université Paris Ouest Nanterre La Défense

UNIVERSITÉ PARIS LUMIÈRES

Ecole Doctorale 139

Connaissances, Langages, Modélisation

THÈSE

Présentée

en vue de l'obtention du

grade de Docteur

de

l'Université Paris Ouest Nanterre La Défense

en Sciences pour l'Ingénieur

Astrid RUBIANO FONSECA

Smart control of a soft robotic hand prosthesis

Contrôle intelligent d'une prothèse de main robotique souple

soutenue le 9 Décembre 2016 devant le jury composé de :

P. WENGER	Prof., Ecole Centrale de Nantes	Président
C. LASCHI	Prof., Scuola Superiore Sant'Anna, Pisa	Rapporteur
Y. AOUSTIN	Prof., Ecole Centrale de Nantes	Rapporteur
T. WALLMERSPERGER	Prof., Dr.-Ing, Technische Universität Dresden	Examinateur
N. JOUANDEAU	Asst. Prof. HDR, Université Paris 8	Examinateur
M. D'OTTAVIO	Asst. Prof., Université Paris Ouest Nanterre La Défense	Examinateur
L. GALLIMARD	Prof., Université Paris Ouest Nanterre La Défense	Directeur
O. POLIT	Prof., Université Paris Ouest Nanterre La Défense	Directeur

Laboratoire Énergétique Mécanique Électromagnétisme
LEME - EA 4416
50 rue de Sèvres - 92410 Ville d'Avray

Leme
Laboratoire Énergétique Mécanique Électromagnétisme

Acknowledgments

I express my sincere gratitude to Universidad Militar Nueva Granada Bogotá - Colombia for the financial support of my Ph.D.

I am deeply thankful to have had the opportunity to work with the Professor Olivier Polit (head of the LEME laboratory) and Professor Laurent Gallimard (head of the theoretical and numeric modeling of structures and systems team), as my advisors. They have been a constant source of knowledge, insights, and motivation, proving me valuable guidance and continuous support. Their enthusiasm will continue to influence my future researchers.

My sincere gratitude to Nicolas Jouandeau, Michele D'Ottavio, and Nabil El Korso. I appreciate our discussions. It has been especially constructive. I thank Professor Thomas Wallmersperger (Chair of Mechanics of Multifunctional Structures at the Institute of Solid Mechanics of the Technische Universität Dresden in Germany) for the wonderful and exciting stage in TU Dresden.

I would also like to acknowledge Professor Cecilia Laschi (Full Professor of Biorobotics and Rector's delegate to Research at the BioRobotics Institute of the Scuola Superiore Sant'Anna in Italy) and Professor Yanick Aoustin (full professor at Ecole Centrale de Nantes in France) for agree to be my examiners, your comments and motivation are enriched my Ph.D. thesis. I would like to thank the president of the committee Professor Phillipe Wenger (Director of Research at CNRS IRCCyN Ecole Centrale de Nantes in France) for his insightful comments and encouragement.

I acknowledge Pol Le Borgne (Managing Director at Cogitobio in France) and Sophie Longet (research coordinator at école supérieure de biomécanique appliquée à l'ostéopathie - OSTEOBIO in france), for your insights and enthusiasm. I enjoyed all our discussions about biomechanics and all the sessions in your motion capture laboratory, without your influence, this project would not have been possible. Very spacial thanks all the master's students that worked with us: Aurélien Scola, Adrian Lyonnet and Quentin Bomberault; my colleagues: Alexander Nikitin, Jonas Maruani, Zhaoyi Jiang, Thi Huyen Cham Le, Mohand Ouarabi, Patrick Ribeiro, Romain Pawelko and El Hadji Boubacar Seck.

I express my gratitude to the Université Paris Lumières UPL for the financial support through the ProMain project. Furthermore, I acknowledge the Université franco-allemande Deutsch-Französische Hochschule for the financial support of my visit to the Institute of Solid Mechanics at TU Dresden.

Last but not the least, I would like to thank my family, my husband, my parents and my brother for supporting me spiritually throughout writing this thesis and my life in general, you are my source of inspiration. Finally, very special thanks to all who contributed in my Ph.D. thesis.

Summary

Robotic hands have been created with the aim to grasp objects and to perform precision tasks. Many of these hands were made of rigid materials, so those have rigid joints and structures. As a consequence, the prehension is non-compliant.

Some of these robotic hands are used in the industrial environment as final effectors, and some other as hand prosthesis. Currently, hand prostheses offer patients a partial solution to objects manipulation.

Soft robotics is a new research field of robotic that deals with flexible joints, structures, and mechanisms. Soft-body robotics systems are expected to allow flexible interaction between robots and grasped objects. This particular characteristic becomes necessary in hand prostheses to improve the manipulation of unknown objects.

The target of this work is to develop a new soft robotic hand prosthesis [1][2] called ProMain Hand, which is characterized by: (i) flexible interaction with grasped object (ii) and friendly-intuitive interaction between human and robot hand.

Flexible interaction results from the synergies between rigid bodies and soft bodies, and actuation mechanism. The ProMain hand has three fingers, each one is equipped with three phalanges: proximal, medial and distal [3]. The proximal and medial are built with rigid bodies, and the distal is fabricated using a deformable material. The soft distal phalange has a new smart force sensor [4], which was created with the aim to detect contact and force in the fingertip, facilitating the control of the hand.

Furthermore, the finger has two degrees of freedom (DoF) driven directly by the actuator and a flexible link that allows additional rotations in all directions when it gets in contact with objects during grasping. It enhances mobility and adaptability during prehension and manipulation to unknown objects.

The actuation mechanism is made of rigid, flexible and smart materials [5]. It is under-actuated and driven by tendons, *i.e.* only one actuator activates the whole finger, and the motor is coupled to the finger mechanism through flexible wires [6]. Tendons are distributed in two drive groups, one for the proximal phalange and the other for the distal phalange.

On the other hand, a friendly intuitive human-hand interaction is developed to facilitate the hand utilization [7][8][9]. The human-hand interaction is driven by a controller that uses the superficial electromyographic signals measured in the forearm employing a wearable device.

The wearable device called MyoArmband^{TM}is placed around the forearm near the elbow

joint. Based on the signals transmitted by the wearable device, the beginning of the movement is automatically detected, analyzing entropy behavior of the EMG signals through artificial intelligence. Then, three selected grasping gesture are recognized with the following methodology: (i) learning patients entropy patterns from electromyographic signals captured during the execution of selected grasping gesture [10][11], (ii) and performing a support vector machine classifier, using raw entropy data extracted in real time from electromyographic signals. All these processes are carried out in real time. The above methodology was tested in a motion caption laboratory and validated using Myoarmband^{TM}, out of laboratory conditions.

Once, the desired motion by the patient is recognized, the soft robotic hand performs the movement driven by a controller that uses object contact detection and a hybrid force-position feedback. The controller is designed based on: (i) the parametric identification of the actuator, (ii) the kinematic and dynamic model of the finger, (iii) and the dynamic behavior of the applied force during experiments. Furthermore, the controller architecture is based on hybrid control theory, which superposes the force closed loop to the position closed loop. Both closed loops are proportional-integrate-derivative controllers tuned to reduce setting time. The control system was validated experimentally performing precision grasping of real objects.

Contents

List of Tables

List of Figures

Introduction

The human dexterous manipulation has long been a question of keen interest in a wide range of fields such as medicine, biomechanics, and robotics. In the area of robotics, the study of dexterous manipulation has led to the development of devices aiming at the automation of industrial processes. Principally, robotic manipulators and grippers have been introduced in the last two decades seeking to: (i) replace human intervention in dangerous environments or speed it up; (ii) improve accuracy in productive processes.

However, recent progress in the field of dexterous robotic manipulation has led to a renewed interest in the development of robotic devices for rehabilitation or improvement of human skills. Studies over the past two decades have provided valuable information on the considerations for development of robotic hands. Two main research axes can be clearly defined around the development of robotic hands: (i) the control and the operation; (ii) the mechanic and the actuation.

The first axis concerns the development of control and automation strategies to improve robots'performance, and the design of facilities to operate robotic devices. The second axis concerns the analysis, improvement, and design of mechanisms and actuation technologies, and the study of materials. In all cases, the target is the enhancement of dexterous manipulation skills of robotic hands.

The current commercial robotic hand prostheses: Bebionic Hand 3 [13], i-limb Ultra [14] and Michelangelo [15] are rigid and not compliant, and follow an approach that attempts to completely mimic the human hand, neglecting its functionality which is the primary target in the rehabilitation of amputated patients. Consequently, the development of new adaptable devices using a bio-inspired approach taking into account not only the mechanics of the hand but also the functionality is crucial for the progress of robotic hand prostheses. Thus, the soft robotic constitutes a fascinating approach developing robotic hand prostheses [16, 17].

The main advantage offered by the soft robotics in the dextrous manipulation is the adaptation capacity, which allows the robotic hand to be used in several uncertain grasping situations. However, it implies to undertake several challenges [18]. One of the greatest challenges concerns the control of the soft robotic hand considering the unpredicted behavior that arises from the utilization of smart and flexible materials. Additionally, there is a critical need to develop suitable solutions for human-machine interface, to allow amputated persons to interact with the soft robotic hand prosthesis.

Consequently, the central topic of this thesis dissertation is the development of a smart control strategy for a soft robotic hand prosthesis, which includes a human-machine interface to operate our soft robotic hand. Our approach takes into account: (i) the definition of the relation between upper limb motion and superficial ElectroMyoGraphic signals (sEMG); (ii) Support Vector Machine (SVM) to classify grasping gesture patterns from sEMG; (iii) the development of a grasping recognition methodology using the wearable device MyoArmband^{TM}; (iv) a hybrid force-position control strategy for our soft robotic hand.

This Ph.D. thesis dissertation discusses a human-machine interface for the control of a soft robotic hand. Thus, we highlight the following contributions: (i) the proposed approach takes into account sEMG signals produced during movement; (ii) the performance of five features during flexion and extension motion is analyzed considering movement positions and speed variations; (iii) the proposition of a criterion to select the best features; (iv) the proposition of SVM as a motion classifier based on sEMG signals produced during movements; (v) an effective classification scheme to recognize upper limb gestures; (vi) a new and accurate movement detection model that allows real-time detection and identification of two hand movements. The entropy and flow entropy measurements permit an automatic movement inception detection and prehension patterns recognition, using the SVM. (vii) development of an interface suitable to work out of laboratory conditions, which is able to identify two hand gestures, using a wearable device; (viii) development and characterization of a new tactile smart sensor which allows to measure the applied force over its soft cover, its performance is evaluated in different operational conditions; (ix) a hybrid control model is proposed aiming to drive, finger position and fingertip force. This thesis dissertation is addressed in five chapters as follows:

1. Relationship between upper limb movements and myoelectric signals
2. Movement classification based on myoelectric signals
3. Embedding grasping recognition
4. ProMain- I robotic hand control
5. ProMain- II Soft Hand

In the first Chapter, we present the methodologies developed in the literature for upper limb movements classification, which are based on sEMG signals. The information is broken down by type of movement (flexion extension, radial ulnar, pronation supination) including mainly hand gestures, studied joints (elbow, wrist, and fingers), muscles and subjects. We identify the most used muscles for each kind of motion, considering amputated and healthy patients.

Moreover, we cluster the methodologies of classification considered in researches studies carried out up to now, grouping the information in four categories: (i.) single feature and single classifier; (ii.) single feature and multiples classifiers; (iii.) multiple features and single classifier; (iv.) multiple features and multiples classifiers. In each category we detail the obtained

accuracy of classification and the kind of patients considered. Then, we focus on the features examined in literature, in which we select five features to be analyzed. We also propose an upper limb kinematic model for the upper limb. The model calculates the angles formed by joints, and the rotation speeds.

Based on the proposed kinematic models, and the selected features, we analyze the influence of the kinematic characteristics over the features from sEMG signals, in order to understand how the features are affected by the way in which the movement is performed. Then, we propose a criterion to select the best couple of features. The criterion takes into account the euclidean distance to measure the separation between the cluster of features; higher the distance, higher the probability of good separation. We show that the features selection has an important role in the classification scheme.

In the second Chapter, we present a powerful tool for pattern classification, socalled SVM. The classifier is addressed towards upper limb motion classification, and we present a summary of the previous research studies key point upper limb motion classification using SVM. Furthermore, we introduce a pattern classification strategy, based on SVM, in which we consider an hyperplane classifier with an optimal separation margin. We describe the non-linear SVM using soft margin and kernel functions, and we detail the solution of the optimization problem.

We introduce this classification model for upper limb motion based on sEMG signals. The classification model is based on non-linear SVM, using kernel functions. In order to assess this classification scheme, we test the model by performing an experiment using a motion capture laboratory, that includes tracking motion and capturing sEMG signals.

In the third Chapter, we present the grasping recognition model using a wearable human interface. We select the MyoArmband^{TM}wearable device. We summarize the research studies based on MyoArmband^{TM}in order to present the state of the art. As consequence, a new movement detection model is introduced, merging the proposed concepts and models about the relationship between sEMG and upper limb kinematics presented in Chapter 1 and the experimental results obtained under laboratory conditions, presented in Chapter 2. Our model allows real-time detection and identification of two hand movements, allowing a user to interact in a natural way with the ProMain[1] robotic hand prostheses.

A suitable and smart interpretation of sEMG signals allows the identification of subject's desired actions. Three main stages are required to extract information embedded in sEMG signals as follows: (i) modeling of sEMG signals; (ii) identifying movement inception; (iii) recognition of prehension patterns.

In the fourth Chapter, we present the ProMain-I hand control. In the first part, the hand characteristics are detailed including a new soft fingertip force sensor and its experimental characterization considering two different flexible materials.

In the second part, we propose the robotic finger parametric model which includes: (i) a parametric model of the actuator; (ii) experimental actuator identification and its control sys-

[1] ProMain-I and ProMain-II hands.

tem; (iii) a robotic finger identification, including position and fingertip force. Based on these models, we describe the optimized finger controller, taking into account motion and force specifications, in order to establish the hybrid force position controller. Moreover, the simulation of the overall system is presented. Subsequently, the final section illustrates the ProMain-I hand software interface, and the grasping gesture achieved by the prototype of the ProMain-I hand.

In the fifth Chapter, we present the new ProMain-II Soft Hand, and we introduce its characteristics, specifically the new soft robotic finger and the new soft link. Moreover, we carry out an experiment to track joints position evaluating the soft link bending behavior and its adapting capacity when finger gets in contact with an obstacle.

Chapter 1

Relationship between upper limb movements and myoelectric signals

1.1 Review

The ElectroMyoGraphy (EMG) signal is a measure of the electrical activity produced by muscles during the movement. These signals are formed via the superposition of individual muscle action potentials, generated by irregular discharges of active motor units in a muscle fiber [19]. As a consequence, it is possible to obtain this biological signal by measuring the superficial voltage over Surface ElectroMyoGraphy (sEMG) or over the intramuscular ElectroMyoGraphy (iEMG), using electrodes. Therefore, it is possible to estimate motion through the analysis of these signals [20]. However, due to the nature of the EMG the estimation is not a simple task [21]. sEMG signals typically have amplitudes range from 0 to 10mV (peak-to-peak) or 0 to 1.5 mV (root mean square) [22]. The bandwidth ranged from 0 to 500 Hz with a mean spectrum frequency of 70 - 130 Hz [23].

Body movements are produced thanks to the interaction of joints. Regarding the upper limb motion, different joints are involved in the movements of shoulder, elbow, wrist, fingers and hand gestures. Specifically, the fingers joints are composed of: Metacarpophalangeal (MP), Proximal InterPhalangeal (PIP) and Distal InterPhalangeal (DIP). Each joint has different degrees of freedom as a consequence, different types of motion. Then, the elbow joint has flexion and extension, as shown in Figure 1.1, the wrist has flexion and extension as illustrated in Figure 1.2, and also it has pronation, supination, abduction, adduction. Finally the finger joint has flexion and extension (see Figure 1.3).

(a) Flexion. (b) Extension.

Figure 1.1: Elbow Motion.

(a) Flexion. (b) Extension.

Figure 1.2: Wrist Motion.

(a) Flexion. (b) Extension.

Figure 1.3: Finger Motion.

Some relevant studies have been carried out to automatically recognize upper limb movement based on sEMG signals, through the measurements of the electrical activity produces in the muscles involve in each movement. In this work, we will present a summary of previous works, as shown in Table 1.2. The citation concerns authors with recognition accuracy higher than 70 %, using superficial electrodes placed on the muscles, the information are broken down by the movements classified, and considering healthy and amputated subjects categorized into:

(i) Flexion/extension,

(ii) Radial/ulnar and

(iii) Pronation/supination.

Furthermore, Table 1.1 presents the abbreviations that will be used throughout the thesis dissertation.

Table 1.1: Upper limb muscle acronyms.

Abbreviation	Description
BWF	Biarticulate Wrist Flexor
TWF	Triarticulate Wrist Flexor
BWE	Biarticulate Wrist Extensor
BB	Biceps Brachii
TB	Triceps Brachii
PR	Pronator
BR	BrachioRadialis
ECR	Extensor Carpi Radialis
FCR	Flexor Carpi Radialis
ECU	Extensor Carpi Ulnaris
FCU	Flexor Carpi Ulnaris
FDS	Flexor Digitorum Superficialis
ED	Extensor Digitorum
PL	Palmaris Longus
FCR	Flexor Carpi Radialis
AC	electrodes Around the Circumference of the upper forearm
EPB	Extensor Pollicis Brevis
FPL	Flexor Pollicis Longus
EIP	Extensor Indicis Proprius
EDQP	Extensor Digiti Quinti Proprius
FDP	Flexor Digitorium Profundus
FPL	Flexor Pollicis Longus
H	Healthy Subject
A	Amputated Subject

Flexion and extension are present in different joints of the upper limb such as the elbow, wrist, and hand fingers. The research on elbow flexion and extension classification [19, 21, 24] typically have considered seven muscles, some of them are located in the arm such as: (i) BWF, (ii) TWF (iii) and BWE. And others are situated in the forearm like: (i) BB, (ii) TB, (iii) PR, (iv) and BR.

Previous studies [24] have reported that the analysis of sEMG from four muscles: BB, TB, PR and BR considering healthy subjects. Similarly, [21] studied BB and TB, but with amputated subjects. By comparing the research [24] and [21], both works measured the same two muscles

BB and TB, furthermore, evaluating their methodology with healthy and amputated subjects; both works reached high accuracy. Thus, the remaining muscles of amputated patients play an important role in the movements classification, as shown in[21].

Table 1.2: Upper limb movements classification.

Movement	Joint	Muscles	Subject	Reference
Flexion and extension	Elbow	BWF, TWF, BWE	11H	[19]
		BB, TB, PR, BR	6H	[24]
		BB, TB	12A	[21]
	Wrist	ECR, FCU, ED, PL	10H	[25]
		FCR, FCU, FDS, BR	7H	[26]
		FCR, FCU, ED, BR	1H	[27]
		FCU, ED	30H	[28]
		ECR, FCR, ECU, ED	18H	[29]
		FCR, FCU, ED, BR	1A	[12]
	Fingers	FCR, FCU, FDS, BR	6H	[26]
		FCR, FCU, FDS, BR	7H	[30]
		FCR, FCU, ED, BR	1H	[27]
		AC	6H	[31]
		AC	3A	[32]
		FCR, FCU, ED, BR	1A	[12]
		AC	10H&6A	[33]
Radial and ulnar	Wrist	ECR, FCU, ED, PL	10H	[25]
		FCR, FCU, ED, BR	1H	[27]
		ECR, FCR, ECU, ED	18H	[29]
		BWF, TWF, BWE	11H	[19]
		FCU, ED	30H	[28]
		FCR, FCU, ED, BR	1A	[12]
Pronation and supination	Wrist	ECR, FCU, ED, PL	10H	[25]
		ECR, FCR, ECU, ED	18H	[29]
		FCR, FCU, ED, BR	1H	[27]
		BB, TB, PR, BR	6H	[24]
		FCR, FCU, ED, BR	1A	[12]
		BB, TB	12A	[21]

Regarding the wrist flexion and extension, research works had considered the follow eight different muscles in forearm: (i) ECR, (ii) FCR, (iii) FCU, (iv) ECU, (v) FDS, (vi) ED, (vii) PL, and (viii) BR.

The muscles most studies in literature have been focused in FCU [12, 25–28], ED [12, 25, 27–29], FCR [12, 26, 27, 29] and BR [26, 27], due to its external location and size. [29] pointed

out two muscles: ECR and it's antagonist muscle FCR. Some authors have also examined muscles acting in opposition, for instance, FCU and ED [12, 25, 27, 28]; FCR and BR [12, 26, 27], even with amputated subjects [12].

Finally, concerning fingers flexion and extension, the considered muscles are: (i) FCR, (ii) FCU, (iii) FDS, (iv) BR, (v) ED, (vi) and PL. Furthermore, FCR and FCU are the most used muscles[12, 25–30], even while other authors are placing the electrodes around the circumference of the forearm [31–33][1]. BR can also be used [26, 27, 30]

On the other hand, radial and ulnar movements are present in the wrist joint. The muscles most used in their movements are: ED [12, 25, 27–29] and FCU [12, 25, 27, 28], similarly for the wrist pronation and supination movements, being ED [12, 25, 27, 29] and FCU [25, 27, 29], furthermore, taking into account FCR [12, 27, 29] and BR [12, 24, 27].

In summary, FCU have been the most used muscle to identify: (i) flexion/extension of the wrist and hand fingers, (ii) radial/ulnar and pronation/supination of the wrist. Then, FCR for flexion/extension and pronation/supination of the wrist, and finally ED plays a meaningful role for radial/ulnar and pronation/supination motion of the wrist. Besides, most of the researchers have considered antagonist muscles such as FCU and FCR and their agonist muscles such as, ED and BR.

Additionally, we considered that hand gesture classification is an important aspect to take into account. Thus, we summarize in Table 1.3 the research studies that are focused on this field.

Some of those studies are considered exclusively healthy subject, analyzing the following gestures: H keeping the hand straight [19], H tip, H hook, H lateral, H point and H spherical [34], or only amputated subjects, considering precision pinch gesture [32].

The open hand movement are studied by authors, in which the most used muscle is ED [12, 25, 27, 29], followed by FCU [12, 25, 27] and FCR [12, 27, 29]. In the case of close hand movement, the muscles most used are: FCR [12, 27, 29], FCU and ED [12, 27]. With reference to the gesture hand during resting condition, the muscles most used are: FCU and ED [25, 28], having the same for key grip gesture adding FCR and BR [12, 27]. Finally, for cylindrical and tripod grasping, the muscles most used are FDL and FDS [32, 34].

Briefly, FCU and ED are the muscle most used to identify: (i) open, (ii) close, (iii) rest condition, (iv) and key grip. In fact, they are the same two most muscles used for the radial/ulnar, pronation/supination movement of the wrist.

[1]We suppose that correspond to FCU, ECU, ED, PL, FCR and ECR muscles, even if there are not clear established in these researchers.

Hand gesture	Muscles	Subject	Reference
Opening	ECR, FCU, ED, PL	10H	[25]
	ECR, FCR, ECU, ED	18H	[29]
	FCR, FCU, ED, BR	1A	[12]
	FCR, FCU, ED, BR	1H	[27]
Closing	ECR, FCR, ECU, ED	18H	[29]
	FCR, FCU, ED, BR	1A	[12]
	FCR, FCU, ED, BR	1H	[27]
	ED	1H1A	[35]
Resting	AC	3A	[32]
	ECR, FCU, ED, PL	10H	[25]
	BB, TB, PR, BR	6H	[24]
	FCU, ED	30H	[28]
	FPL, FDS	6H	[34]
Keeping the hand straight	BWF, TWF, BWE	11H	[24]
Precision pinching	AC	3A	[32]
Tip griping	FPL, FDS	6H	[34]
Stretching one's hand	AC	3A	[32]
Key griping	FCR, FCU, ED, BR	1A	[12]
	FCR, FCU, ED, BR	1H	[27]
Cylindrical griping	FPL, FDS	6H	[34]
	AC	3A	[32]
Hook griping	FPL, FDS	6H	[34]
Lateral griping	FPL, FDS	6H	[34]
Point griping	FPL, FDS	6H	[34]
Spherical griping	FPL, FDS	6H	[34]
Tripod griping	FPL, FDS	6H	[34]
	AC	3A	[32]

Table 1.3: Hand gesture classification.

Methodologies			Reference
Filter	Feature extraction	Classification	[12, 26, 29, 30, 32] [24, 27, 36]
	Feature extraction & Feature projection		[25]
Filter & Data segmentation	Feature extraction		[19, 21, 28, 34, 37]
	Feature extraction & Feature reduction		[33]

Table 1.4: Movement classification methodologies.

Once the movements and the involved muscles are analyzed, we examine the methodology

that researchers studies have proposed in order to identify upper limb motion based on sEMG signals. Thus, Table 1.4 presents an overview of the applied techniques in the analyzed studies.

The most used methodologies in order to classify movement, follow the next steps: (i) filtering signals (ii) extracting features from signals and (iii) classifying the movement [12, 24, 26, 27, 29, 30, 32, 36]. Some authors added one step after the feature extraction that consists in carrying out a projection [25] or reduction features [33]. Feature extraction consists in the obtain relevant information about the movement which is taken from sEMG signals, that is developed in section 1.3.1.

There are several methods for features extraction and also for classification. Therefore, with the aim to compare the different types of features and classifiers, we proposed a cluster of the state of the art as follows:

(i.) Single feature and single classifier summarized in Table 1.6,

(ii.) Single feature and multiples classifiers summarized in Table 1.7,

(iii.) Multiple features and single classifier summarized in Table 1.8,

(iv.) Multiple features and multiples classifiers summarized in Table 1.9.

Furthermore, Table 1.5 presents the abbreviations used throughout the present section.

There are several works that had been chosen to use only one feature with one classifier, as is shown in Table 1.6. [25] mentions that their methodology reached 97.4% recognition accuracy, using a WAVET to extract a feature, reducing the feature space through LDA and finally classifying with a MLPT.

[19] presents a performance comparison between different features, which are classified using a SVM. The features considered were: MAV, RMS, WL, ZC, VAR, SSC, WAMP, AR2, AR6, PSD, FMN, FMD. The highest accuracy recognition was found considering WL reaching 96% following by MAV and RMS with 95%.

Likewise [31], identified WL as best feature taking into account a MLPT, reaching 92.41% recognition accuracy. In the same way, [30] used ANN, testing the two features ICA and FD, finding that the best feature was ICA with 96%. ICA feature is also tested by [26] considering a SVM. Thus, the accuracy recognition increases to 99.34%. Moreover, [32] considers the same classifier SVM and PCA feature, validating their model with three amputated subjects. As a result, 95% are correctly classified.

Table 1.5: Abbreviations feature extraction and classifier.

Abbreviation	Description
WAVET	Wavelet Packet Transform
MAV	Mean Absolute Value
RMS	Root Mean Square
WL	Waveform Length
ZC	Zero Crossings
VAR	Variance
SSC	Slope Sign Changes
WAMP	William Amplitude
AR	Autoregressive Coefficients
AR2	Autoregressive Coefficients of second order
AR6	Autoregressive Coefficients of sixth order
PSD	Power Spectrum
FMN	Frequencies of Mean
FMD	Frequencies of Median
ICA	Independent Component Analysis
FD	Fractal Dimension
DMAV	Difference Absolute Mean Value
DVARV	Difference Variance Value
DASDV	Difference Absolute Standard Deviation Value
MYOP	Myopulse Percentage Rate
PCA	Principal Component Analysis
LDA	Linear Discriminant Analysis
QDA	Quadratic Discriminant Analysis
KNN	k-nearest Neighbor
MLPT	Multilayer Perceptron
DT	Decision Tree
NB	Naive Bayes
MD	Mahalanobis Distance
SVM	Support Vector Machine
ANN	Neural Network
SampEn	Simple Entropy
NDAMV	Normalized Difference Variance Value
MLPT1	Multilayer Perceptron first Modification
MLPT2	Multilayer Perceptron second Modification
RFS	Random Forests
MAV2	Modified Mean Absolute Value Second Definition
MAVS	Mean Absolute Value Slope
CC	Cepstral Coefficients
MLP	Multilayer Perceptron
kNN	k-th Nearest Neighbor
IAV	Integral Absolute Value
OFNDA	Orthogonal Fuzzy Neighborhood
DA	Discriminant Analysis

Table 1.6: Single feature extraction and single classifier.

