. The highest values per classifier are highlighted in bold. See Fig. 20 for a graphic display and statistical significance.

Table 7 shows the EOF averaged over 30 people with trans-radial amputation and the corresponding standard deviation (s) for the three algorithms. Hence, a comparative analysis among NLR, MLP, and SVM was carried out (first for TS, then for GS).

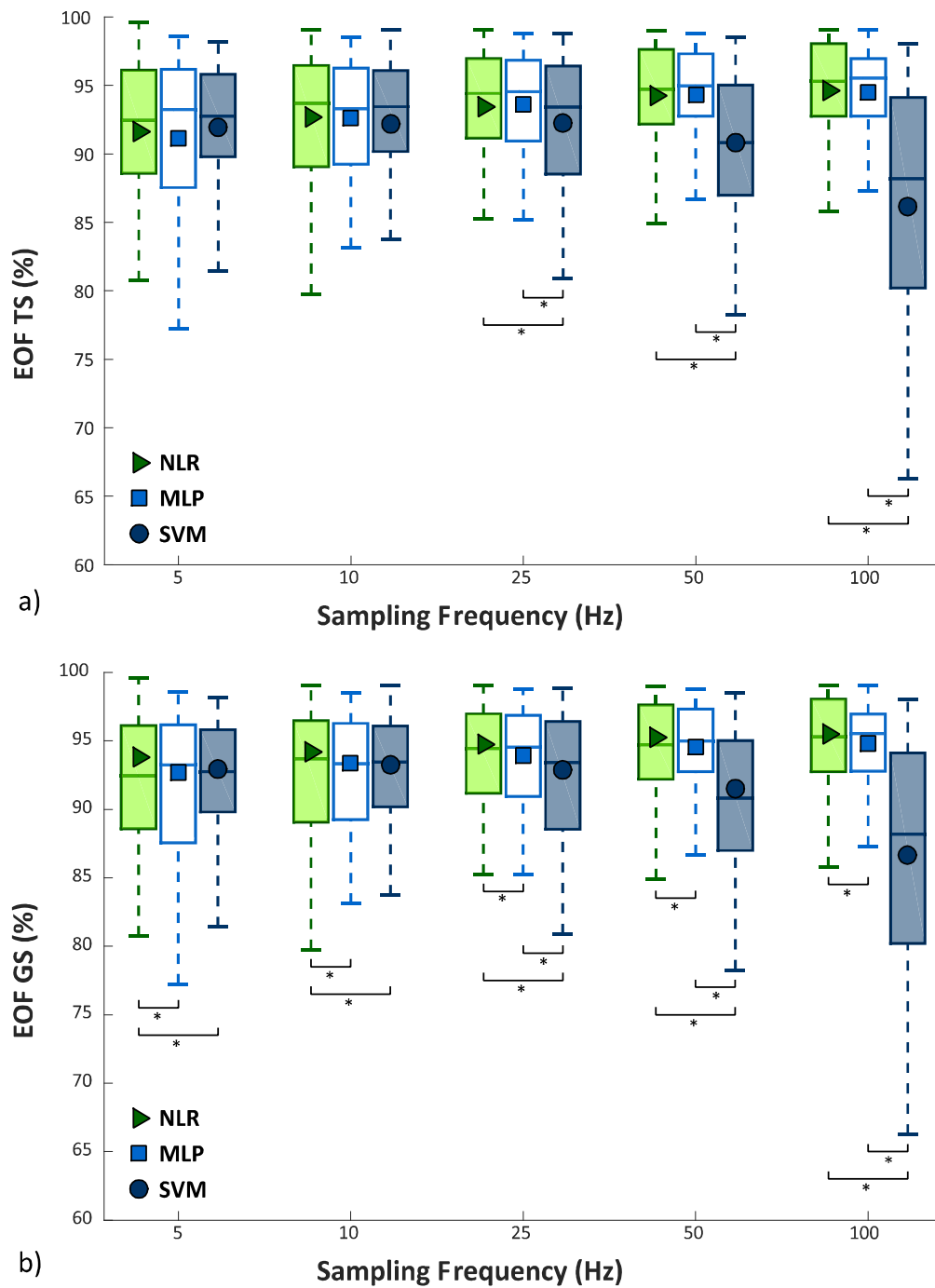


Figure 20. EOF values from 30 people with trans-radial amputation increasing the sampling frequency of the dataset with 5 classes used to train and cross validate the NLR, MLP, and SVM algorithms. Statistical significance is shown by “ * ”. a) EOF values for Test Set; b) EOF values for Generalization Set.

Except that for TS at 5Hz sampling frequency (where SVM has obtained the maximum value of EOF) among the three classifiers NLR attained the maximum EOF value for both TS and GS and perhaps, the result means that for people with

trans-radial amputation NLR and MLP classifiers represent the best compromise between classification performance and computational burden. The result is even more valuable considering the trend of the value of EOF increasing the sampling rate. In fact, for the NLR and the MLP classifier the value of this index tends to slightly increase, while for the SVM classifier it decreases more and more.

3.3.7 NLR, MLP, SVM and LDA Comparison

In this section the results of the comparative analysis of LDA with NLR, MLP, and SVM classifiers are reported. For comparative purposes, NLR, MLP, and SVM models that obtained the highest EOF values on GS were used. The LDA classifier was considered as ground truth, in terms of performance, number of parameters and EOF index. Figure 21 shows the values of F1Score of GS for NLR, and MLP on TR sampled at 100 Hz and SVM, on TR sampled at 25 Hz, and of TS_{30%} for LDA on TR_{70%} sampled at 1 kHz.

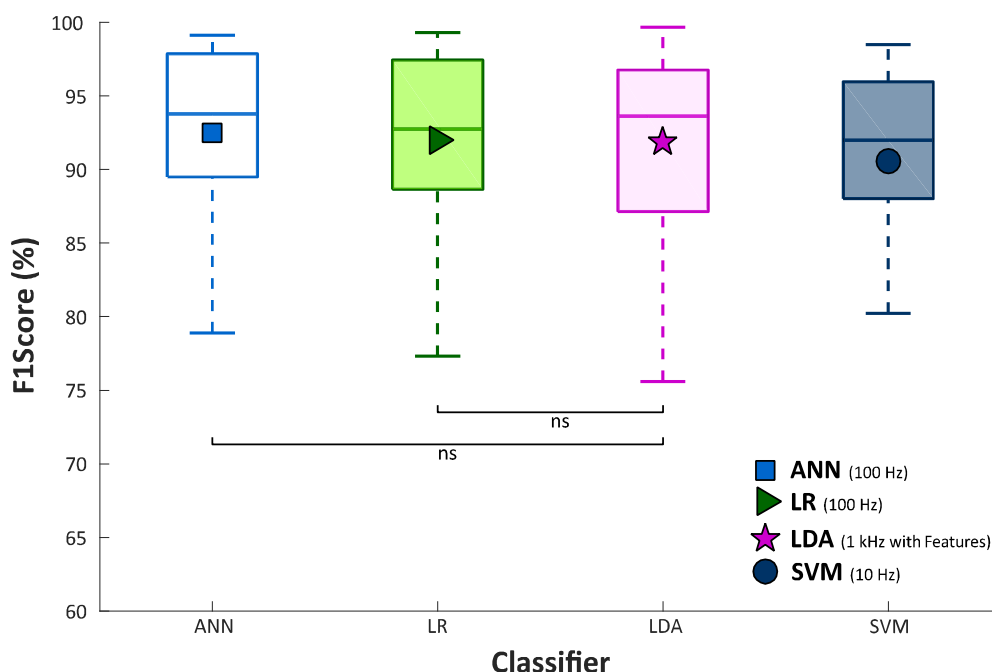


Figure 21. F1Score values from 30 people with trans-radial amputation for MLP, NLR, SVM, tested on GS, and LDA with 5 time domain features, on a 5 classes dataset. NLR and MLP were trained using data sampled at 100Hz, while SVM using data sampled at 10Hz. Statistical non-significance is shown by " ns ".

By exploiting the previously obtained optimization results, D value was limited to 5 for NLR, while the maximum number of layers and neurons was limited to 5 and

28 for MLP. Table 8 shows the numeric values of F1Scores averaged over 30 subjects with trans-radial amputation and the corresponding standard deviation (s) for all the four algorithms. A Wilcoxon Signed-Rank test was adopted for the statistical analysis of comparison between NLR, MLP, and SVM and LDA.. The analysis reports no statistically significant difference between LDA and both NLR and MLP classifiers, while SVM achieved significantly lower value than the others. Figure 22 displays the number of classification parameters ($n\theta$ and $n\eta$). Table 8 shows the number of classification parameters averaged over 30 subjects with trans-radial amputation and the corresponding standard deviation (σ) for the four algorithms. The analysis showed that LDA obtained the minimum number of parameters, and no statistically significant difference was observed only between MLP and SVM.

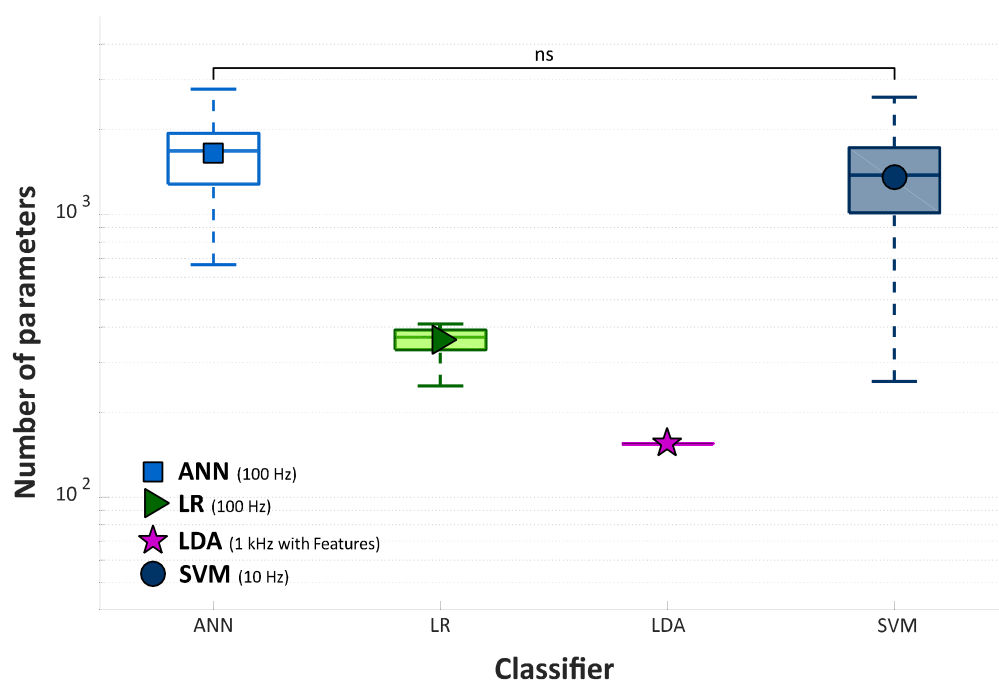


Figure 22. Number of classification parameters from 30 people with trans-radial amputation for MLP, NLR, SVM, and LDA with 5 time domain features, on a 5 classes dataset. NLR and MLP were trained using data sampled at 100Hz, while SVM using data sampled at 10Hz. The y axis in logarithmic scale. Statistical non-significance is shown by "ns".

Finally, the EOF index for LDA was evaluated and compared with NLR, MLP and SVM, as showed in Fig. 23 and Table 8. While SVM achieved significantly lower value than the other classifiers, MLP, NLR and LDA showed similar EOF score.

The Wilcoxon Signed-Rank showed no statistically significant difference only between the NLR and LDA classifier.

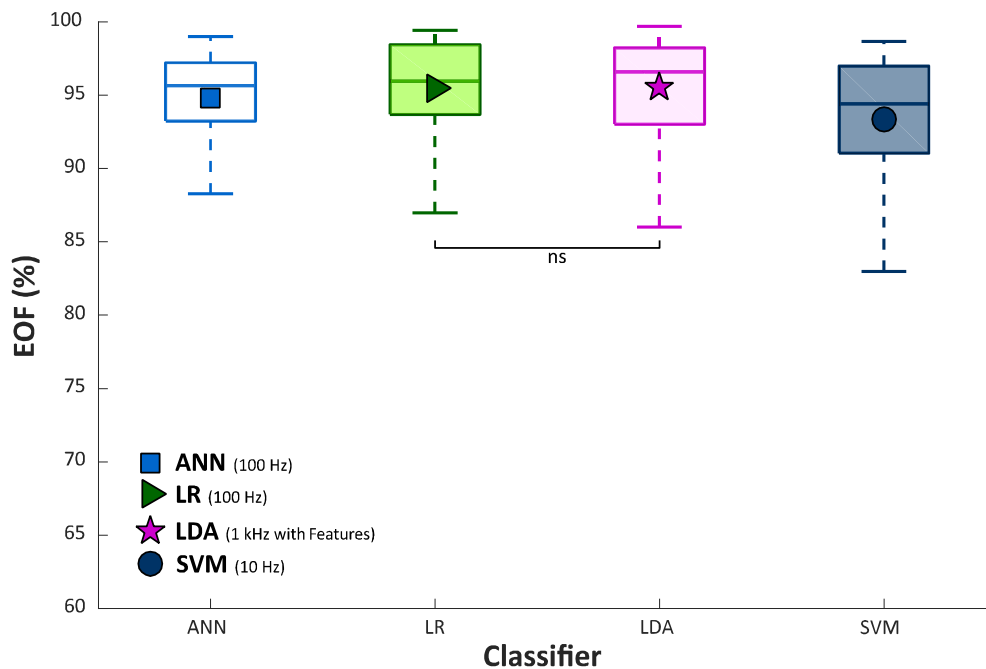


Figure 23. EOF values from 30 people with trans-radial amputation for MLP, NLR, SVM, tested on GS, and LDA with 5 time domain features, on a 5 classes dataset. NLR and MLP were trained using data sampled at 100Hz, while SVM using data sampled at 10Hz. Statistical non-significance is shown by “ ns ”.

CLASSIFICATION PERFORMANCE AND COMPUTATIONAL BURDEN FOR NLR, MLP AND SVM MODELS WITH HIGHEST EOF VALUE ON GS AND LDA SAMPLED AT 1 KHz WITH FEATURES

Classification Algorithm	F1Score	Number of Classification Parameters	EOF
NLR (100Hz)	92.0 (6.1s)	362 (41s)	95.5 (3.4s)
MLP(100Hz)	92.5 (5.9s)	1654 (605s)	94.8 (3.2s)
SVM (10Hz)	89.5 (7.3s)	1361 (648s)	93.3 (4.4s)
LDA (1kHz with features)	91.9 (6.5s)	155	95.5 (3.7s)

Table 8. Mean values and standard deviation of F1Score values, classification parameters and EOF values from 30 people with trans-radial amputation for each classifier involved in this study on a 5 classes dataset. The EOF and F1Score highest values and the lowest number of parameters are highlighted in bold. See Fig. 21-22-23 for a graphic display and statistical significance.

3.4 Discussion

In this study an in-depth analysis has been carried out of three of the most adopted classifiers for sEMG signals, i.e. NLR, MLP, and SVM using LDA with time domain feature extraction as ground truth for the final validation of the performed analysis. The choice fell on these because of the extensive discussion in the literature and because of the high performance notwithstanding the extremely different number of classification parameters. In particular, an intensive analysis on data acquired from 30 people with trans-radial amputation was conducted and performance were assessed, with special attention to the problem of developing embedded classifier solutions. Although the type and number of recruited subjects was not sufficient to generalize the results to all kinds of trans-radial amputations, this study wants to provide a solid basis for reflecting upon the trade-off between performance and computational burden of these classifiers.

Six commercial sEMG sensors produced analog signals that were sampled at 1kHz and used as “raw” input features of the classifiers. In order to speed up the training and the cross validation of NLR, MLP and SVM classification algorithms, downsampling was applied to the data creating one downsampled dataset (TR, CV, and TS) and one dataset containing all the remaining data (GS). While the TR and CV were used to train and cross validate, TS and GS have been used to test the performance of the classifiers.

The performance of NLR and MLP algorithms were firstly evaluated and then analysed with the Wilcoxon Signed-Rank test for both TS and GS. The results showed that for NLR no significant improvement of performance can be obtained for a degree of polynomial features greater than 5 and that for MLP no significant improvements can be achieved by increasing the complexity of the network up to 5 layers and 23 neurons for TS and 28 neurons for GS, respectively (Fig.17, Table 4). This result is very important because sets a boundary on the complexity of the classifier, allowing to reduce the training and cross-validating times when applying these algorithms on raw sEMG data recorded from people with trans-radial amputation. Furthermore, it is also relevant to observe that NLR in the linear case analysis (polynomial features of grade 1) obtained the lowest F1Score value with respect of the other higher grade of polynomial features, suggesting the use of a

non-linear classifier when as input features the raw outputs of the Ottobock sEMG sensors are used.

After this preliminary investigation, a comparative analysis among the NLR, MLP, and SVM algorithms was performed using data at different frequencies (5 Hz, 10 Hz, 20 Hz, 40 Hz, and 100 Hz) as TR, CV and TS. The comparison pointed out that the sampling rate and the classification performance increased at the same time (Fig. 18, Table 5). In fact, for all the algorithms the maximum performance was obtained with 100 Hz sampling rate, however, increasing the sampling rate also tends to elevate the number of classification parameters, used as index of computational burden of the classifier. The analysis showed that, for both classification performance and number of classification parameters (Fig. 19, Table 6), SVM attains the highest values followed by MLP, and then by NLR. Although downsampling causes a loss of information, classification performance was still high (ranging from 91.1% to 94.5%) meaning that the signals kept the main content related to the gesture. The reason is that, for constructing a decision boundary, it is not necessary to use high frequency sampled data during the classifier training phase; data with similar range, dispersion and redundancy are required. This also explains why GS systematically reports higher performance value than TS. GS contains a larger number of data than TS and, consequently, leads to higher performance scores. Hence, the results carried out from it might better represents the real behaviour of the classifiers when data sampled up to 1 kHz are provided as input.

Although when implementing these algorithms on PC systems it is reasonable to choose the one with the highest classification performance, when moving to embedded systems for prosthetic devices, the computational burden is no longer negligible. Hence, in order to investigate the best compromise between performance and computational burden, the EOF index was presented. Using as unique constraint the memory usage the EOF has been evaluated referring to a standard microcontroller 256 KB memory at different frequencies of TR, CV and TS. As previously reported, this is just an application example but the same method can be applied taking into account different memory values and/or other constraints, such as the available RAM memory and/or the evaluation time for a single classification for any microcontroller. The analysis performed showed that, for people with trans-radial amputation and using sampled sEMG signals to more than

5 Hz as input, the algorithm that produces the best compromise is NLR, with the highest values of EOF (95.5 %), closely followed by MLP (94.8 %). Conversely, SVM algorithm, which obtained the highest classification performance, presents considerably lower values of EOF (93.3 %) than the other two algorithms (Fig. 20, Table 7); this means that high performance is achieved at the expenses of a sharp increase of the computational burden and memory usage. Hence, it is possible to summarize that in order to choose the most suitable classifier in a real application with data sampled at the same frequency used for train and cross validate the algorithm, there is no difference between NLR, MLP, and SVM up to 10 Hz, while from 10 to 100Hz SVM becomes significantly disadvantageous with respect to the other two classifiers, which did not show significant difference. On the other hand, for use in a real application with data sampled at higher frequency (up to 1kHz) than the ones used to train and cross validate the algorithms, NLR resulted to be the most suitable clearly representing the best compromise between classification performance and computational burden. Furthermore, the analysis suggests, among the tested cases, a downsampling step equal to 10 (100Hz) for the training and the cross validation of NLR and MLP algorithms, and equal to 100 (10 Hz) for SVM. Finally, a comparison between each of the three non-linear classifiers and LDA was carried out. Since LDA was trained and tested with data sampled at 1 kHz (without downsampling), NLR, MLP and SVM models with the highest EOF values on GS for performance, number of parameters and EOF index were used for the comparative analysis. This analysis pointed out no statistically significant difference between NLR and LDA in terms of performance and EOF index (Fig. 19-20-21, Table 8) by confirming the results of the previously showed comparisons (Fig. 16-17-18, Table 5-7) despite LDA reported the minimum computational burden. Therefore, this result is also more appreciable if we consider that NLR was trained and tested using raw sEMG data. So, this study shows that it is possible to use non-linear classification algorithms on raw sEMG signals recorded from people with trans-radial amputation also for embedded applications. Furthermore, since LDA and NLR retained statistically similar value for both performance and computational burden, it is possible to speculate that the features extraction step linearizes the classification problem at the expense of a delay on the class evaluation time and on the readiness of the system during the transition between two different gestures. Indeed, using raw sEMG signals as input features the class evaluation time

and system readiness approximate the sampling time; on the other hand, using features based on time windowing, the class evaluation time equals the window shift and the readiness delay is around the half of the time window length.