Reference	Feature Extraction	Classifier	Recognition Accuracy	Subject
[25]	WAVET	MLPT	97.4	10H
[19]	MAV, RMS, WL, ZC, VAR, SSC, WAMP, AR2, AR6, PSD, FMN, FMD	SVM	MAV=95 RMS=95 WL=96 ZC=94 VAR=94 SSC=90 WAMP=94 AR2=85 AR6=90 PSD=61 FMN=61 FMD=62	11H
[31]	MAV, VAR, WL, WAMP	MLPT	MAV=88.65 VAR=86.95 WAMP=87.48 WL=92.41	6H
[30]	ICA, FD	ANN	FD= 58 ICA= 96	6H
[26]	ICA	SVM	99.34, 59.34	7H
[32]	PCA	SVM	≈95	3A

The second cluster we propose is related to the researches which have performed comparison between classifiers, as presented in Table 1.7. Taking into account MAV feature [19, 21] reported that the best classifier is SVM. The recognition accuracy is from 95% to 99%, even with a high number of amputated subjects 12A and healthy subjects 11H.

Moreover, [12] only considers amputee patients and choosing SampEn as feature and LDA, RF and QDA as classifiers. the best accuracy recognition is 97% with QDA, followed by 93% with LDA, and finally 87% with RFS.

On the other hand, considering only healthy patients, [28] has tested NDAMV as feature and KNN, QDA and LDA as classifiers, achieving accuracy recognition from 81.1% with LDA to 84.9% with KNN. [29] compared several features including MAV, DMAV, VAR, DVARV, RMS, DASDV, MYOP and WAMP, comparing each features with LDA, QDA, KNN, DT, NB, MD classifiers. The best combination was DAMV and QDA reaching recognition accuracy of 90.12%.

Briefly, the highest accuracy recognition is found taking into account MAV and SVM reaching from 95% to 99% , followed by SampEn and LDA with 93% even taking into account amputee patients.

Table 1.7: Single feature extraction and multiple classifier.

Reference	Feature	Classifier Accuracy	Subject
[19]	MAV	SVM=95 LDA=91.8 MLP1=85 MLP2=91.5	11H
[28]	NDMAV	KNN=84.9 QDA=82.4 LDA=81.1	30H
[21]	MAV	LDA=97.75 Diaglinear DA=97.25 Quadratic DA=97.75 Diagquadratic DA=98.00 Mahalanobis DA=96.00 SVM=99.00	12A
[12]	SampEn	LDA≈ 93 RFS≈ 88 QDA≈ 87	1A
[29]	MAV, DMAV, VAR, DVARV, RMS, DASDV, MYOP, WAMP	DMAV + QDA = 90.12 DASDV + QDA = 89.42 DMAV + KNN = 89.60	18H

The third cluster that we proposed is related to the research that have considered to use multiple features and one classifier, as shown in Table 1.8. [19] mentioned that the highest accuracy of 96.5% is reached using the group of features composed with RMS and AR6, using LDA classifier, following by the combination of MAV, WL, ZC and SSC, considering LDA classifier and achieving 95.5%. In contrast, [27] established that considering the same group of feature MAV, WL, ZC and SSC and also the same classifier LDA, the accuracy recognition decreases until 72.46% but considering SVM the accuracy recognition increases until 89.66%.

Similarly, [37] chose the same features but adding MAVS, reaching an accuracy recognition lower than the others authors of 70%. [34] describes that the best feature group is MAV, VAR, AR4, ZC, FMN and FMD, using SVM (96.9%) and LDA (97.3%). [12] compares features combinations between SampEn and others features, for instance: (i) SampEn with CC, (ii) SampEn, CC with MAV2, (iii) SampEn, CC, RMS with WL. As a result, LDA has the best performance between all of the feature groups, reaching recognition accuracy from 97% to 99%.

On balance, there is a high correlation between the feature group and the classifier selected with the recognition accuracy reached by the methodology selected, although establish the best combination is not an easy task, due the particular condition of each experiment.

Table 1.8: Multiple feature extraction and simple classifier.

Reference	Feature	Classifier Accuracy	Subject
[19]	G1: MAV WL ZC SSC G2: RMS AR2 G3:RMS AR6 G4: MAV WL	G1: SVM=82, LDA=95.5, MPL1=90.5, MPL2=95 G2: SVM=90 G3: SVM=85, LDA=96.5 MLP1=89, MLP2=93 G4: SVM=70	11H
[27]	MAV ZC SSC WL	DT= 86.72, KNN= 83.97, MLPT=88.85, LDA=72.46, SVM=89.66	1H
[37]	MAV MAVS ZC & SSC & WL	ANN=70%	1H
[34]	G1: MAV VAR AR4 ZC FMN FMD	G1: SVM= 96.9, LDA=97.3	6H
[12]	G1:SampEn CC G2: SampEn CC MAV2 G3: SampEn&CC RMS WL	G1:LDA$\approx$ 97, RFS$\approx$ 92, QDA$\approx$ 92 G2: LDA$\approx$ 98, RFS$\approx$ 92 QDA$\approx$ 83 G3: LDA$\approx$ 99, RFS$\approx$ 89, QDA$\approx$ 78	1A

Finally, the last cluster that we proposed is related to research that have considered use a group of features which is reduced and a group of classifiers, as is shown in Table 1.9. Thus, the author that we highlight is [33] whose develop a high accurate offline processing of 90% using AR feature then OFNDA features reduction and finally LDA classifier.

Table 1.9: Single feature extraction, feature reduction and multiple classifier.

Reference	Feature	Feature Reduction	Classifier Accuracy	Subject
[33]	AR6, RMS, WL, ZC, IAV, SSC	PCA PCA OFNDA OFNDA	AR OFNDA LDA=98% (H) 90% (A)	10H&6A

1.2 Features of myoelectric signals

Relevant information about the body movements is obtained from myoelectric signals; this information is embedded into the signal [19, 38]. Thus, characteristics of signals are useful to identify the produced movements, as a function of the measured electrical activity. These characteristics are called features.

Table 1.10: Feature definition.

Feature	Abbreviation	Reference
Mean Absolute Value	MAV	[19, 31] [21, 25]
Willison Amplitud	WAMP	[19, 31] [29]
Variance	VAR	[19, 31] [29]
Autoregresive Coefficients	AR2 or AR6	[19, 36] [33]
Root Mean Square	RMS	[19, 29]
Waveform Length	WL	[19, 31]
Zero Crossing	ZC	[19, 39]
Slope Sign Changes	SSC	[19, 39]
Independent Component Analysis	ICA	[26, 30, 31]
Fractal Dimension	FD	[30, 31]
Power Spectrum Dimension	PSD	[19]
Frequency MeaN	FMN	[19]
Frequency Median Dimension	FMD	[19]
Principal Component Analisys	PCA	[32]
Normalized Difference variance value	NDAMV	[28]
Sample Entropy	SampEn	[12]
Wavelet packet transform	WAVET	[25]
Difference Mean Absolute Value	DMAV	[29]
Difference Variance Value	DVARV	[29]
Difference Absolute Standard Deviation Value	DASDV	[29]
Myopulse Percentage Range	MYOP	[29]

The most used features have a low computational cost [23], which is a relevant consideration in real time applications, such as myoelectric control of a prosthesis or a manipulator robot. Moreover, the most used features are: MAV, WAMP, VAR, AR2 or AR6, following by RMS, WL, ZC, SSC, ICA and FD. Table 1.10 summaries some features that had been considered by

authors, taking into account individual feature in their classification schemes.

Similarly, some authors have considered the combination of several features, including from two until seven. The most used combination is MAV, WL, ZC and SSC [19, 27, 37]. The group of features most used are WL and ZC, as shown in Table 1.11.

Comparing the performance classification scheme using one feature or several features, [19] mentioned that the best recognition accuracy is reached using the individual feature.

Table 1.11: Feature Group time and frequency domain.

Feature Groups															References
MAV	MAVS	MAV2	WL	ZC	SSC	RMS	AR2	AR4	AR6	VAR	FMN	FMD	SampEn	CC	
✓			✓	✓	✓										[19]
✓			✓	✓	✓										[27]
✓	✓		✓	✓	✓										[37]
						✓	✓								[19]
						✓			✓						
✓				✓				✓		✓	✓	✓	✓		[34]
													✓	✓	[12]
			✓										✓	✓	
			✓	✓									✓	✓	

In contrast, [12] compared the recognition accuracy considering the following set: (i) SampEn, (ii) SampEn and CC, (iii) SampEn, CC and MAV2, (iv) and, SampEn, CC, RMS and WL. As a result, they found that taking into account LDA classifier, when the number of feature increases the recognition accuracy also increases from 93% to 99%. In contrast, choosing QDA the recognition accuracy decreases from 87% until 78%, as illustrated in Figure 1.4.

Based on the previous studies, the recognition accuracy not only depends on the feature or the group of features selected, but also on the classifier which has a high influence on the final accuracy.

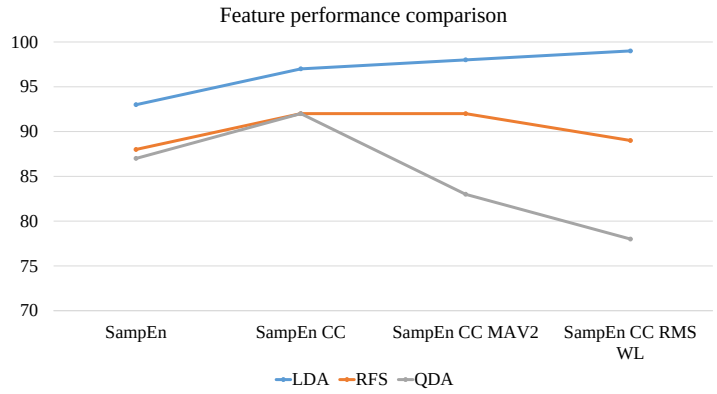


Figure 1.4: Recognition accuracy comparison based on [12] results.

In summary, it is difficult to compare the performance recognition reached in previous studies due to: the different used schemes in the literature as shown in Table 1.4 and the differences in the experiments [12]. Furthermore, several authors have studied the performance of different methodologies measuring the accuracy through the movement classification, as shown in Tables 1.6, 1.7, 1.8, and 1.9.

On the other hand, few authors have analyzed the performances of the features during movements and their different characteristics, *e.g.* speed and acceleration.

[40] analyzed separately the effect of the joint angle and the force level variation when estimating EMG mean frequency. They reported that the joint angle variation is more influent than all other joints. Similarly, [41] reported that the average EMG signal power increases with the hand grip force. Finally, the relationship between features and upper limb motion characteristics are not clearly established. These answers improve the recognition of the motion intention in amputated patients, so out of laboratory conditions. Based on the previous argues, we propose to analyze the performance of some selected features while movements are performed with different position and speed, with the aim to establish relations between feature and motion characteristics.

Consequently, in the following subsection 1.3, we will propose the kinematic model of the upper limb, with the aim to determine position and speed of elbow and wrist joints. Then, considering the behavior features and the kinematic model of the upper limb we will find the best feature set, taking into account, features with the highest changes.

1.3 Mathematical model of upper limb movement

In the following section, it is described the bio-mechanical model and the methodology proposed. To analyze the behavior of some selected features of sEMG and the upper limb motion. First at all, we chose five features of the sEMG signals then we describe the upper limb motion, followed by an experiment, and finally the formulation of kinematic and the ElectroMyoGraphic.

1.3.1 Features

The muscular contraction in the musculoskeletal system are coming from the impulsive electrical stimuli, which is transmitted from the α-motoneurons to muscular fibers. Thus, a Motor Units Action Potential (MUAP) is the particular impulsive response in each fiber, see more details in the section 3.3. As consequence, the voltage picked up by the electrodes represents the sum of the activity of all MUAPs.

The number of MUAPs per muscle in humans are between 100 to 1000, for instance, a small hand muscle has 100 MUAPs and the large limb muscles has 1000 MUAPs [42]. As consequence, a higher number of MUAPs active have more elements in the summation of the voltage detected by electrodes. It has a meaningfully influence in the resultant signal, changing its magnitude and frequency components.

The behavior of features extracted from sEMG signal is analyzed, for the purpose of finding the features that describe in the best manner the relationship between the activity of sEMG and the upper limb movements. The chosen features are: MAV, MV, Entropy H, HM and MF.

The mathematical model of MAV is described in equation (1.1) [43], where x_k represents the k-th signal sample, N is the number of samples in segment. Furthermore, MV is presented in equation (1.2) and Entropy H is described in equation (3.5); the complete model is presented in section 3.3.2.1.

$$MAV(\boldsymbol{x}) = \frac{1}{N}\sum_{k=1}^{N} \mid x_k \mid \tag{1.1}$$

$$MV(\boldsymbol{x}) = \frac{1}{N}\sum_{k=1}^{N} x_k \tag{1.2}$$

$$H(\boldsymbol{x}) = \sum_{l=1}^{N_i} p_l \log_2 p_l \tag{1.3}$$

Moreover, Harmonic Mean (HM) is given by the equation 1.4 and finally Mean Frequency (MF) is represents by 1.5, where C_f denotes the number of harmonics in the spectrum, I_j represents the magnitude of the j-th harmonic, and f_j is the frequency of the j-th harmonic.

$$HM(\boldsymbol{x}) = \frac{1}{\frac{1}{N}\sum_{k=1}^{N}\frac{1}{x_k}} \tag{1.4}$$

$$MF(\boldsymbol{x}) = \frac{\sum_{j=1}^{C_f} I_j f_j}{\sum_{j=1}^{C_f} I_j} \tag{1.5}$$

1.3.2 Kinematics of upper limb

The upper limb chain is mainly composed of the following joints: (i) shoulder, (ii) elbow, (iii) and wrist. Each joint have different mobility, for example shoulder joint has flexion, extension, adduction, abduction, internal, external rotation. Then, elbow joint have only flexion and

extension. Finally wrist joint have flexion, extension, radial, ulnar, pronation, and supination. Thus, from these characteristics, a methodology was formulated, for tracking upper limb motion, considering the following steps: (i) labeling markers, that are positioned in body, (ii) clustering the markers, (iii) fixing a coordinate system fixed to each marker, (iv) and setting vectors which link the markers between them, with the same order of the markers clusters. Moreover, the complete clusters and markers are fully presented in section 1.4.1 and in Figure 1.9.

Once the kinematic is known, it is necessary to find the angle performed by each joint. Particularly, in this study we point out to find elbow angle φ.

For instance, marker 19 is linked to marker 20 by a vector and marker 21 is linked to marker 22 by another vector, etc., as is shown in Figure 1.5

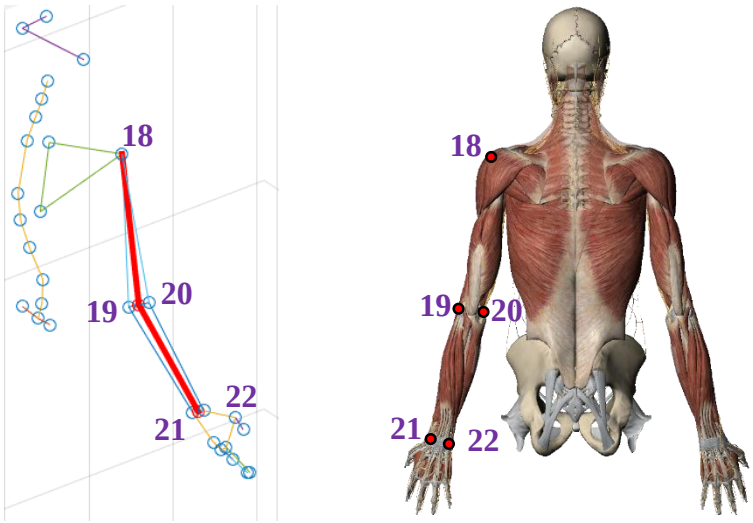


Figure 1.5: Kinematic upper limb.

For this purpose, we use two vectors: (i) from the elbow to the wrist, and (ii) from the elbow to the shoulder, which are represented by red lines in Figure 1.5. First vector was established finding the middle point p_1 between the coronoid process ulna and radius (markers 19 and 20). Similarly, second vector was determined by locating the middle point p_2 in the wrist, between the markers placed on the styloid process of ulna and radius (*i.e.* markers 21 to 22).

The arm vector denoted by $\boldsymbol{v}_a \in \mathbb{R}^3$ is the vector that links the point p_3 (marker placed on the epicondyle of humerus) to the point p_1. The forearm vector denoted by $\boldsymbol{v}_{fa} \in \mathbb{R}^3$ links points p_1 and p_2 (see Figure 1.5), in which these vector is represented as red lines. With these two vectors, the elbow angle φ is calculated as:

$$\varphi = \arccos\left(\langle \boldsymbol{v}_a, \boldsymbol{v}_{fa} \rangle / \|\boldsymbol{v}_a\| \|\boldsymbol{v}_{fa}\|\right) \tag{1.6}$$

1.4 Experiments of flexion and extension movements

The experiment seeks to measure the sEMG produced during upper limb movements. Thus, we propose an experiment for: (i) analyzing the behavior of the features from sEMG signals produced during upper limb motion, (ii) analyzing the kinematics characteristics of the movement such as position and angle speed.

For the purpose of minimizing the complexity of the experiment, we choose the elbow joint: performing flexion and extension and the muscle selected are bicep and triceps. The workflow of the experiment is presented in Figure 1.6. Biceps are one of the most important muscles involved in the elbow flexion, and triceps muscle is its antagonist muscle.

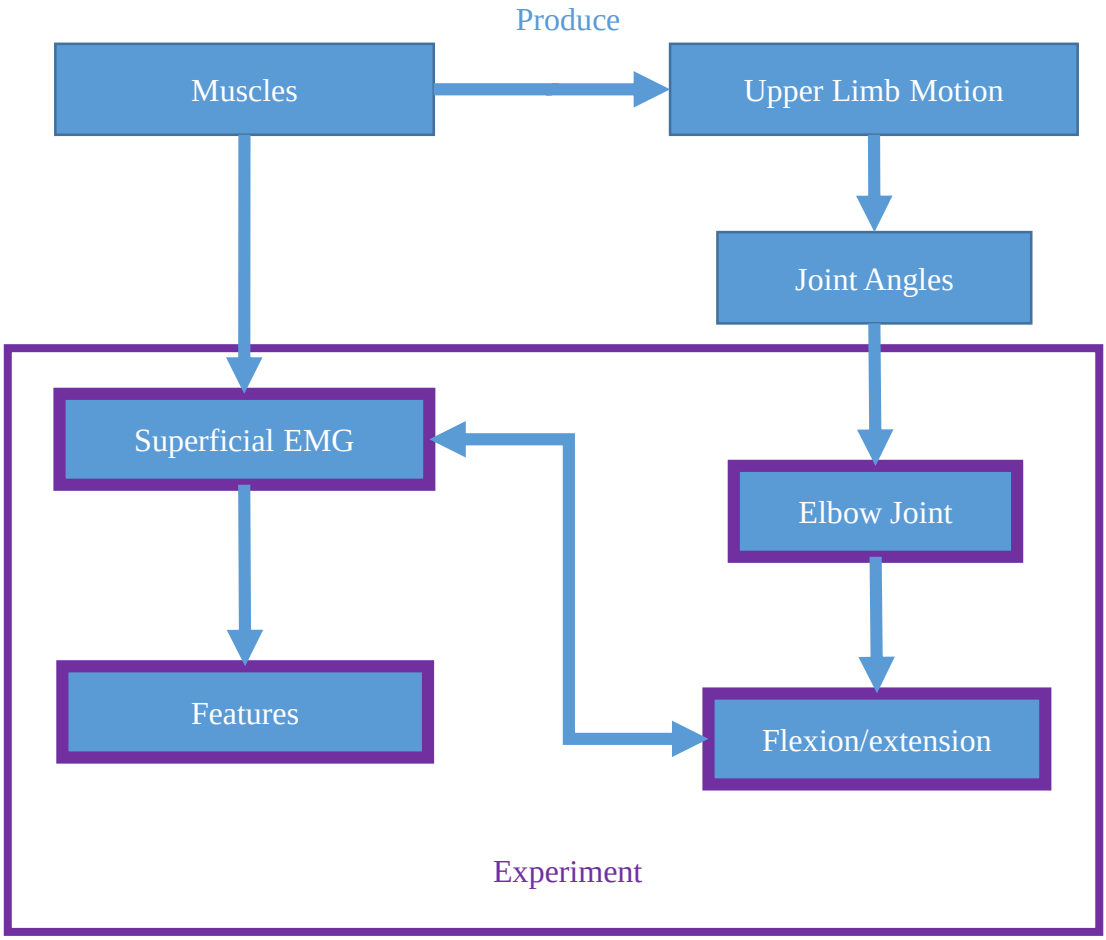
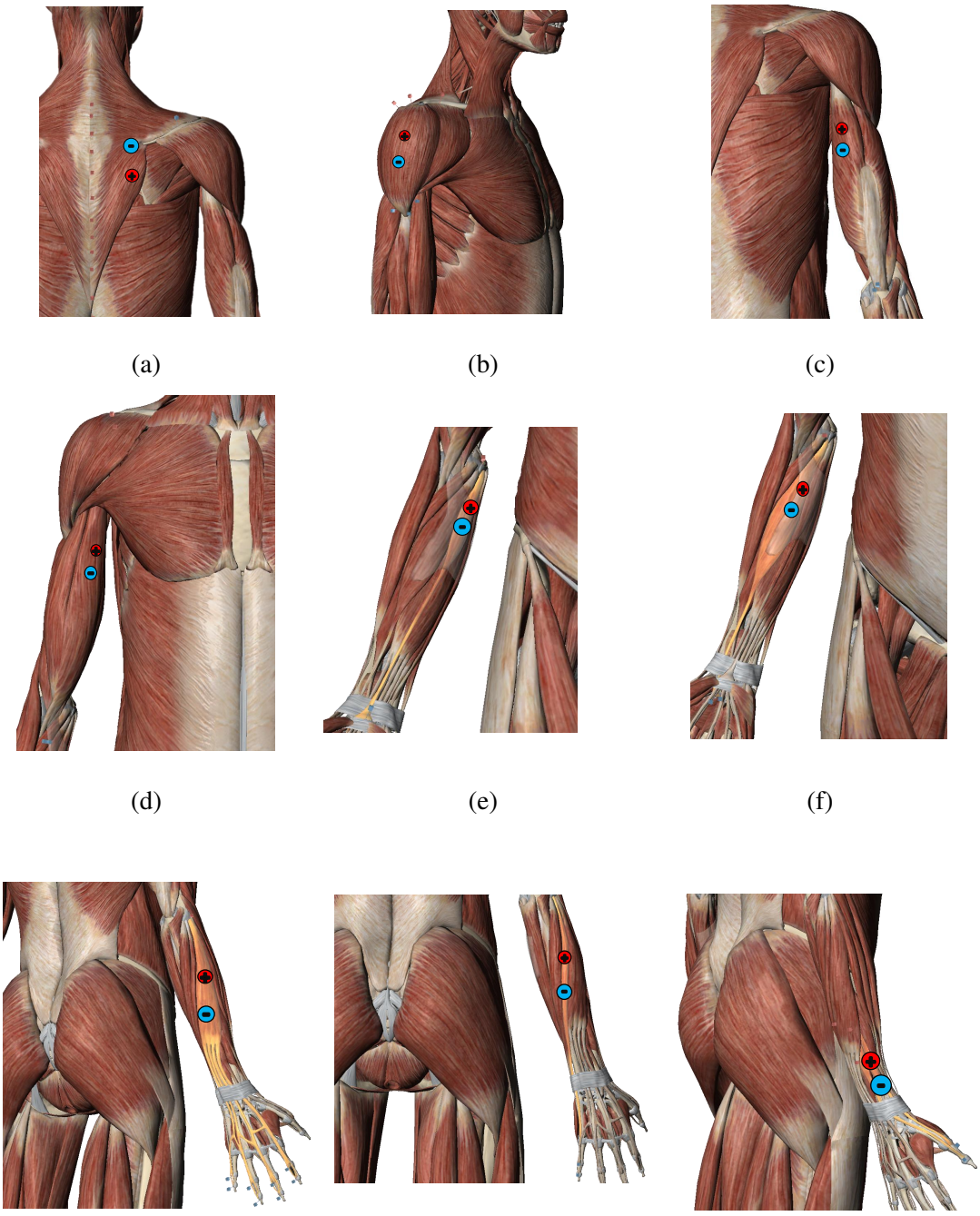


Figure 1.6: Experiment work-flow.

1.4.1 Materials

The sEMG signals have been recorded using the ZEROWIRE wireless capture system, with a sampling frequency of 1 kHz. The electrodes are placed over the upper limb muscle. Figure 1.7 illustrates muscle position, the images are courtesy of Visible Body (www.visiblebody.com).

Moreover, muscles and their functions are detailed in Table 1.12.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 1.7: Electrodes position over upper limb muscles.

Id Muscle	Muscle	Function
Figure 1.7a	Trapezum	stabilization shoulder
Figure 1.7b	Deltoid	flexion/extension abduction/adduction shoulder humerus rotation
Figure 1.7c	Triceps (TR)	extensor muscle of the elbow joint
Figure 1.7d	Biceps (BB)	flexion/supination of the forearm
Figure 1.7e	Palmaris longus (PL)	flexion of the hand
Figure 1.7f	Flexor carpi radialis (FCR)	flexion and abduction of the hand
Figure 1.7g	Extensor carpi radialis (ECR)	extension of the wrist joint
Figure 1.7h	Extensor digitorum communis (ED)	extension of the phalanges, the wrist, and the elbow
Figure 1.7i	Extensor muscles of the thumb	extension of phalanges

Table 1.12: Measured upper limb muscles.

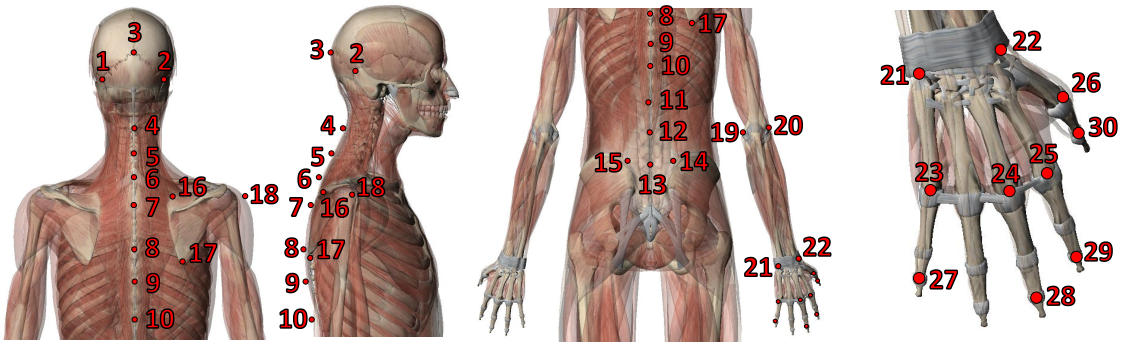


Figure 1.8: Distribution of markers in the upper limb kinematic chain.

Table 1.13: Markers distribution.

Id Clusters	Id Marker	Marker Position
A	1 to 3	Occipital bone
B	4 to 13	Cervical and lumbar vertebra
C	13 and 15	Top part of sacrum
D	16 and 18	Acromion of scapula
E	18	Medial epicondyle of humerus
F	19 and 20	Coronoid process of ulna and radius
G	21 to 22	Styloid process of ulna and radius
H	23 to 26	Metacarpophalangeal joint of fingers
I	27 to 29	Proximal interphalangeal joint of the fingers
J	30	Interphalangeal joints of finger 1

The tracking of the upper limb motion was performed using 7 3D VICON cameras and Nexus software, with a sampling frequency of 200 Hz. 30 retro-reflective markers were used and positioned all over the upper limb kinematic chain including the spinal column, as shown in Figure 1.8 and in Table 1.13.

Figure 1.9 shows one session of the experiment, in which the subject was equipped with retro-reflective markers and wireless superficial electrodes.