It is worth noticing that, when transient sEMG signals are included in classifier training, system controllability and performance are shown to improve [63]; conversely, offline classification accuracy degrades. This comparative study was grounded on steady state sEMG signals, however, this does not affect our comparative analysis, since the experimental data were the same for all the analysed classifiers.

3.5 Conclusions

In this study the NLR, MLP and SVM classification algorithms were developed, tested and optimized on a dataset of 5 hand gestures classes composed of the data recorded from 30 people with trans-radial amputation, using 6 commercial sEMG sensors. After evaluating the maximum complexity of the NLR and MLP algorithms needed to apply PR on this population, the comparative analysis among the three algorithms was carried out. It pointed out that, for both classification performance and number of classification parameters, SVM attains the highest values followed by MLP, and then by NLR. Hence, in order to investigate the best compromise between performance and computational burden, the EOF index was presented. The analysis performed showed that, for people with trans-radial amputation and using sampled sEMG signals to more than 5 Hz as input, the algorithm that reached the best compromise is NLR (with the highest value of EOF) closely followed by MLP. This result was also confirmed by the comparative analysis with LDA with time domain features, which showed no statistically significant difference with NLR. The proposed analysis would provide innovative engineering tools and indications on how to choose the most suitable classifier, and its specific internal settings, based on the application and the desired results for prostheses control. As the research has reached an advanced grade of accuracy, these algorithms were proved and the embedding is necessary for the realization of prosthetic devices.

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Chapter 4

Embedded Control of Prosthetic Hands via Pattern Recognition

4.1 Introduction

Based on the results showed in chapter 3 NLR without feature extraction has been identified as the most suitable classifier for the embedded control of multi-fingered prosthetic hands. Avoiding the features extraction step enables an extreme reduction of the classification time response time of the system without significant loss of system performance [9]- [10]- [11]- [20]. The saved time is used to improve the stability of the classification by means of post processing techniques as voting and voting acceptance threshold [12]- [13]. The gestures to reproduce were selected among the eight canonical hand postures [7] and were “Rest” (relaxed hand), “Spherical” (hand with all fingers closed), “Tip” (hand with thumb and finger touching to pick up a small object), “Platform” (hand completely open and stretched), and “Point” (hand with all fingers closed, except for the index finger that is pointing).

As afore mentioned, the main issue related to PR algorithms is the robustness and reliability of the classification and it is unlikely to use directly the classification outputs for the online control of the prosthetic device. Hence, a control strategy able to combine robustness, reliability and less cognitive effort has to be included between the classification algorithm and the internal control of the hand as showed in Fig. 24.

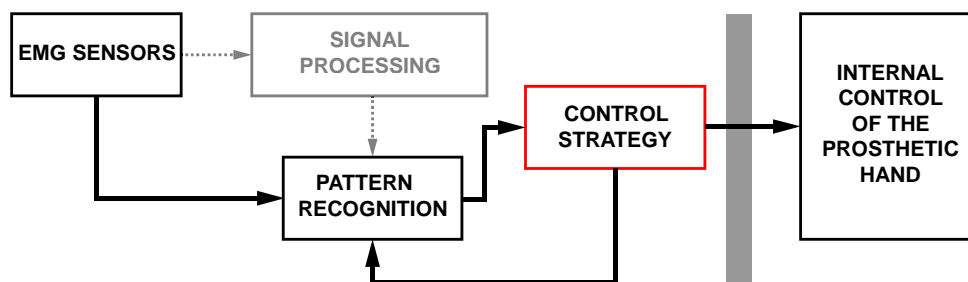


Figure 24. Schematization of the control flow. The signal processing is necessary just to filter and amplify the sEMG signal when the used sensor cannot provide it themselves.

In the following, an embedded control system (hardware, firmware, and software) and a new control strategy for hand prostheses capable to handle up to five different grasping movements via PR algorithms are presented. The functions have been allocated as schematized in Fig. 25.

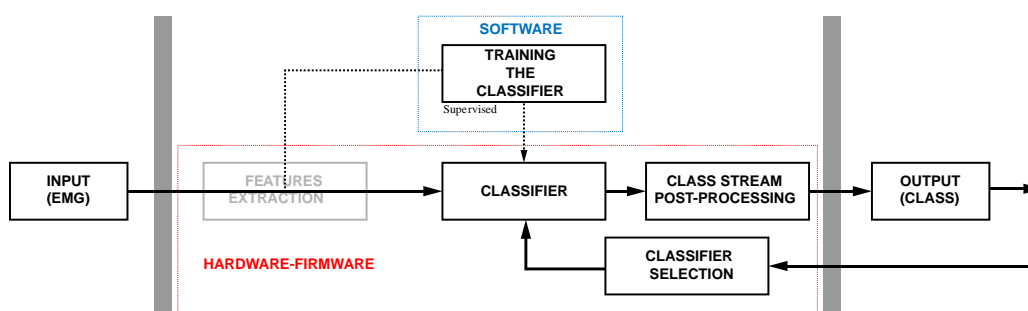


Figure 25. Schematization of the system functions allocation. Based on the results obtained in chapter 3 the features extraction step has been skipped in lieu of a post-processing applied to the output of the classification. The control strategy interacts with the pattern recognition block (fig. 24) changing internal settings of the classifier block.

This Chapter is structured as follows:

- Section 4.1 presents in details the new PR control strategy and the designed hardware, software, and firmware of the prosthetic control system;
- Section 4.2 shows the device made and the preliminary results on one subject with trans-radial amputation.

Unfortunately, it is not possible to show the detailed results of 3 months of testing of the entire system on one amputee with neural interfaces, being the study still unpublished.

4.2 Design and Development

4.2.1 The Proposed Myoelectric Control Strategy for Hands Prostheses

As reported in Sect. 2.2, the most clinically adopted control strategy relies on a cyclic selection (Fig. 10) and the switch among the states of the cycle is achievable by means of a trigger command (e.g. co.-contraction). This serial operation, despite being extremely robust, is slow and unnatural; in addition, it requires considerable training and cognitive effort on behalf of the user [2]. On the other hand, control based on PR algorithms can overcome these issues but it is unlikely to use directly the classification outputs for the online control of the prosthetic device (Fig. 26).

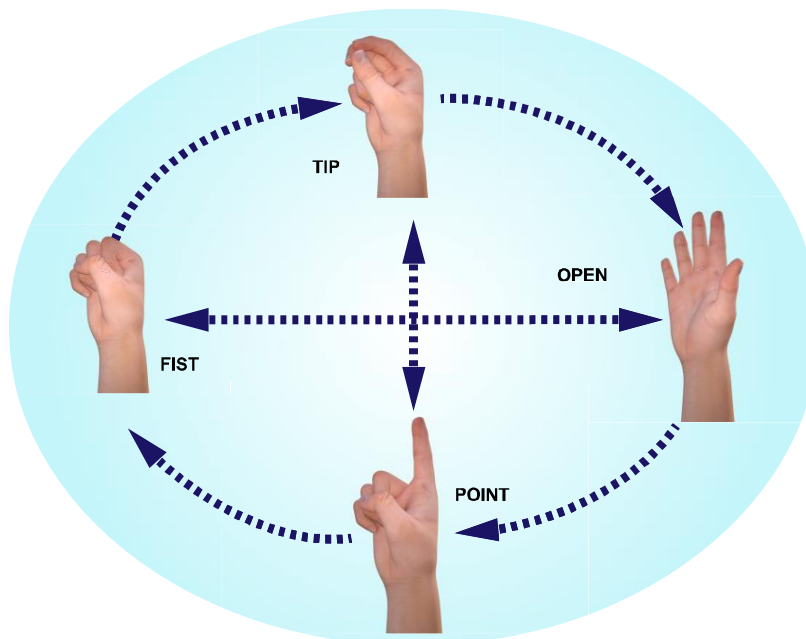


Figure 26. Direct pattern recognition control strategy. Once a class output is available the prosthetic hands moves to the corresponding gesture.

Because of the deficiency in robustness and reliability of the classification the application of this strategy can result in snappy movements and instability of the control, making difficult to perform grasp operations. Hence, a new control strategy able to combine robustness, reliability and less cognitive effort has been proposed (Fig. 27).

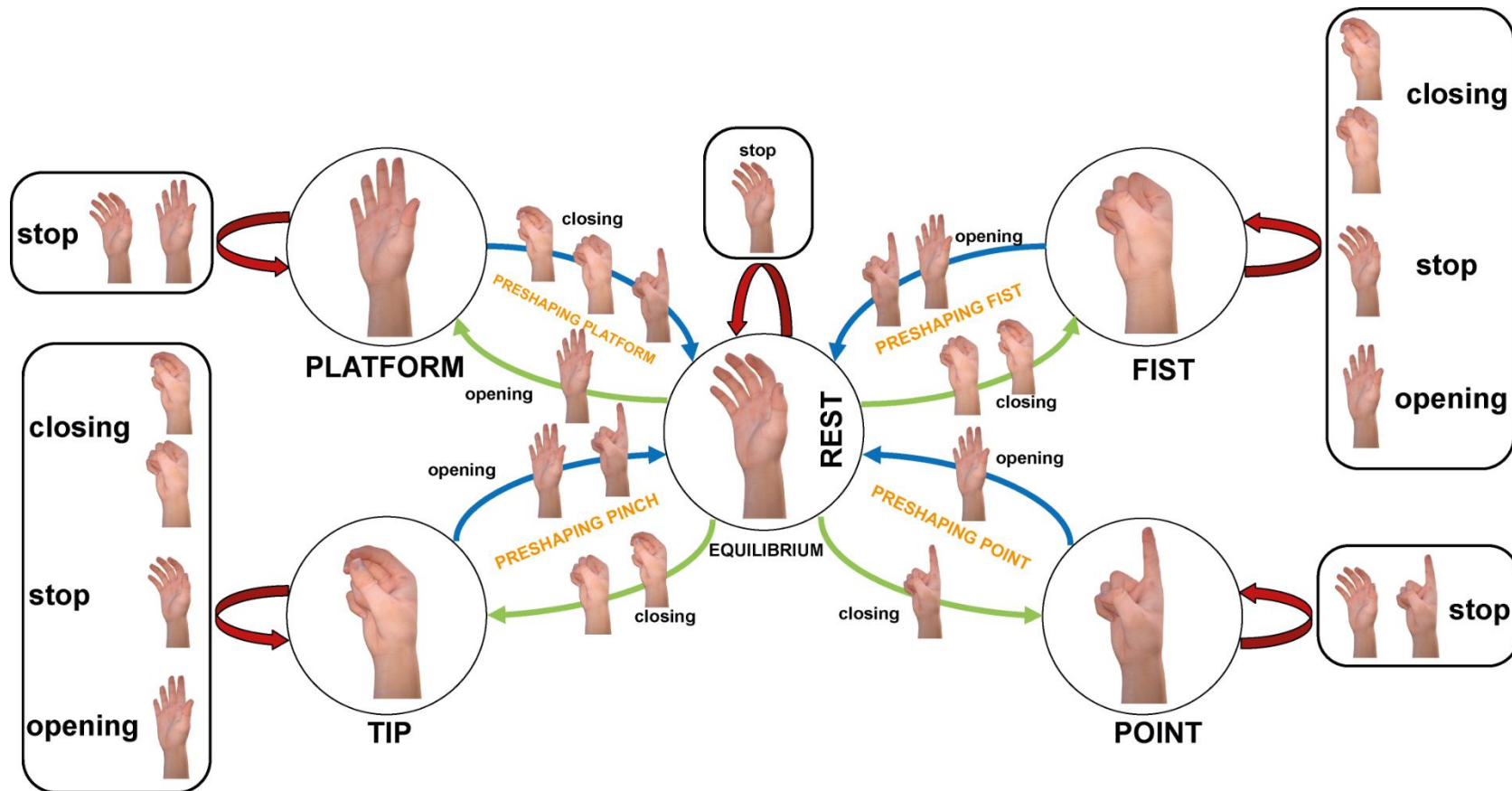


Figure 27. Finite state machine control strategy for 5 different gestures. It is possible to observe that this strategy is a hybrid between the cyclic control strategy (Fig. 10) and the direct pattern recognition one (Fig.24).

According to this control strategy, there are 5 different states (one for each gesture to reproduce) each one is flanked by a specific optimized classifier. The structure presents a central state called “*equilibrium*” associated to the rest gesture representing the starting point of the finite state machine (FSM), and 4 satellite states dedicated to the grasp execution. As it will be seen below, the classifier is closely linked to the control strategy. Indeed, the classifier generates the input class (hand posture to be reproduced) for the strategy; on the other hand, the classifier to be used depends on the present state of the control strategy. In fact, the classifier block in Fig. 25 is actually composed of several classifiers, one for each state of the control strategy as shown in Fig. 28.

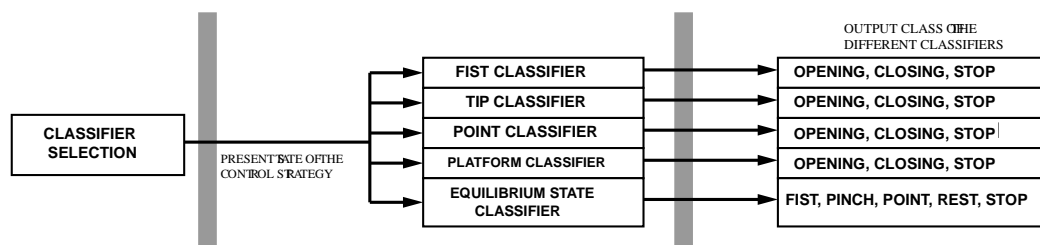


Figure 28. Structure of the Classifier Block.

Depending on the present state of the control strategy, both the classification algorithm of the sEMG signal and the output generated by the classifier block vary. The switch between the satellites states can only come through the equilibrium state and the transitions are dedicated to the preshaping of the prosthetic hand. This lead, to a clear simplification of the classification by restricting the issues related to direct PR control strategy to the equilibrium state only. Indeed, for the satellite states we move from a 5 to a 3 classes classification problem thus yielding an increase of robustness and reliability of the system. Each classifier associated with each one of the satellite states is trained with a dataset obtained by grouping the electromyographic signals into 3 sets: a set for the poses that express a closure (e.g. fist and tip), one for the poses that express an opening (e.g. open and point) and one for the rest. For a detailed explanation of this strategy, consider to be in the equilibrium state. In this state the “equilibrium state classifier” (Fig. 28) is able to recognize all the considered gestures (FIST, TIP, POINT, OPEN and REST). If the REST class is classified the hand remains stationary, while in case of the classification of one of the other four classes (e.g. FIST) the prosthetic hand starts

to preshape. During the preshaping state the classifier associated with the corresponding status is active, for example during the "preshape fist" state the "fist classifier", which will no longer recognize 5 classes but only 3 (Fig. 28). Once the preshaping has been completed, the hand is ready to perform a grasping task (in this case a power grasp) and the satellite fist state has been reached. Here, the "fist classifier" remains active and it will be possible to control the prosthesis for grabbing or realising an object. Again, if the REST class is classified the stop command is performed. This feature in behaviour is very important because allows the user to relax his/her forearm muscles when holding an object as in clinically available myoelectric controls (Section 2.2). Once, the task has been completed it is possible to move back to the equilibrium states keeping the opening contraction pattern.

4.2.2 Control Strategy Variant for Force Management

The control strategy presented in section 4.2.1 can be modified in order to control of the force exerted when holding an object during a grasping task. This can be achieved by means of a 3 -level (*High, Medium and Low*) force classifier cascading the classifiers associated to the Fist and Tip gestures (Fig. 28) as showed in Fig. 30.

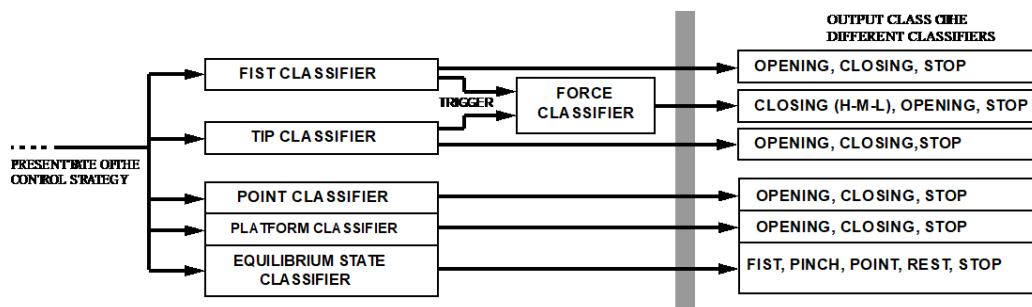


Figure 29. Variant of the Classifier Block. The trigger can be achieved by means of touch sensors placed on the prosthetic hand.

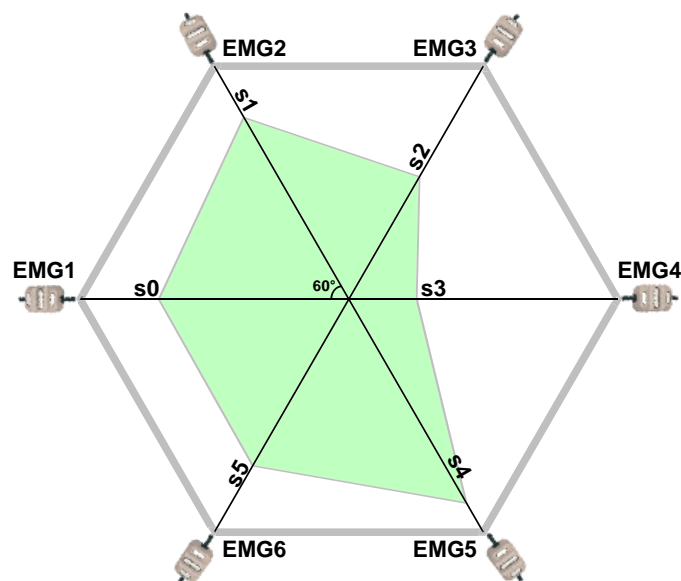
In order to avoid instability of the classification this classifier has been designed to be active only when holding an object thus enabling manipulation functionality. The information of taken object (trigger) can be achieved by means of touch sensor placed on the fingers of the prosthetic device. The force classifier, based on the NLR algorithm (Section 3.2.2), is trained new on sEMG data expressing different muscular contraction levels for the closing action and the pre-existent data for the

opening and stop movements. Table 9 reports an example on how to subdivide the sEMG data in for the training of the Force Classifier.

SUBDIVIDING THE sEMG DATA FOR THE TRAINING OF THE FORCE CLASSIFIER				
1	2	3	4	5
Closing H	Closing M	Closing L	Opening	Stop
5 FIST H	5 FIST M	5 FIST L	10 OPEN	10 REST
5 TIP H	5 TIP M	5 TIP L		

Table 9. Subdividing the sEMG data for the training of the force classifier.

Depending on user ability it is also possible modulate the exerted force in a proportional way as presented in section 2.2. In this case, a threshold can be used to identify the STOP command, and a binary classifier specialized in recognizing the Opening action. The closing proportionality can be thus achieved by considering the sum (ore the mean) of the amplitudes recorded from the sEMG sensors. Figure 30 shows an example in evaluating the overall intensity (I_{EMG}) of the sEMG signal exploiting a star diagram.



$$I_{EMG} = \left(\frac{s_5 \cdot s_0}{2} + \sum_{i=0}^4 \frac{s_i \cdot s_{i+1}}{2} \right) \cdot \sin \frac{\pi}{3} \quad (17)$$

Figure 30. Example of a 6 sEMG signal star diagram and the formula to evaluate the overall intensity (I_{EMG}) of the sEMG signal.

Figure 31 shows the schematization of the variant of the proposed control strategy enabling also manipulation functionality.

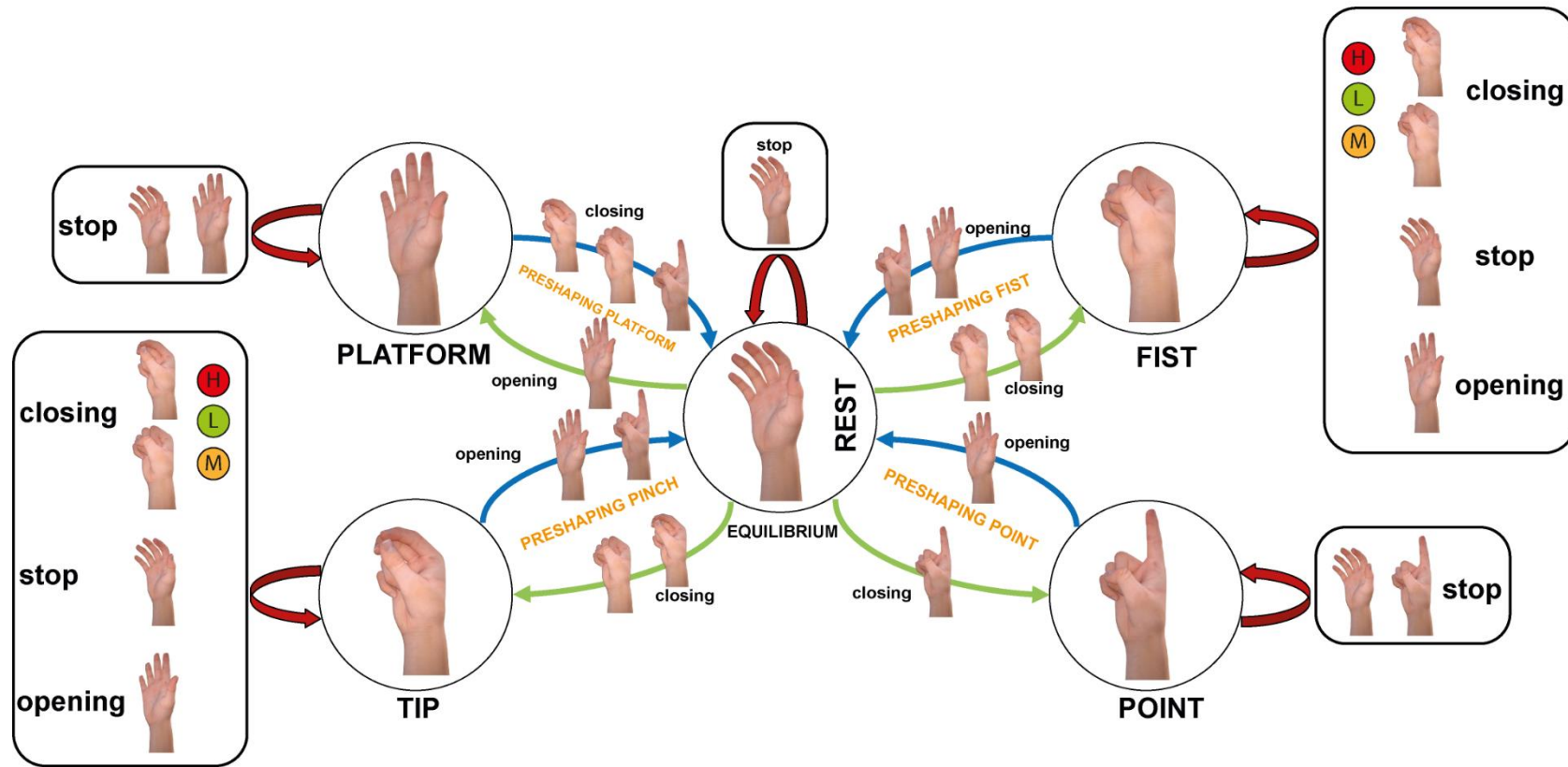


Figure 31. Finite state machine control strategy for 5 different gestures and force management.

4.2.3 Hardware

The hardware system, object of study, must guarantee the execution of the algorithms and the proper functioning of sEMG sensors, prosthetic hand and PC software interfaces representing the classic case in which a microcontroller system is implemented. Based on the results obtained in chapter 3 and having available the Robo-Limb [25] prosthetic hand an ARM4 32bit NXP [64] microcontroller with a 128Kb flash memory and 100MHz clock frequency has been chosen as core element of the hardware. Also, in this case six commercial active sEMG sensors (Ottobock 13E200=50), representing the state of art about myoelectric sensors for prostheses are used to measure the intensity of the sEMG signal. Figure 32 reports the hierarchical electronic schematic of the hardware. The hardware has been designed for the use in a complete prosthetic device, hence as power supply requires an 8.1V Li-ion battery as commonly used in clinics for trans-radial myoelectric prostheses.

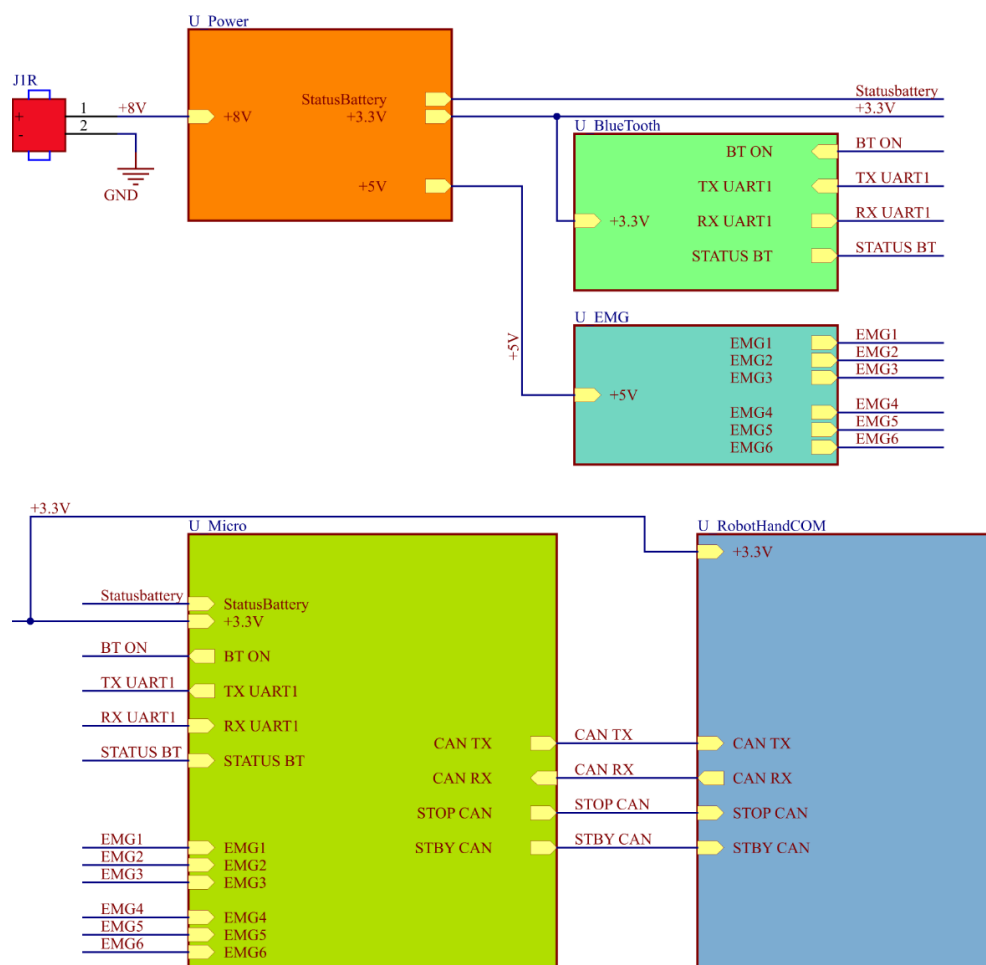


Figure 32. Electronic hierarchical schematic of the hardware.

The prosthetic hand is controlled via high speed CANbus communication protocol (ISO 11898-2) and the hardware is able to connect to the software part dedicated to the training of the PR algorithm (Fig.25) by a wireless bluetooth communication. The PCB counts 234 SMD components and has been designed as a 4-routing layer layout (Fig. 33) in order to keep the overall dimensions reduces.

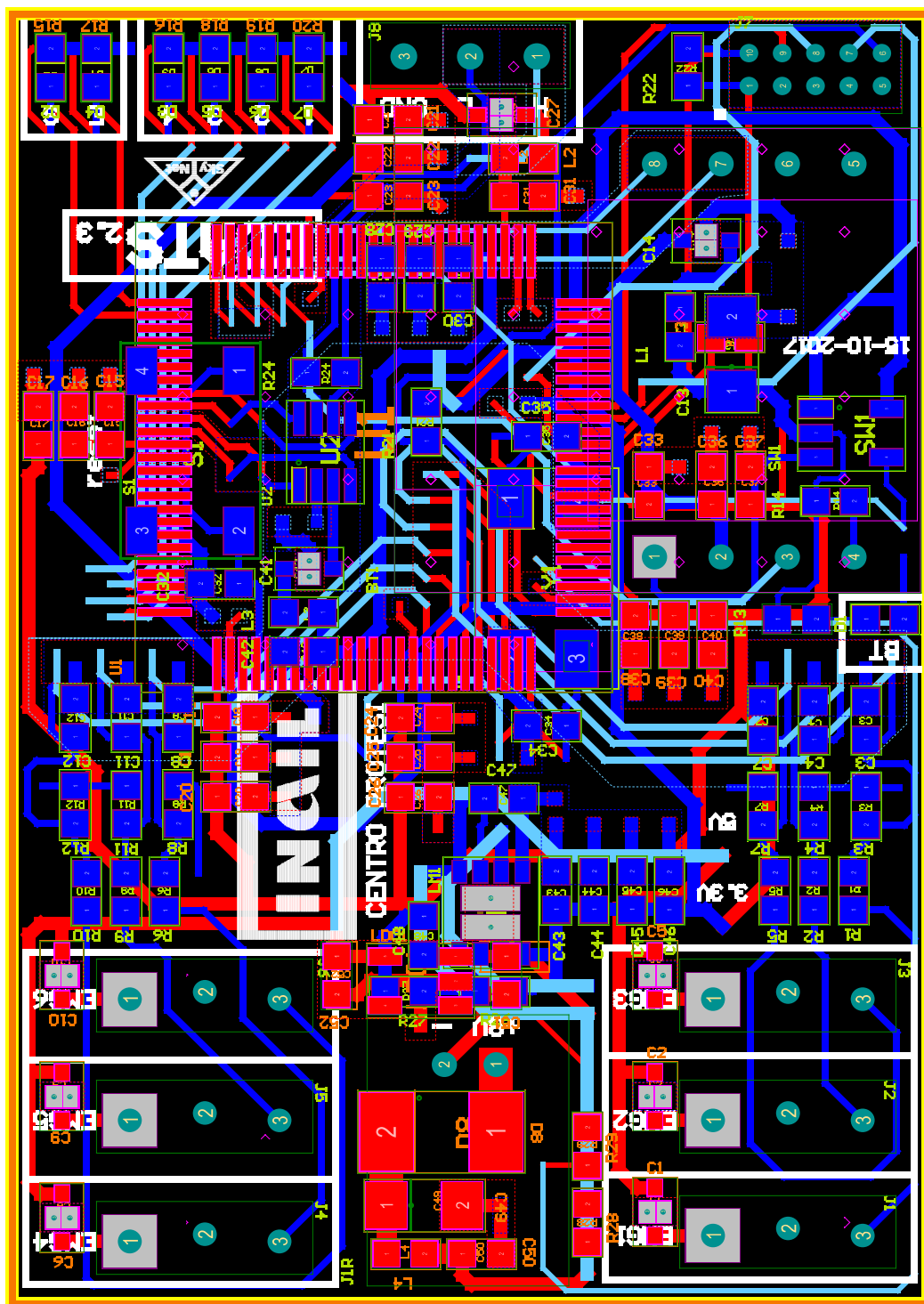


Figure 33. PCB layout.

Despite integrating electrostatic discharge protections, high frequency filters and all the components to achieve the desired functions, it was possible to obtain PCB of 50x35x1.6mm and a no-load consumption of 13mA, perfectly in line with the control units commonly used in clinics.

Figure 34 show a detailed picture of the result after the manufacturing stage.

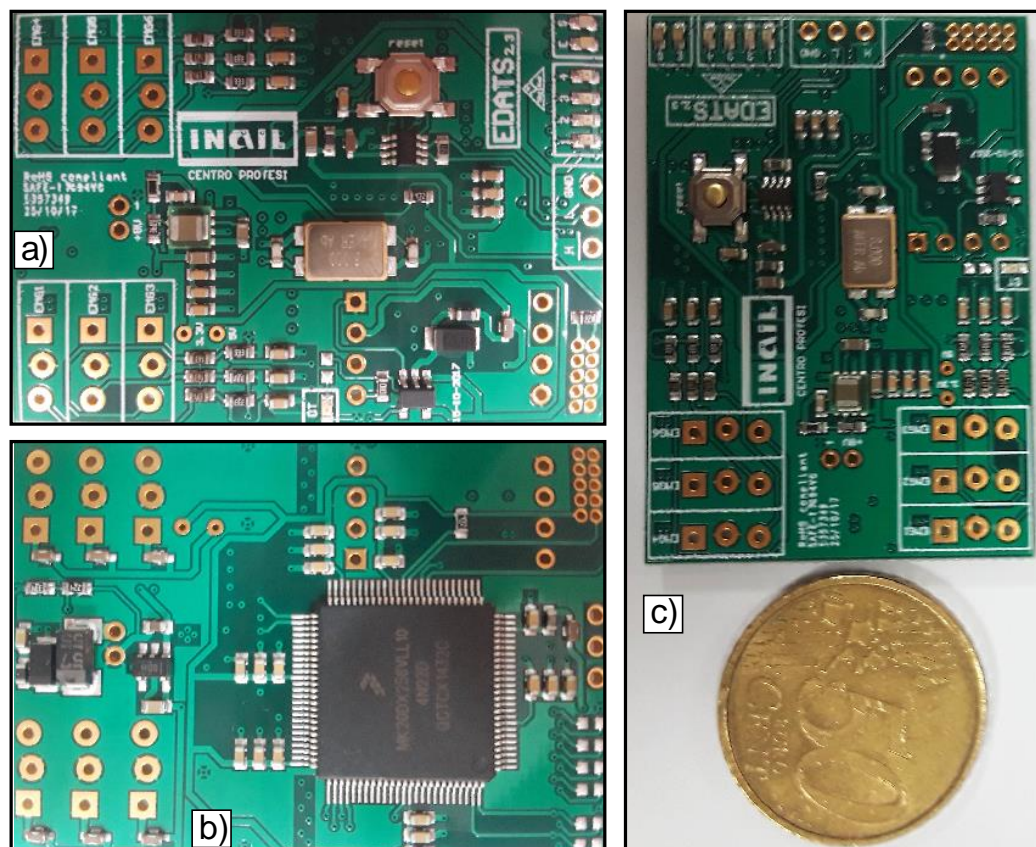


Figure 34. Pattern recognition control unit PCB; a) Top view; b) Bottom view; c) Overall sizing placing the PCB near a 0.5€ coin.

4.2.4 Software

As show in Fig. 25 as mid-layer interface, between the user layer and the hardware layer, an on-purpose built software on LabView platform was developed. The software, called EMG-Data Acquisition Software (EDATS) communicates with the hardware through a wireless Bluetooth (BT) connection and with the user by different graphic user interfaces (GUI). The main purpose of this software is to implement an off-line training of the NLR classification algorithm. When launched, EDATS starts with a main menu, which offers 3 different but intuitive choices:

TRAINING & SIMULATION & UPLOAD; SIMULATION & UPLOAD; UPLOAD MODEL. Figure 35 shows the GUI sequence for the selection of the first button. Once the user's ID is entered and adjusted all the settings about number of sEMG sensor to use, number of classes to identify, sampling frequency and recording time window, a new GUI dedicated to the data acquisition appears. Then, once a desired number of acquisition for each class is reached clicking on the button "TRAIN THE CLASSIFIER" the training of the algorithm presented in chapter 3 begins. In order to test the on-line performance of the trained classifier another new window appears with which is possible to test different combinations of voting number and decision threshold (TH) values for each one of the classifier describe in section 4.1.1. Once the classifiers meet the expectations is possible to proceed to the final GUI for the upload of the settings and the classification parameters (model) into the embedded system.

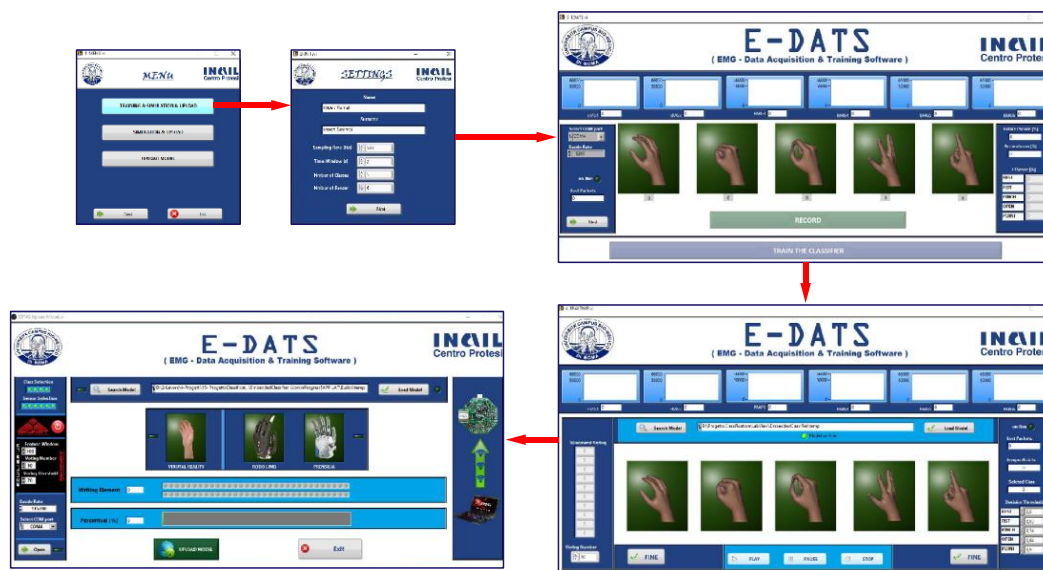


Figure 35. EDATS, TRAINING & SIMULATION & UPLOAD flow chart: in this example has been decided to classify 5 hand gestures (Rest, Fist, Tip, Open and Point).