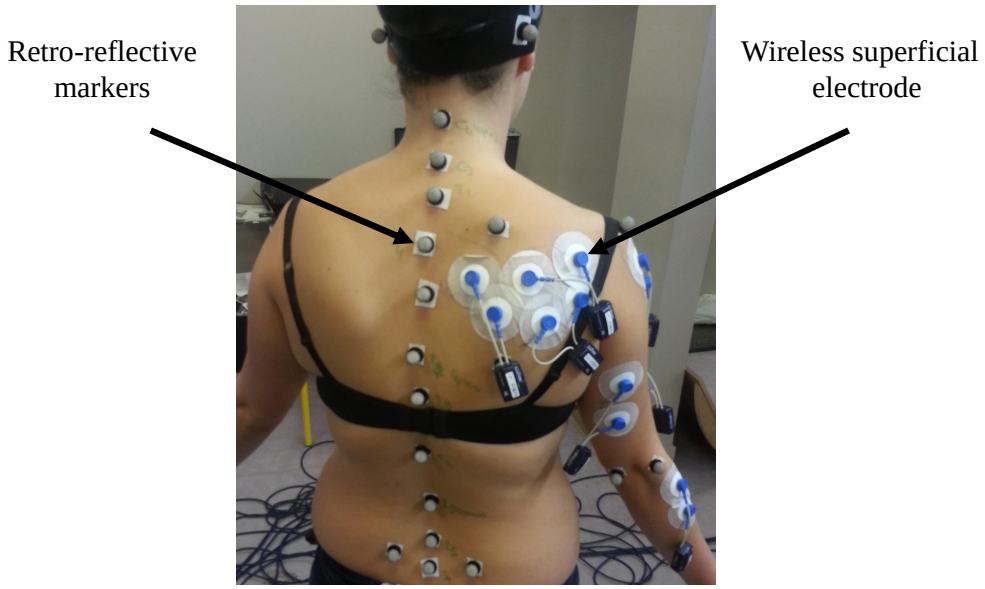


Figure 1.9: Distribution of markers in the upper limb kinematic chain.

In the proposed experiment we considered three Healthy Subjects (HS), with an average age of 24 years. Each subject performs flexion and extension of elbow joint. The first HS performed one trial, the second HS four trials and third HS six trials. Flexion and extension of the elbow joint was performed by subjects with different angles and speeds.

1.4.2 Data processing

To analyze data obtained from the experiments performed, we propose a methodology that consists in: (i) determine the angle of each joint, in this specific experiment only flexion and extension of the elbow joint was calculated, (ii) synchronize tracking position samples with sEMG signals, (iii) and crop the sEMG signals when the flexion and extension were produced. This last step is performed with the aim of analyzing the sEMG signals only during flexion and extension, neglecting the stabilization of the movement.

To implement the previous methodology, we developed a software which :

(i) displays tracking motion of the upper limb data,

(ii) visualizes the corresponding sEMG signals,

(iii) synchronizes in real time the sEMG signals and the tracking motion of the upper limb chain data,

(iv) calculates the angle joints based on the quaternion model, in this case, we are interested in the elbow flexion and extension,

(v) and crops the sEMG signals.

All the methodology was developed in the graphical user interface development environment (GUIDE) of MATLAB®, the Figure 1.10 shows the software interface developed.

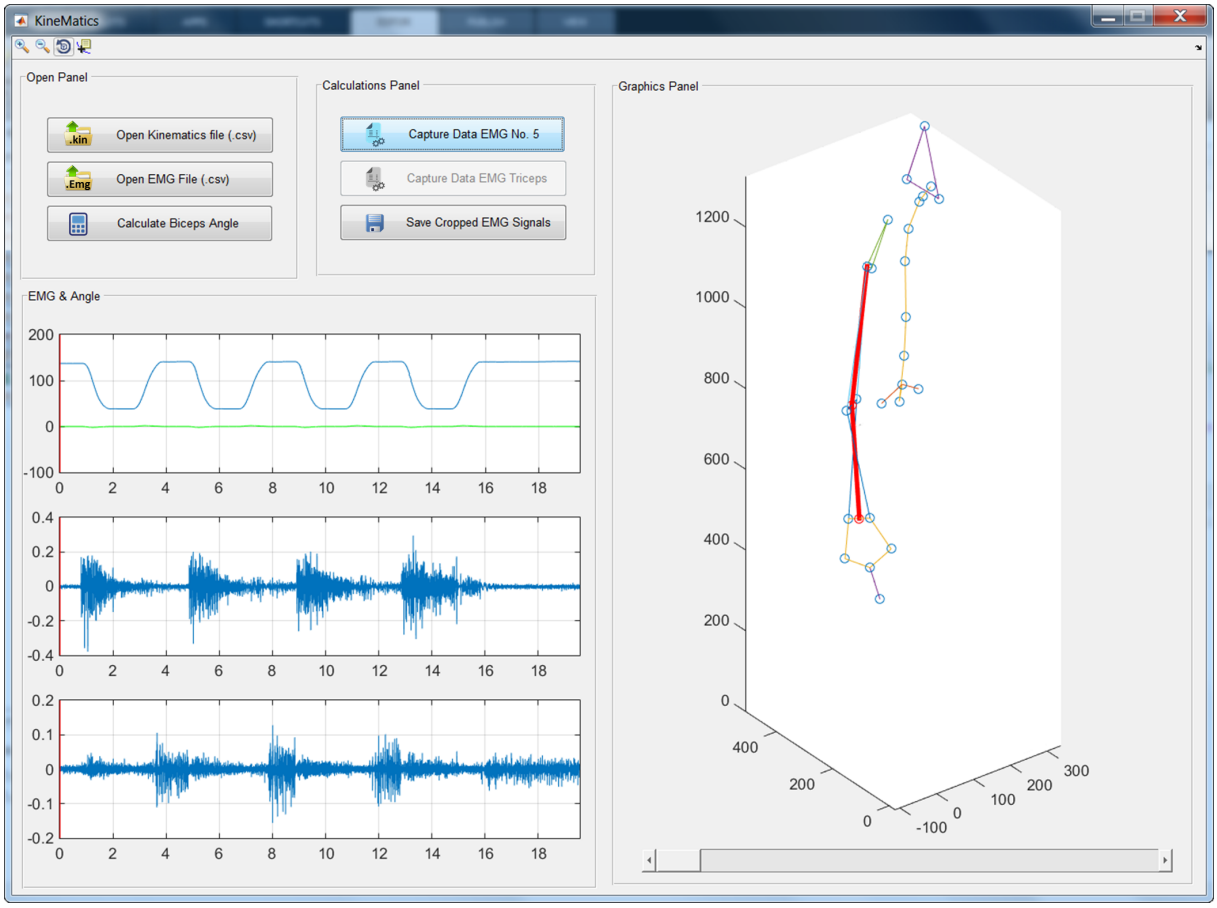


Figure 1.10: Software interface.

The signal synchronization is required because both signal are recorded at different sampling frequency. Kinematic tracking is sampling to 200 Hz and sEMG signals is sampling to 1kHz.

On the other hand, based on the kinematics of the upper limb, the rotation of joint is determined. Ones the rotation of joint is found, the beginning and the ending of motion is estimated. Moreover, the procedure is explained in section 1.3.2.

Ones the beginning and the ending of the movement was determined, the sEMG signal was accurately cropped, obtaining the signals when a movement exist.

1.4.3 Results

Table 1.14 shows the performed angle and speed during elbow flexion and extension, considering all trials performed by subjects. Because of the selected motion (joint elbow), we considered two muscles: Biceps Branchii (BB) and Triceps Branchii (TB).

The BB and TB sEMG was synchronized with upper limb motion tracking, Then, we have cropped the sEMG of BB and TB during flexion and extension, furthermore we applied the feature extraction models to all trials; getting a set of scalar values that corresponds to the flexion and the extension. Considering that the values have different scales, we applied a normalization.

Table 1.14: Angle and speed of movements.

Subjects Id.	Trial	Angle[°]		Speed[rad/s]	
		Flexion	Extension	Flex	Ext
1	1	057.92	-071.29	1.80	-2.04
2	1	097.37	-102.28	1.24	-1.30
	2	108.69	-117.79	1.93	-1.76
	3	117.27	-117.87	8.15	-6.22
	4	117.27	-117.87	1.49	-1.32
3	1	048.54	-038.74	2.72	-1.77
	2	043.18	-039.88	1.71	-1.22
	3	036.80	-041.40	2.56	-2.18
	4	046.60	-044.44	2.62	-1.52
	5	046.76	-046.56	2.54	-1.84
	6	045.63	-043.73	2.41	-1.49

Table 1.15: The normalised values of the features.

Flexion										Extension									
Biceps					Triceps					Biceps					Triceps				
$HMAV$		M	HM	MF	$HMAV$		M	HM	MF	$HMAV$		M	HM	MF	$HMAV$		M	HM	MF
−0.47	−0.57	0.01	−0.31	−0.69	−0.07	−0.94	1.36	−2.50	−0.66	2.25	2.81	−1.60	3.02	2.14	2.55	2.90	−0.57	0.45	1.65
−0.26	−0.49	−0.46	−0.31	−0.42	−0.57	−0.82	0.03	0.52	−0.88	−1.05	−0.66	0.52	−0.30	−1.00	−0.86	−0.62	−0.40	0.45	−1.10
−0.03	−0.34	−0.37	−0.31	−0.19	−0.52	−0.80	0.03	0.53	−0.86	−0.92	−0.61	0.47	−0.30	−0.79	−0.74	−0.56	−0.73	0.45	−1.09
−0.63	−0.03	−1.55	−0.32	0.28	0.47	0.07	1.28	−0.87	−0.31	0.19	−0.01	−1.06	−0.30	0.04	0.16	−0.04	−0.34	−2.02	0.07
0.00	−0.42	−0.41	2.43	−0.55	−0.01	−0.59	0.30	0.53	−0.09	−0.76	−0.59	0.74	−0.30	−1.28	−1.20	−0.81	0.04	0.45	−1.71
2.03	2.20	0.41	−0.31	2.67	1.38	1.32	−0.33	0.53	1.49	0.91	0.34	−0.81	−0.30	1.16	0.45	−0.03	2.61	0.45	0.34
−1.02	−0.89	0.33	−0.31	−0.57	−0.74	−0.11	−0.48	0.53	0.66	−0.51	−0.42	0.80	−0.30	−0.12	−0.19	−0.14	0.15	0.45	1.12
−0.63	−0.03	−1.55	−0.32	0.28	0.47	0.07	1.28	−0.87	−0.31	0.19	−0.01	−1.06	−0.30	0.04	0.16	−0.04	−0.34	−2.02	0.07
−0.35	−0.56	1.40	−0.31	−0.64	−1.80	−0.71	−0.73	0.54	−1.25	−0.68	−0.57	0.44	−0.30	−0.71	−0.19	−0.27	0.27	0.45	0.50
1.84	1.69	1.03	1.32	0.62	1.77	2.27	−1.62	0.53	1.89	0.77	0.23	−0.07	−0.30	0.52	0.42	−0.02	0.56	0.44	0.47
−0.48	−0.58	1.17	−1.24	−0.77	−0.38	0.22	−1.12	0.53	0.31	−0.39	−0.52	1.64	−0.30	0.01	−0.54	−0.35	−1.25	0.45	−0.32

The row one contains the values for trial one, the row two represents trial two, and so on

Table 1.15 shows the normalized values of the features; it is divided into two mainly sections: flexion and extension. Then, each section has the data corresponding of biceps and triceps muscles. The row one contains the values for trial one, the row two represents trial two, and so on.

Figure 1.11 presents the behavior of the normalized values of the features extraction during elbow extension as a function of the angle performed. First line presents five features: H, MAV, M, HM, M. Furthermore, taking into account the similar waveform pattern features extracted, we selected only three features: H, MAV and MF, see second line in Figure 1.11.

Similarly, we presents the performance of the normalized features extracted during flexion, as shows in Figure 1.12.

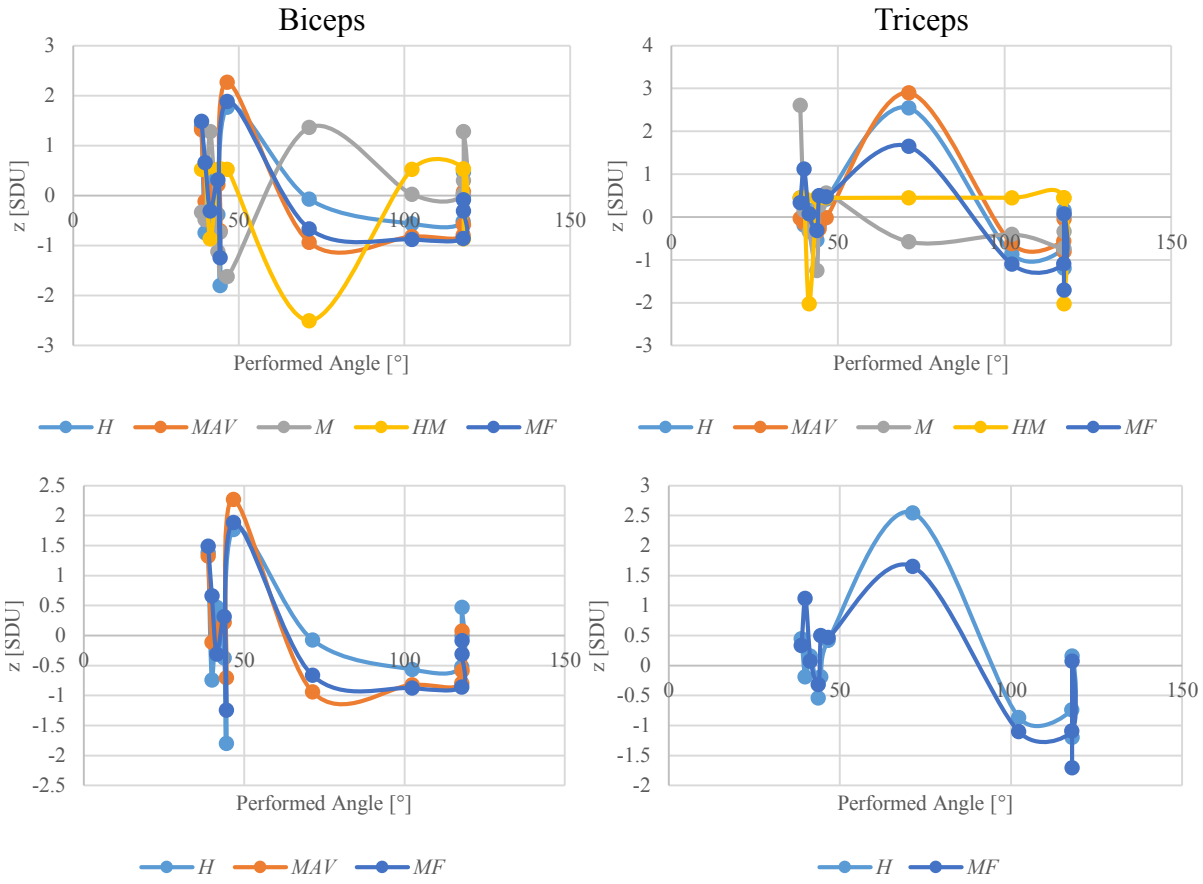


Figure 1.11: Features behavior biceps and triceps muscle during elbow extension considering angle performed.

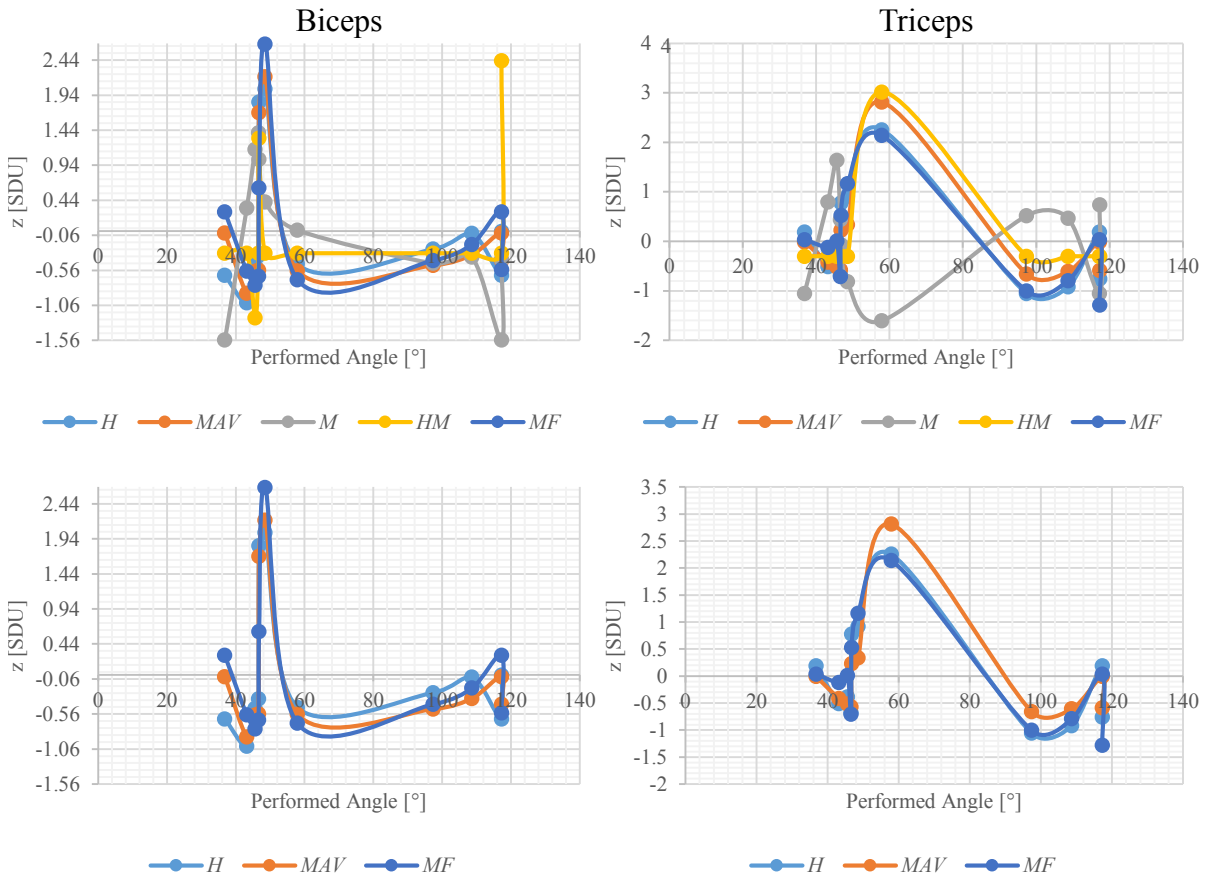


Figure 1.12: Features behavior biceps and triceps muscle during elbow flexion considering angle performed.

Moreover, comparing the normalized values Figures 1.11 and 1.12, in both cases we found that M and HM have significant variations, few local minimal and maximal values. These long fluctuations could be useful to model small angle. But in the opposite, the identification of the complete range will be more difficult.

Furthermore, H, MAV and MF show a similar pattern, a more regular behavior and several local minimal and maximal values (see Figure 1.12 or 1.11 right). These particularities make possible to model and consequently describes long range movement.

Consequently, we also analyzed the behavior features pointing out the speed performed during elbow flexion, as show in Figure 1.13 and elbow extension as presented in Figure 1.14.

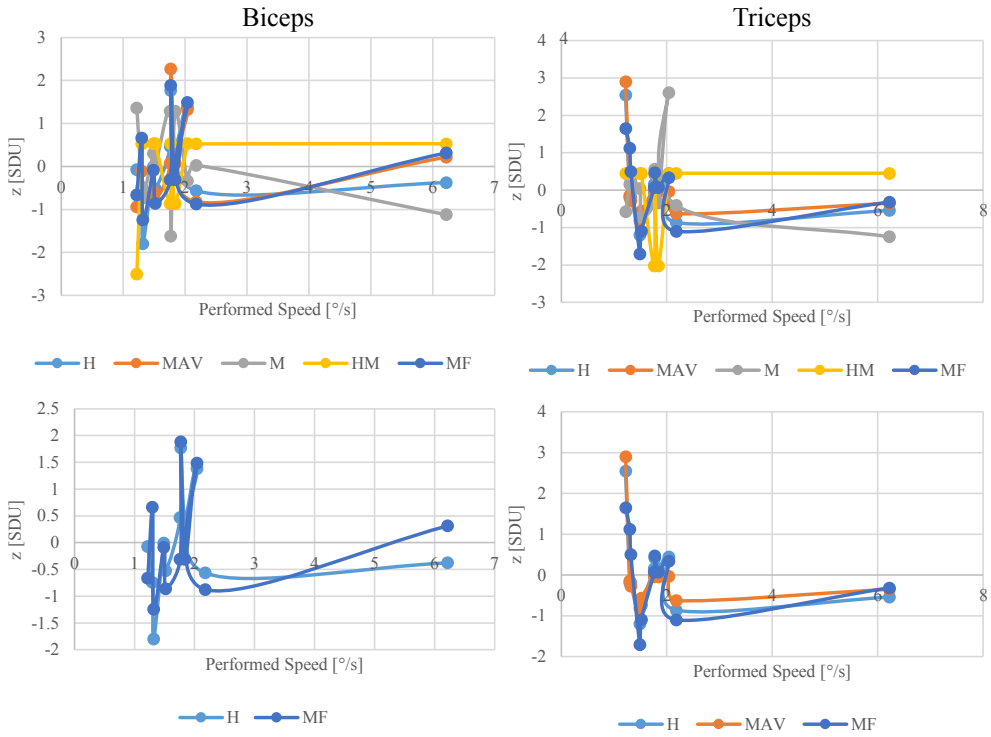


Figure 1.13: Features behavior biceps and triceps muscle during elbow extension considering speed performed.

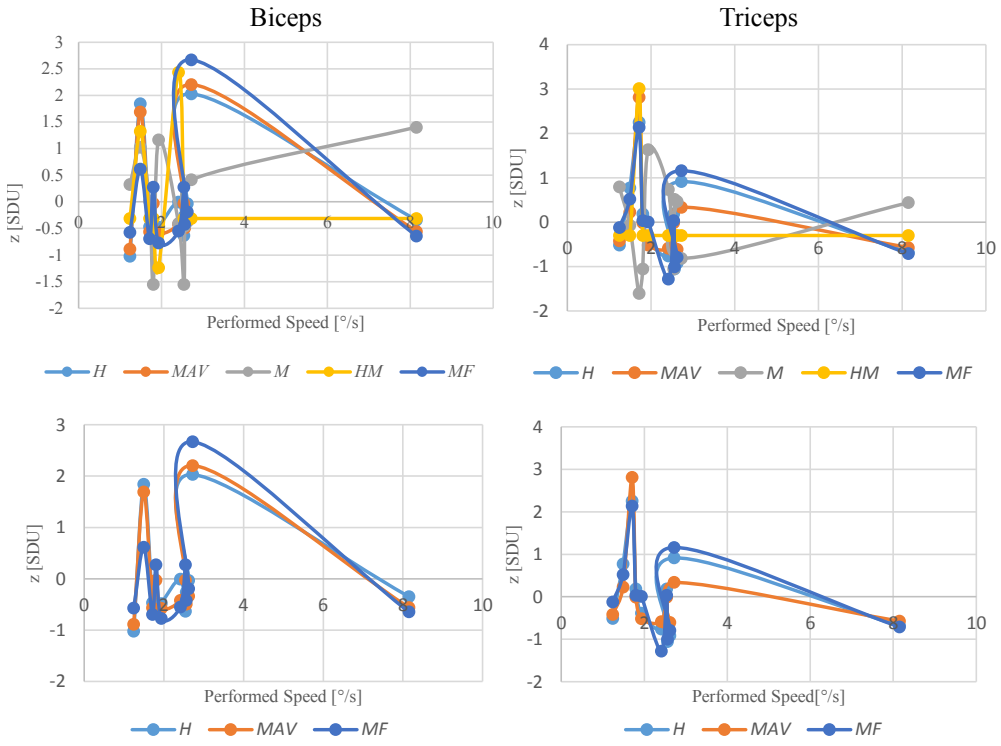


Figure 1.14: Features behavior biceps and triceps muscle during elbow flexion considering speed performed.

All features have significant variation at lower speeds. Moreover, the maximum values was found in the range of $\approx$ 1m/s and $\approx$ 2m/s. In contrast, the features variation at higher speeds are not significant. Moreover, H, MAV and MF present similar and more regular behavior compared with H and M. Thus, we propose a criterion to choose the best features to identify the flexion and extension movements.

1.5 Formulation of kinematic / ElectroMyoGraphic relation

Distinguishing upper limb movements based on sEMG signals, implies to consider the following factors: (i) higher variations of feature values during movements, particularly elbow joint (ii) similar behavior of the feature during movements. Focused on these factors, the features selected are H, MAV, and MF, it was chosen from experiments results presented in the last subsection 1.4.3.

For the purpose of distinguishing two movements, particularly elbow joint, we propose to find the couple of normalized features that presents the higher Euclidean distance between them.

Based on the features selected, we created two sets of them, one for flexion, and one for the extension. These sets are used to characterize the difference between the features which is evaluated using a Euclidean distance. The generated sets of features are affected by the variations in kinematics parameters, such as speed and angle range of the performed movements.

To chose the best set of features, we propose a criterion that maximizes distance between the nearest values of each couple of features. For this purpose, we propose the criterion presents in equation (1.7).

$$(b,u) = \arg\max_{b,u} \left(\min\left(|F_b^{\text{ext}} - F_b^{\text{fle}}| + |F_u^{\text{ext}} - F_u^{\text{fle}}|\right)\right) \tag{1.7}$$

for $b \neq u$ where F_b^{ext} and F_b^{fle} are the b-th features for extension and flexion, F_u^{ext} and F_u^{ext} are the u-th features for extension and flexion. Based on the proposed criterion, the selected features are found to be the entropy and the mean frequency features. As a result of the proposed criterion in equation1.7, the couple (H, MF) shows the best fitness among all the features previously introduced.

1.6 Conclusion

We have analyzed the performance of five selected features while movements are performed with different positions and speeds, with the aim to establish relations between features and motion. The compared features are: MAV, MV, H, HM and MF. Additionally, we have presented a kinematic model of the upper limb in order to establish the performed joint angles and rotation speeds.

Analyzing the joint angle variation, we find that M and HM have significant variations, few local minimal and maximal values. These long fluctuations are be useful to model small angles. Although, the identification of the complete range will be more difficult.

Furthermore, H, MAV and MF have a similar pattern, a more regular behavior and several local minimal and maximal values. These particularities make possible to model and consequently describes long range movements. Moreover, all features have significant variations at lower speeds. The maximum values are found in the range [1,2]m/s. In contrast, the features variation at a higher speeds are not significant.

Finally, we propose a criterion to chose the best set of features. Based on the proposed criterion, the selected features are entropy and mean frequency features, which show the best fitness among the introduced features.

Chapter 2

Movement classification based on myoelectric signals

2.1 Review

Support Vector Machine (SVM) have been proved to be a powerful tool to solve classification problems. It has been tested in different applications such as: (i) automated estimation of human emotion from electroencephalography [44], achieving accuracy of nearly 80%, (ii) obstacle recognition system from visible and infrared images [45] getting accuracy from 87.6% to 96.9%, (iii) speaker identification [46] finding high scores of 90%.

Furthermore, previous studies have reported several advantages when SVM is used as classifier regression and optimizer, for instance: (i) the high generalization capability, even if patterns change in time [46, 47], (ii) the ability of classifying linearly non-separable patterns [47], (iii) the low computational cost, (iv) and the real time pattern recognition probabilities.

These characteristics are essential for solving classification problems, particularly biomedical and especially for sEMG analysis that can be safely collected in nowadays life. Thus, SVM is a successful method for hand gesture identification and finger motion patterns recognition from sEMG.

Several studies have documented the SVM classification accuracy. A summary is introduced in Table 2.1, which presents studies that have been performed measuring the electrical activity in the upper limb muscles. The muscles most used by authors are: (i) FCU and (ii) FCR. recognizing between 6 and 7 grasping gestures, with 93% to 97% accuracies [47–49]. Moreover, [50] have considered to used different muscles, such as: EPB, FPL, EIP, FDS and EDQP, predicting effectively 18 finger motions, with 82 to 93% accuracy.

Individual finger motion have been also studied by [51], predicting the index flexion and pinching of index against thumb, achieving 80% accuracy. Finally, the identification of the wrist motion including opening and closing hand is studied by [52], reaching 97% accuracy. Additionally, closing and opening hand are the most studied hand gesture by authors [47–50, 52].

Table 2.1: SVM toward grasping recognition gestures.