Furthermore, the software offers also online-test functionality of the system thanks to an interface called "FIRMWARE TEST" (Fig. 36). The software allows the user to control a virtual reality limb using the output class provided by the FW-HW parts displaying meanwhile the cycle time, the current settings of the classifiers and eventually errors detected by the FW diagnostic system.

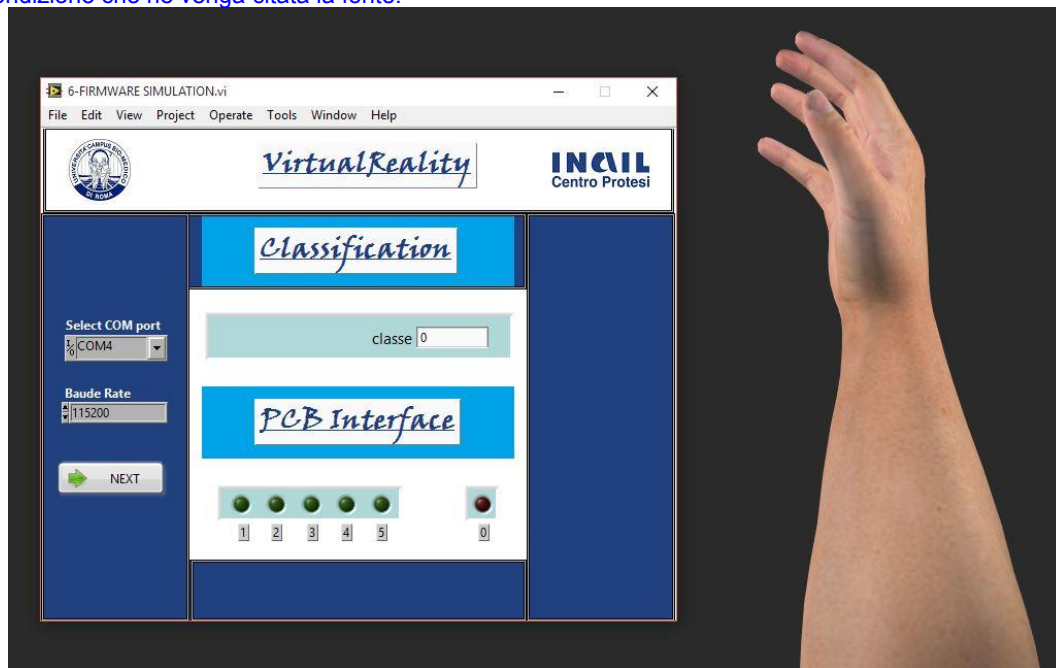


Figure 36. FIRMWARE TEST software Interface.

4.2.5 Firmware

The firmware is the software that is installed on the microcontroller and that, through a loop system, manages all the operations that the electronics of the control unit must perform in response to electrical input signals, such as electromyographic data. The firmware was developed in C language exploiting the Kinetis Design Studio IDE; as showed in Fig.25 this part performs: the NLR classification algorithm; the post processing classification policies; the communication with the PC software interface; and the control strategy described in section 4.1.1.

Figure 37 reports a simplified version of the firmware flowchart. When the circuit is powered an initialization procedure is firstly executed. Then, if there is no model of the classifier saved on flash memory, a BT streaming session of all the sEMG signals is started, otherwise the settings and the classification parameters on flash memory are used to classify the sEMG signal and send actuation commands to a Robotic Hand through the wired UART connector. Anytime, when a specific BT command is received, an interrupt routine is launched in order to download a model from the EDATS. The last instruction of this interrupt is a firmware reset. The developed firmware has proved to be very performant obtaining an 800 μ s cycle time in EMG streaming mode and employing less than 2 ms for a single sEMG classification (without voting).

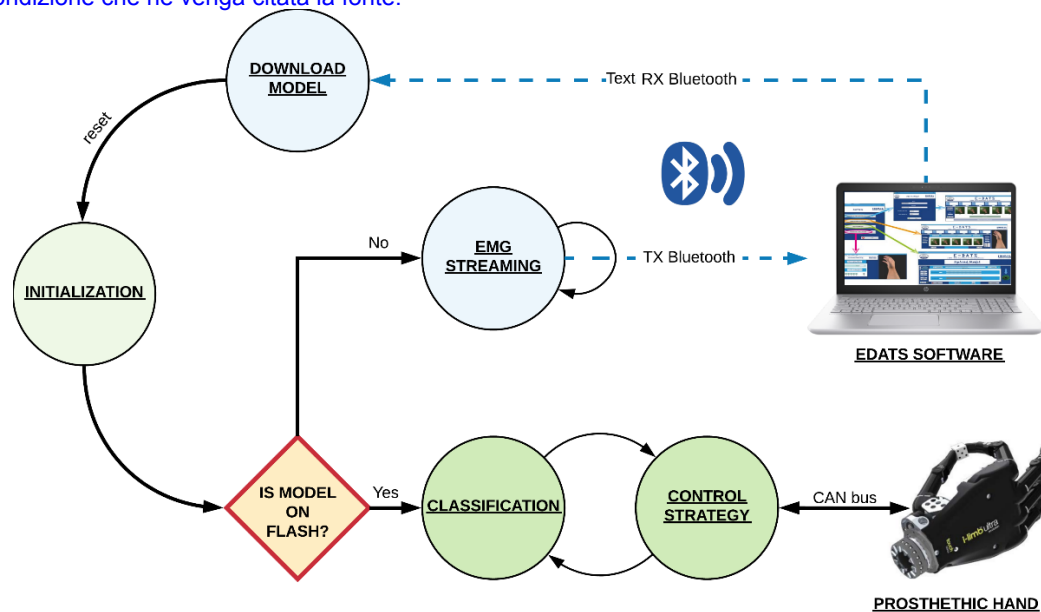


Figure 37. Simplified firmware flowchart.

Figure 38 presents a detailed description of the Firmware via sequence diagram. The Firmware presents main loop flanked by several interrupt routines grouped by priority ID. In system programming, an interrupt is a signal to the processor emitted by hardware or software indicating an event that needs immediate attention. The processor responds by suspending its current activities, saving its state, and executing a function called an interrupt handler (or an interrupt service routine, ISR) to deal with the event. This interruption is temporary, and, after the interrupt handler finishes, the processor resumes normal activities. In this case, the code has been designed with 3 level of priority:

1. *High-Priority*: occurs every 100 μ s and it checks if any fatal error is occurred and eventually responds by suspending all the activities and turning on a bright red led;
2. *Medium-Priority*: Occurring every 1ms this interrupt handles the AD converter, the UART and the CAN peripheral (see table 10 for the settings);
3. *Low-Priority*: Every 60 seconds this interrupt checks the status of the battery.

Each function created for the realization of the firmware presents a self-diagnostic instruction able to recognize both the most frequents errors (e.g. low battery) and the most serious ones (e.g. overcurrent, or system failure).

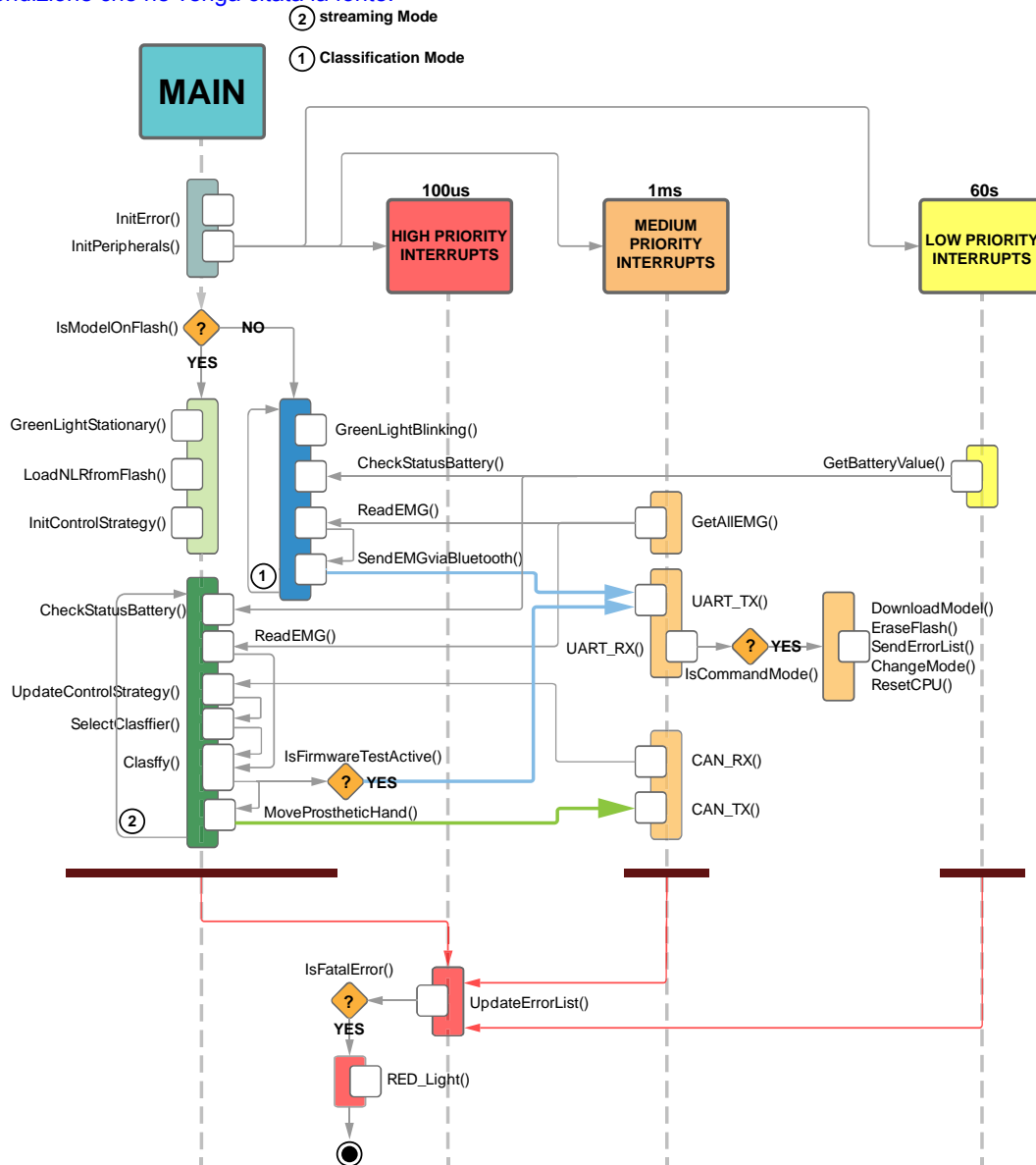


Figure 38. Firmware Sequence Diagram

CAN AND UART PERIPHERAL SETTINGS

UART	
PROPERTIES	SETTINGS
Data Width	8 bits
Parity	None
Baud Rate	115200 bps
Stop Bits	1
CAN	
PROPERTIES	SETTINGS
Bit Rate	1 Mbps
Time quanta per bit	10
Propagation Segment	1
Time Segment 1	6
Time Segment 2	2
Resync Jump Width	1

Table 10. CAN and UART Peripheral Settings.

4.3 Results and Conclusion

Once all the parts described in section 4.1 have been completed, the prosthetic device has been assembled in the Prosthesis Centre in Vigorso Budrio for a male subject with a right trans-radial amputation (Fig. 39), and already experienced in myoelectric control of prosthetic hands.



Figure 39. Building the socket from the stump of the subject who took part to the experiment.

The six-commercial active sEMG sensors (Ottobock 13E200=50), were equidistantly positioned in appropriate locations created on the socket (Fig. 40)



Figure 40. Inside view of the socket of the prosthesis.

Battery, cables and the control unit reside in the space between the socket and the outer shell of the prosthetic device (Fig. 41)

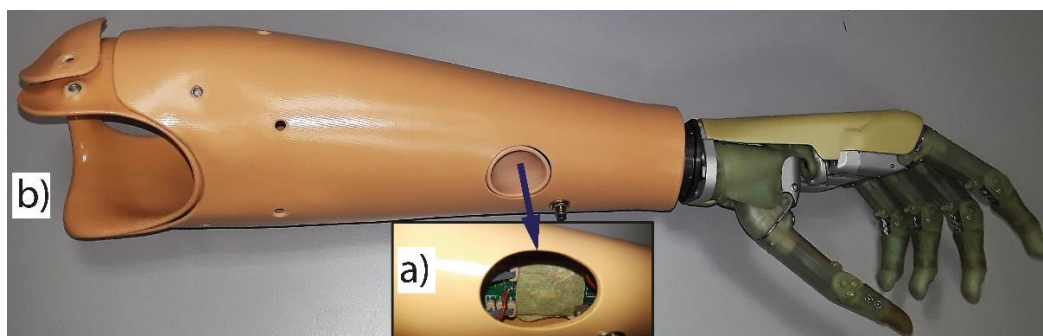


Figure 41. The Complete Prosthetic device. a) Detail of the components placed inside the Prosthesis. b) Lateral view of the Prosthesis.

Using an on-purpose built software developed in LabView environment the maximum frequency of the sEMG signal that is available when the system is in streaming mode resulted 850Hz. Furthermore, exploiting the measure functions on an oscilloscope emerged that 1.2ms is the minimum time required for a single classification (no post-processing applied). Regarding the power consumption: 45mA is the average current through the control board in classification mode (with bluetooth and CAN transceiver active) when only the prosthetic hand is powered with a different supply. On the other hand, turning off the bluetooth module the average current trough the control board in classification mode settles on 27mA. Finally, 25mA is the average current trough the control board in streaming mode when the CAN transceiver is turned off. Table 11 summarizes all these results.

ELECTRONIC PERFORMANCE TEST RESULTS	
RESULTS	
Maximum sEMG frequency in streaming mode	850Hz
Minimum cycle time for a classification (no post-processing applied)	1.2ms
Maximum depth of the buffer for real-time applications	250
Current consumption in streaming mode (CAN transceiver off)	25mA
Current consumption in classification mode (Bluetooth module off)	27mA
Current consumption in classification mode	45mA

Table 11. Results of the electronic performance tests.

On April 2017, the first training session of the complete system was carried out on the aforementioned volunteer subject.

The subject gave informed consent before performing the experiments, which were approved by local scientific and ethical committees. Again, the gestures to reproduce were “Rest” (relaxed hand), “Fist” (hand with all fingers closed), “Tip” (hand with thumb and finger touching to pick up a small object), “Platform” (hand completely open and stretched), and “Point” (hand with all fingers closed, except for the index finger that is pointing). Using an 80-deep voting buffer and a 70% voting acceptance threshold (which were evaluated by means of the optimization algorithm described in section 3.1.2) as settings of the post-processing stage (Fig. 25) the system was able to achieve an astonishing 99.3% for both off-line accuracy and F1Score, proving the subject’s capability to use the developed myoelectric control. Figure 42 shows the confusion matrix associated to the training session and table 12 summarizes the obtained results reporting the F1Score achieved for each single class.

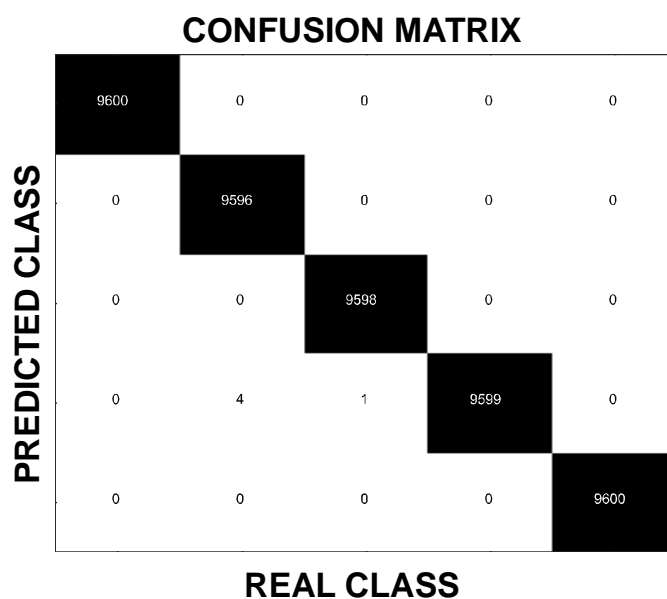


Figure 42. Confusion matrix associated to the April 25 training session

APRIL 25 2017 RESULTS TABLE

Gesture	F1Score	Average F1Score	Average Accuracy
Rest	100 %		
Fist	100 %		
Tip	99.1 %	99.3%	99.3%
Open	99.2 %		
Point	100 %		

Table 12. April 25 2017 training session results table

After uploading the model inside the control unit, the subject was immediately able to control all the gestures of the prosthesis and the system appeared sufficiently reliable even on-line (Fig. 43) allowing to perform ADL tasks as grasping a bottle of water and drink.

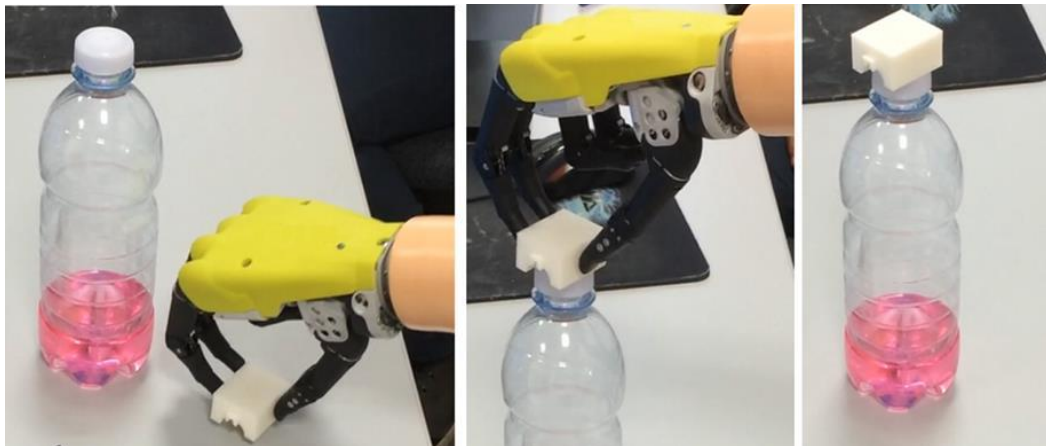


Figure 43. Subject performing a complex task.

Using an 80-deep buffer voting the time between two subsequent classifications is about 100ms. Indeed, the subject was not able to perceive any delay in the actuation of the prosthesis. The only instability of the system was observed when the arm was raised beyond the subject's head. After the test session, the prosthesis was put away and a second meeting with the subject was set after 2 weeks in order to check the stability of the system over time. Despite a slight but noticeable reduction in the stability of the system, after 2 weeks the subject was still able to control all the gesture suggesting that the combination of sensors and socket, rather than a silicone belt (Fig. 13a), greatly limits the electrode shift issue. Furthermore, the stability drop could be related to the long-time of system inactivity, which caused the subject a difficulty in remembering the specific muscular contractions used to train the system during the first experimental session, rather than to a modification of the experimental setup.

Even though preliminary, these results are encouraging and provided the foundations for a 3-months experimentation on one amputee subject for the bidirectional control of a prosthetic hand by means of neural interfaces and myoelectric control. However, the results of such an experimental study cannot be reported here being still unpublished.