Number of Movements	Description	Accuracy	Muscle	Reference
6	Flexion, Extension, Grasping, Opening, Pronation, Supination, Neutral	93%	FCU, ECR, FDP, ED	[47]
7	Neutral, Flexion, Extension, Grasping, Opening, Pronation, Supination	97%	FCU, ECR, FDP, ED	[48]
6	Resting, Grasping, Index Finger Pinching, Middle Finger Pinching, Ring Finger Pinching, Little Finger Pinching	95%	PL, FCU, FDS,FCR	[49]
18	Resting, Opening, Flexion and Extension all Fingers	82% – 93%	EPB, FPL, EIP FDS (proximal, distal), EDQP	[50]
2	Index Flexion Pinching of the Index Thumb	80%	-	[51]
6	Wrist Flexion, Wrist Extension, Hand Closing, Hand Opening, Radial Deviation, Ulnar Deviation	97%	-	[52]

2.2 Pattern classification

Pattern Classification (PC) is a field of Artificial Intelligence (IA). IA is defined as systems that think like humans and can think rationally [53]. PC investigates the development of algorithms to distinguish specimen inside a group. Typically the differentiation is based on an information set composed of distinctive specimen's features. A full suite of features describing specimen is usually known as a pattern. In the state of the art, three kinds of pattern classification are clearly identified: (i) labeling data also called supervised learning, (ii) separating of data into classes are also called unsupervised learning (iii) identifying relevant information also called feature selection [54, 55].

In order to define the movement recognition algorithm, we take into account the following considerations: (i) the sEMG signals are information carriers that correspond to movement intention, (ii) the features extracted from the sEMG signals are directly linked to the movement,

and (iii) the number of movements to be identified are finite. Consequently, the identification of movements is assessed through a classification problem in which features extracted from sEMG signals constitute the set of patterns, and the movements themselves constitute the labels.

2.2.1 Perceptron

The perceptron is an algorithm for supervised learning of binary classifiers [56]. It means that the algorithm can decide if the new input presented belongs to one class or another. The data or input values are called $\boldsymbol{x_i} = \{x_{1i}, x_{2i}, ..., x_{ni}\}$, each input has a connection that is designated weight value $\boldsymbol{w_i} = \{w_{1i}, w_{2i}, ..., w_{ni}\}$ respectively. The weight values represent the connection importance. The product of each input by its corresponding weight are added. Then a decision function is performed b_i, which result y_i indicates if the input presented belongs to one class (1) or another (0). Figure2.1 illustrates the perceptron.

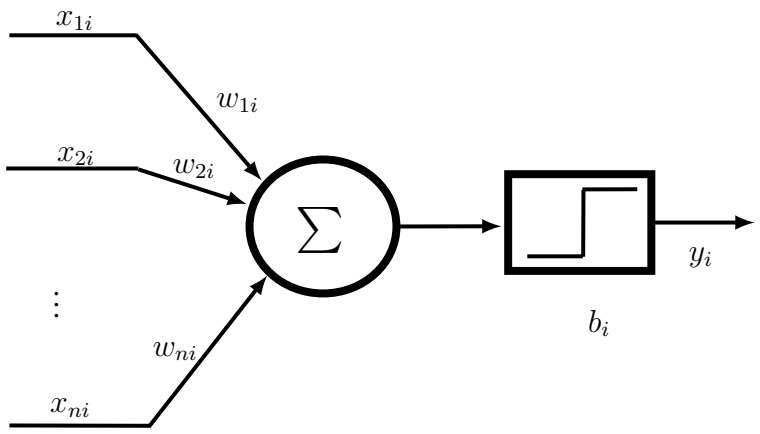


Figure 2.1: Diagram perceptron.

$$y_i = \begin{cases} 1 & \text{if} \quad w_{1i}x_{1i} + w_{2i}x_{2i} + \cdots + w_{ni}x_{ni} \geq b_i \\ 0 & \text{if} \quad w_{1i}x_{1i} + w_{2i}x_{2i} + \cdots + w_{ni}x_{ni} < b_i \end{cases}$$

In matrix notation, the corresponding expression is:

$$y_i = \begin{cases} 1 & \text{if} \quad \boldsymbol{w^T} \cdot \boldsymbol{x} \geq b_i \\ 0 & \text{if} \quad \boldsymbol{w^T} \cdot \boldsymbol{x} < b_i \end{cases}$$

In order to understand the procedure performed by a perceptron, we present an example of a binary classifier. AND gate behavior is in Table 2.2, and illustrated in Figure 2.2.

Considering, $\boldsymbol{x_i} = \{x_{11}, x_{21}\}$ inputs, $\boldsymbol{w_i} = \{w_{11} = 1, w_{21} = 1\}$ weights and $b_i = 1.5$ value,

as consequence the output function is:

$$y_i = \begin{cases} 1 & \text{if} \quad x_{1i} + x_{2i} \geq 1.5 \\ 0 & \text{if} \quad x_{1i} + x_{2i} < 1.5 \end{cases}$$

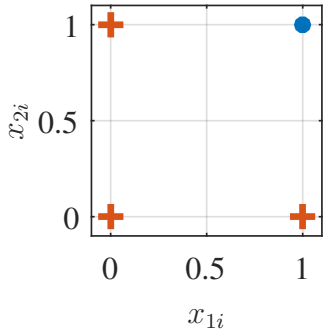


Figure 2.2: AND gate class.

x_{ii}	x_{2i}	y_i
0	0	0
0	1	0
1	0	0
1	1	1

Table 2.2: AND gate truth table.

Two class are illustrated in Figure 2.3, rose circles represents first class (0) and purple cross plots the second class (1). Moreover, the procedure is described in the Table 2.3.

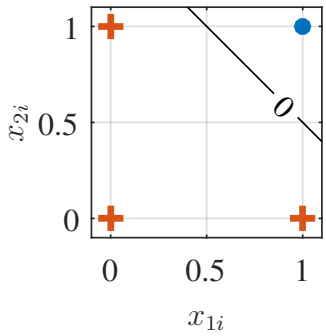


Figure 2.3: Perceptron representation.

$w_{1i} = 1$	$w_{2i} = 1$	$b_{1i} = 1.5$
x_{1i}	x_{2i}	$w_{1i}x_{1i} + w_{2i}x_{2i}$	y_i
0	0	0	0
0	1	1	0
1	0	1	0
1	1	2	1

Table 2.3: Perceptron as a classifier.

Classification is highly influenced by the weight values, the line separation relation is related to the weight values, as illustrated in Figure 2.4 and Table 2.4.

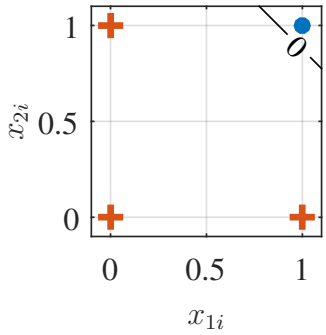


Figure 2.4: Perceptron representation: weight influence.

$w_{1i} = 0.8$	$w_{2i} = 0.8$	$b_{1i} = 1.5$
x_{1i}	x_{2i}	$w_{1i}x_{1i} + w_{2i}x_{2i}$	y_i
0	0	0	0
0	1	0.8	0
1	0	0.8	0
1	1	1.6	1

Table 2.4: Perceptron as a classifier: weight influence.

Furthermore, the b_i has also a high influence on the classification, because it allows to shift the decision function moving the line up and down. The shift contributes to fit the prediction with the data in a better way. Figure 2.5 and Table 2.5 illustrate the influence on the parameters.

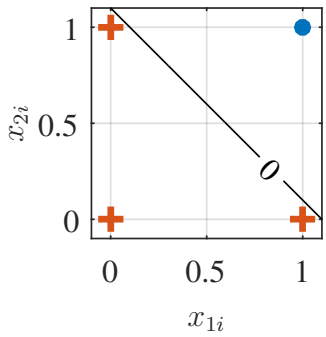


Figure 2.5: Perceptron representation: b_i influence.

$w_{1i}=1$	$w_{2i}=1$		$b_{1i}=1.1$
x_{1i}	x_{2i}	$w_{1i}x_{1i}+w_{2i}x_{2i}$	y_i
0	0	0	0
0	1	1	0
1	0	1	0
1	1	2	1

Table 2.5: Perceptron as a classifier: b_i influence.

On the other hand, if the input values change then the perceptron classifier could distinguish the different classes, as illustrates in Figure 2.6 and Table 2.6.

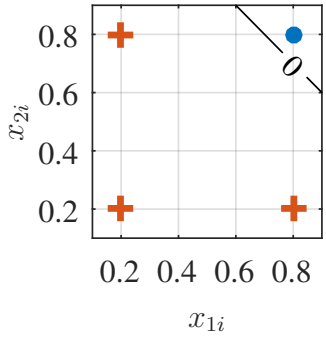


Figure 2.6: Perceptron representation: inputs influence.

$w_{1i}=1$	$w_{2i}=1$		$b_{1i}=1.5$
x_{1i}	x_{2i}	$w_{1i}x_{1i}+w_{2i}x_{2i}$	y_i
0.2	0.2	0.4	0
0.2	0.8	1	0
0.8	0.2	1	0
0.8	0.8	1.6	1

Table 2.6: Perceptron as a classifier: inputs influence.

The perceptron decision rule previously introduced could be rewritten, adding a term $w_{0i}x_{0i}$ to the input values, in which $w_{0i}=b_i$ and $x_{0i}=1$. As consequence $\boldsymbol{x_i}=\{1,x_{1i},x_{2i},...,x_{ni}\}$.

$$y_i=\begin{cases}1 & \text{if} \quad w_{0i}x_{0i}+w_{1i}x_{1i}+w_{2i}x_{2i}+\cdots+w_{ni}x_{ni}\geq 0\\ 0 & \text{if} \quad w_{0i}x_{0i}+w_{1i}x_{1i}+w_{2i}x_{2i}+\cdots+w_{ni}x_{ni}<0\end{cases}$$

In matrix notation, the corresponding expression is:

$$y_i=\begin{cases}1 & \text{if} \quad \boldsymbol{w^T}\cdot\boldsymbol{x}\geq 0\\ 0 & \text{if} \quad \boldsymbol{w^T}\cdot\boldsymbol{x}<0\end{cases}$$

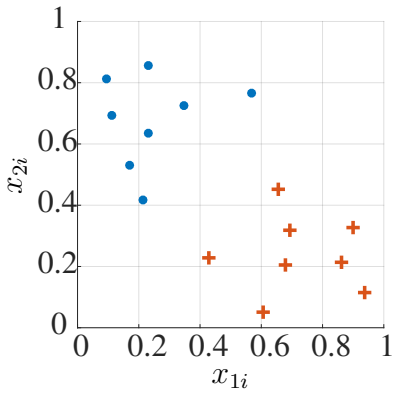
2.2.2 Support Vector Machines

The classification process consists in find an hyperplane that separates classes, as presented in Figure 2.7a where a distribution of patterns ($\boldsymbol{x}_i=\{x_{1i},\ x_{2i}\}^T$ for $i=1,\ldots,16$) is divided in two groups (*i.e* blue circles and orange crosses), it is noticeable that multiple separation solutions (*i.e* hyperplanes) are possible.

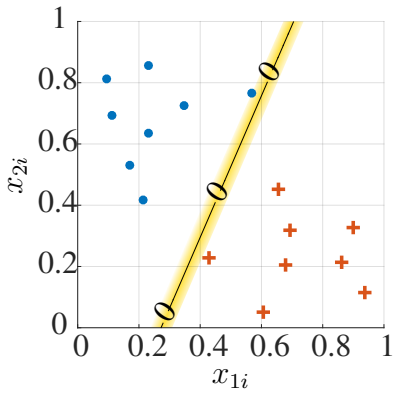
Figures 2.7b, 2.7c and 2.7d show three possible separator hyperplanes, represented by black lines. Likewise, the yellow bands that surround hyperplanes depict the margins, which correspond to the minimal distance between the hyperplane and the closest patterns.

Intuitively, large margins drive to better accuracies. Small margins drive to more miss classification. Higher margins drives to less dichotomy classification, and more generalization capacity.

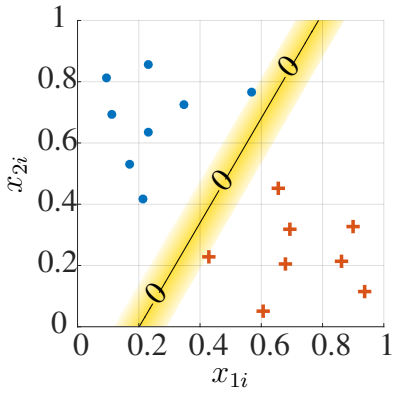
Accordingly, Support Vector Machines (SVMs) are a kind of classifiers that attempt to find a separator hyperplane that has the largest possible margin [57, 58]. In the following, SMV are introduced in three main stages: (i) definition of hyperplane, (ii) formulation of optimal margin, and (iii) solution to the optimization problem.



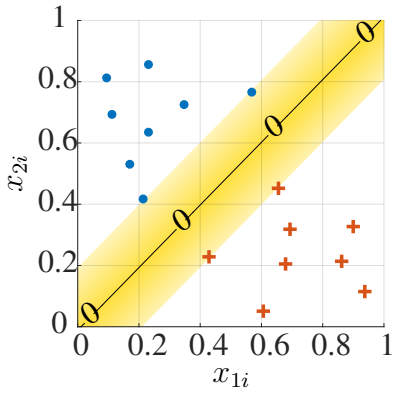
(a) Example of patterns distribution.



(b) Thin margin classification.



(c) Intermediate margin classification.



(d) Large margin classification.

Figure 2.7: Example of patterns distribution and three possible separator hyperplanes.

2.2.2.1 Definition of hyperplane

A hyperplane $\mathcal{H}$ in the $\mathbb{R}^n$ space (also noted $\mathcal{X}$ space) is an $(n-1)$ dimensional subspace of $\mathbb{R}^n$. Thus, we can consider it as the solution of the linear equation $\boldsymbol{w}^T\boldsymbol{x} = 0$, being $\boldsymbol{x}$ a vector inside the plane in $\mathbb{R}^n$. Similarly, from a geometric point of view, the hyperplane can be defined as the group of vectors in $\mathbb{R}^n$ that are perpendicular to the vector $\boldsymbol{w}$. Thus, the hyperplane is

defined plainly by the vector $\boldsymbol{w}$ as follows:

$$\mathcal{H}(\boldsymbol{w}) = \{\boldsymbol{x} \in \mathbb{R}^n \mid \boldsymbol{w}^T\boldsymbol{x} = 0\} \tag{2.1}$$

As introduced in the previous section, in the labeling stage of classification, a bias value b is required for the identification of patterns. Typically, to calculate the b value, the vector $\boldsymbol{w} = \{w_1, \ldots, w_n\}^T$ is modified by adding a scalar value w_0 at the beginning that multiplies the first position of the pattern vector $x_{0i} = 1$. Even that, in the present case the vector $\boldsymbol{w}$ is operating an entirely different role in the calculation of margins and is no longer convenient to have this notation. Thus, the hyperplane equation is rewritten including bias value b as follows:

$$\mathcal{H}(\boldsymbol{w}, b) = \{\boldsymbol{x} \in \mathbb{R}^n \mid \boldsymbol{w}^T\boldsymbol{x} + b = 0\} \tag{2.2}$$

2.2.2.2 Formulation of optimal margin

The margin is the distance $\boldsymbol{d}$ between the hyperplane $\mathcal{H}(\boldsymbol{w}, b)$ and the closest pattern defined by the point $\boldsymbol{x}_c$. Thus, the distance $\boldsymbol{d}$ is maximized to calculate hyperplane parameters $\mathcal{H}(\boldsymbol{w}, b)$ that classify the patterns correctly and achieve classes separation with the biggest possible margin. For that purpose, the following considerations are required:

1. If patterns $\boldsymbol{x}_i$ are linearly separable, then $\mid \boldsymbol{w}^T\boldsymbol{x}_i + b \mid > 0$ for all points.

2. If the point $\boldsymbol{x}_c$ is considered as the tip of a vector whose origin is the same of vector $\boldsymbol{w}$ (see figure 2.8), then the hyperplane $\mathcal{H}(\boldsymbol{w}, b)$ is normalized with respect to the margin by introducing $\mid \boldsymbol{w}^T\boldsymbol{x}_c + b \mid = 1$.

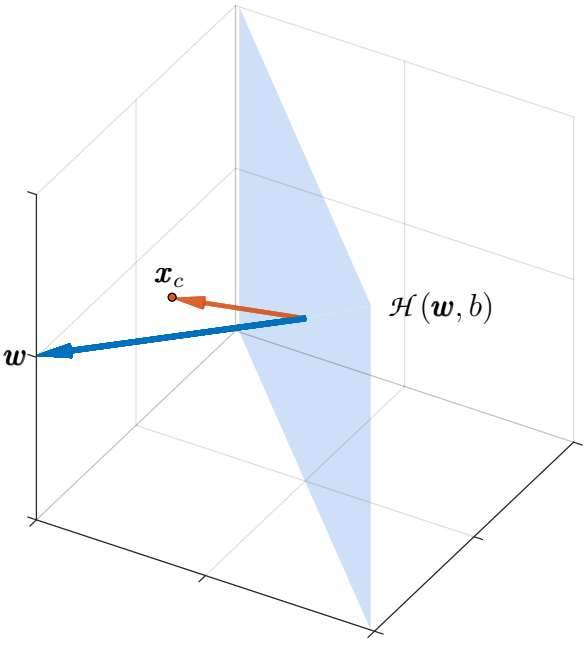


Figure 2.8: Vector $\boldsymbol{x}_c$.

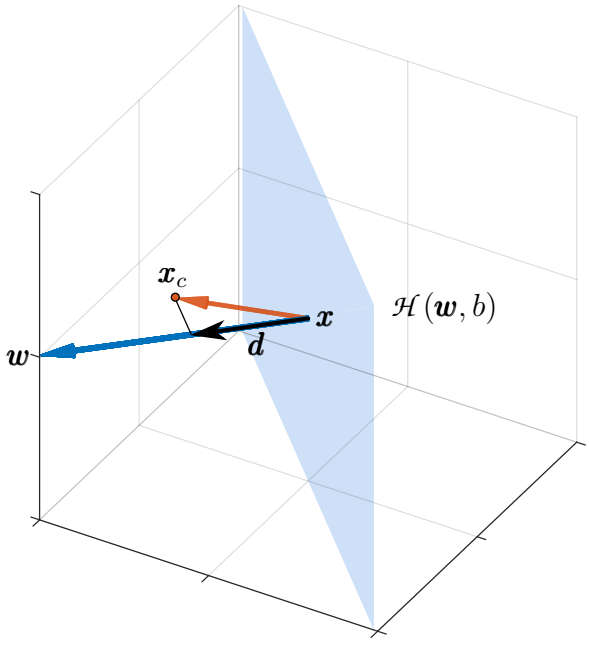


Figure 2.9: Distance $\boldsymbol{d} = \text{proj}_{\boldsymbol{w}}\,\boldsymbol{x}_c$.

Subsequently, the distance $\boldsymbol{d}$ can be formulated as the absolute value of the projection of the vector $\boldsymbol{x}_c$ onto the vector $\boldsymbol{w}$ as $\boldsymbol{d} = \mid \text{proj}_{\boldsymbol{w}}\, \boldsymbol{x_c} \mid$. Thus, calculating dot product between unit vector $\widehat{\boldsymbol{w}} = \boldsymbol{w} / \parallel \boldsymbol{w} \parallel$ and the vector $(\boldsymbol{x}_c - \boldsymbol{x})$ (see figure 2.9), the distance $\boldsymbol{d}$ can be computed as $\boldsymbol{d} = \mid \widehat{\boldsymbol{w}}^T(\boldsymbol{x}_c - \boldsymbol{x}) \mid$. Then, substituting the unit vector $\widehat{\boldsymbol{w}}$ we obtain $d = \mid \boldsymbol{w}^T\boldsymbol{x}_c - \boldsymbol{w}^T\boldsymbol{x} \mid / \parallel \boldsymbol{w} \parallel$. Moreover, if we consider that $\mid \boldsymbol{w}^T\boldsymbol{x}_c - \boldsymbol{w}^T\boldsymbol{x} \mid$ can be rewritten as $\mid \boldsymbol{w}^T\boldsymbol{x}_c + b - (\boldsymbol{w}^T\boldsymbol{x} + b) \mid$ without changing the result, we can formulate the distance as follows:

$$\boldsymbol{d} = \frac{1}{\parallel \boldsymbol{w} \parallel} \mid \boldsymbol{w}^T\boldsymbol{x}_c + b - (\boldsymbol{w}^T\boldsymbol{x} + b) \mid \tag{2.3}$$

Taking into account the normalization $\mid \boldsymbol{w}^T\boldsymbol{x}_c + b \mid = 1$, and equation (2.2), which states that $\boldsymbol{w}^T\boldsymbol{x} + b = 0$ for the hyperplane, the distance $\boldsymbol{d}$ becomes:

$$\boldsymbol{d} = \frac{1}{\parallel \boldsymbol{w} \parallel} \tag{2.4}$$

On the other hand, the maximization of the margin can be formulated as an optimization problem aiming to maximize the distances, subject to the normalization condition $\mid \boldsymbol{w}^T\boldsymbol{x}_c + b \mid = 1$. Thus, for a complete set of N patterns, the optimization problem must be written as:

$$\begin{aligned} \max \quad & \frac{1}{\parallel \boldsymbol{w} \parallel} \\ \text{subject to} \quad & \min_{i=1,\ldots,N} \mid \boldsymbol{w}^T\boldsymbol{x}_i + b \mid = 1 \end{aligned} \tag{2.5}$$

This is not a simple optimization problem because the constraints depend on an absolute value and the minimization of $\boldsymbol{x_i}$ inputs thus the problen is reformulated. Bearing in mind that during the training phase of classification each pattern $\boldsymbol{x}_i$ match an etiquette $y_i \in \{-1, 1\}$, then the expression $\mid \boldsymbol{w}^T\boldsymbol{x}_i + b \mid$ is equivalent to $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b)$. Likewise, if $\mid \boldsymbol{w}^T\boldsymbol{x}_i + b \mid = 1$, then the statement $y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1$ is also true. Furthermore, maximizing the expression $1/ \parallel \boldsymbol{w} \parallel$ is equivalent to minimize $\boldsymbol{w}^T\boldsymbol{w}/2$. As a result, the optimization problem can be reformulated as follows:

$$\begin{aligned} \min \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} \\ \text{subject to} \quad & y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 \text{ for } i = 1, \ldots, N \end{aligned} \tag{2.6}$$

Despite the fact that both approaches are not exactly equivalent, the obtained solution solving the optimization problem proposed in equation (2.6) satisfies the problem proposed in equation (2.5). Thus, the solution of equation (2.6) corresponds to the hyperplane $\mathcal{H}(\boldsymbol{w}, b)$ that separates patterns with the best possible margin.

2.2.2.3 Solution to the optimization problem

The optimization proposed in equation (2.6) is known as primal problem, and has a domain given by $\boldsymbol{w} \in \mathbb{R}^n$ and $b \in \mathbb{R}$. This problem can be solved using Quadratic Programing Problem

(QPP) algorithm with linear restrictions, and will have a solution within general theory of convex minimization [59]. For that purpose, we use the Lagrangian formulation which states that a minimization of a function $f(x)$, subject to equality constraints $h_j(x) = 0,\ \forall\ 1 \leq j \leq J$ and inequality constrains $g_i(x) \leq 0,\ \forall\ 1 \leq i \leq I$, can be rewritten as the following dual problem:

$$\begin{aligned} &\min_{x} \max_{\lambda,\mu} \quad \mathcal{L}(x, \lambda, \mu) \\ &\text{subject to} \quad \lambda \geq 0 \end{aligned} \tag{2.7}$$

where $\mathcal{L}(x, \lambda, \mu)$ is the following Lagrangian function:

$$\mathcal{L}(x, \lambda, \mu) = f(x) + \sum_i \lambda_i g_i(x) + \sum_j \mu_j h_j(x) \tag{2.8}$$

Moreover, the Karush–Kuhn–Tucker (KKT) conditions state that the gradient of the Lagrangian function $\nabla_x \mathcal{L}(x, \lambda, \mu)$ is equal to zero and the product between the inequality constrains and the Lagrange multipliers is also zero. These conditions can be written as:

$$\nabla_x \mathcal{L}(x, \lambda, \mu) = 0 \tag{2.9}$$

$$\lambda g_i(x) = 0 \text{ for all } i \tag{2.10}$$

Consequently, to formulate the Lagrangian dual problem for the SVM optimization problem, it is necessary to transform the inequality constraint proposed in equation (2.6) into the form $g_i(x) \leq 0$. Thus, the restriction equation can be written as $-(y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) - 1) \leq 0$ and the Lagrangian for the SVM primal problem is as follows:

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\lambda}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} - \sum_{i=1}^{N} \lambda_i \left(y_i \left(\boldsymbol{w}^T \boldsymbol{x}_i + b\right) - 1\right) \tag{2.11}$$

Taking into account that the Lagrangian is function of $\boldsymbol{w}$ and b, the condition established in equation (2.9) is calculated as the gradient of the Lagrangian with respect to $\boldsymbol{w}$, and the partial derivative regarding b. Thus, the following conditions are introduced:

$$\nabla_{\boldsymbol{w}} \mathcal{L} = \boldsymbol{w} - \sum_{i=1}^{N} \lambda_i y_i \boldsymbol{x}_i = 0 \tag{2.12}$$

$$\frac{\partial \mathcal{L}}{\partial b} = -\sum_{i=1}^{N} \lambda_i y_i = 0 \tag{2.13}$$

Furthermore, solving equation (2.12), a solution for the vector $\boldsymbol{w}$ as function of the Lagrangian multipliers $\boldsymbol{\lambda}$ are formulated as follows:

$$\boldsymbol{w} = \sum_{i=1}^{N} \lambda_i y_i \boldsymbol{x}_i \tag{2.14}$$

Thereafter, substituting equations (2.13) and (2.14) in equation (2.11), we obtain a Lagrangian formulation that is independent of $\boldsymbol{w}$ and b as follows:

$$\mathcal{L}(\boldsymbol{\lambda}) = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y_i y_j \lambda_i \lambda_j \boldsymbol{x}_i^T \boldsymbol{x}_j \tag{2.15}$$

As a consequence, the dual problem of SVM can be formulated as presented below:

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y_i y_j \lambda_i \lambda_j \boldsymbol{x}_i^T \boldsymbol{x}_j - \sum_{i=1}^{N} \lambda_i \\ \text{subject to} \quad & \lambda_i \geq 0 \text{ for } i = 1, \ldots, N \\ & \sum_{i=1}^{N} \lambda_i y_i = 0 \end{aligned} \tag{2.16}$$

The optimization problem presented in equation (2.16) can be reformulated in matrices notation isolating the Lagrange coefficients λ of the equation (2.15) as follows:

$$-\mathcal{L}(\boldsymbol{\lambda}) = \frac{1}{2} \boldsymbol{\lambda}^T \boldsymbol{Q} \boldsymbol{\lambda} + (-\mathbf{1}^T) \boldsymbol{\lambda} \tag{2.17}$$

where $-\mathbf{1}$ is a vector whose elements are -1, and $\boldsymbol{Q}$ is the quadratic element, which is formulated as follows:

$$\boldsymbol{Q} = \begin{bmatrix} y_1 y_1 \boldsymbol{x}_1^T \boldsymbol{x}_1 & \ldots & y_1 y_N \boldsymbol{x}_1^T \boldsymbol{x}_N \\ \vdots & \ddots & \vdots \\ y_N y_1 \boldsymbol{x}_N^T \boldsymbol{x}_1 & \ldots & y_N y_N \boldsymbol{x}_N^T \boldsymbol{x}_N \end{bmatrix} \tag{2.18}$$

As a result, the dual SVM optimization problem is formulated as following:

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & \frac{1}{2} \boldsymbol{\lambda}^T \boldsymbol{Q} \boldsymbol{\lambda} + (-\mathbf{1}^T) \boldsymbol{\lambda} \\ \text{subject to} \quad & \boldsymbol{y}^T \boldsymbol{\lambda} = 0 \\ & \mathbf{0} \leq \boldsymbol{\lambda} \leq \boldsymbol{\infty} \end{aligned} \tag{2.19}$$

Then, the problem can be solved using a classical QPP solver, which receives as inputs: (i) the quadratic coefficients $\boldsymbol{Q}$, (ii) the linear coefficients $-\mathbf{1}$, (iii) the linear constraints $\boldsymbol{y}^T \boldsymbol{\lambda} = 0$, and (iv) the upper and lower bounds $\mathbf{0} \leq \boldsymbol{\lambda} \leq \boldsymbol{\infty}$. The result of the QPP algorithm is a vector containing $\boldsymbol{\lambda}$ values. Once the solution for the Lagrange multipliers is done, equation (2.14) is used to calculate the vector $\boldsymbol{w}$. The last KKT condition, introduced in equation (2.10), states that the product of the Lagrange multipliers and the inequality restriction function is equal to zero. Taking into account that in the SVM problem the inequality constraint is given by $(y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b) - 1)$, the last KKT condition is formulated as follows:

$$\lambda_i (y_i (\boldsymbol{w}^T \boldsymbol{x}_i + b) - 1) = 0 \tag{2.20}$$

As presented before, the normalization $\mid \boldsymbol{w}^T \boldsymbol{x}_c + b \mid = 1$ is equivalent to $y_c(\boldsymbol{w}^T \boldsymbol{x}_c + b) = 1$, for points $\boldsymbol{x}_c$ lying the margin. Thus, for the points $\boldsymbol{x}_c$, equation (2.20) becomes $\lambda_i(1 - 1) = 0$, showing that any positive value of λ_c fulfill the last KKT condition. On the other hand, for

points $\boldsymbol{x}_o$ that do not lay the margin, the normalization states that $y_o(\boldsymbol{w}^T\boldsymbol{x}_o + b) > 1$. Hence, λ_o values must be zero to meet the condition formulated in equation (2.20). As a result, the vector $\boldsymbol{\lambda}$ issue of the QPP solution will be mainly composed of zeros, and the non zero values are Lagrange multipliers of the so-called support vectors $\boldsymbol{x}_c$.

For instance, table 2.7 shows the Lagrange multipliers calculated for the linear separable problem proposed in figure 2.7a. The $\boldsymbol{\lambda}$ values are zero, and only three of them have non zero value. Thus, in this example the support vectors are the patterns $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_{10}$.

Table 2.7: Example patterns and Lagrange multipliers.

i	Pattern $\boldsymbol{x}_i$		Class y_i	Lagrange multipliers λ_i
	x_{1i}	x_{2i}		
1	**0.213**	**0.418**	**1**	**7.754**
2	**0.568**	**0.765**	**1**	**16.996**
3	0.346	0.724	1	0.000
4	0.231	0.634	1	0.000
5	0.169	0.529	1	0.000
6	0.111	0.692	1	0.000
7	0.093	0.812	1	0.000
8	0.231	0.855	1	0.000
9	0.427	0.228	-1	0.000
10	**0.655**	**0.453**	**-1**	**24.750**
11	0.694	0.319	-1	0.000
12	0.678	0.205	-1	0.000
13	0.604	0.051	-1	0.000
14	0.860	0.214	-1	0.000
15	0.899	0.328	-1	0.000
16	0.938	0.115	-1	0.000

The equation (2.14) is used in order to calculate parameters $\boldsymbol{w}$ of the hyperplane $\mathcal{H}(\boldsymbol{w}, b)$. Moreover, to calculate the b value, we use the normalization $y_c(\boldsymbol{w}^T\boldsymbol{x}_i + b) = 1$, substituting $\boldsymbol{x}_i$ by one of the found support vectors; the value of b is the same using any of the support vectors. Furthermore, figure 2.10 shows a graphical representation of the classification results, the figure includes: (i) a black line, with a zero over it, representing the separation hyperplane, (ii) two black lines, with the etiquettes -1 and 1 over them, representing the positive and negative limits of the margin, and (iii) the three support vectors highlighted (annotated SV) by blue circles (where 2 concern the blue class and 1 concerns the red class).

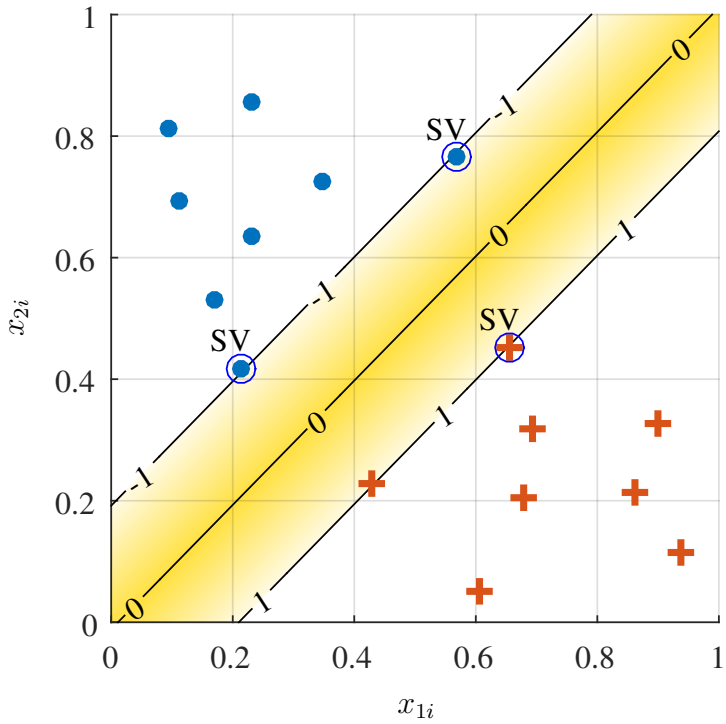


Figure 2.10: Classification results of the SVM classifier.

2.2.3 Non-linear SVM classifiers

Non-linear SVM classifiers deal with the problematic of non-separable patterns, which can be mainly classified into two categories: slightly and heavily non-separable. For instance, figure 2.11 illustrates the slightly non-separable case, where the patterns are divided into two classes, which are clearly located at the top-left and lower-right sides of the feature space; due to those misplaced patterns a linear classification is not possible. However, handling the error produced by misplaced patterns as deviated points, the so-called soft margin SVM can perform a non-linear separation.

Furthermore, figure 2.12 presents an example of heavily non-separable, where orange cross patterns are surrounded by blue points patterns. In such situations, a representation of patterns in hyperspace is used; this representation is performed through kernel functions. In the following, the soft margin SVM and the kernel functions are introduced.

2.2.3.1 Soft margin SVM

When patterns are distorted, *i.e.* are coming from noised data, some points could diverge with respect to the class, resulting in an apparently non-separable distribution. The first logical attempt, during the learning process, could be the utilization of a non-linear classifier (see figure 2.13), even that, considering that the deviation is produced by noise, the classifier learns useless information, which is known as over-fitting.

In order to improve generalization and reduce over-fitting during the training of the SVM,

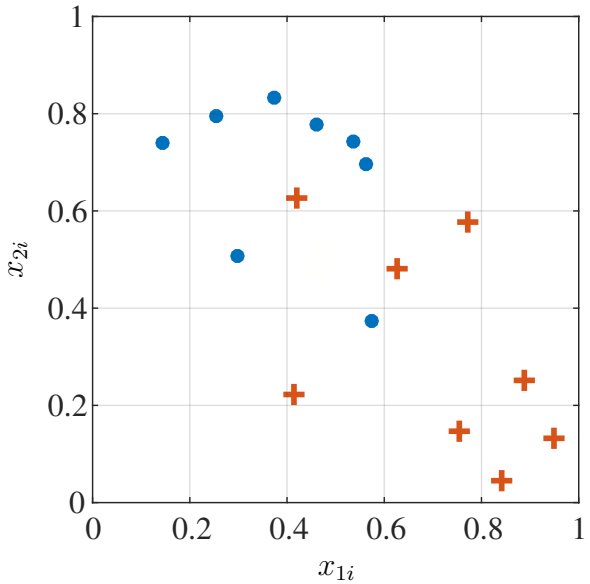


Figure 2.11: Slightly non separable data.

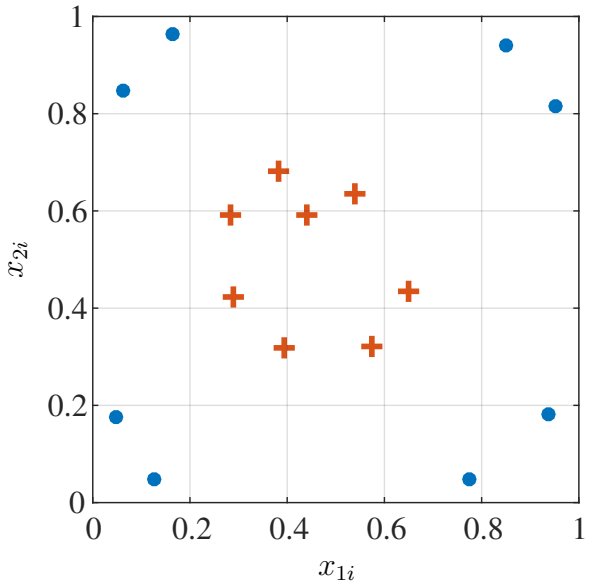


Figure 2.12: Heavily non separable data.

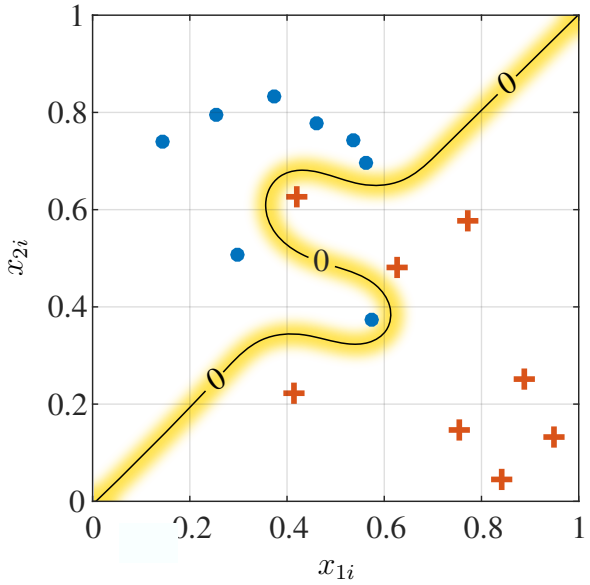


Figure 2.13: Non-linear separation.

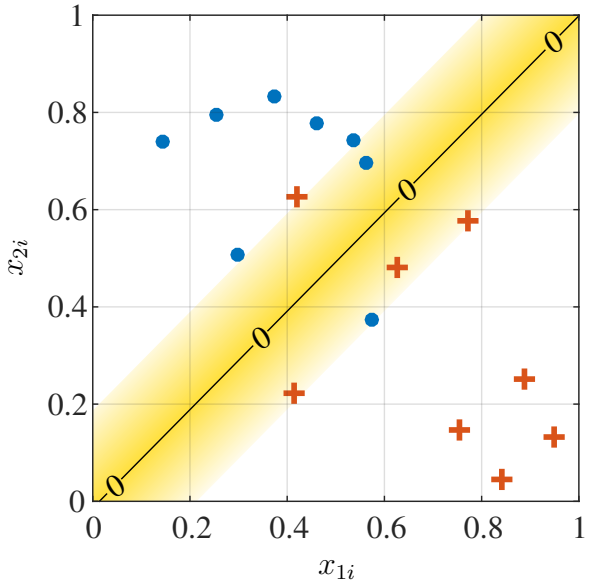


Figure 2.14: Linear separation accepting margin violations.

each pattern is evaluated, and when it introduces an error in the classification, it is penalized. As a result, the impact of distorted patterns in the margin is reduced. Figure 2.14 shows a separator hyperplane, calculated to classify patterns despite the two misplaced points.

The implementation of the soft margin SVM requires a measure of the margin violation, *i.e.* a numerical value ξ_i, called slack that indicates how deep a point is inside the margin, or even how deep is on the wrong side of the hyperplane, as shown in figure 2.15. Besides, this slack ξ_i influence the normalization that must be formulated as:

$$y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \xi_i, \text{ for } \xi_i \geq 0 \tag{2.21}$$

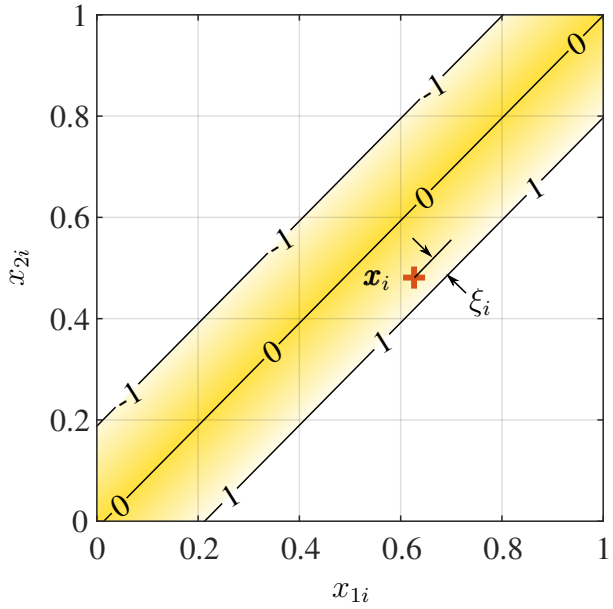


Figure 2.15: Measure of slack ξ_i with respect to the margin.

The weighted total violation of the margin can be quantified as the summation of slack $C\sum\xi_i$, for $i = 1, \ldots, N$ being N the total number of patterns and C is a constant that gives the relative importance of the slack with respect the margin. Thus, the primal problem of the SVM proposed in equation (2.6), must be modified to include the slack penalization as follows:

$$\begin{aligned} \min \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i \\ \text{subject to} \quad & y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \qquad\qquad \text{for } i = 1, \ldots, N \end{aligned} \tag{2.22}$$

where $\boldsymbol{w} \in \mathbb{R}^N$, $b \in \mathbb{R}$, and $\xi \in \mathbb{R}^N$.

The new Lagrangian formulation for the soft margin, includes the weighted total violation of the margin and two different Lagrange multipliers: $\boldsymbol{\lambda}$ for the first inequality constraint and $\boldsymbol{\varsigma}$ for the second one. Hence, the Lagrangian is formulated as follows:

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\varsigma}) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\lambda_i\left(y_i\left(\boldsymbol{w}^T\boldsymbol{x}_i + b\right) - 1 + \xi_i\right) - \sum_{i=1}^{N}\varsigma_i\xi_i \tag{2.23}$$

Similarly, the Lagrangian must be: (i) maximized with respect to: $\boldsymbol{w}$, b, and $\boldsymbol{\xi}$ and (ii) minimized regarding each $\lambda \geq 0$ and $\varsigma \geq 0$. Applying the KKT conditions, we find the following expressions:

$$\nabla_{\boldsymbol{w}}\,\mathcal{L} = \boldsymbol{w} - \sum_{i=1}^{N}\lambda_i y_i \boldsymbol{x}_i = 0 \tag{2.24}$$

$$\frac{\partial\,\mathcal{L}}{\partial b} = -\sum_{i=1}^{N}\lambda_i y_i = 0 \tag{2.25}$$

$$\nabla_{\boldsymbol{\xi}} \mathcal{L} = \mathbf{1}C - \boldsymbol{\lambda} - \boldsymbol{\varsigma} = 0 \tag{2.26}$$

Thereafter, substituting equations (2.24) to (2.26) in equation (2.23), we obtain a Lagrangian formulation that is independent of $\boldsymbol{w}$, b and $\boldsymbol{\xi}$ shown in equation (2.27) below, which is exactly the same as for the classic SVM introduced in equation (2.15).

$$\mathcal{L}(\boldsymbol{\lambda}) = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y_i y_j \lambda_i \lambda_j \boldsymbol{x}_i^T \boldsymbol{x}_j \tag{2.27}$$

Moreover, according to equation (2.26), and taking into account that $\boldsymbol{\varsigma} \geq 0$, $\boldsymbol{\lambda}$ is lower than $\mathbf{1}C$. Then, the dual problem can be formulated in matrix notation just adding $\mathbf{1}C$ as the upper bound of $\boldsymbol{\lambda}$. As a result, the complet dual problem for soft margin SVM is as follows:

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & \frac{1}{2} \boldsymbol{\lambda}^T \boldsymbol{Q} \boldsymbol{\lambda} + (-\mathbf{1}^T) \boldsymbol{\lambda} \\ \text{subject to} \quad & \boldsymbol{y}^T \boldsymbol{\lambda} = 0 \\ & \mathbf{0} \leq \boldsymbol{\lambda} \leq \mathbf{1}C \end{aligned} \tag{2.28}$$

Then, the problem can be solved using a classical QPP solver, which receives as inputs: (i) the quadratic coefficients $\boldsymbol{Q}$, (ii) the linear coefficients $-\mathbf{1}$, (iii) the linear constraints $\boldsymbol{y}^T \boldsymbol{\lambda} = 0$, and (iv) the upper and lower bounds $\mathbf{0} \leq \boldsymbol{\lambda} \leq \mathbf{1}C$. Concerning the value of the parameter C, high values lead to linear rigid SVM problem because the limits of the Lagrange multipliers go back to $\mathbf{0} \leq \boldsymbol{\lambda} \leq \boldsymbol{\infty}$, and small values produce very soft margins. Parameters $\boldsymbol{w}$ and b of the hyperplane $\mathcal{H}(\boldsymbol{w}, b)$, are calculated in the same way as for the linear SVM problem.

Finally, we have three types of support vectors $\boldsymbol{x}_c$: (i) margin support vectors when $0 < \lambda_c < C$ and $\xi_c = 0$, (ii) non-margin support vectors when $\lambda_c = C$ and $\xi_c > 0$, and (iii) misclassified non-margin support vectors also $\lambda_c = C$ and $\xi_c > 0$. Figure 2.16 exemplary shows the results of a separator hyperplane calculated using a soft margin and examples of the three kind of support vectors.

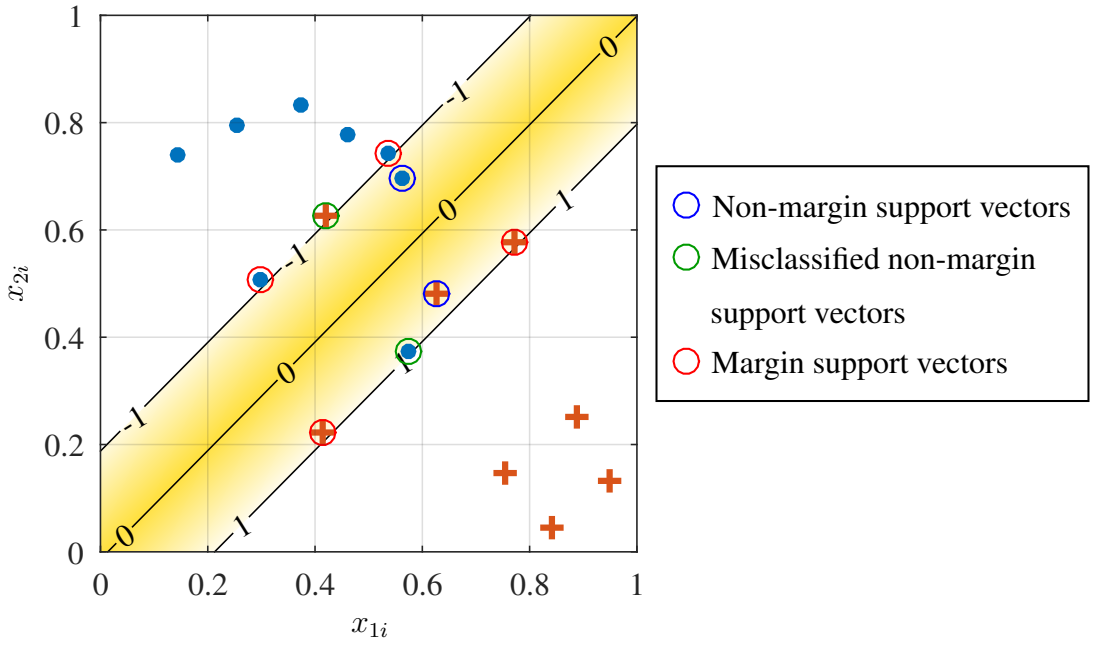


Figure 2.16: Type of support vectors issued of soft margin SVM classification.

2.2.3.2 Non-linear SVM classifier based on kernel functions

Heavily non-separable classification are problems on which none of the possible dichotomy can linearly separate the classes. The approach in that cases tries to map patterns into a higher dimensional space. The central concept is to let the SVM perform a linear classification but in a hyperspace where patterns are separable. Hitherto, patterns were considered as vectors belonging the $\mathcal{X}$ space. Now, it is necessary to map those vectors into a high-dimensional space $\mathcal{Z}$. As consequence, a function Φ is introduced such that $\Phi : \mathcal{X} \mapsto \mathcal{Z}$. For example, figure 2.17 shows a constellation of points belonging a 2-dimensional space that are non-linearly separably, then using the mapping function, vectors are transformed into an higher space where they are linearly separable through a hyperplane $\mathcal{H}(\boldsymbol{w}, b) = \{\boldsymbol{z} \in \mathcal{Z} \mid \boldsymbol{w}^T \boldsymbol{z} + b = 0\}$.

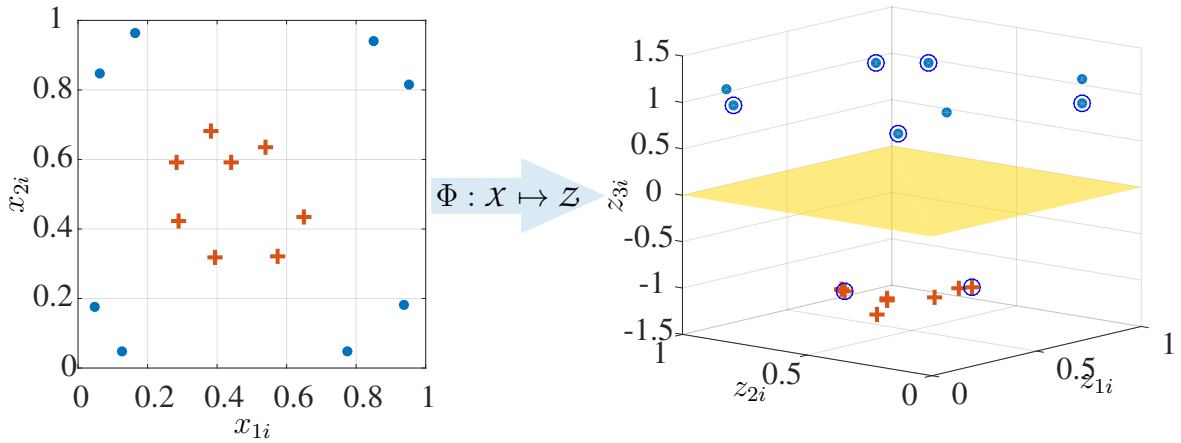


Figure 2.17: Mapping vectors into a higher-dimensional space.

Considering the dual problem of the soft margin SVM, as presented in Figure 2.28, the only essential modification, that arises from mapping $\mathcal{X} \mapsto \mathcal{X}$, is the necessity of substituting the inner product of $\boldsymbol{x}_i^T \boldsymbol{x}_j$ by $\boldsymbol{z}_i^T \boldsymbol{z}_j$. Thus the Lagrangian is formulated as follow:

$$\mathcal{L}(\boldsymbol{\lambda}) = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y_i y_j \lambda_i \lambda_j \boldsymbol{z}_i^T \boldsymbol{z}_j \tag{2.29}$$

Likewise, to formulate the Lagrangian in matrices notation, the quadratic element $\boldsymbol{Q}$ is modified using the same consideration with respect to the inner product. As a result, the following quadratic element $\boldsymbol{Q}$ is introduced:

$$\boldsymbol{Q} = \begin{bmatrix} y_1 y_1 \boldsymbol{z}_1^T \boldsymbol{z}_1 & \dots & y_1 y_N \boldsymbol{z}_1^T \boldsymbol{z}_N \\ \vdots & \ddots & \vdots \\ y_N y_1 \boldsymbol{z}_N^T \boldsymbol{z}_1 & \dots & y_N y_N \boldsymbol{z}_N^T \boldsymbol{z}_N \end{bmatrix} \tag{2.30}$$

As a result, the dual problem of the non-linear SVM, is formulate as follows:

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & \frac{1}{2} \boldsymbol{\lambda}^T \boldsymbol{Q} \boldsymbol{\lambda} + (-\mathbf{1}^T) \boldsymbol{\lambda} \\ \text{subject to} \quad & \boldsymbol{y}^T \boldsymbol{\lambda} = 0 \\ & \mathbf{0} \leq \boldsymbol{\lambda} \leq \mathbf{1} C \end{aligned} \tag{2.31}$$

The dual problem is the same as for the soft margin SVM. Consequently, the constant C, which gives the relative importance of the slack with respect the margin, can be manipulated. Thus, if C approaches infinity, then the non-linear SVM has a rigid margin; and for finite values of C, the non-linear SVM can be considered to have a soft margin. On the other hand, it is evident that it is not necessary to transform the complete set of patterns into the high dimensional space. Instead, it is enough to have a function κ to calculate inner product of $\boldsymbol{z}_i^T \boldsymbol{z}_j$. Usually in the state of the art those functions κ are known as kernels.

Kernels are functions that allow calculating the inner product of two vectors (formed by points or patterns) represented in a high dimensional space without going to that space. Therefore, the inputs of the Kernel function κ are patterns in the space $\mathcal{X}$, and the output of the function is the inner product of the equivalent patterns in the space $\mathcal{Z}$. Thus, the function is formulated as $\kappa(\boldsymbol{x}_i, \boldsymbol{z}_j) = \boldsymbol{x}_i^T \boldsymbol{z}_j$ and the quadratic element $\boldsymbol{Q}$ of the Lagrangian becomes:

$$\boldsymbol{Q} = \begin{bmatrix} y_1 y_1 \kappa(\boldsymbol{x}_1, \boldsymbol{x}_1) & \dots & y_1 y_N \kappa(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ y_N y_1 \kappa(\boldsymbol{x}_N, \boldsymbol{x}_1) & \dots & y_N y_N \kappa(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{bmatrix} \tag{2.32}$$

To be valid, a kernel function should accomplish two conditions: (i) considering that its result is an inner product, it must be symmetric $\kappa(\boldsymbol{x}_i, \boldsymbol{z}_j) = \kappa(\boldsymbol{x}_j, \boldsymbol{z}_i)$ and (ii) according to the Mercer's condition a kernel matrix $\boldsymbol{\varrho} = (\kappa(\boldsymbol{x}_i, \boldsymbol{z}_j))_{ij}$ must be positive semidefinite (*i.e.* it should be greater or equal to zero for any point $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$). The following are two classic examples of kernel functions:

1. Degree–d polynomial kernel:

$$\kappa(\boldsymbol{x}_i, \boldsymbol{z}_j) = (\boldsymbol{x}_i^T \boldsymbol{z}_j + 1)^d \tag{2.33}$$

2. Radial basis function (RBF):

$$\kappa(\boldsymbol{x}_i, \boldsymbol{z}_j) = \mathrm{e}^{-\frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{2\sigma^2}} \tag{2.34}$$

where σ denotes a positive parameter for controlling the radius

On the other hand, applying the KKT condition $\nabla_{\boldsymbol{w}} \mathcal{L} = 0$, the expression to compute $\boldsymbol{w}$, based on the N_{SV} presents support vectors, as follows:

$$\boldsymbol{w} = \sum_{c=1}^{N_{SV}} \lambda_c y_c \boldsymbol{z}_c \tag{2.35}$$

Once the Lagrange multipliers and $\boldsymbol{w}$ are calculated, we define a decision function to evaluate a pattern $\boldsymbol{x}_i$. Nevertheless, taking into account that $\boldsymbol{w}$ was calculate using a $\boldsymbol{Q}$ matrix depending on $\boldsymbol{z}_i$, the decision function is formulated as regarding $\boldsymbol{z}_i$ instead $\boldsymbol{x}_i$ as follows:

$$\Gamma(\boldsymbol{x}_i) = \text{sign}\left(\boldsymbol{w}^T \boldsymbol{z}_i + b\right) \tag{2.36}$$

The main disadvantage of equation (2.36) is that $\boldsymbol{z}_i$ are unknown representations of $\boldsymbol{x}_i$ in the hyperspace $\mathcal{Z}$. Even that, substituting equation (2.35) into equation (2.36), we obtain the following expression:

$$\Gamma(\boldsymbol{x}_i) = sign\left(\sum_{c=1}^{N_{SV}} \lambda_c y_c \boldsymbol{z}_c \boldsymbol{z}_i + b\right) \tag{2.37}$$

Subsequently, the inner product $\boldsymbol{z}_c \boldsymbol{z}_i$ is substituted by the kernel function $\kappa(\boldsymbol{x}_c, \boldsymbol{x}_i)$ and equation (2.37) is reformulated as follows:

$$\Gamma(\boldsymbol{x}_i) = sign\left(\sum_{c=1}^{N_{SV}} \lambda_c y_c \kappa(\boldsymbol{x}_c, \boldsymbol{x}_i) + b\right) \tag{2.38}$$

The constant b of the separator hyperplane, is calculated with the support vectors $\boldsymbol{x}_c$ as follows:

$$b = y_c - \sum_{\lambda_m > 0} \lambda_m y_m \kappa(\boldsymbol{x}_m, \boldsymbol{x}_c) \tag{2.39}$$

As an illustration, the pattern distribution, displayed in figure 2.17, is solved using: (i) the dual SVM problem presented in equation (2.31), (ii) the RBF kernel function described in equation (2.34), and (iii) rigid margin *i.e.* ($C \to \infty$). As shown in figure 2.18, the separator hyperplane $\mathcal{H}(\boldsymbol{w}, b) = \{\boldsymbol{z} \in \mathcal{Z} \mid \boldsymbol{w}^T \boldsymbol{z} + b = 0\}$ and the margins $\mid \boldsymbol{w}^T \boldsymbol{z} + b \mid = 1$ delineate a circular trajectory. Likewise, all patterns are on the right side of the hyperplane; red crosses are inside, and blue circles are outside. Regarding support vectors that are surrounded by blue

circles, some of them seem to violate the margin, which is unusual considering the imposed rigid margin $C \to \infty$. However, this behavior of support vector can be explained by analyzing the classification in the hyperspace.