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Chapter 5

sEMG and eENG pattern recognition for Prosthetic Hand Control

5.1 Introduction

As just told in the previous chapters, the hand loss has enormous repercussions at both anatomical and psychological level and causes great interest in the field of upper-limb prosthetics. Since they allow decoding motion intentions in a more intuitive way than proportional sEMG control, pattern recognition techniques have gained growing interest in the field of upper limb prostheses. However, although solutions based on pattern recognition on sEMG signals are promising, they suffer from the limitation that the subject cannot be provided with a natural sensory feedback. On the other hand, it has been shown that invasive solutions based on neural electrodes allow directly stimulating the Peripheral Nervous Systems (PNS) and eliciting in the patients close-to-natural tactile feedback [16]- [17]- [18]- [19]. This kind of interfaces can be used with the twofold purpose of stimulating the PNS and recording the neural information coming from the brain to drive the residual muscles. Hence, assuming that it is possible to de-code human motion intention from the neural signals, i.e. the electroneurogram (ENG), neural interfaces offer the huge advantage that they can both record from and stimulate the PNS in a more natural way than the other interfacing systems. To date, only a few studies have

provided evidence of the feasibility to use the ENG recorded from peripheral nerves to control a prosthetic device [16]- [65]- [66]. In Figure 44, a functional scheme of the neural control of a prosthesis is shown. Indeed, this control approach is difficult to be pursued for a number of reasons.

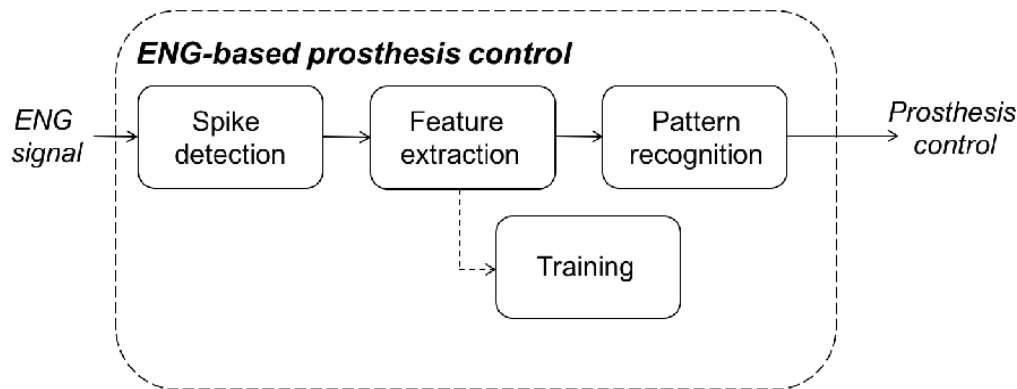


Figure 44. Block scheme of a neural-controlled prostheses approach.

First, the recorded ENG signal is very noisy because of different noise sources, e.g. the muscular activity, the devices and the wires used for recording. Furthermore, the ENG signal processing for identifying specific neural patterns is highly demanding in terms of both complexity and computational burden. Spike Sorting Algorithms (SSA) are typically adopted to recognize action potentials that are supposed to be the depolarization of a particular neuron. SSA consists in three main steps: Spike detection, Feature Extraction and PR. The high number of neurons that can be involved in muscle activation can notably rise up the computational burden of the algorithm: correspondingly, SSA performance could not improve and, in some cases decrease. Moreover, the algorithm can hardly be implemented on-line if high accuracy of the system is demanded. This paper wants to propose a new method for processing ENG signals specifically aimed at controlling hand prostheses. The proposed method intends to overcome limitations of traditionally adopted techniques of ENG processing by (i) computing the ENG envelope (eENG), starting from the recorded neural signals; (ii) resorting to sEMG PR techniques applied to the eENG. The main advantages of the proposed approach are related to the reduced computational burden and the on-line processing (which are paramount for closed loop interfaces). The authors recently proposed a mathematical formulation to compute the ENG envelope (eENG) and showed that

it is possible to relate concurrently recorded ENG and sEMG signals [67]. Here, the author wants to show that pattern recognition algorithms applied to the ENG envelope can be used for gesture decoding. To this purpose, neural and muscular signals simultaneously recorded from one human amputee are used in this analysis. They are acquired respectively with intraneural and sEMG electrodes and are related to two gestures, i.e. little finger extension and open hand. The envelope of the ENG signals is computed and a SVM algorithm is adopted to decode user's intention. SVM was chosen to be consistent with the pattern recognition algorithm adopted in the literature on neural signals on hand control [68]- [66]. For comparison purpose, the same neural signal was processed with a standard SSA. Finally, PR algorithm based on SVM was also applied to the simultaneously recorded sEMG signals for comparison purpose. Furthermore, performance decay of SSA with the increase of gestures was studied by means of synthetic data. In section 5.1, signal recording, algorithm formulations and data analysis techniques are described. Results are presented in section 5.2 and discussed in section 5.3. Conclusions are finally reported in section 5.4. The following contents are taken from the paper that the candidate has submitted on *Journal of Neuroscience Methods* [21].

5.2 Methods

5.2.1 *Simultaneous ENG and sEMG recordings from Amputee Subject*

Simultaneous ENG and sEMG signals were acquired during a human experimentation on one amputee [66]. Intraneural tf-LIFE4 electrodes were implanted in the median and ulnar nerves of a male amputee and were used to record the user intention on the efferent pathway and to stimulate the peripheral nerves on the afferent pathway in the attempt to restore tactile feedback. The study was approved by the local Ethics Committee and by the assigned office of Italian Ministry of Health and an informed consent was signed by the patient in the presence of a witness from his family. The amputee was seated on a chair during the trials and 6 repetitions for each gesture were performed. An example of

simultaneous recording is shown in Fig. 45. We will refer as reference the sEMG signals recorded when the subject was asked to perform a particular movement. ENG and sEMG signals were recorded with a sampling frequency of 48 kHz. The ENG signals were filtered between 700 Hz and 5000 Hz by means of a 4th-ordered Butterworth filter. Because of the quality of the recorded signals, only two gestures were analysed, i.e. the open hand and the little finger movement. All the subsequent processing on these data were performed off-line.

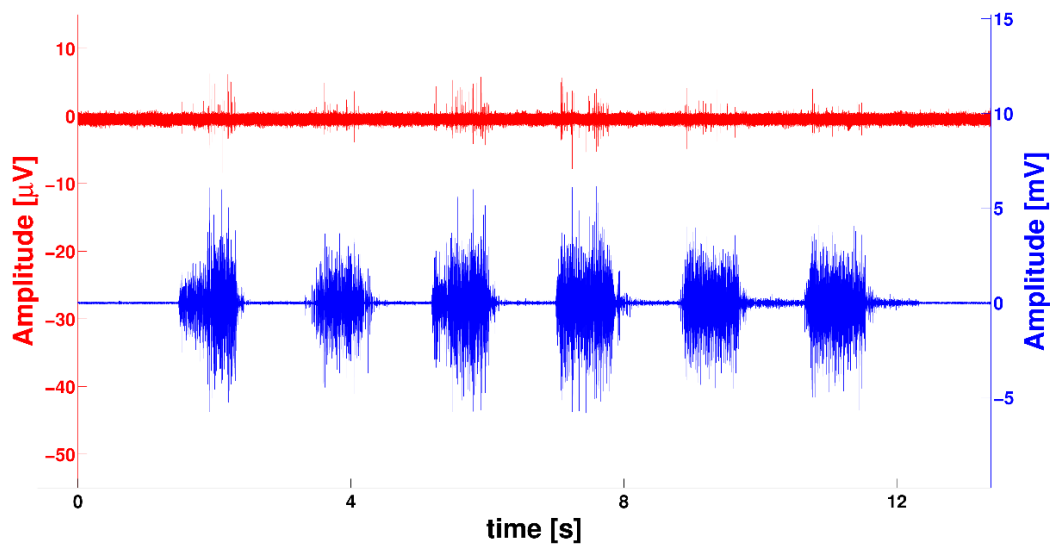


Figure 45. Neural (in red) and muscular (in blue) recordings during little finger extension.

5.2.2 Synthetic neural Data

Synthetic data offer a number of advantages: it is known which action potentials lay in the recording, how many spikes can be detected and their exact positioning on the temporal axis. The database adopted is created in order to accurately replicate a neural recording both for the features of the background noise and for the shapes of the action potentials. The complete description can be found in [69]. Synthetic signals were sampled at 96 kHz and, later, downsampled at 24 kHz. They were filtered between 300 Hz and 6000 Hz. The database is composed of 95 simulations and the number of different spikes recognizable inside the recording ranges from 2 to 20.

5.2.3 The proposed ENG classification Method

The proposed method wants to ensure on-line classification of ENG signals for identifying different hand gestures, without resorting to traditional methods based on SSA. Peripheral neural data convey the information on a desired task to the muscles; the neural signal can be regarded as the driving force that makes the muscles contract accordingly. Hence, it is reasonable to assume the existence of a relationship between ENG and sEMG. The authors recently proposed a mathematical model relating envelopes of surface sEMG and ENG signals [67]. It accounts for the amplitude and the relative positioning of the action potentials fired during a motion task and extracts the eENG that is related to the sEMG envelope. In order to compute the eENG, a spike detection algorithm, is applied to the neural recording. The detection of the spikes is achieved by means of two steps. During the first one, the energy E of the neural recording is computed by means of a Moving Average Algorithm (MMA) [70], then the envelope is finally computed as

$$eENG[i] = \frac{\alpha A_{i+1} + \beta A_i}{2} \cdot e^{-\frac{O_{i+1}}{O_i}}, \quad (18)$$

where the weight factors α and β are $0 < \alpha, \beta < 1$ and A_i and O_i are the amplitude and the occurrence of the i -th spike, respectively. The weight factors are chosen so that the ratio between the areas under the two curves is as close as possible to 1. For α and β set to 0.5 and 0.4 respectively the muscular and the neural signals were highly correlated [67] as illustrated in Fig. 46.

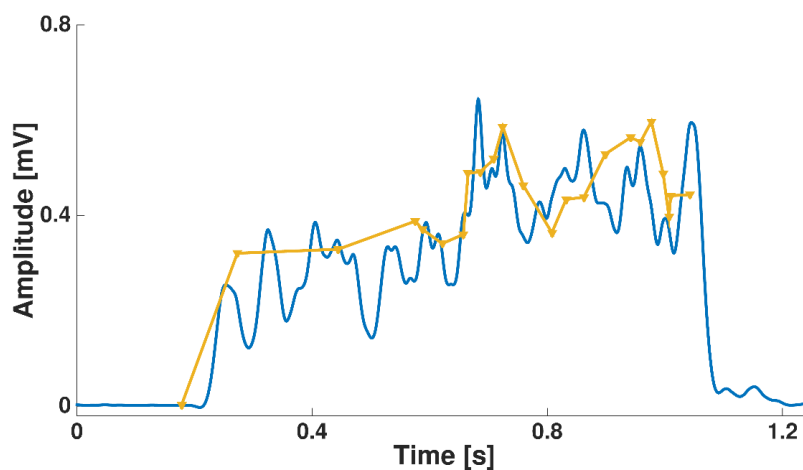


Figure 46. sEMG envelope (in blue) and ENG envelope (in yellow) computed with

The obtained eENG was classified by means of the SVM algorithm (5)(6)(7) (Dreiseitl and Ohno-Machado, 2002) [55] without feature extraction as describe in Section 3.1.2. The same algorithm was also applied to classify the muscular recordings.

5.2.4 Spike Sorting

Spike sorting is performed by means of three main steps, namely spike detection, feature extraction and pattern recognition. Spike sorting is typically adopted for identifying the activity of different neurons inside a neural recording and is described in the following subsections. This approach was also used in [68] and [66] to analyse neural recordings respectively from animals and from an amputee subject. In this paper, the standard classification method (i.e. the SSA) will be applied to the recorded ENG signals for comparing the results with the proposed method, based on eENG and SVM. SSA will also be applied to synthetic recordings in the database [69] available at [71] for studying classification performance over the number of classes.

Spike detection algorithm has the purpose of identifying the presence of neural activity in-side a recording. Since it is paramount to increase the performance of the SSA, it is necessary to increase the number of detected Real Positive (RP) and reduce the number of False Positive (FP). Amplitude thresholding can be an easy and computational efficient method to discriminate spikes from the noise. The threshold is generally calculated by the SD of the background noise multiplied by a factor N in the range between 3 and 5 [72]. Whenever a sample of the recording overtakes the threshold, a window extracts the action potential and stores this information for the subsequent steps of the SSA. Based on the quality of the recording, a simple amplitude threshold cannot be sufficient. Increasing the value of N can be useful for noisy recordings, because it reduces the number of FPs but it also reduces the number of RPs. Conversely, decreasing N allows increasing the number of RPs at the cost of higher FPs. It is possible to increase the performance of the detector by means of specific algorithms that enhance the presence of the spikes by reducing the noise in the recording. They include, for example, the Non-Linear Energy Operator [73] that takes into account the instantaneous amplitude

and frequency of the signal, the MMA [70], wavelet denoising [68] that filters the signals by means of scaled mother wavelet and matched filters [74]. By analogy with our choice for eENG, the MMA has been used for the spike detection algorithm with the same parameters adopted in Section 5.3.1.

Feature extractions aims at separating the activity of different neurons by means of the traits of the detected action potentials, thus reducing the data dimensionality and algorithm complexity. Defined as S the number of samples of a detected spike, a set of M representative feature are found and used to discriminate between the different action potentials. It has been heavily demonstrated that a few repertoires of unique features are necessary to reach high levels of classification accuracy both for muscular and neural recordings [30]- [72]. Two set of features have been derived from the action potentials, temporal and wavelet ones. The temporal features descend from the first and second derivative of the action potentials, calculated as

$$\begin{cases} FD(i) = s(i) - s(i - 1) \\ SD(i) = FD(i) - FD(i - 1) , \end{cases} \quad (19)$$

where s is the action potential and FD and SD are, respectively, its first and second derivative. The maximum of FD and the maximum and the minimum of SD have been selected as features [75]. We opted for this set of features for the straightforward computation and for their use in implantable device for peripheral neural recordings [76]. Wavelet decomposition was performed to calculate the wavelet coefficients of the action potentials. The mother wavelet Symlet was chosen due to its similarity with the shape of the action potentials [77]. The action potentials were decomposed into 3 levels in order to obtain one subset of approximation coefficient and 3 subsets of detail coefficients. The coefficients that could allow a better discrimination are selected by means of the Lilliefors modification of the Kolmogorov-Smirnov test [78]. It is calculated as

$$L(i) = \max |F(i) - G(i)| , \quad (20)$$

where $F(i)$ is the cumulative distribution of the i -th wavelet coefficient and $G(i)$ is the Gaussian distribution with the mean and variance of the i -th data. The three

coefficients with the highest value of $L(i)$ were selected as features for the classification algorithm.

Regarding the PR step, for comparison purposes, SVM has been chosen and used to classify ENG signals processed with the SSA. The algorithm is described in section 3.1.2.

5.2.5 Data Analysis

The ultimate goal of this study is to demonstrate that is possible to decode user intention by means of pattern recognition algorithms applied to the eENG and, consequently, pave the way to closed-loop hand prosthesis control grounded on ENG signals. To do that, neural recordings from the amputee subject were analysed in two ways. Firstly, the eENG was extracted from the recorded signals as described in section 5.1.3 and the SVM was applied to classify the gestures. Afterwards, the SSA explained in section 5.1.4 was applied to the same recorded neural signals. A comparison with the eENG patten recognition was then carried out. To be sure that the recorded information from the nerves actually represented the two different gestures little finger and open hand, the simultaneously acquired muscular signals were processed and classified by means of the SVM [20]. In this way, we set the accuracy reference for the neural recordings. Moreover, since the sEMG pattern recognition is a well-established procedure, an accuracy very close to 100% is expected for the two-gesture classification. Feature extraction was avoided both for sEMG and for eENG data. The choice to avoid feature extraction generates large datasets, therefore a 200 Hz downsampling was applied to all sEMG and eENG datasets. Although we may experience some loss of information, 200 Hz was empirically determined representing a good trade-off between evaluation time to train the algorithm and the classification performance. The optimization strategy used to train the SVM algorithm is the same described in section 3.1.3. sEMG data were thresholded using an amplitude of 0.05 mV in order to discard all the samples that are not related of a muscle activation. eENG, instead, was not thresholded since the computation of the envelope already removed all the samples not involved in the electrical activity risen from the movement intention. The accuracy of the PR based on eENG and SSA was measured and compared with the defined muscular reference. The accuracy is intended as the correct classification ratio. Recordings

of neural data were limited to one subject and two gestures. Therefore, a further analysis was performed in order to evaluate the classifier performance evolution with the increase of the number of gestures. For SSA, simulated neural recordings were used and performance up to 10 classes were evaluated. We assume that each class is representative of a gesture similarly to what was presented in [66]. The signals were segmented in order to have the same number of action potentials for each synthetic recording (i.e. 4000). The detected action potentials were randomly divided into a Training and a Testing Set with the same number of action potentials. Data were processed following the steps described in Section 5.1.4. In order to evaluate also the effect of the feature extraction on the algorithm performance, both temporal and wavelet features were used for this analysis. The SVM one-vs-one was coded in Matlab. The computation of the accuracy of the algorithm was repeated 20 times for each simulation. Classifier performance evolution with the increase of gestures was described through a decay curve fitted by a polynomial function of 2nd degree. The fitting constraint are defined as

$$\left\{ \begin{array}{l} D(n) = \theta_0 + \theta_1 n + \theta_2 n^2 \\ -1000 < D(n) < 100 \quad \forall n > 1 \in \mathbb{Q}^+ \\ D(2) = 0 \\ \lim_{n \rightarrow \infty} D(n) = 100 \end{array} \right. , \quad (21)$$

where $D(n)$ is the accuracy decay and n is the number of classes of the datasets.

5.3 Results

5.3.1 Algorithm performance for simultaneous ENG and sEMG recordings

Figure 47-48 report the results of algorithm accuracy for the real simultaneous ENG and sEMG recordings. As expected, the accuracy of the SVM applied to sEMG signal was very close to 100% ($99.98\% \pm 0.08\%$); consequently, the assumption of

using SMV with sEMG signals as reference for the two gestures was confirmed. The algorithm was also tested with the data in the GS. Again, a classification of $100\% \pm 0.02\%$ was reached, as shown in Fig. 47 and confirmed by the confusion matrix in Fig. 48a.

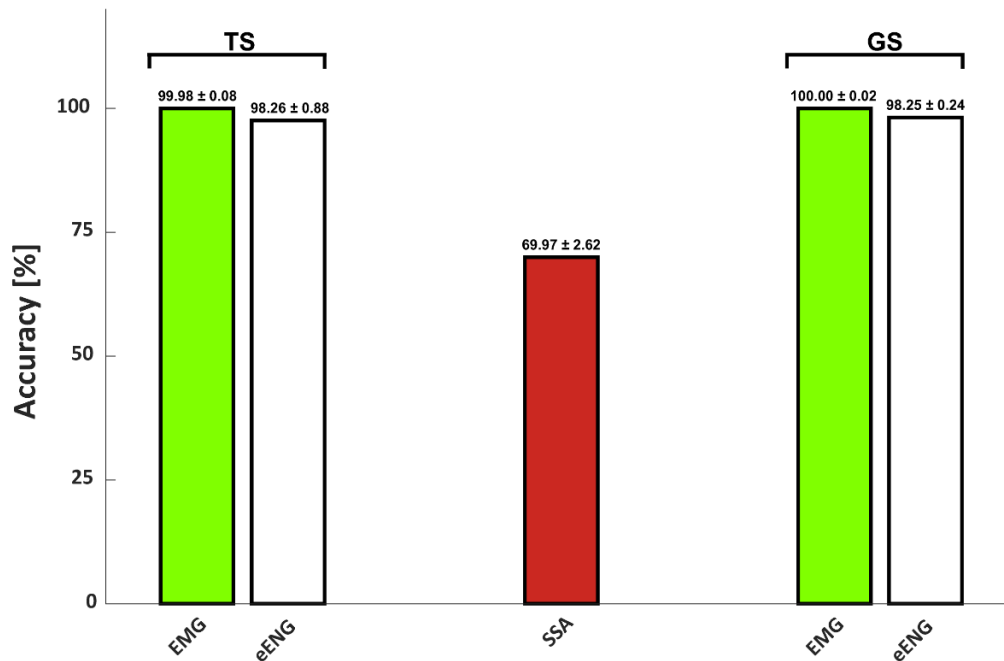


Figure 47. Performance of the algorithms applied to real data. Columns on the left indicates the results obtained with the data downsampled at 200 Hz. On the right, when all the information was used.