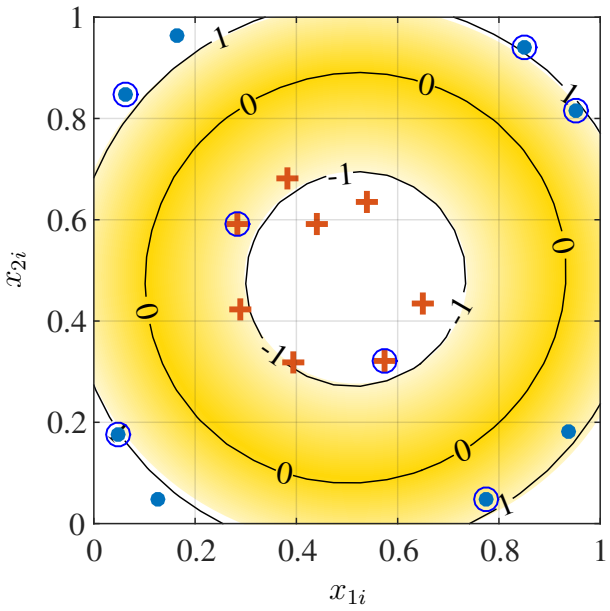


Figure 2.18: Hyperspace top view.

Bearing in mind that the calculation of the inner product in the hyperspace $\mathcal{Z}$ is performed using an RBF kernel function, the arrangement of patterns $\boldsymbol{z}_i$ reflects a Gaussian distribution, in which classes are projected to opposite sides of the belt as shown in figure 2.19. In the same figure, we can observe the hyperplane $\mathcal{H}(\boldsymbol{w}, b) = \{\boldsymbol{z} \in \mathcal{Z} \mid \boldsymbol{w}^T\boldsymbol{z} + b = 0\}$ and the normalized limits of vast optimal margin where $\mid \boldsymbol{w}^T\boldsymbol{z} + b \mid = 1$.

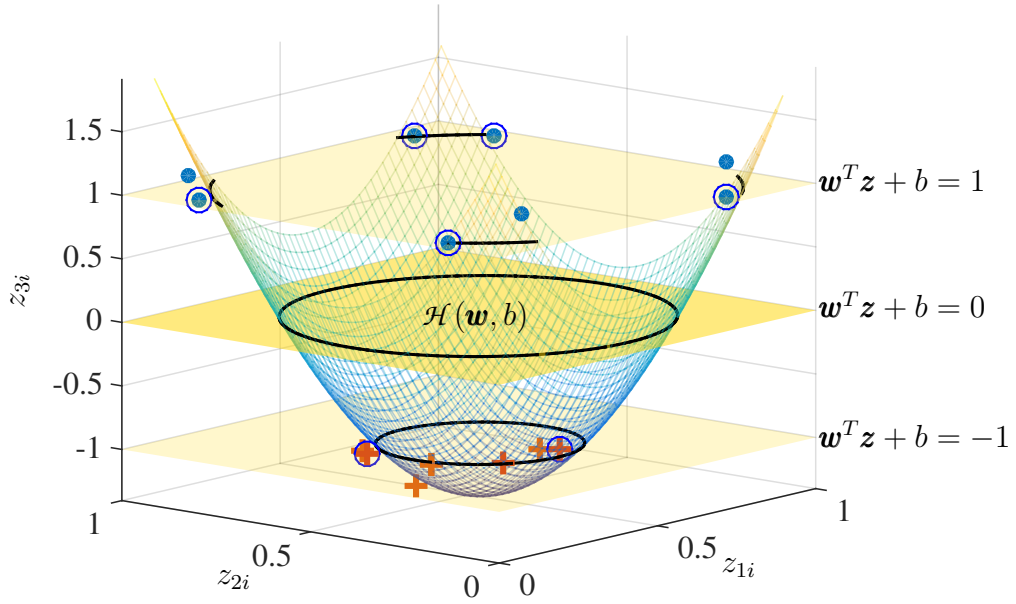


Figure 2.19: Hyperspace 3D view.

Besides Figure 2.20, which corresponds to a side view of Figure 2.19, shows the margin and the support vectors which are in laying the margin. Summing-up, the non-linear support vector machines use the kernel functions as a method to map patterns into a high dimensional space. Thus, classification is possible using the same linear formulation of linear SVM, but substituting the inner product of features by the kernel function. Moreover, the obtained representation of the hyperplane, in the case of 2D, represents the intersection between the separator hyperplane and the surface produced by the kernel function. Finally, the support vectors appear in the wrong position of the margin, but in the space $\mathcal{Z}$, they are correctly located.

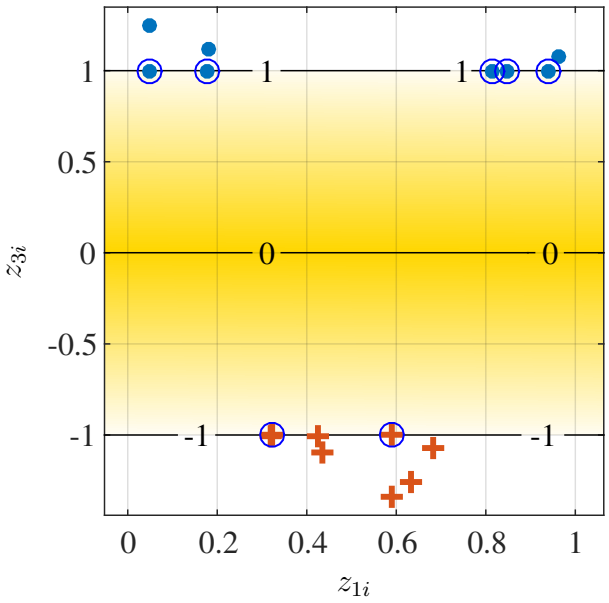


Figure 2.20: Classification results.

2.3 Movement SVMs classifier using sEMG signals

We have considered SVMs to classify upper limb motion based on patterns, because of their above mentioned advantages. The patterns is the features extracted from sEMG signals from upper limb muscles[1]. Thus, the proposed methodology consist in: (i) obtaining sEMG signal from muscles, (ii) extracting sEMG produced during upper limb motion (iii) getting features from sEMG (iv) establishing patterns (v) applying SVMs classifier in order to identify classes, which correspond to the motion. In order to illustrate the procedure, Figure 2.21 presents the methodology used.

[1]We only consider sEMG produced during movements in order to obtain properly patterns

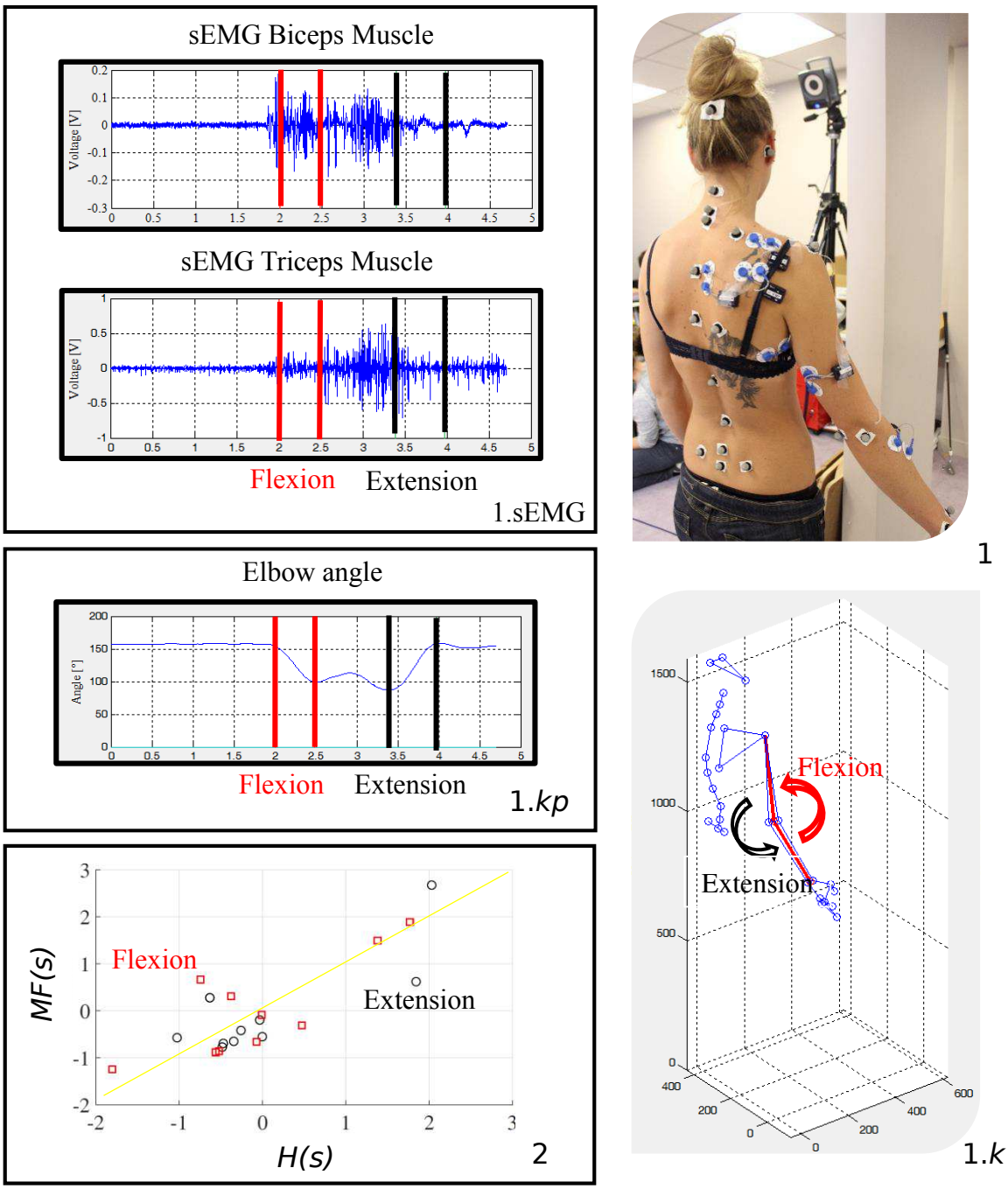


Figure 2.21: SVM scheme: (1) illustrates real environment motion capture laboratory in which the subject are equipped with sEMGsensors and retro-reflective markers; (1.*k*) plots tracking kinematic; (1.sEMG) plots B and T sEMG obtained during flexion and extension, which are represents by red and black lines respectively; (1.*kp*) illustrates elbow angle calculated based on (1.k); (2) shows the features space in which red squares and black circles are elbow flexion and extension.

2.4 Experiments

We carried out an experiment with the aim to assess the sSVM performance in upper limb motion classification, considering the same material and data processing as in sections 1.4.1 and 1.4.2.

In addition, entropy and mean frequency were selected in Section 1.5 as the couple features with the highest Euclidean distance between them. Entropy is indicated as H, mean frequency as MF, biceps and triceps muscles are respectively denoted by B and T.

Therefore, with these selected features, we define matrix $\boldsymbol{X}$ which is composed of:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x_1} & \cdots & \boldsymbol{x}_N \end{bmatrix} \tag{2.40}$$

where, x_i correspond to different trial, that are elbow flexion in the first part (where i is verying between 1 and $N/2$) and elbow extension in the second part (where i is varying between $N/2 + 1$ and N) Furthermore, it is given by:

$$\boldsymbol{x_i} = \begin{Bmatrix} H(\boldsymbol{s}_{Bi}) \\ MF(\boldsymbol{s}_{Bi}) \\ H(\boldsymbol{s}_{Ti}) \\ MF(\boldsymbol{s}_{Ti}) \end{Bmatrix} \tag{2.41}$$

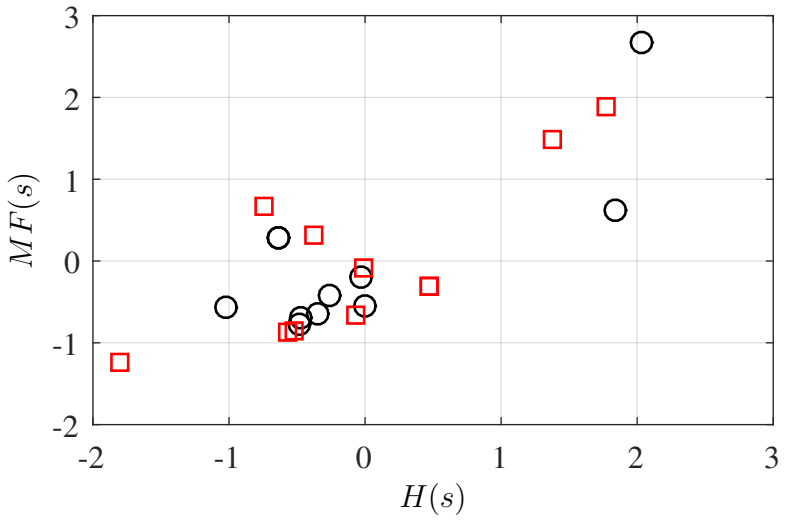
where, $H(\boldsymbol{s}_{Bi})$ corresponds to entropy value of $\boldsymbol{s}_{Bi}$. Moreover, $\boldsymbol{s}_{Bi}$ and $\boldsymbol{s}_{Ti}$ are respectively the biceps and triceps recorded sEMG signals .

Table 2.8 shows the extracted values of the selected features stored in the matrix $\boldsymbol{X}$ for 3 subjects and 22 trials. Subject 1, 2 and 3 correspond to (x_1),(x_2 ˘ x_5),(x_6 ˘ $x_2$2) respectively.

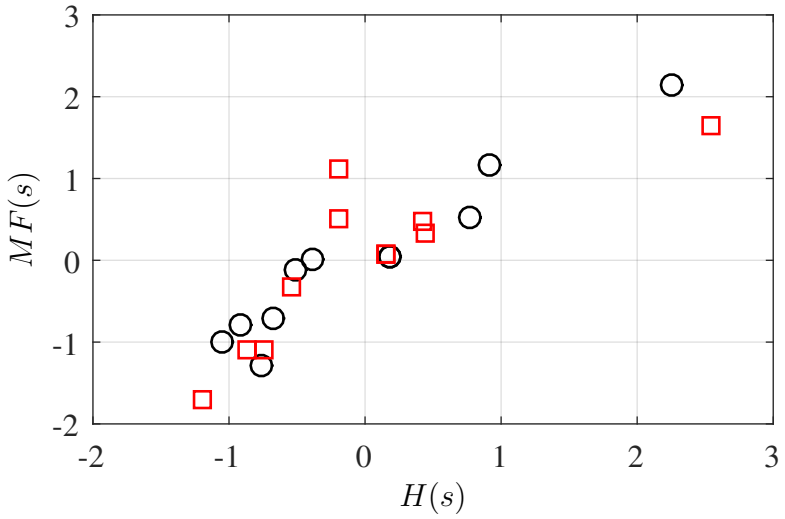
Flexion					**Extension**				
Elements of x_i	**Biceps**		**Triceps**		Elements of x_i	**Biceps**		**Triceps**	
	H	MF	H	MF		H	MF	H	MF
x_1	-0.47	-0.69	-0.07	-0.66	x_{12}	2.25	2.14	2.55	1.65
x_2	-0.26	-0.42	-0.57	-0.88	x_{13}	-1.05	-1	-0.86	-1.1
x_3	-0.03	-0.19	-0.52	-0.86	x_{14}	-0.92	-0.79	-0.74	-1.09
x_4	-0.63	0.28	0.47	-0.31	x_{15}	0.19	0.04	0.16	0.07
x_5	0	-0.55	-0.01	-0.09	x_{16}	-0.76	-1.28	-1.2	-1.71
x_6	2.03	2.67	1.38	1.49	x_{17}	0.91	1.16	0.45	0.34
x_7	-1.02	-0.57	-0.74	0.66	x_{18}	-0.51	-0.12	-0.19	1.12
x_8	-0.63	0.28	0.47	-0.31	x_{19}	0.19	0.04	0.16	0.07
x_9	-0.35	-0.64	-1.8	-1.25	x_{20}	-0.68	-0.71	-0.19	0.5
x_{10}	1.84	0.62	1.77	1.89	x_{21}	0.77	0.52	0.42	0.47
x_{11}	-0.48	-0.77	-0.38	0.31	x_{22}	-0.39	0.01	-0.54	-0.32

Table 2.8: Description of $\boldsymbol{X}$ elements from x_1 to x_{22}.

Features distribution from B muscle is shown in Figure 2.22a, considering the feature selected, the black circles represents elbow flexion and red squares illustrates elbow extension. Moreover, Figure 2.22b plots MF(s) and H(s) feature values obtained from T muscle. Considering the dispersion of the data classes a linear separation is not possible.



(a) Features of B.



(b) Features of T.

Figure 2.22: Distribution of features for flexion and extension, red circles presents flexion movement and black squares extension movement.

The matrix $\boldsymbol{X}$, is considered as a set of points in a 4-th dimensional features space, wherein each i-th pattern $\boldsymbol{x}_i \in \mathbb{R}^4$, has an etiquette $y_i \in \{1, -1\}$. The etiquette classifies the pattern into two categories: flexion or extension.

Thus, the features extracted from B and T are separate into two classes flexion and extension. This classification has been performed using a non-linear SVM machine with soft margin, as explained in section 2.2.3.

The support vectors are selected as explained in section 2.2.2.3, the decision function $\Gamma(\boldsymbol{x}_i)$ is applied to the remaining points $\boldsymbol{x}_i$ of the matrix $\boldsymbol{X}$. Then the result of $\Gamma(\boldsymbol{x}_i)$ is compared with y_i. If the values match we consider that the SVM has correctly classified the point $\boldsymbol{x}_l$. As a result, using a non optimal settings of σ and C, the algorithm may achieves a classification percentage of 90.91%.

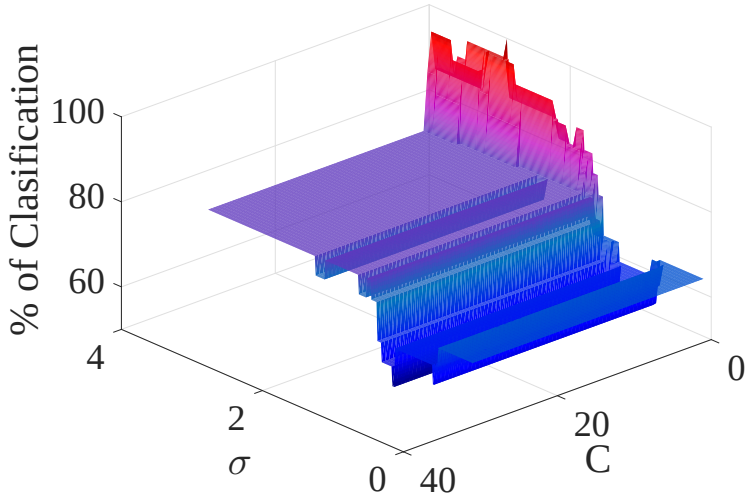


Figure 2.23: Percentage of classification as function of σ and C.

2.5 Conclusion

We implemented a soft margin SVM classifier to separate patterns extracted from sEMG signals, corresponding to two movements (elbow flexion and extension). We analyze the effect of the parameters σ and C in order to identify their influence on the final percentage of classification. As result, we found that there is an optimal area for values σ near to 1.1 and C close to 2.5, where the percentage of classification reaches 100% as shown in Figure 2.23.

The successful classification indicates that high accuracy is obtained in laboratory conditions, consequently, the proposed model can be extrapolated to: (i) out laboratory conditions with healthy and amputee patients, (ii) amputates patients even when they have low level of sEMG produced in the remained muscle.

Chapter 3

Embedding grasping recognition

3.1 Review

The MyoArmband™ is a wearable device used to control applications and systems, that is based on sEMG and Inertial Measurement Units (IMU). It was recently introduced by Thalmic Labs [60].

[38] performed the comparison of five time domain sEMG features. The features compared were MAV, VAR, WAMP, WL and ZC. The research tested five motions: (i) closing the hand, (ii) resting, (iii) mid-finger folding, (iv) gun-pointing and (v) half closing the hand. They consider MAV the best feature as it is constant behavior over all experiments. Therefore they proposed that time domain features have better performance with respect to frequency domain features.

Three-wheeled omni-directional robot was controlled using a muscle gesture-computer interface system. They considered only five gestures supported by the armband. Some other related work describes the architectural model of a system that addresses MyoArmband™ to control home automation systems [61].

In the field of augmented reality, for instance, [62] proposed a neuro-rehabilitation game for children, and [63] proposed a system to train physical rehabilitation situations.

[64] evaluated the level of acceptability of MyoArmband™, according to two situations, (i) when doctors use the device as diagnostic tool and (ii) when patients use the device in their daily life. As a result, the device has good acceptability by patients and doctors, considering 23 participants the average score is 69.21, where the maximum score is 100.

Finally, [61] described an architectural model of control for a home automatic systems, controlled by hand gestures. As a result, they mentioned the feasibility of using the gesture as a remote control interface braking the barrier of using touch user interface.

3.2 Wearable human interface concept

Human interface devices are proposed with the aim of having an interpreter between human and machines in real-time. Thus, wearable human interface is a device which is attached to the user as a piece of clothing *(e.g.* smart watches, jewellery and intelligent eyewear). Those devices have multiple applications, for instance, blind persons assistance [65], ambulation, transportation, exercise, fitness and military activities [66]. Measuring several attributes such as motion, location, temperature and vital signals.

One of the most successful wearable companies is Fitbit®[67], whereby several devices for measuring motion patterns are produced. The device studied by [68] measures steps and distance traveled. The collected data is captured from triaxial accelerometers based on Micro-ElectroMechanical systems (MEMS) [67]. Similarly, new innovative devices have been developed not only for measuring the patterns but also for identifying motion and more specifically gestures, considering gestures as a non-verbal communication [61].

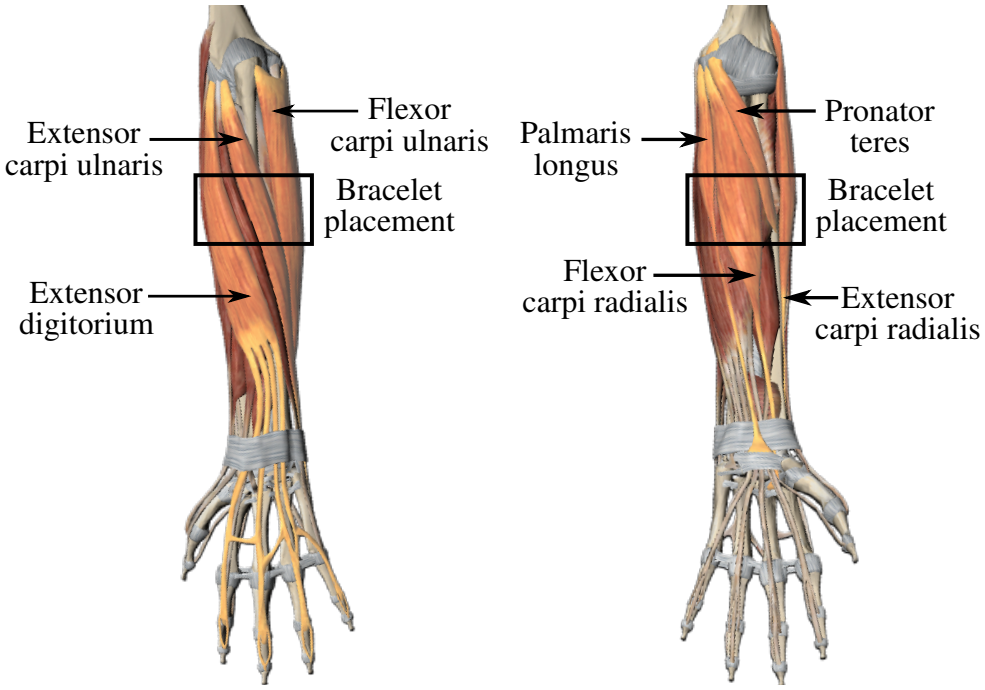
For the present research work, we use the MyoArmband™ bracelet, which is a wearable device for gesture recognition. It comprises a set of eight sEMG sensors and one inertial measurement units sensors that are composed by: (i) a three-axis gyroscope (ii) a three-axis accelerometer and (iii) a three-axis magnetometer. These capture are used to sense motion in all directions, obtaining the reference position using Euler angles and Quaternions formulation.

The data from sEMG and movements are transmited via Bluetooth Low Energy (BLE) wireless connection. Furthermore, it is equipped with an ARM Cortex-M4 microprocessor of low consumption. The MyoArmband™ electrodes are labeled with IDs from 1 to 8 and are disposed as shown Figure 3.1. The fourth channel (CH4) has a blue marker, and is placed in lower forearm followed by third channel (CH3) in clockwise and fifth channel (CH5) in counter clockwise. The sampling frequency is 200Hz.

Figure 3.1: MyoArmband™ sEMG sensors.

Once a subject is wearing the bracelet, sEMG sensor are located radially around a circum-

ference of the forearm. According to the placement of the bracelet, different muscles can be measured. Thalmic Labs suggest sliding bracelet into the forearm until it is just below the elbow. Following this instruction, the muscles closer to the sensors are those presented in figure 3.2.



(a) Frontal muscles.

(b) Posterior muscles.

Figure 3.2: Forearm muscles considering the MyoArmband™ placed in.

3.3 Movement detection model

Merging the concepts and models about the relationship between ElectroMyoGraphic and upper limb kinematics, and the experimental results obtained under laboratory conditions a new movement detection model is introduced. Our model allows real-time detection and identification of two hand movements, allowing a user to interact in a natural way with the ProMain robotic hand prostheses.

In order to define the movements to be detected by our method, we take into account experimental results, analyzing daily living grasps [69], which suggest that a limited number of gestures are enough for dexterous manipulation of a wide range of objects. The idea is that a Prosthetic hand does not need to mimic human hand movements, but it should be functional. Thus, we attempt to identify two simple movements inside the sEMG signal, hand open and close, aiming to allow a user to control the ProMain soft robotic hand prosthesis in a more natural way.

One important consideration for our movement detection model is that human motion process can be equated to a communication system, where messages (desired action) are trans-

mitted from α–motoneurons to muscular fibers. Consequently, sEMG signals take the role of information carriers, containing the subject's intention. Therefore, a suitable and smart interpretation of sEMG signal allows the identification of subject's desired actions. Three main stages are required to extract information embedded in sEMG signals as follows: (i) Modeling of surface ElectroMyoGraphic signals, (ii) Identifying movement inception and (iii) Recognition of prehension patterns. The following section present these stages in detail.