The SVM algorithm applied to eENG reached an accuracy of $98.26\% \pm 0.88\%$. When the SVM was tested with the GS, the accuracy slightly decreased to $98.25\% \pm 0.24\%$. This result is shown in Fig. 47. In this case, conversely to what we have found for the sEMG data, the number of FPs and False Negatives (FNs) are not 0 (respectively 1200 and 4080 in Fig. 48b).

This number is however very low with respect to the number of RPs, thus a high classification accuracy is obtained. Finally, SSA was applied to the neural recorded data from the amputee. The classification was repeated 50 times and the accuracy was recalculated each time with new TRs and TSs. Although the classes to discriminate were only 2, the overall performance of the algorithm was $69.97\% \pm 2.62\%$. These results are reported in Fig. 47 where it is clearly visible the high

discrepancy between the standard SSA accuracy (red bar) and the accuracy of the eENG (white bar).

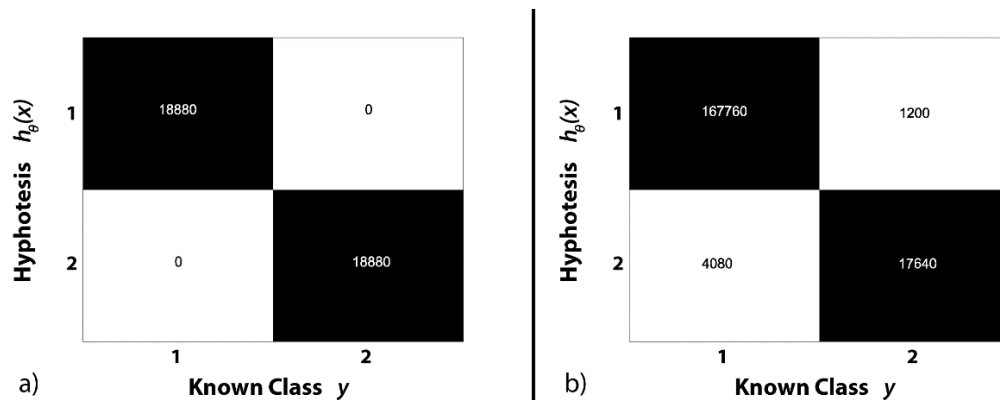


Figure 48. a) Confusion matrix for the sEMG recordings. On the main diagonal is reported the cardinality of the correct classifications; in the top left dial is reported the cardinality of the misclassified data as little finger extension (class 1) and in the bottom right dial the cardinality of the misclassified data as open hand (class 2); b) Confusion matrix for the ENG recordings. On the main diagonal is reported the cardinality of the correct classifications; in the top left dial is reported the cardinality of the misclassified data as little finger extension (class 1) and in the bottom right dial the cardinality of the misclassified data as open hand (class 2).

5.3.2 Performance decay with the number of classes

In order to study SSA, performance decay with the increase of classes, synthetic data were used. Figures 49 show the decay curves when temporal features and wavelet features are used. Again, the starting point of the curve is for 2 classes, where the Δ_{Accuracy} is 0. For temporal features the decay is slower than for the wavelet ones. The blue line is the average accuracy decay while the red dotted lines represent the percentiles. The green boxes show the standard deviation of the data. The big dots are the mean value obtained for each class, while the dark green line inside the boxes stands for the median. In particular, for 10 classes, SSA with temporal features lost 2.85% in accuracy whereas SSA with wavelet features lost 15.95%. Moreover, in case of temporal features accuracy variability is lower than the case of wavelet features; this probably due to the chose features that are more robust to the shape of the action potentials.

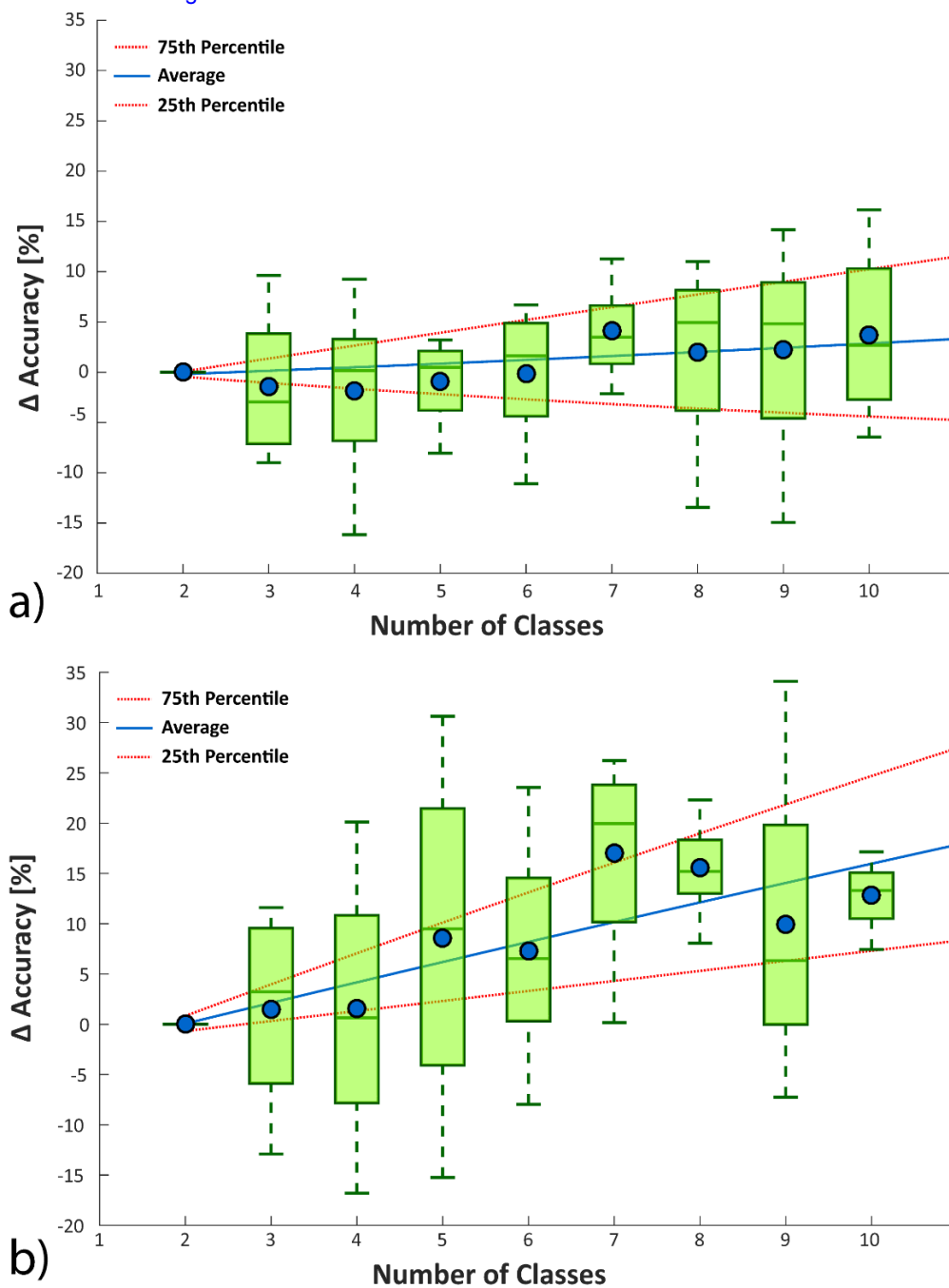


Figure 49. a) Decay of the accuracy of the SSA with temporal features. The data used to evaluate the decay curves in blue and red are presented in boxplots where the central line represents the median value; the edges of the box are the 25th and the 75th percentiles; the whiskers give the range of the data without outliers; solid markers represent the mean value; b) Decay of the accuracy of the SSA with wavelet features. The data used to evaluate the decay curves in blue and red are presented in boxplots where the central line represents the median value; the edges of the box are the 25th and the 75th percentiles; the whiskers give the range of the data without outliers; solid markers represent the mean value.

5.4 Discussion

In this chapter is presented a new approach for decoding user intention from neural recordings, with the specific purpose of controlling a prosthetic hand. Hence, online processing and low computational burden are fundamental constraints to address in addition to classification accuracy. To pursue this objective, the study was organized in three parts: (i) A muscular reference was identified in order to measure the classification accuracy of the proposed algorithm. It consisted of a SVM classifier applied to sEMG data for discriminating two classes. As shown in Fig. 47, the SVM classifier had an accuracy of 100%. (ii) A neural classifier based on eENG and SVM was developed and experimentally validated on two classes and performed offline due to the nature of the data. A comparative analysis with a SSA on the same recorded data and 2 classes was carried out. (iii) The decay of classification performance with the increase of motion classes was investigated for both the SVM and the SSA (up to 10 classes) by means of synthetic neural data. To date, neural recordings are processed by means of the SSA. The use of this algorithm to classify gestures suggests implicitly that one movement is driven by one or two neurons at most. The SSA was used in this very study to discriminate two gestures in a real context of neural prosthesis hand control. The accuracy obtained was quite low (69.97%). This because muscle activation involves a remarkable number of neurons with action potentials very different between each other. Moreover, it is likely that some neurons can be active in both the gestures; therefore, classification through SSA increases the computation burden without providing an increase in the performance. Additionally, SSA is conceived to recognize different neurons that have different spike shapes by identifying the set of features that best characterize the action potentials. This approach is hardly compatible with online application of SSA (as in the case of closed loop prosthesis control, where the features may not be well optimized) resulting in low performance of the classification algorithm. The dependence of the algorithm performance with respect to the chosen features is shown in Fig. 49. The level of decay of classification performance, for the same number of classes is different for temporal features (accuracy decay is 2.85%) and wavelet features (accuracy decay is 15.95%). As an alternative, matched filters can be used. Unfortunately, this

approach requires a great computational burden, again unsuitable for online applications. Moreover, a great variability emerges from the plots. This can be explained as follows. Each outcome was obtained by 5 different synthetic recordings for each number of classes. Since the SSA is strictly dependent on the shape on the shape of the action potentials in the recording, it may happen that, even increasing the number of the classes, the chosen set of features is more performing in that particular case than in one another. It is also worth considering that both temporal and wavelet features are not optimized for the specific recording. On the other hand, the computation of the ENG envelope allows considering the neural activation as a whole, by overcoming the specificity of the shape of the action potentials that is accounted for in the choice of the features. Additionally, the computational burden is much reduced with respect of SSA. What is really needed for decoding a gesture and driving a prosthetic hand is the amplitude of the action potential and the firing rate of neurons, as they are connected to the elicited muscular contraction. Having this in mind, in [67] the authors identified a mathematical relationship between the envelope of ENG and sEMG signals. Consequently, it is expected that the pattern recognition techniques applied to sEMG can also be successfully applied to ENG. This paper was built on this assumption and wanted to verify the feasibility of using the envelope of the neural signals and the pattern recognition algorithm typically adopted in the sEMG studies. The results on neural and muscular signals simultaneously acquired on one amputee subject were encouraging (Fig. 47). In fact, even though the number of classes was low, classification accuracy highly depended on the type of classification. The SVM applied on the eENG reaches an accuracy of 98.26% whereas with the SSA it was obtained only 69.97%. Moreover, it is worth noticing that content of information from nerve and muscle could not be always the same because of variations in available muscle or the type and location of neural electrodes. This variability can be managed by acting on the parametric relationship in (17). Parameters α and b can be optimized in order to maximize the correlation between the ENG and the sEMG envelopes. It is plausible that this relationship should be further optimized through more data, possibly taken from more subjects. Because of the bad quality of the neural recordings, the analysis on real neural signals could not be extended to a number of classes higher than 2. Consequently, the analysis of the decay of classification performance with the increase of the number of the classes was

carried out on synthetic neural signals [69]. The analysis for SSA applied on synthetic signals showed that accuracy decay depends on the type of features. In this analysis, the wavelet features behave worse than the temporal features. Hence, for 10 classes, the classification accuracy decreases by 2.85% for the temporal features and by 15.95% for the wavelet features from the starting reference point (2 classes). The choice of the features, then, becomes a not trivial task to perform and a wrong selection could lead to very low performance in terms of accuracy. Finally, the computational burden of the algorithms was calculated in terms of computational time. In hand prosthesis control, online classification is mandatory for the patient to perceive no delays between the motion intention and the actuation of the prosthesis. This aspect, along with sensory feedback, is compulsory for the complete acceptability of the device and for the naturalness of the control. A promising approach to achieve naturalness would involve TMR. In this procedure, the nerves of the stump are connected to different anatomical muscle, e.g. chest or dorsal, and use them as a biological amplifier. This way, it is possible to classify a great number of classes allowing the patient to control simultaneously different Degrees of Freedom (DoF) of a prosthetic device recording the signals from the reinnervated muscles [79]. As regards sEMG and ENG signals, it was found that the pattern recognition algorithm is able to classify an input every 1.32 μ s and 32.6 μ s respectively. Conversely, with temporal features, SSA needs 3.37 ms to classify each spike. These tests were performed in Matlab 2015b under Windows 10 OS on an Intel(R) Core(TM) i7 – 4710HQ CPU @ 2.5GHz with 16 GB RAM. These values are suitable for online application in neural control of a hand prosthesis, since the delay between the intention and the actuation of a movement should not exceed 150 ms [80].

5.5 Conclusion

In this chapter, a new method for classifying hand gestures by means of neural signals was proposed. It relies on the computation of the envelope of neural signals and the application of pattern recognition algorithms typically adopted for sEMG. The proposed method was applied to neural signals recorded from a male amputee

during an experiment of prosthetic hand control and compared to standard spike sorting algorithm. In the classification of two gestures, the comparative analysis showed that SSA reached lower classification accuracy (69.97%) than SVM applied to eENG (98.26%). Moreover, it was verified that the eENG method can be used for an online application. In fact, the pattern recognition algorithm is able to classify the inputs every 32.6 μ s. The analysis performed on 10 classes showed that classification performance of neural recordings can reach a decay of 2.85% from the reference point of 2 classes with the SSA performed with temporal features. This one was obtained by means of simulated data, arranged ad hoc to test the performance of the SSA. It is expected that the achieved results can only get worst in the case of real neural recordings. Hence, these results strongly encourage to further investigate solutions of pattern recognition applied to neural signals (without resorting on SSA) and possibly apply them to prosthesis control where gesture classification accuracy and reduced computational burden are demanding.

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Chapter 6

Conclusions

This thesis aims to provide useful insights into the choice of the suitable classifier (and its specific internal settings) for the embedded control of multi-fingered hand prostheses and to use them for designing a fully embedded control unit able to recognize the user will, via PR algorithms and to control a multi-fingered prosthetic hand exploiting a new control strategy.

The analysis performed on sEMG data recorded from 30 people with trans-radial amputation showed that the algorithm representing the best compromise between performance and computational burden is NLR (with the highest value of EOF) and showing no statistically significant difference with LDA with time domain features (Chapter 3). Therefore, this result is also more appreciable if we consider that NLR was trained and tested using raw sEMG data, demonstrating that it is possible to use non-linear classification algorithms for embedded applications. As just mentioned before, the main issue related to PR algorithms in clinical application is the robustness and reliability of the classification. Hence, a new control strategy trying to combine robustness, reliability and less cognitive effort has been proposed and included between the classification algorithm and the internal control of the prosthetic hand (section 4.2.2). Then, a complete system FW-HW-SW for the control of multi-fingered prosthetic hand has been designed, evaluated and integrated into a prosthetic device (section 4.1). Unfortunately, it is not possible to show the results of 3 months of on-line testing of the entire system carried out during the PPR2 experimentation as part of an article in preparation. Nevertheless, these first preliminary and qualitative results are encouraging. The device was tested on a male subject with a right trans-radial amputation already experienced in myoelectric control of prosthetic hands. Since the first use, the system appears

sufficiently reliable allowing to perform ADL tasks as grasping a bottle of water and drink, and even after 2 weeks the subject was still able to control all the gesture. In conclusion, the device designed and built meets the specifications, the constraints and fully functional so the objectives set can be said achieved.

Future development will regard a power consumption review of the HW FW parts by implementing energy save policies and will exploit the full system in order to perform long-term test on a pool set of trans-radial amputation. In the case of a positive outcome, the system will be re-engineered to comply with the normative concerning medical devices and will be subjected to clinical validation trials.

Long is the way but hopefully this work encourages to prosecute the studies in this field. Indeed, if the expectations are confirmed, there could be considerable repercussions regarding not only technological advances but also the limb amputation surgery, and the rehabilitation path of people with amputations.

On the other hand, Solutions based on patten recognition on sEMG suffer from the limitation that the subject cannot be provided with a natural sensory feedback. However, it has been shown that invasive solutions based on neural electrodes allow directly stimulating PNS and can be used with the twofold purpose of stimulating and recording the neural information coming from the brain to drive the residual muscles. Hence, as complementary activity, a new method for processing ENG signals specifically aimed at controlling hand prostheses has been proposed in order to overcome the limitations of traditionally adopted techniques of ENG processing resorting to sEMG PR techniques applied to the eENG. The main advantages of the proposed approach are related to the reduced computational burden and the on-line processing (which are paramount for closed loop interfaces). The obtained results strongly encourage to further investigate solutions of pattern recognition applied to neural signals (without resorting on SSA) and possibly apply them to prosthesis control. Indeed, future work will be addressed to collect additional neural data form amputees in order to more in depth analyse the performance decay of the proposed classification algorithms over the number of classes and further validate the proposed method in the online neural closed-loop control of a hand prosthesis by means of brand new human experimentation.

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

List of Publications

1. E. Noce, A. Dellacasa Bellingegni, A. Ciancio, A. Davalli, R. Sacchetti, E. Guglielmelli, L. Zollo. “*Pattern Recognition of Nerve Signal for Prosthetic Hand Control*”, “*Journal of Neuroscience Methods*”, 2017 (submitted).
2. Dellacasa Bellingegni, E. Gruppioni, G. Colazzo, A. Davalli, R. Sacchetti, L. Zollo, E. Guglielmelli. “NLR, MLP, SVM, and LDA a Comparative Analysis on EMG Data from People with Trans-Radial Amputation”, “*Journal of NeuroEngineer and Rehabilitation*”, 2017; 14:82.
3. L. Ciancio, F. Cordella, R. Barone, R. A. Romeo, A. Dellacasa Bellingegni, R. Sacchetti, A. Davalli, G. Di Pino, F. Ranieri, V. Di Lazzaro, E. Guglielmelli, L. Zollo. “Control of Prosthetic Hands via the Peripheral Nervous System”, “*Frontiers in Neuroscience*”, 2016; 10: 116.