3.3.1 Modeling of sEMG signals

Human movements result from muscular contractions in the musculoskeletal system, which are induced by impulsive electrical stimuli δ transmitted from the α–motoneurons to muscular fibers. For a total number of muscular fibers N_f, the Stimuli $\delta_i(t)$, $\forall\ i \in \{1, \cdots, N_f\}$, are modeled as Dirac delta impulse train $\delta_i(t)$, as shown in the following equation [70]:

$$\delta_i(t) = \sum_{j=1}^{N_p} \delta(t - t_j) \tag{3.1}$$

where N_p is the total number of impulses and t_j are impulse's time occurrences.

Each muscular fiber has its own impulsive response $\psi_i(t)$, which is known as a Motor Unit Action Potential (MUAP). In the state of the art, MUAPs have been estimated using wavelet analysis, *e.g.* examples of MUAPs, adapted from the proposed by [71], are shown in Figure 3.3. Taking into account that δ_i is the convolution modulus, the response of muscular fibers is a set of $u_i(t)$ composed of N_p occurrences of MUAPs at t_j instants. These signals are known as Motor Unit Action Potential Trains (MUAPTs) and are formulated as follows [70]:

$$u_i(t) = \sum_{j=1}^{N_p} \psi_i(t) * \delta(t - t_j) = \sum_{j=1}^{N_p} \psi_i(t - t_j) \tag{3.2}$$

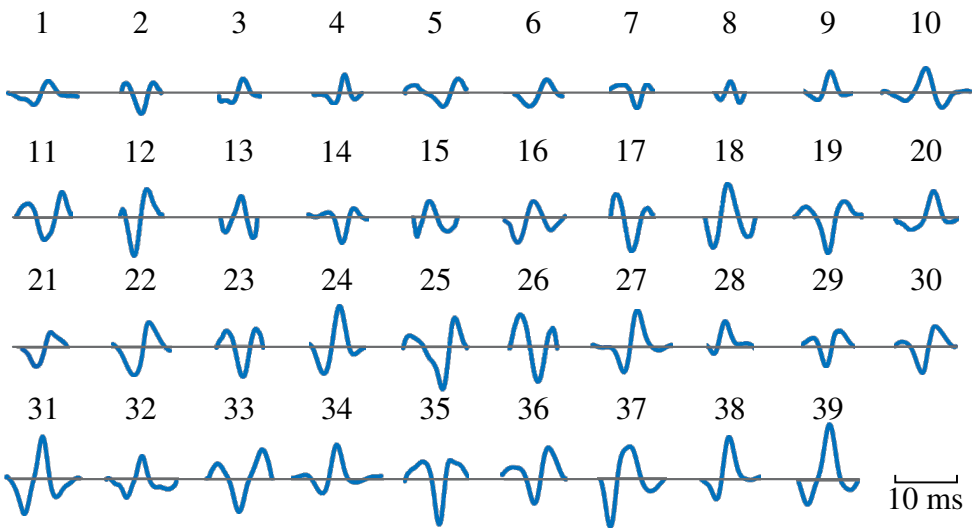


Figure 3.3: Example of impulsive response waveforms of motor unit action potentials $\psi_i(t)$.

The sum of the complete set of MUAPTs $u_i(t)$ define to the real EMG signals. Additionally, due to the multiple biological and environmental conditions a component of additive white Gaussian noise $\eta(t)$ is added, to constitute sEMG signals. Figure 3.4 shows the sEMG generation scheme. After discretization the model of the generated sEMG signals $s(k)$, being k the discret time index, is as follows:

$$s(k) = \sum_{i=1}^{N_f} u_i(k) + \eta(k) \tag{3.3}$$

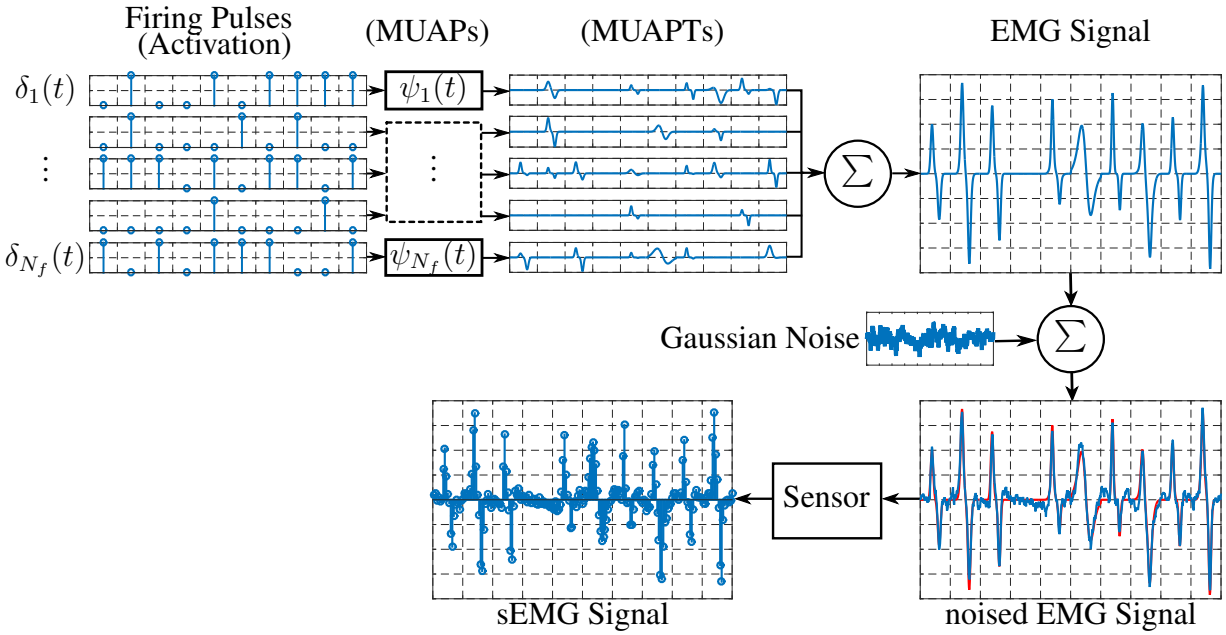


Figure 3.4: sEMG signal generation scheme.

In summary, the sEMG signal $s(k)$ is a discrete representation of $s(t)$, which is the response of subject's muscular fibers while electrical stimuli $\delta_i(t)$ exist. Furthermore, the stimuli $\delta_i(t)$ are considered as a coded message, embedded in the ElectroMyoGraphic, containing the subject's movement intention. Thereupon, based on the sEMG signal model, considered as shown in equation 3.3, we identify the presence of information, and moreover, extract and classify patterns embedded in the sEMG signals.

3.3.2 Identifying movement inception

In the state of the art [20], the movement recognition is frequently addressed using artificial intelligence algorithms to classify patterns extracted from sEMG signals. Those methods have shown a high efficiency under controlled conditions using offline analysis, but the problem is more complicated when real-time processing is required out of laboratory conditions. Hence, we introduce a procedure that improves the efficiency of movement identifications out of laboratory conditions and in real time.

3.3.2.1 Measure of produced information rate

The procedure is based on the analysis of a quantity H that measures at what rate the information, embedded in a sEMG signal, is produced during a movement. Each sample of sEMG signal results from the combination of a finite number of MUAPTs $u_i(k)$. Thus, the samples are directly related to muscular activity and movements. Likewise, samples are mapped into a set of finite events, whose probabilities of occurrence p_l are $p_1, p_2, \ldots, p_{N_i}$, being N_i the number of finite events. Consequently, the quantity H is calculated examining the signal samples in a segment of the signal.

Furthermore, to measure how much information is produced, the quantity H is required to be: (i) continuous in the p_l, (ii) a monotonic increasing function over the number of samples if all p_l values are equal to $1/N_i$, and (iii) the weighted sum of the individual values of H if a choice is broken down into two successive choices. All these assumptions are satisfied with:

The only H quantity satisfying the above assumptions is of the form:

$$H = \sum_{l=1}^{N_i} p_l \log p_l \tag{3.4}$$

The form of H is recognized as entropy which is defined in formulations of statistical mechanics, in Boltzmann's H–theorem and in the Shannon's entropy of information theory [72].

In vectorial notation, a sEMG signal $s_i(k)$, with a total number of samples W, is represented as $\boldsymbol{s}_i = \{s_{1\,i}\ ,\ \ldots\ ,\ s_{W\,i}\}^T$. Consequently, for this sEMG signal $\boldsymbol{s}_1$, the entropy will be written as $H(\boldsymbol{s}_1)$; thus $\boldsymbol{s}_1$ is not an argument of a function but a label for a number, to differentiate it from $H(\boldsymbol{s}_2)$ which is the entropy of the sEMG signal $\boldsymbol{s}_2$. The entropy has several interesting properties and two of them characterize its maximum and minimum values: (i) H is zero if and only if one of all probabilities of occurrence p_l is almost equal to zero (ii) for a given number of events N_i, entropy is a maximum $H = log(N_i)$, when all $p_l = 1/N_i$. The last property represents the most uncertain situation. In our case, since we use binary digits the selected base is 2. As a result, the entropy for an sEMG signal fragment $\boldsymbol{s}$ with W samples is given by:

$$H(\boldsymbol{s}) = \sum_{l=1}^{N_i} p_l \log_2 p_l \tag{3.5}$$

The probabilities of occurrence $p_1, p_2, \ldots, p_{N_i}$ is estimated based on the absolute occurrence frequency f_l normalized by the total of events as follows:

$$p_l = \frac{f_l}{N_i},\ \text{for } l = 1, 2, \ldots, N_i \tag{3.6}$$

Classically, absolute occurrence frequencies f_l are described as the amount of favorable outcomes over the number of all possible outcomes. In that case, we consider favorable outcomes as the number of signal samples belonging to one discrete event l, and the number of all possible outcomes as the total number of discrete events N_i. Taking into account that sEMG signal is

acquired using the MyoArmband™ bracelet, which employs a 8-bit (1-byte) analog to digital converter, it is reasonable to set the total number of discrete intervals N_i to 256. Likewise, to evaluate if a sample belongs to a particular discrete interval l, the following membership function is introduced:

$$\varepsilon(s_k, l) = \begin{cases} 1 & \text{if } \frac{l-1.5}{N_i-1} \leq s_k < \frac{l-0.5}{N_i-1} \\ 0 & \text{otherwise} \end{cases} \tag{3.7}$$

where s_k is the kth sample of the sEMG signal. Thereupon, absolute occurrence frequencies f_l, for a signal with W samples, are calculated as:

$$f_l = \sum_{k=1}^{W} \varepsilon(s_k, l), \text{ for } l = 1, 2, \ldots, N_i \tag{3.8}$$

3.3.2.2 Entropy flow and movement inception

Considering that the entropy $H(s)$, is a measure of the information rate, we analyze its value to identify the information contained in the sEMG signal. These variations are extracted through the calculation of the entropy flow $\dot{H}(s)$, which corresponds to the first derivative of the over time. In consideration of the discrete nature of the sEMG signals, we consider to use a numerical approximation of the derivative, thus we adopt the following discrete-time differential operator D:

$$D = \frac{\delta}{\mu\delta T + 1} \tag{3.9}$$

where, $\delta = (z-1)/T$, z is the usual shift-operator, and T is the sampling period.

The parameter μ is any real number, and it is usually chosen between three options: (i) $\mu = 0$ if the derivative is on future and current values using forward difference, (ii) $\mu = 1/2$ if the derivative is computed in a middle point between two samples and is operated using future or past values merged with the current measures. The method is typically known as Tustin or bilinear transform, and (iii) $\mu = 1$ if the derivative is on the past and present values using backward difference. The following equation summarizes the criterion to choose the μ parameter:

$$\mu = \begin{cases} 0 & \text{Forward difference} \\ 1/2 & \text{Tustin transform} \\ 1 & \text{Backward difference} \end{cases} \tag{3.10}$$

In our case, the parameter μ is set to 1, assuring that the derivative is causal (*i.e* it is calculated based on the past and current values, and not on future values of entropy). Consequently, the discrete-time differential operator is rewritten as follows:

$$D = \frac{1 - z^{-1}}{T} \tag{3.11}$$

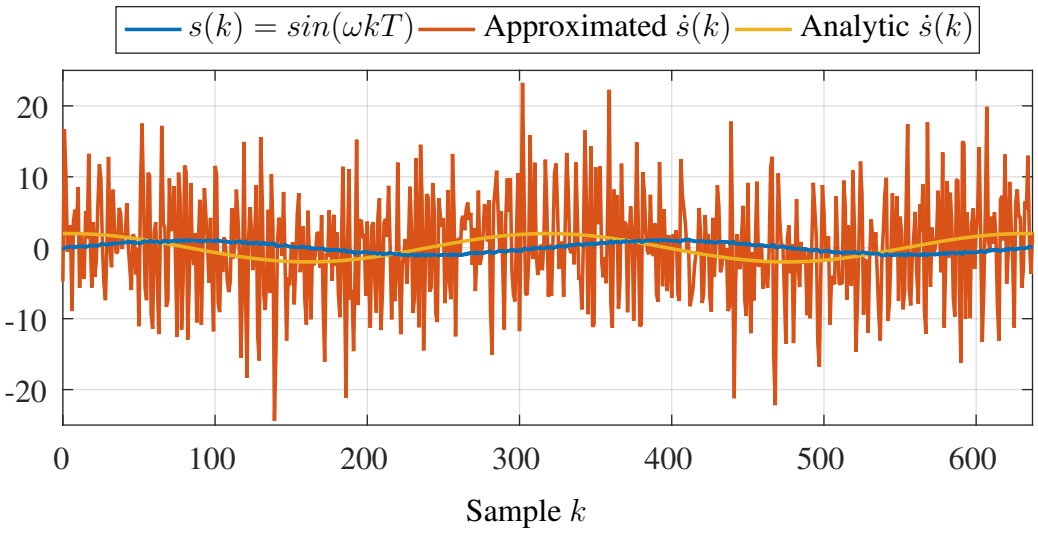


Figure 3.5: Backward derivative of a noisy signal $s(k)$.

In time series analysis, the shift operator z leads the kth sample of a time series to produce the following $(k+1)$th element. When the shift operator is raised to negative power, *e.g* z^{-1} in equation 3.11, it lags the kth sample of a time series to produce the previous $(k-1)$th element. Therefore, when the D operator is applied to a discrete signal $D\boldsymbol{s}$, the numerator $(1 - z^{-1})s_k$ becomes $s_k - s_{k-1}$. As a result, the first derivative of the signal $s(k)$ presented in equation 3.3 written as follows:

$$\dot{s}(k) = \dot{\boldsymbol{s}} = \frac{s_k - s_{k-1}}{T} \tag{3.12}$$

This numeric approximation, using backward difference is simple, fast, and accurate, even that, it is not robust in the presence of noise. Figure 3.5 shows this disadvantage when the derivative of a noised signal $\boldsymbol{s} = sin(\omega kT) + \eta(k)$ is approximated using backward difference; the frequency ω belongs to the sEMG spectrum and $\eta(k)$ is the additive white Gaussian noise.

The problem of sensibility to noise can be overcome analyzing separated samples instead consecutive samples. Hence, we modify the power of the shift operator to z^{-K_W}, producing a lag of K_W samples. Thus, the D operator is equal to $D = (1 - z^{-K_W})/(T\ K_W)$ and the formulation of the derivative, so-called sliding backward derivative, is rewritten as follows:

$$\dot{\boldsymbol{s}} = \frac{s_k - s_{k-K_W}}{T\ K_W} \tag{3.13}$$

The numeric derivative, introduced in equation 3.13, was tested with the same noised signal $\boldsymbol{s} = sin(\omega kT) + \eta(k)$ using $K_W = 80$. The choose of $K_W = 80$ is done considering the MyoArmband™ sampling frequency and the experimental results obtained in the motion capture laboratory (see section 2.4). Indeed, the approximation is much more accurate as shown in Figure 3.6.

The entropy is analyzed continuously, to maximize the identification of changes that represents the movement intention. However, the analysis is performed in a limited number of

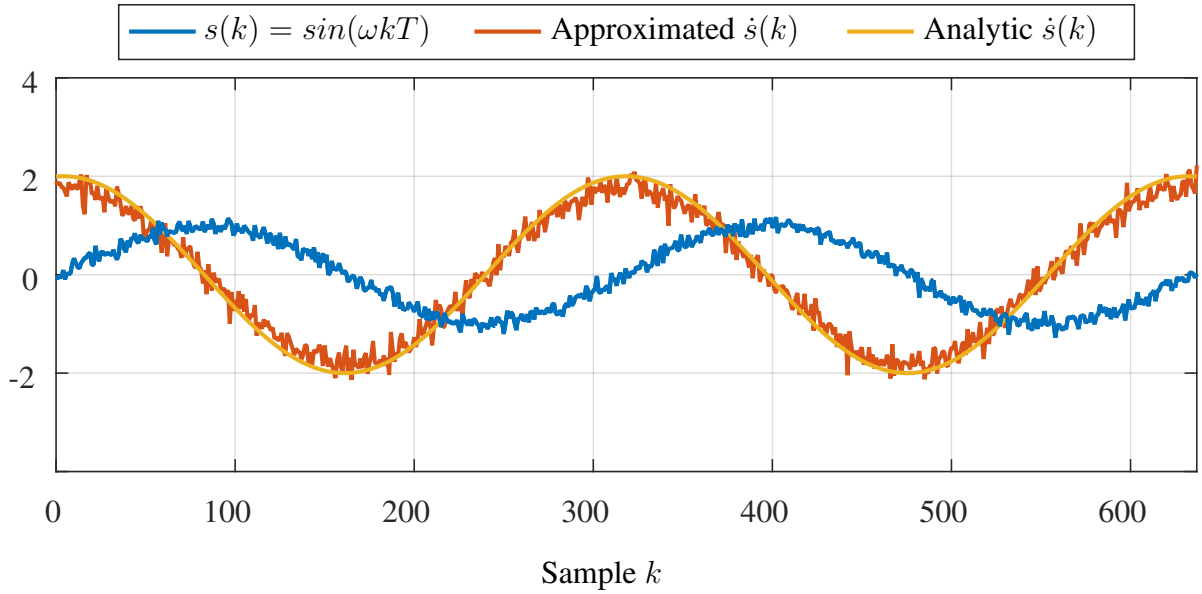


Figure 3.6: Backward sliding derivative of a noisy signal $s(k)$.

samples. Thus, the proposed method consists in calculating the entropy not along all the sEMG signal, but along segments matching the size of the sliding derivative

interval. So that, for an sEMG signal $\boldsymbol{s}_i$, the entropy $H(k, \boldsymbol{s}_i)$ is calculated along the sEMG signal segment $\overline{\boldsymbol{s}}_i = \{s_{i\,k-K_W}\ ,\ \dots\ ,\ s_{i\,k}\}^T$. It is to note that k and $\boldsymbol{s}_i$ are not arguments of the entropy, the notation $H(k, \boldsymbol{s}_i)$ means the entropy of the sEMG signal at the discret-time k calculated along the segment $\overline{\boldsymbol{s}}_i$.

Considering that the entropy $H(k, \boldsymbol{s}_i)$ is a time series, its variation is calculated applying the sliding backward derivative proposed in equation (3.13). As a result, we obtain the expression of $\dot{H}(k, \boldsymbol{s}_i)$, where $\dot{H}(k, \boldsymbol{s}_i)$ means the entropy flow of the sEMG signal at the discret-time k calculated in the discrete-time interval $[k - K_W, k]$. The expression of $\dot{H}(k, \boldsymbol{s}_i)$ is shown equation (3.14).

$$\dot{H}(k, \boldsymbol{s}_i) = \frac{H(k, \boldsymbol{s}_i) - H(k - K_W, \boldsymbol{s}_i)}{T\ K_W} \tag{3.14}$$

The obtained experimental results, presented in section 1.5, have shown that the entropy of an sEMG signal remains constant when no movement is produced. Hence, when a movement is performed, the entropy of the corresponding sEMG signal presents visible variations, causing sharp peaks in the entropy flow. Therefore, our method tracks the significant changes of entropy by detecting peaks in the entropy flows.

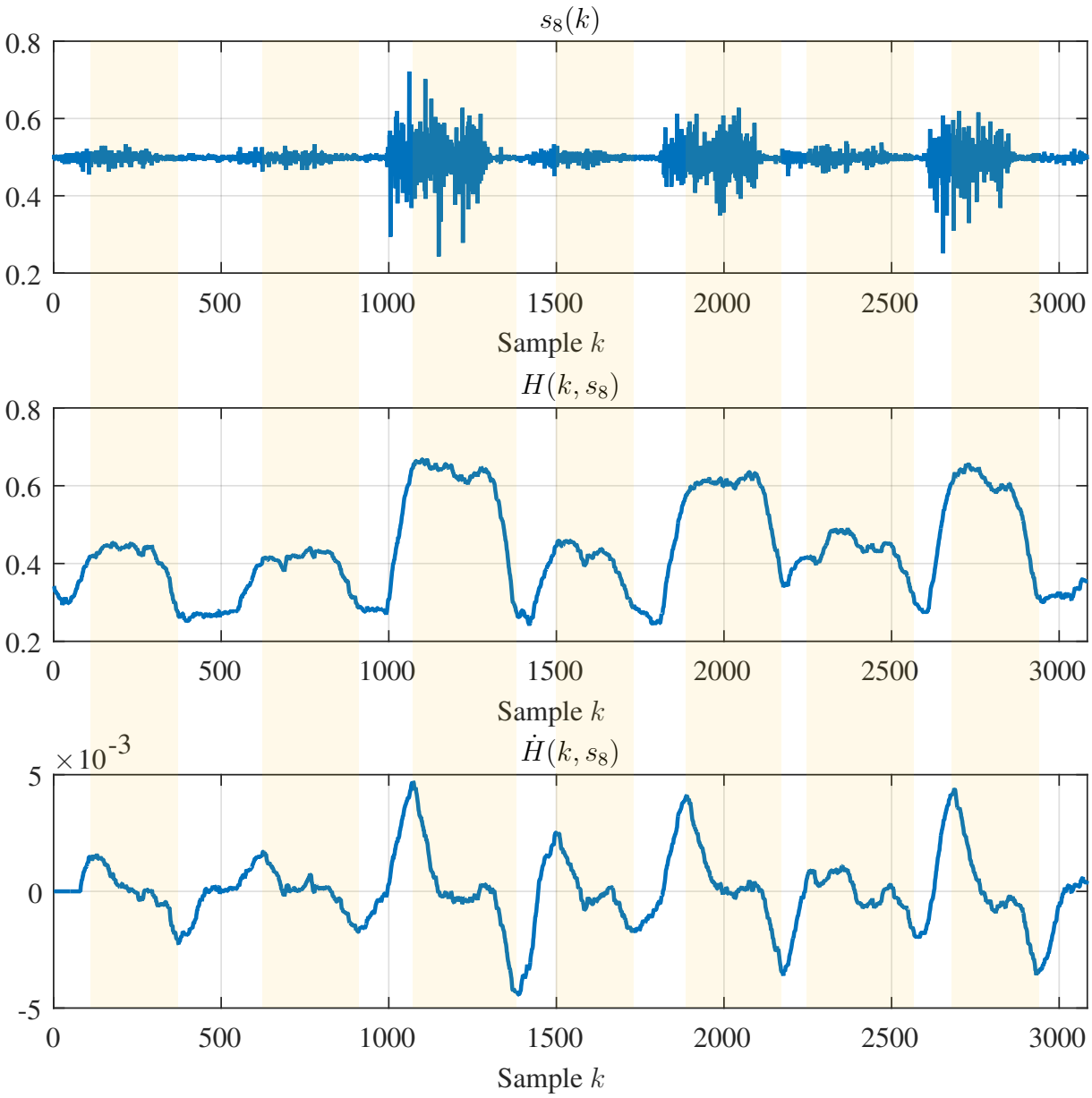


Figure 3.7: Backward sliding derivative of a noisy signal $s(k)$.

The occurrence time k of each positive peak in the entropy flow corresponds to a movement inception and the occurrence time k of each negative peak corresponds to a movement termination. Figure 3.7 exemplary shows the calculation of the entropy and the entropy flow of an sEMG signal $\boldsymbol{s}_8$ captured from the 8th sensor of the MyoArmband™ during several open and close movements of the hand. Clearly, the peaks in the entropy flow $\dot{H}(k, \boldsymbol{s}_8)$ overlap the beginning of the steady state of the entropy for each movement. The same result has been found with all other sensors of the MyoArmband™, as shown in Figure 3.8.

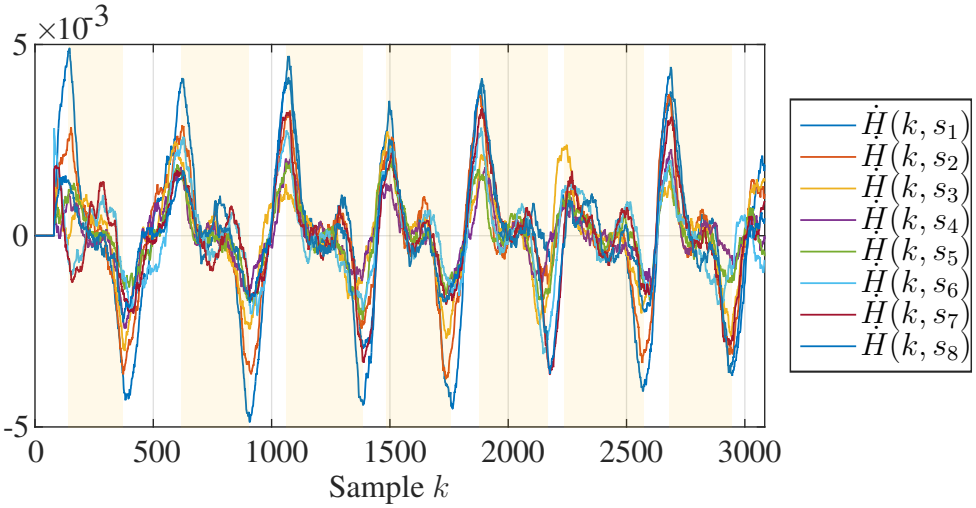


Figure 3.8: Entropy flow of the complete set of sEMG signals.

It can be seen in Figure 3.7 that the entropy $H(k, \boldsymbol{s}_i)$ reaches two different steady-state values after positive peaks, which is reasonable considering that the signal contains information of two different movements. In the following, we introduce the classification method, which is applied subsequently to the detection of movement inception.

3.3.3 Recognition of prehension patterns using Support Vector Machines

As presented in section 2.2.2, Support Vector Machines (SVMs) are a kind of classifiers that attempt to find a separator classes using an hyperplane that has the largest possible margin with respect to patterns. According to the classification problem there are three possibilities of separation: (i) rigid margin SVM for linearly separable data, (ii) soft margin SVM for slightly non-linearly separable data, and (iii) kernel function based SVM for heavily non-linear separable data.

Furthermore, the achieved result regarding the classification of elbow flexion and extension, as is shown in sections 2.3 and 2.4, whereby classification rate is 100%, let presume that: (i) the problem has a feasible solution out of laboratory conditions and (ii) the machinery of a soft-margin SVM classifier could be suitable to separate two classes, (*i.e.* open and close hand). Moreover, in consideration of the signal acquisition technology used in MyoArmband™, which delivers eight signals in real time, using of multiple features or mapping into a hyperspace are not necessary. Therefore, the proposed approach to the classification of prehension patterns recalls the following soft margin SVM model:

$$\begin{aligned} \min \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i \\ \text{subject to} \quad & y_i(\boldsymbol{w}^T\boldsymbol{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \qquad\qquad \text{for } i = 1, \ldots, N \end{aligned} \tag{3.15}$$

where $\xi \in \mathbb{R}^N$ is a slack measuring margin violation, C is a constant that gives the relative importance of the slack with respect the margin, and $\boldsymbol{w} \in \mathbb{R}^n$ and $b \in \mathbb{R}$ are the separator hyperplane parameters.