Tesi di dottorato in Bioingegneria e bioscienze, di Alberto Dellacasa Bellingegni,
discussa presso l'Università Campus Bio-Medico di Roma in data 08/05/2018.
La disseminazione e la riproduzione di questo documento sono consentite per scopi di didattica e ricerca,
a condizione che ne venga citata la fonte.

this page is intentionally left blank

Appendix

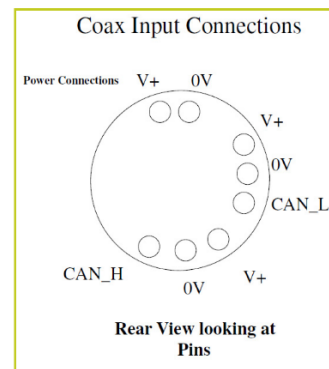
4.0 Power Supply and CAN bus Protocol

4.1 Power Supply (V+):

Power to the hand should be supplied by a 7.4V (7A Peak) bench PSU or use the battery and charger supplied. 7.4V is the normal operating voltage for the hand



Do not at any time exceed voltages of 8.3V as this will cause permanent damage



4.2 Connect to PC: Termination

A high-speed CAN bus ISO 11898-2 must be terminated on both ends with 120 Ohms. Otherwise there will be interfering signal reflections and the transceivers of the connected CAN nodes will not work.

The CAN connection is supplied un-terminated and requires to be connected to suitably terminated CAN bus.

A simple CAN Connection is shown in Fig. 5. In this example the Hand is connected to the interface by a cable that is terminated at both ends.

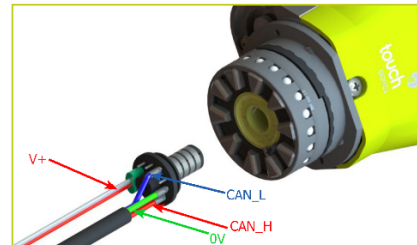


Figure. 5

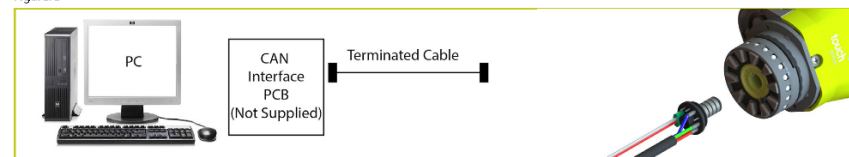


Figure 50. Power Supply and CAN bus Protocol specifications from robo-limb datasheet.

4.3 CANbus Configuration

- Standard CAN Mode has been implemented.
- Baud rate is set to 1 Mbit/sec
- Bit Time = 10, TSEG1 = 6, TSEG2 = 1, Sampling Point = 80%
- Each message has a 4 byte payload

4.4 CANbus Communication Protocol

To get the connected hand serial number:

- Send CAN message from ID 0x402 with no relevant data (i.e. 4 bytes all 0) (fig. 6).

Figure. 6

0x402	0x00	0x00	0x00	0x00
I.D	Byte 0	Byte 1	Byte 2	Byte 3

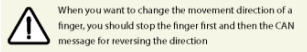
- The hand will respond with 0x403 and 4 bytes of data being the hand type "CU" or "CR" and the serial number of the hand as a UINT16, e.g. CR1234 would return "CR" (0x43, 0x52) and 1234 (0x04D2) or 0x435204D2 (fig. 7).

Figure. 7

0x403	0x43	0x52	0x04	0xD2
I.D	Byte 0	Byte 1	Byte 2	Byte 3

To drive the individual motors:

- Each digit has a unique mailbox ID, 0x101, 0x102, ..., 0x106 (thumb, index, middle, ring, little, rotator).
- Message payload is set to 2 words, that is 2x16 bits.
- High Word contains the direction of finger or thumb
 - 0 for STOP
 - 1 for CLOSE
 - 2 for OPEN



- Low WORD contains PWM level to apply to motor to adjust speed.
 - Valid values are 10 to 297

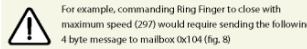


Figure. 8

0x104	0x00	0x01	0x01	0x29
I.D	Not Used	Closed	PWM Value	

Note: 297 = 0x129

Feedback from motors:

- Is sent every 20 milliseconds
 - One mailbox used per digit with ID of 0x201, 0x202 ... 0x206
- Message payload of 2 WORDS (2x16 bits)
 - High byte of High WORD contains thumb rotator switch status
 - 0 is thumb not fully palmar or lateral
 - 1 is thumb is fully palmar or lateral
 - Low byte of High WORD contains digit status:
 - 0 is stop
 - 1 is closing
 - 2 is opening
 - 3 is stalled closed
 - 4 is stalled open
 - Low WORD contains raw A/D value (12bit) of measured motor current draw.
 - Value is 12 bit, left justified
 - Divide by 21825 to convert to Amps

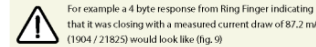
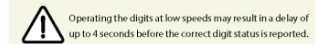


Figure. 9

0x204	0x00	0x01	0x07	0x70
I.D	Not Used	Closed	Value	

CANbus communication messages (PWM value = 277)			
ID	Data (Byte)	Comment	Digit Mapping
0x101	0 0 1 15	Idle Thumb	Thumb
0x101	0 1 1 15	Close Thumb	
0x101	0 2 1 15	Open Thumb	
0x102	0 0 1 15	Idle Index	Index
0x102	0 1 1 15	Close Index	
0x102	0 2 1 15	Open Index	
0x103	0 0 1 15	Idle Middle	Middle
0x103	0 1 1 15	Close Middle	
0x103	0 2 1 15	Open Middle	
0x104	0 0 1 15	Idle Ring	Ring
0x104	0 1 1 15	Close Ring	
0x104	0 2 1 15	Open Ring	
0x105	0 0 1 15	Idle Little	Little
0x105	0 1 1 15	Close Little	
0x105	0 2 1 15	Open Little	
0x106	0 0 1 15	Idle Rotator	Rotator
0x106	0 1 1 15	Close Rotator	
0x106	0 2 1 15	Open Rotator	

Figure 51. CAN bus settings and instruction list from robo-limb datasheet.

5.2 Battery Charger



Only use supplied Touch Bionics charger to charge battery. Depending on your location, you will receive one of the below chargers (fig. 18, or fig. 19)

robo-limb should only be charged using the Touch Bionics charger supplied.

Insert the charger (fig. 18 or fig. 19) into the power outlet. The charger will need to be inserted into the power outlet prior to connecting to the charge port. To charge, insert the charger lead connector into the charge port. A "click" should be heard on connection. If the green light is on when you first plug in the device, ensure the switch block is off.

The light display for fig. 18 is:

- Solid Red – charging
- Solid Green – fully charged or idle
- Continuous flashing red – fault condition
- Rapid flashing amber – Threshold state between charging and fully charged (should only last for 1-2 seconds)
- Continuous flashing red or green – Connection Error. Remove charger lead connector from the charge port. Ensure charger is plugged in and switched on at the mains. Re-insert the charger lead connector into the charge port.

The light display for fig. 19 is:

- Solid Amber – on standby
- Slow flashing amber – pre-charge mode
- Rapid flashing amber – Error
- Slow flashing green – maintenance charge
- Rapid flashing green – rapid charge
- Solid green – fully charged

Insert the charger lead connector into the charge port (fig.20). A "click" should be heard on connection.

Insert the charger into the power outlet.

Figure. 18



Figure. 19



Figure.20



6.0 Connecting robo-limb

6.1 Wrist Connection

The Quick Wrist Disconnect (QWD) wrist option is supplied with robo-limb (fig. 21).

Figure. 21



Note: Please contact Touch Bionics if you require a QWD female connector (fig 22). This connection is normally laminated into a structural support and is used to lock/unlock the hand into position. The CANbus or RS232 cable can then be connected as an electrical interface.

Figure. 22(not supplied)



6.2 Quick Wrist Disconnect (QWD)

The QWD is supplied by Touch Bionics. Connection or disconnection of robo-limb fitted with a QWD from the socket is completed as follows:

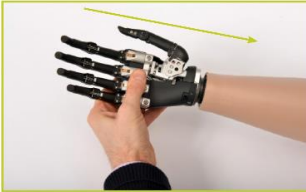
Connecting robo-limb using the QWD

- 1 Ensure the robo-limb is switched off (only applies to CANbus version).




Figure 52. Battery recharger and prosthesis connection specifications from robo-limb datasheet.

- 1 Align the QWD connection of robo-limb with the connection.
- 2 Engage the coupling.
- 3 Test the connection is fully engaged with a slight rotation.

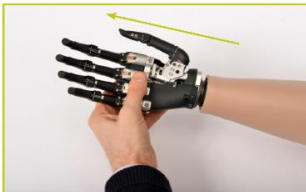


Disconnecting robo-limb using the QWD

- 1 Ensure robo-limb is switched off.
- 2 Support robo-limb in the palm of the hand.
- 3 Rotate the robo-limb through 360° in either direction until a click is heard



- 4 The robo-limb will now disengage from the socket. Support the hand and withdraw away from the socket



7.0 Adjustments

7.1 robo-limb Digit Reference Guide

Individual digits for robo-limb are manufactured as sizes 2, 3, and 5. Sizes 2 and 3 contain a small motor, while size 5 contains a larger motor. The standard digit configuration of robo-limb is outlined in the table.

robo-limb digit guide	
Digit	Size number
Thumb	5
Index	5
Middle	5
Ring	3
Little	2

7.2 Digit Maintenance




robo-limb is only compatible with Touch Bionics robo-limb digits. To install a digit, ensure that the correct digit size is selected. Remove the digit by the following steps:

Instruments required: T10 Screwdriver (not included). Contact Customer Support for ordering information

- 1 Ensure robo-limb is switched off.
- 2 Support robo-limb in the palm of the hand with the digits in the fully open position. Insert the T10 Screwdriver in the screw of the Knuckle block.
- 3 Loosen the Knuckle block screw while supporting the digit, remove the digit.
- 4 Select the appropriate sized replacement digit and follow the steps in reverse order to replace.



Figure 53. Maintenance specifications from robo-limb datasheet.

-  It is recommended that you discard the used screws as the anti-vibration pad on the screw shaft will be deformed during use. Digit screws should be replaced using the new screws provided.
-  Do not over tighten screws.
-  If there is resistance while tightening the screw check for cross threading by removing and re-inserting the screw.

7.3 Thumb Maintenance

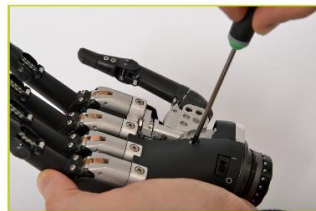
robo-limb is only compatible with a Touch Bionics' robo-limb thumb. To exchange a thumb ensure the correct size has been selected.

Instruments required: T10 Screwdriver (not included). Contact Customer Support for ordering information.

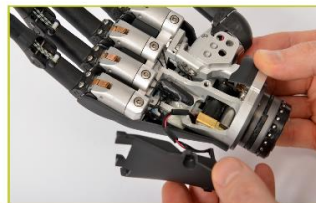
1 Fully abduct the rotating thumb.



2 Disconnect the palmar fairing using the T10 screwdriver to loosen the palmar T10 screw.



3 Gently move the palmar fairing to the ulnar side to allow access to the exposed T10 screw at the base of the thumb.



4 Using the T10 Screwdriver access the screw from the medial to lateral direction to loosen.




5 The thumb is now easily removed from the knuckle block.



6 Position the replacement thumb in the knuckle block and follow the above steps in the reverse order to reconstruct the hand. When replacing the palmar fairing, ensure wires are not pinched between the palmar fairing and the chassis.



 It is recommended that you discard the used screws as the anti-vibration pad on the screw shaft will be deformed during use. Digit screws should be replaced using the new screws provided.

 Do not over tighten screws.

 If there is resistance while tightening the screw check for cross threading by removing and re-inserting the screw.


 Touch Bionics recommends T10 Screwdriver for use with all T10 screws in robo-limb. T10 Screwdriver is available for order from Touch Bionics. Contact Customer Support for ordering information.

Figure 54. Maintenance specifications from robo-limb datasheet.

this page is intentionally left blank

Bibliography

- [1] A. L. Ciancio, et al, "Control of prosthetic hands via the peripheral nervous system," *Frontiers in neuroscience*. 2016, vol. 10, p. 116, 2016.
- [2] M. Ortiz Catalan, B. Håkansson and R. Brånemark, "Real-time and simultaneous control of artificial limbs based on pattern recognition algorithms," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, pp. 756-764., 2014.
- [3] P. Parker, K. Englehart and B. Hudgins, "Myoelectric signal processing for control of powered limb prostheses," *Journal of electromyography and kinesiology*, vol. 16, pp. 541-548, 2006.
- [4] D. Farina and O. Aszmann, "Bionic limbs: clinical reality and academic promises," *Science translational medicine*, vol. 12, pp. 257-272, 2014.
- [5] A. Roche, et al, "Prosthetic myoelectric control strategies: a clinical perspective," *Current Surgery Reports*, vol. 2, pp. 1-11, 2014.
- [6] C. Castellini, et al, "Fine detection of grasp force and posture by amputees via surface electromyography," *Journal of Physiology-Paris*, vol. 103, pp. 255-262, 2009.
- [7] M. Zecca, S. Micera, et al, "Control of multifunctional prosthetic hands by processing the electromyographic signal," *Critical Reviews™ in Biomedical Engineering*, vol. 30, pp. 4-6, 2002.
- [8] L. H. Smith, B. A. Lock and L. Hargrove, "Effects of window length and classification accuracy on the real-time controllability of pattern recognition myoelectric control," *Proceedings of the 18th Congress of the International Society for Electrophysiology and Kinesiology*, 2010.
- [9] S. Benatti, et al, "Analysis of Robust Implementation of an EMG Pattern Recognition based Control," *Biosignals*, 2014.

- [10] K. Nazarpour, "Surface EMG signals pattern recognition utilizing an adaptive crosstalk suppression preprocessor," *ICSC Congress on ma pc Computational Intelligence Methods and Applications*, 2005.
- [11] P. Dohnalek, "Human activity recognition on raw sensors data via sparse approximation," *International Conference on Telecommunications and Signal*, 2013.
- [12] A. D. C. Chan and K. Englehart, "Continuous classification of myoelectric signals for powered prostheses using Gaussian mixture models," *Engineering in Medicine and Biology Society*, 2003.
- [13] L. Zhijun, et al, "Boosting-based EMG patterns classification scheme for robustness enhancement," *IEEE Journal of Biomedical and Health Informatics*, vol. 17, pp. 545-552, 2013.
- [14] A. Cloutier and J. Yang, "Design, control, and sensory feedback of externally powered hand prostheses: a literature review," *Critical Reviews™ in Biomedical Engineering*, vol. 2, no. 2, pp. 161-181, 2013.
- [15] E. Scheme and K. Englehart, "Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use," *Journal of rehabilitation research and development*, pp. 643-659, 2011.
- [16] G. S. Dhillon, S. M. Lawrence, D. T. Hutchinson and K. W. Horch, "Residual function in peripheral nerve stumps of amputees: implications for neural control of artificial limbs," *The Journal of hand surgery*, vol. 29, no. 4, pp. 605-615, 2004.
- [17] Raspopovic, M. Bonizzato, J. Rigosa, et al, "Restoring natural sensory feedback in real-time bidirectional hand prostheses," *Science translational medicine*, vol. 6, 2014.
- [18] D. W. Tan, M. A. Schiefer, M. W. Keith, J. R. Anderson, J. Tyler, D. J. Tyler and et al, "A neural interface provides long-term stable natural touch perception," *Science translational medicine*, vol. 6, pp. 257-265, 2014.

- [19] C. M. Oddo, S. Raspopovic, F. Artoni, et al, "Intraneural stimulation elicits discrimination of textural features by artificial fingertip in intact and amputee humans," *Elife*.
- [20] A. Dellacasa Bellingegni, E. Gruppioni, et al, "NLR, MLP, SVM, and LDA a Comparative Analysis on EMG Data from People with Trans-Radial Amputation," *Journal of NeuroEngineer and Rehabilitation*, 2017.
- [21] E. Noce, A. Dellacasa Bellingegni, A. Ciancio, A. Davalli, R. Sacchetti, E. Guglielmelli and L. Zollo, "Pattern Recognition of Nerve Singal for Prosthetic Hand Control," *Journal of Neuroscience Methods*, 2017 (submitted).
- [22] M. Troncossi, "A procedure for the synthesis of upper limb prostheses. A case study: prototype manufacturing of a Novel Two-DoF Myoelectric Shoulder," in *Diss. PhD Thesis, University of Bologna*, 2006.
- [23] Ottobock, "Ottobock upper-limbprosthetics," [Online]. Available: <http://www.ottobockus.com/prosthetics/upper-limbprosthetics/>. [Accessed 27 Dicembre 2017].
- [24] RLS Steeper, "bebionic hand," RLS Steeper, [Online]. Available: <http://www.bebionic.com>. [Accessed 27 Dicembre 2017].
- [25] Touch Bionics, "Touch Bionics Ultra-Limb," [Online]. Available: <http://www.touchbionics.com/products/active-prostheses/i-limb-ultra>. [Accessed 27 Dicembre 2017].
- [26] N. Jiang and D. Farina, "Myoelectric control of upper limb prosthesis: current status, challenges and recent advances.," *Front. Neuroeng. Conference*, 2014.
- [27] J. Hijjawi, T. A. Kuiken, R. Lipschutz, L. Miller and K. Stubblefield, "Improved myoelectric prosthesis control accomplished using multiple nerve transfers.," *Plast.Reconstr.Surg*, vol. 118, pp. 1573-1578, 2006.

- [28] O. C. Aszmann, A. Roche, S. Salminger, T. Paternostro-Sluga, M. Herceg, et al, "Bionic reconstruction to restore hand function after brachial plexus injury: a case series of three patients," *Lancet*, vol. 385, pp. 2183-2189, 2015.
- [29] L. Miller, K. Stubblefield, R. Lipschutz, B. Lock and T. Kuiken, "Improved myoelectric prosthesis control using targeted reinnervation surgery: a case series," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 16, pp. 45-50, 2008.
- [30] B. Hudgins, P. Parker and R. Scott, "A new strategy for multifunctioning control," *IEEE Trans. Biomed. Eng.*, vol. 40, pp. 82-94, 1993.
- [31] M. Zardoshti-Kermani, B. Wheeler, K. Badie and R. N. Hashemi, "EMG feature evaluation for movement control of upper extremity prostheses," *IEEE Trans. Rehab. Eng.*, vol. 3, pp. 324-333, 1995.
- [32] S. Park and S. Lee, "EMG pattern recognition based on artificial intelligence techniques," *IEEE Transac. Rehabil. Eng.*, vol. 6, pp. 400-405, 1998.
- [33] K. Farry, I. D. Walker and R. Baraniuk, "Myoelectric teleoperation of a complex robotic hand," *IEEE Trans. Rob. Autom.*, vol. 12, pp. 775-788, 1996.
- [34] D. Graupe and W. Cline, "Functional separation of EMG signals via ARMA identification methods for prosthesis control purposes," *IEEE Trans. Syst. Man Cybernet.*, vol. 2, pp. 252-258, 1975.
- [35] M. Vetterli and J. Kovacevic, "Wavelets and Subband Coding," *Englewood Cliffs NJ: Prentice Hall PTR*, 1995.
- [36] M. Akay, "Wavelet Application in Medicine," *IEEE Spectr.*, vol. 34, pp. 50-56, 1997.
- [37] K. Englehart, "Signal Representation for Classification of the Transient Myoelectric Signal," *Ph.D thesis, University of New Brunswick (Canada)*, 1998.
- [38] C. S. Pattichis and M. Pattichis, "Time-scale analysis for motor unit action potential," *IEEE. Trans. Biomed. Eng.*, vol. 46, pp. 1320-1329, 1999.

- [39] S. Karlsson, J. Yu and M. Akay, "Time-frequency analysis of myoelectric signals during dynamic contractions: a comparative study," *IEEE Trans. Biomed. Eng.*, vol. 47, pp. 228-238, 2000.
- [40] P. Sparto, M. Parnianpour, E. Barria and J. Jagadeesh, "Wavelet and short-time fourier transform analysis of electromyography for detection of back muscle fatigue," *IEEE Trans. Rehab. Eng.*, vol. 8, pp. 433-436, 2000.
- [41] R. Coifman and M. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. Inform. Theory*, vol. 38, pp. 713-718, 1992.
- [42] L. Hargrove, L. Guanglin, K. Englehart and B. Hudgins, "Principal components analysis processing for improved classification accuracies in pattern recognition-based myoelectric control," *IEEE Transac. Biomed. Eng.*, vol. 56, pp. 1407-1414, 2009.
- [43] A. Chan and G. Green, "Myoelectric control development toolbox," *Proceedings of 30th Conference of the Canadian Medical and Biological Engineering Society*, 2007.
- [44] C. Bishop, *Pattern Recognition and Machine Learning*, New York: Springer, 2006.
- [45] S. Kotsiantis, I. Zaharakis and P. Pintelas, "Supervised machine learning: a review of classification techniques," *Emerging Artificial Intelligence Applications, in Computer Engineering*, pp. 3-24, 2007.
- [46] L. J. Hargrove, K. Englehart and B. Hudgins, "A comparison of surface and intramuscular myoelectric signal classification," *IEEE Trans. Biomed. Eng.*, vol. 54, pp. 847-853, 2007.
- [47] B. Lock, K. Englehart and B. Hudgins, "Real-time myoelectric control in a virtual environment to relate usability vs. accuracy," *Proceedings of MyElectric Controls/Powered Prosthetics Symposium Fredericton*.
- [48] F. Finley and R. Wirta, "Myocode studies for multiple myopotential response," *Arch. Phys. Med. Rehabil.*, vol. 48, pp. 598-601, 1967.

- [49] J. Lyman, A. Freedy and P. Prior, "Fundamental and applied research related to the design and development of upper limb externally powered prostheses," *Bull. Prosthet.*, vol. 13, pp. 184-195, 1976.
- [50] COAPT Engineering, "COAPT Complete Control," COAPT Engineering, [Online]. Available: <https://www.coaptengineering.com/>. [Accessed 28 Dicembre 2017].
- [51] A. Simon, L. Hargrove, B. Lock and T. Kuiken, "The Target Achievement Control Test: Evaluating real-time myoelectric pattern recognition control of a multifunctional upper-limb prosthesis," *Journal of rehabilitation research and development*. 2011; 48(, vol. 48, 2011.
- [52] A. Young, L. Smith, E. Rouse and L. Hargrove, "A comparison of the real-time controllability of pattern recognition to conventional myoelectric control for discrete and simultaneous movements," *Journal of neuroengineering and rehabilitation*, vol. 11, no. 1, pp. 1-5, 2014.
- [53] F. Riillo and e. al, "Optimization of EMG-based hand gesture recognition Optimization of EMG-based hand gesture recognition: Supervised vs. unsupervised data preprocessing on healthy subjects and transradial amputees," *Biomedical Signal Processing and Control*, vol. 14, pp. 117-125, 2014.
- [54] S. Dreiseitl and L. Ohno Machado, "Logistic regression and artificial neural network classification models: a methodology review," *Journal of biomedical informatics*, vol. 35, pp. 352-360, 2002.
- [55] N. Chaiyaratana, A. Zalzala and D. Datta, "Myoelectric signals pattern recognition for intelligent functional operation of upper-limb prosthesis," *Department of Automatic Control and Systems Engineering*, 1996.
- [56] C. Hsu and C. Lin, "A comparison of methods for multi-class support vector machine," *IEEE Transactions on 680 Neural Networks*, vol. 13, pp. 412-425, 2002.

- [57] LIBSVM, “LIBSVM FAQ,” [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm/faq.html#Q04:_Training_and_prediction. [Accessed 29 Dicembre 2017].
- [58] B. Ripley, Pattern recognition and neural networks, Cambridge university press, 2007.
- [59] N. Baykal and A. Erkmen, “Resilient backpropagation for RBF networks,” *Knowledge-Based Intelligent Engineering-Systems and Allied Technologies*, 2000.
- [60] Y. Ding, E. Lushi and Q. Li, “Investigation of quasi-Newton method for unconstrained optimization,” *Simon Fraser - University*, 2004..
- [61] D. Powers and D. Martin, “Evaluation from Precision, Recall and F-measure to ROC, informedness, markedness 689 and correlation,” *Journal of Machine Learning Technologies*, vol. 2, pp. 37-63, 2011.
- [62] J. Demšar, “Statistical comparisons of classifiers over multiple data sets,” *Journal of Machine learning research*, vol. 1, p. 30, 2006.
- [63] L. Hargrove, et al, “A real-time pattern recognition based myoelectric control usability study implemented in a virtual environment,” *Engineering in Medicine and Biology Society EMBS 29th Annual International Conference of the IEEE*, 2009.
- [64] NXP, “NXP mk20dx256vll10,” [Online]. Available: <https://www.nxp.com/part/MK20DX256VLL10>. [Accessed 2 Gennaio 2018].
- [65] X. Jia, M. A. Koenig, X. Zhang, J. Zhang, T. Chen and Z. Chen, “Residual motor signal in long-term human severed peripheral nerves and feasibility of neural signal-controlled artificial limb,” *The Journal of hand surgery* , vol. 5, no. 32, pp. 657-666, 2007.

- [66] P. M. Rossini, S. Micera, A. Benvenuto, J. Carpaneto, et al, “Double nerve intraneural interface implant on a human amputee for robotic hand control,” *Clinical neurophysiology*, vol. 5, no. 121, pp. 777-783, 2010.
- [67] E. Noce, L. Zollo, A. Davalli, R. Sacchetti and E. Guglielmelli, “Experimental analysis of the relationship between neural and muscular recordings during hand control,” *Biomedical Robotics and Biomechanics (BioRob)*, 2016.
- [68] L. Citi, J. Carpaneto, K. Yoshida, K. Hoffmann, K. P. Koch, P. Dario and S. Micera, “On the use of wavelet denoising and spike sorting techniques to process electroneurographic signals recorded using intraneural electrodes,” *Journal of neuroscience methods* , vol. 2, no. 172, pp. 294-302, 2008.
- [69] J. Martinez, C. Pedreira, M. J. Ison and R. Q. Quiroga, “Realistic simulation of extracellular recordings,” *Journal of neuroscience methods*, vol. 2, no. 184, pp. 285-293, 2009.
- [70] U. Rutishauser, E. M. Schuman and A. N. Mamelak, “Online detection and sorting of extracellularly recorded action potentials in human medial temporal lobe recordings, in vivo,” *Journal of neuroscience methods*, vol. 1, no. 154, pp. 204-224, 2006.
- [71] Martinez, “Database for ENG study,” [Online]. Available: <http://www135.lamp.le.ac.uk/hgr3/>. [Accessed 15 Gennaio 2018].
- [72] H. G. Rey, C. Pedreira and R. Q. Quiroga, “Past, present and future of spike sorting techniques,” *Brain research bulletin*, vol. 119, pp. 106-117, 2015.
- [73] K. Kim and S. J. Kim, “Neural spike sorting under nearly 0-db signal-to-noise ratio using nonlinear energy operator and artificial neural-network classifier,” *IEEE Transactions on Biomedical Engineering*, vol. 10, no. 47, pp. 1406-1411, 2000.

- [74] D. Farina, O. F. Do Nascimento, M. F. Lucas and C. Doncarli, "Optimization of wavelets for classification of movement-related cortical potentials generated by variation of force-related parameters," *ournal of neuroscience methods*, vol. 1, no. 162, pp. 357-363, 2007.
- [75] S. E. Paraskevopoulou, D. Y. Barsakcioglu, M. R. Saberi, A. Eftekhar and T. G. Constandinou, "Feature extracion using first and second derivative extrema (fsde) for real-time and hardware-efficient spike sorting," *Journal of Neuroscience methods*, vol. 1, no. 215, pp. 29-37, 2013.
- [76] M. Zamani and A. Demosthenous, "Feature extraction using extrema sampling of discrete derivatives for spike sorting n implantable upper-limb neural prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 4, no. 22, pp. 716-726, 2014.
- [77] A. Diedrich, W. Charoensuk, R. J. Brychta, A. C. Ertl and R. Shiavi, "Analysis of raw microneurographic recordings based on wavelet de-noising technique and classification algorithm: wavelet analysis in microneurography," *IEEE Transactions on Biomedical Engineering*, vol. 1, no. 50, pp. 41-50, 2004.
- [78] R. Quiroga, Z. Nadasdy and Y. Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering," *Neural computation*, vol. 8, no. 16, pp. 1661-1687, 2004.
- [79] D. Farina, I. Vujaklija, M. Sartori, T. Kapelner, et al, "an/machine interface based on the discharge timings of spinal motor neurons after targeted muscle reinnervation," *Nature Biomedical Engineering*, vol. 1, 2017.
- [80] T. R. Farrell and R. F. Weir, "The optimal controller delay for myoelectric prostheses," *IEEE Transactions on neural ,* vol. 1, no. 15, pp. 111-118, systems and rehabilitation engineering.
- [81] The Southampton Hand Assessment Procedure, "SHAP," [Online]. Available: <http://www.shap.ecs.soton.ac.uk/index.php>. [Accessed 4 Gennaio 2018].

- [82] P. Agnew, "Functional effectiveness of a myo-electric prosthesis compared with a functional split-hook prosthesis: A single subject experiment," *Prosthetics and Orthotics International*, vol. 5, pp. 92-96, 1981.

Figure Captions

Figure 1. Block diagram of a generic pattern recognition system based on sEMG signals.

Figure 2. Cosmetic Prosthesis.

Figure 3. Kinematic elbow Prosthesis.

Figure 4. Myoelectric Prosthesis.

Figure 5. Single degree of freedom hands. a) Ottobock myoelectric speed hand without external cover. b) Ottobock myoelectric speed hand with external cover. c) Ottobock electric Greifer terminal device.

Figure 6. Michelangelo hand prosthesis by Ottobock.

Figure 7. a) Bebionic3S prosthetic hand by RLS Steeper. b) Ultra-Limb prosthetic hand by Touch Bionics

Figure 8. a) Single Threshold control trend. b) Double Threshold control trend.

Figure 9. Proportional control trend.

Figure 10. Cyclic Selection Control Strategy representation for the selection of the desired grasp from a predefined set.

Figure 11. TMR prosthetic setup.

Figure 12. COAPT-Engineering prosthetic Setup.

Figure 13. Experimental Setup a) sEMG bracelet and NI DAQ USB 6002; b) Subject positioning and acquisition Software.

Figure 14. Graphic display of the selected gestures and of the raw recording for the six different channels at the same time for all the imagined movements of a single acquisition session from one of the subjects who took part to the experiment.

Figure 15. F1Score of Test Set (smaller boxes) and Generalization Set (bigger boxes) of 5 classes over the maximum value of variable D calculated from 30 people with trans-radial amputation. The figure also shows the trend of the mean value for both Sets. Statistical non-significance over value 5 is shown by “ns”.

Figure 16. F1Score of Test Set (smaller boxes) and Generalization Set (bigger boxes) of 5 classes over the maximum number of layers having fixed at 30 the maximum number of neurons for each hidden layer calculated from 30 people with trans-radial amputation. The figure also shows the trend of the mean value for both Sets. Statistical non-significance over value 5 is shown by “ns”.

Figure 17. F1Score of Test Set (smaller boxes) and Generalization Set (bigger boxes) of 5 classes over the maximum number of neurons for each layer. The maximum number of hidden layers calculated from 30 people with trans-radial amputation has been fixed at 5. The figure also shows the trend of the mean value for both Sets. Statistical non-significances over value 23 for and overvalue 28 for GS are shown by “ns”.

*Figure 18 F1Score values from 30 people with trans-radial amputation increasing the sampling frequency of the dataset used to train and cross validate the NLR, MLP, and SVM algorithms and 5 classes. Statistical significance is shown by “ * ”. a) F1Score values for Test Set; b) F1Score values for Generalization Set.*

Figure 19. Number of classification parameters from 30 people with trans-radial amputation increasing the sampling frequency of the dataset used to train and cross validate the NLR, MLP, and SVM algorithms and 5 classes. The y axis in logarithmic scale. Statistical non-significance is shown by “ns”.

*Figure 20. EOF values from 30 people with trans-radial amputation increasing the sampling frequency of the dataset with 5 classes used to train and cross validate the NLR, MLP, and SVM algorithms. Statistical significance is shown by “ * ”. a) EOF values for Test Set; b) EOF values for Generalization Set.*

Figure 21. F1Score values from 30 people with trans-radial amputation for MLP, NLR, SVM, tested on GS, and LDA with 5 time domain features, on a 5 classes dataset. NLR and MLP where trained using data sampled at 100Hz, while SVM using data sampled at 10Hz. Statistical non-significance is shown by “ ns ”.

Figure 22. Number of classification parameters from 30 people with trans-radial amputation for MLP, NLR, SVM, and LDA with 5 time domain features, on a 5 classes dataset. NLR and MLP where trained using data sampled at 100Hz, while SVM using data sampled at 10Hz. The y axis in logarithmic scale. Statistical non-significance is shown by “ ns ”.

Figure 23. EOF values from 30 people with trans-radial amputation for MLP, NLR, SVM, tested on GS, and LDA with 5 time domain features, on a 5 classes dataset. NLR and MLP where trained using data sampled at 100Hz, while SVM using data sampled at 10Hz. Statistical non-significance is shown by “ ns ”.

Figure 24. Schematization of the control flow. The signal processing is necessary just to filter and amplify the sEMG signal when the used sensor cannot provide it themselves.

Figure 25. Schematization of the system functions allocation. Based on the results obtained in chapter 3 the features extraction step has been skipped in lieu of a post-processing applied to the output of the classification. The control strategy interacts with the pattern recognition block (fig. 24) changing internal settings of the classifier block.

Figure 26. Direct pattern recognition control strategy. Once a class output is available the prosthetic hands moves to the corresponding gesture.

Figure 27. Finite state machine control strategy for 5 different gestures. It is possible to observe that this strategy is a hybrid between the cyclic control strategy (Fig. 10) and the direct pattern recognition one (Fig.24).

Figure 28. Structure of the Classifier Block.

Figure 30. Variant of the Classifier Block. The trigger can be achieved by means of touch sensors placed on the prosthetic hand.

Figure 30. Example of a 6 sEMG signal star diagram and the formula to evaluate the overall intensity (I_{EMG}) of the sEMG signal.

Figure 31. Finite state machine control strategy for 5 different gestures and force management.

Figure 32. Electronic hierarchical schematic of the hardware.

Figure 33. PCB layout.

Figure 34. Pattern recognition control unit PCB; a) Top view; b) Bottom view; c) Overall sizing placing the PCB near a 0.5€ coin.

Figure 35. EDATS, TRAINING & SIMULATION & UPLOAD flow chart: in this example has been decided to classify 5 hand gestures (Rest, Fist, Tip, Open and Point).

Figure 36. FIRMWARE TEST software Interface.

Figure 37. Simplified firmware flowchart.

Figure 38. Firmware Sequence Diagram

Figure 39. Building the socket from the stump of the subject who took part to the experiment.

Figure 40. Inside view of the socket of the prosthesis.

Figure 41. The Complete Prosthetic device. a) Detail of the components placed inside the Prosthesis. b) Lateral view of the Prosthesis.

Figure 42. Confusion matrix associated to the April 25 training session

Figure 43. Subject performing a complex task.

Figure 44. Block scheme of a neural-controlled prostheses approach.

Figure 45. Neural (in red) and muscular (in blue) recordings during little finger extension.

Figure 46. sEMG envelope (in blue) and ENG envelope (in yellow) computed with

Figure 47. Performance of the algorithms applied to real data. Columns on the left indicates the results obtained with the data downsampled at 200 Hz. On the right, when all the information was used.

Figure 48. a) Confusion matrix for the sEMG recordings. On the main diagonal is reported the cardinality of the correct classifications; in the top left dial is reported the cardinality of the misclassified data as little finger extension (class 1) and in the bottom right dial the cardinality of the misclassified data as open hand (class 2); b) Confusion matrix for the ENG recordings. On the main diagonal is reported the cardinality of the correct classifications; in the top left dial is reported the cardinality of the misclassified data as little finger extension (class 1) and in the bottom right dial the cardinality of the misclassified data as open hand (class 2).

Figure 49. a) Decay of the accuracy of the SSA with temporal features. The data used to evaluate the decay curves in blue and red are presented in boxplots where the central line represents the median value; the edges of the box are the 25th and the 75th percentiles; the whiskers give the range of the data without outliers; solid markers represent the mean value; b) Decay of the accuracy of the SSA with wavelet features. The data used to evaluate the decay curves in blue and red are presented in boxplots where the central line represents the median value; the edges of the box are the 25th and the 75th percentiles; the whiskers give the range of the data without outliers; solid markers represent the mean value.

Figure 50. Power Supply and CAN bus Protocol specifications from robo-limb datasheet.

Figure 51. CAN bus settings and instruction list from robo-limb datasheet.

Figure 52. Battery recharger and prosthesis connection specifications from robo-limb datasheet.

Figure 53. Maintenance specifications from robo-limb datasheet.

Figure 54. Maintenance specifications from robo-limb datasheet.

Table Captions

Table 1. Encoding the variable D.

Table 2. Mean values and standard deviation of F1Score of Test Set and Generalization Set of 5 classes over the maximum value of variable D calculated from 30 people with trans-radial amputation. The highest values per Set are highlighted in bold. See Fig. 15 for a graphic display and statistical significance.

Table 3. Mean values and standard deviation of F1Score of Test Set and Generalization Set of 5 classes over the maximum number of layers having fixed at 30 the maximum number of neurons for each hidden layer calculated from 30 people with trans-radial amputation. The highest values per Set are highlighted in bold. See Fig. 16 for a graphic display and statistical significance.

Table 4. Mean values and standard deviation of F1Score of Test Set and Generalization Set of 5 classes over the maximum number of neuron having fixed at 5 the maximum number of layers calculated from 30 people with trans-radial amputation. The highest values per Set are highlighted in bold. See Fig. 17 for a graphic display and statistical significance.

Table 5. Mean values and standard deviation of F1Score of Test Set and Generalization Set of 5 classes from 30 people with trans-radial amputation varying the frequency of the dataset used to train and cross validate the NLR, MLP and SVM classifier. The highest values per classifier are highlighted in bold. See Fig. 18 for a graphic display and statistical significance.

Table 6. Mean values and standard deviation of classification parameters from 30 people with trans-radial amputation varying the frequency of the dataset used to train and cross validate the NLR, MLP and SVM classifier and 5 classes. The highest values per classifier are highlighted in bold. See Fig. 19 for a graphic display and statistical significance.

Table 7. Mean values and standard deviation of EOF of Test Set and Generalization Set of 5 classes from 30 people with trans-radial amputation varying the frequency of the dataset used to train and cross validate the NLR, MLP and SVM classifier. The highest values per classifier are highlighted in bold. See Fig. 20 for a graphic display and statistical significance.

Table 8. Mean values and standard deviation of F1Score values, classification parameters and EOF values from 30 people with trans-radial amputation for each classifier involved in this study on a 5 classes dataset. The EOF and F1Score highest values and the lowest number of parameters are highlighted in bold. See Fig. 21-22-23 for a graphic display and statistical significance.

Table 9. Subdividing the sEMG data for the training of the force classifier.

Table 10. CAN and UART Peripheral Settings.

Table 11. Results of the electronic performance tests.

Table 12. April 25 2017 training session results table