Likewise, the dual formulation of the above primal soft margin SVM problem is formulated using the Lagrangian multipliers $\boldsymbol{\lambda}$ as follows:

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & \frac{1}{2}\boldsymbol{\lambda}^T \boldsymbol{Q}\boldsymbol{\lambda} + (-\mathbf{1}^T)\boldsymbol{\lambda} \\ \text{subject to} \quad & \boldsymbol{y}^T\boldsymbol{\lambda} = 0 \\ & \mathbf{0} \leq \boldsymbol{\lambda} \leq \mathbf{1}C \end{aligned} \tag{3.16}$$

In order to calculate the b value, we use the normalization $y_c(\boldsymbol{w}^T\boldsymbol{x}_i + b) = 1$, substituting $\boldsymbol{x}_i$ by one of the found support vectors; the value of b is the same using any of the support vectors. To calculate parameter $\boldsymbol{w}$ of the hyperplane $\mathcal{H}(\boldsymbol{w}, b)$, equation (3.17) below is used:

$$\boldsymbol{w} = \sum_{i=1}^{N} \lambda_i y_i \boldsymbol{x}_i = 0 \tag{3.17}$$

Each pattern $\boldsymbol{x}_i$ is calculated extracting the entropy from the group of sEMG signals captured simultaneously by the MyoArmband™ bracelet, (see Figure 3.9). The adopted notation to represent the group of incoming signals is $\boldsymbol{S}_i = \{\boldsymbol{s}_{1i}, \boldsymbol{s}_{2i}, \ldots, \boldsymbol{s}_{8i}\}^T$, being $\boldsymbol{s}_{1i}$ the signal captured by the first sensor of the bracelet, $\boldsymbol{s}_{2i}$ the signal measured by the second sensor, and so on. Afterward, the entropy is calculated for each signal belonging $\boldsymbol{S}_i$, resulting in a set of entropies from $H(\boldsymbol{s}_{1i})$ until $H(\boldsymbol{s}_{8i})$. Taking into account that the target is to find the optimal margin during separation, it is a good practice to normalize inputs. The normalization emphasizes the differences or similarities (whichever the case) between patterns. As a result, the feature space $\mathcal{X} \in \mathbb{R}^8$ is composed of the normalized entropies $\hat{H}(\boldsymbol{s}_{1i})$ to $\hat{H}(\boldsymbol{s}_{1i})$, and the pattern $\boldsymbol{x}_i$ can be written as:

$$\boldsymbol{x}_i = \begin{Bmatrix} \hat{H}(\boldsymbol{s}_{1i}) \\ \hat{H}(\boldsymbol{s}_{2i}) \\ \vdots \\ \hat{H}(\boldsymbol{s}_{7i}) \\ \hat{H}(\boldsymbol{s}_{8i}) \end{Bmatrix} \tag{3.18}$$

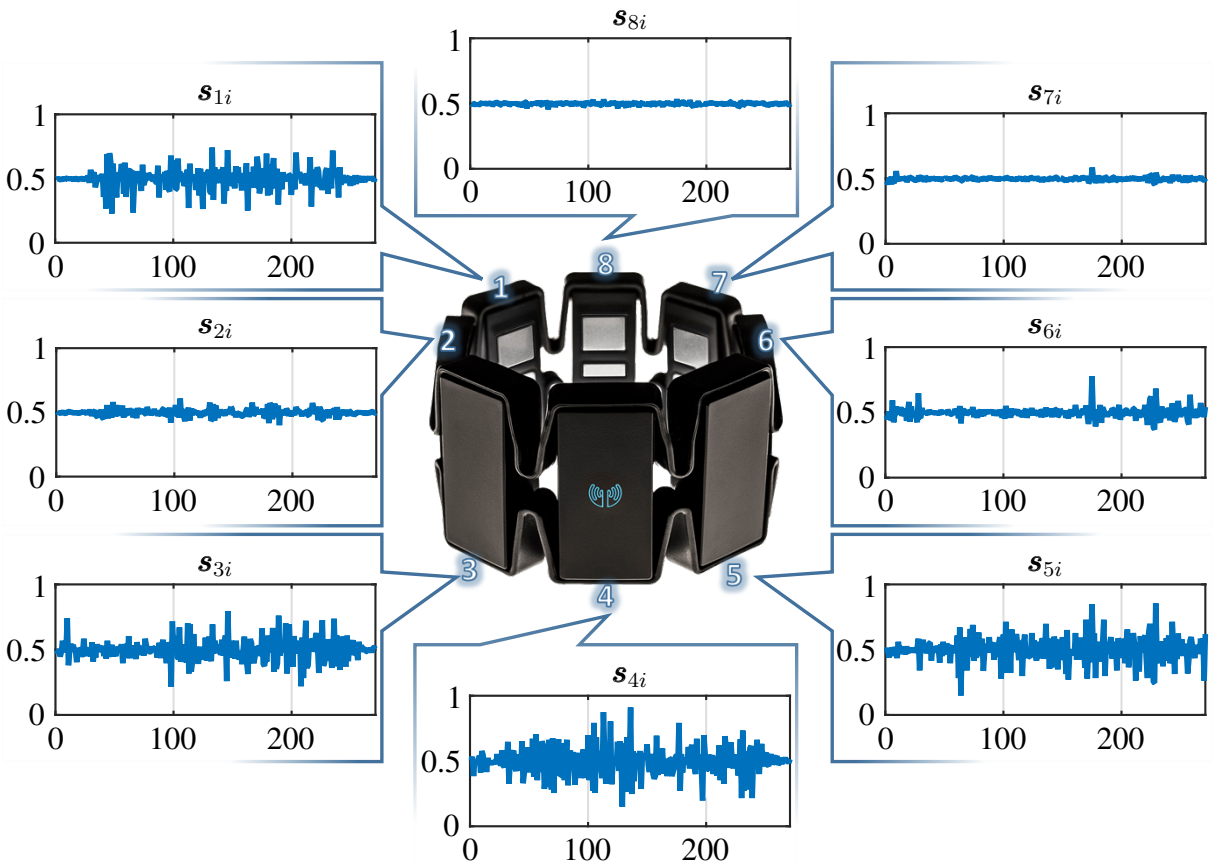


Figure 3.9: Complete set of sEMG signals captured using MyoArmband™.

3.4 Experimental analysis and model validation

One of the main obstacles in the classification or recognition of human movements, out of laboratory conditions, is related to the uncertainty of the movement occurrence time. To solve this problem, we have introduced a methodology based on the analysis of the entropy flow, which allows to detect the movement inception. Thus, in the following, we present an experiment that aims to verify the performance of the proposed methods in the classification of two prehension patterns, that are grasp (close hand) and release (open hand).

The proposed methodology is tested using MyoArmband™ bracelet, and the introduced models are implemented in software developed using C++. The software is composed of: (i) a connection with bracelet via Bluetooth , (ii) a graphic interface to display the hand gesture, and (iii) a principal cycle that iterates every 5 ms. In the beginning, the software gets connected to MyoArmband™, display the first gesture, and initializes with zero parameters k and i, which respectively correspond to the discrete time and pattern index.

Once the cycle is active, the sEMG signals acquisition is started, each received sample produces an increment of discrete-time stamp k. When k reaches a limit of K_W samples (value defined in section 3.3.2.2 to calculate sliding derivative), the normalized entropy is calculated

along the signal segments $\overline{\boldsymbol{X}}_i$.

Afterwards, when k reaches a limit of $2K_W$ the entropy flow calculation is launched. The entropy flow is analyzed to find peaks; if a positive peak is found, the present set of values is saved and tagged with the present label. Entropy values are saved until negative peak rises. After a negative peak, the signal, the entropy, the entropy flow and the label are stored to hard disk. Subsequently, a new visual indication is displayed and the process restarts. The number of instruction to displayed must be set before executing the software. Figure 3.10 shows the capture process flow chart.

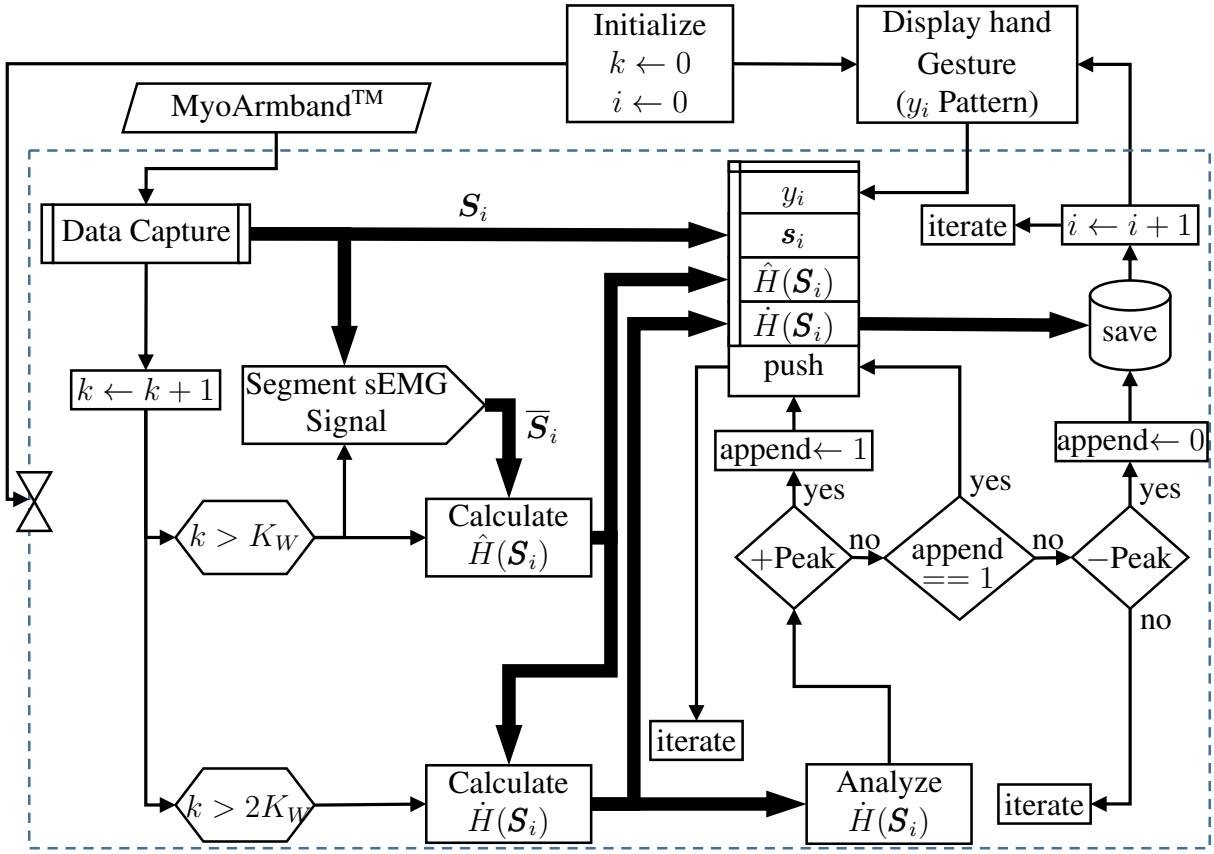


Figure 3.10: Pattern capture flow chart.

3.4.1 Materials and methods

The experiment, was performed with three subjects with the MyoArmband™, which was placed as recommended by Thalmic Labs to capture signals from the following principal muscles: (i) Extensor carpi radialis longus, (ii) Palmaris longus, (iii) Flexor carpi radialis, (iv) Flexor carpi ulnaris, (v) Pronator teres, and (vi) Extensor carpiradialis. Likewise, subjects received instructions directly from a laptop screen, in which images concerning the movement that they should perform were displayed.

Subsequently, the software captures the set of sEMG signals $\boldsymbol{S}_i$, processes the entropies $H(\boldsymbol{s}_{1i})$ to $H(\boldsymbol{s}_{8i})$, and when a positive peak is found in the entropy flow, it pushes values into a

matrix, which is stored to hard disk with the labels that correspond to the image shown in the screen when a negative peak is detected in the entropy flow.

During the experiment, the subjects are asked to open five times the hand from a relaxed posture and close five times the hand from a carefree position. Figure 3.11 summarizes the experimental set-up and the used methodology.

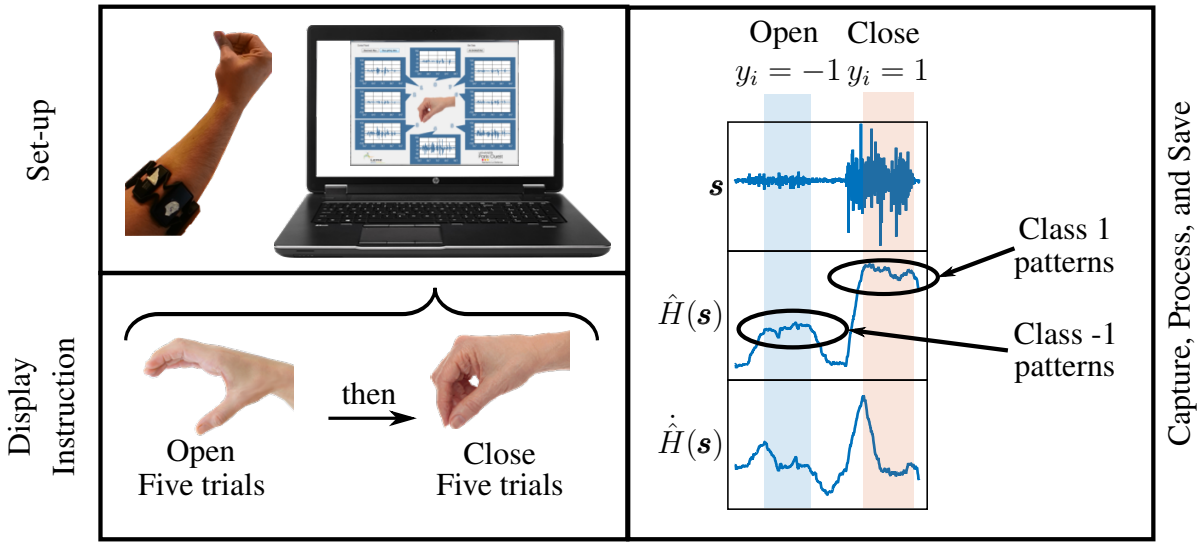


Figure 3.11: Summary of experiment materials and methods.

3.4.2 Analysis of experimental entropy data

In the following, we present the analysis of obtained experimental data issued of the experiment. Subjects performed two prehension patterns, say grasp (close hand) and release (open hand). Thus, in the following, the experimental data is analyzed to:

1. Identify the lag time of the positive peak of entropy flow with respect the sEMG signal activation.

2. Compare cross relation of entropy data to fix the most adapted classification amongst the introduced SVM algorithms.

The lag time t_l of the entropy flow's positive peak can be considered as: 1. the rising time required for the entropy to achieve its steady state value from a stable reference or 2. the time required to identify the presence of a pattern. In both cases, this lag will have a considerable influence in the classification of movements. The value of t_l is measured directly from the experimental data as:

$$t_l = t_{pa} - t_{sa} \tag{3.19}$$

Where t_{pa} is the peak occurrence time and t_{sa} is the sEMG signal activation start.

Figure 3.12 shows the measures of t_{sa} and t_{pa} based on the 8th sEMG signal of subject 1 during trial 1. Thereafter, t_l is measured for the ten trials (five for grasp and five for release)

performed for each subject applying equation (3.19). Taking into account that the lag t_l depends on several phenomena (*e.g.* during a specific movement, the dominant muscle has less slack than the auxiliary ones, as explained in section 1.4.1), the lag is measured over all signals captured from different muscles and the values during an specific movements.

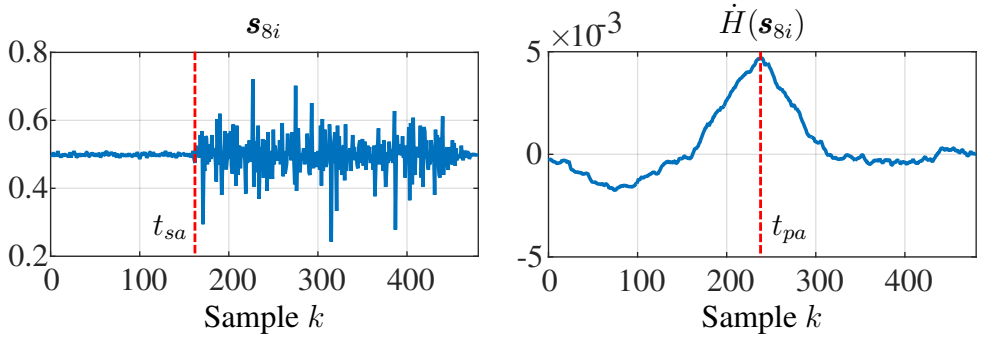
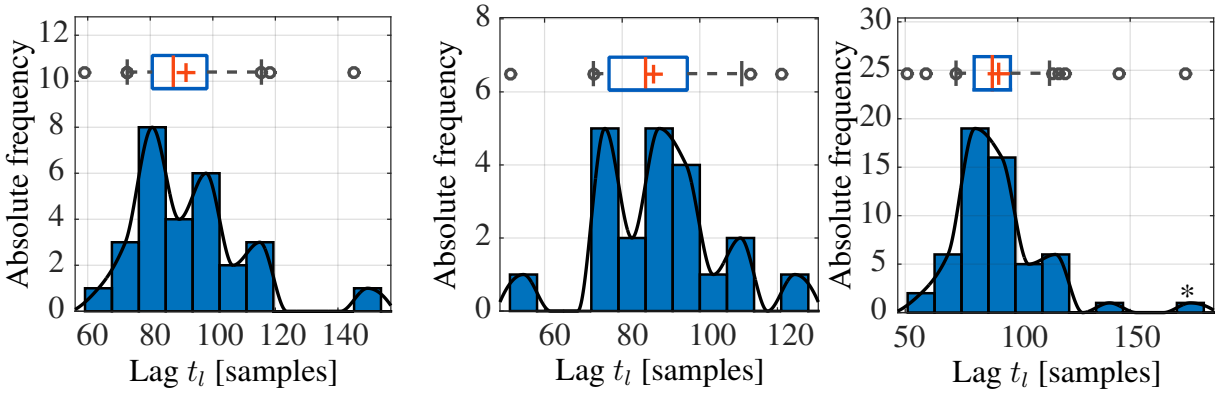


Figure 3.12: Graphical representation of the peak occurrence time t_{pa} and the sEMG signal activation start t_{sa}.

The lag obtained from the different muscles are averaged to obtained three quantities per subject: 1. the mean value during grasp trials, 2. the mean value during release trials, and 3. the overall mean value considering grasp and release trials. As a result, the global mean value of lag is $t_l = 91.58$ with a standard deviation of 19.792. The mean value during grasp trial is $t_l = 87.941$ with a standard deviation of 16.042. The mean value during release trials is $t_l = 91.589$ with a standard deviation of 17.65.

Figures 3.13a to 3.13c summarize the obtained results, wherein vertical axis is the total number of observations within a given interval (absolute frequency), and the horizontal axis corresponds to t_l measured in number of samples. Furthermore, the red crosses are the mean values and red vars the median.



(a) Lag t_l for release trials. (b) Lag t_l for grasp trials. (c) Lag t_l for all trials.

Figure 3.13: Forearm muscles activities tracked with MyoArmband™placed in.

Table 3.1 summarizes the results for all subjects. Considering that the sample frequency of the MyoArmband™ is 200Hz, the maximal global lag is 457.9ms. According to the re-

Table 3.1: Summary of lag values for all subjects.

Subject	release trials		grasp trials		overall	
	avg	std_dev	avg	std_dev	avg	std_dev
1	$t_l = 91.58$	17.64	$t_l = 87.9416$	16.0424	$t_l = 91.58$	19.7929
2	$t_l = 110.33$	5.13	61.33	14.50	85.83	28.54
3	94.33	7.50	80.66	3.78	87.50	9.18

the values are measured in number of samples and STD is the acronym for standard deviation.

sults, introduced in section 1.4.1, the sEMG signal gets activated 212ms[1] before the movement inception. Therefore, we can consider that our methodology allows to detect a movement approximatively in 245.9ms out of laboratory conditions.

Regarding the classification problem, the target is to verify if the patterns, captured after inception detection, are separable or not. Figures 3.14 and 3.15 show two constellation of first subject's patterns. Clearly, the feature space is non-linearly separable with respect to entropies of signals arising from sensors three and four. In contrast, despite some overlapped patterns, the entropy extracted from sEMG signals coming from sensors one and eight seems linearly separable through a soft margin SVM classifier.

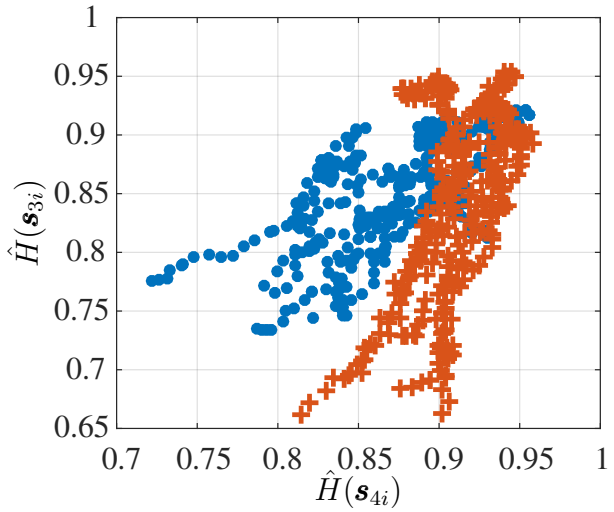


Figure 3.14: Values from electrodes three and four.

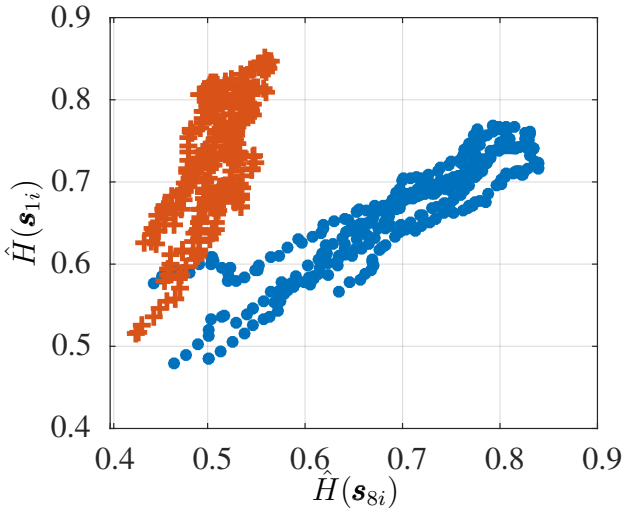


Figure 3.15: Values from electrodes one and eight.

The formed feature space is composed of eight dimensions, one for each sEMG signal

[1]This values is measured comparing the movement kinematic registered with Vicon cameras and the sEMG signal that are rerecord in real time, the sampling frequency are 200Hz for kinematic tracking and $1K$ Hz for sEMG

captured. Figure 3.16 shows the complete feature space, that contains in some dimensions is linearly separable. Thus, the separation problem consists in training the soft-margin SVM classifier proposes in equation (3.16).

The separator hyperplane $\mathcal{H}(\boldsymbol{w}, b) = \{\boldsymbol{x} \in \mathbb{R}^8 \mid \boldsymbol{w}^T\boldsymbol{x} + b = 0\}$ will be composed of a vector $\boldsymbol{w} \in \mathbb{R}^8$ and the scalar value b. The solution for the subject one, obtained solving equation (3.17), is $\boldsymbol{w} = \{1135.4, -958.7, 388.2, -135.1, -542.6, 557.3, -275.5, -478.1\}^T$ and $b = 233.9$. The last parameter result from the normalization $y_c(\boldsymbol{w}^T\boldsymbol{x}_i + b) = 1$, substituting $\boldsymbol{x}_i$ by one of the found support vectors.

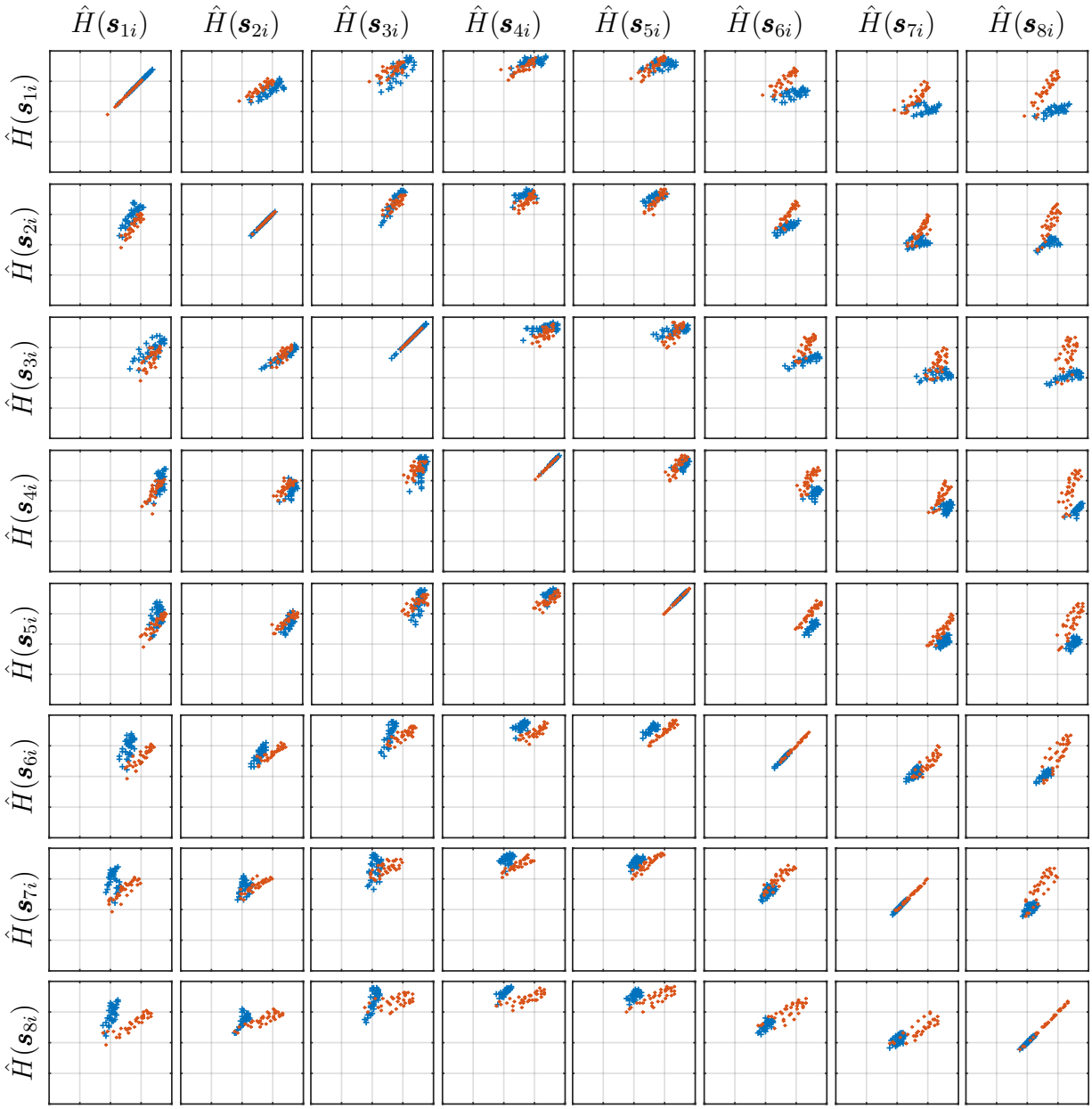


Figure 3.16: Experimental materials and methods.

3.4.3 Method validation

The proposed methodology to validate classifier results using MyoArmband™ bracelet, is based on the test software used in the previous section, but replacing the storage phase by a movement recognition phase wherein the SVM classifier is implemented. Thus, the software for validation is composed of: (i) a connection via Bluetooth with bracelet, (ii) a graphic interface to display recognized gesture, and (iii) a principal cycle that iterates every 5 ms. In the beginning, the software gets connected to MyoArmband™, and initialize with zero parameters k and j, which correspond to the discrete time and pattern index respectively. Once the cycle is active, the sEMG signals acquisition is started, each received sample produces an increment of discrete-time stamp k. When k reaches a limit of K_W then the entropy are calculated. Therefore, when k reaches a limit of $2K_W$ then entropy flow are calculated.

The entropy flow is analyzed to find peaks; if a positive peak is found, the SVM classifier is launched and the following decision function is applied to recognize the pattern $\boldsymbol{x}_i$:

$$\Gamma(\boldsymbol{x}_i) = \text{sign}\left(\boldsymbol{w}^T\boldsymbol{x}_i + b\right) \tag{3.20}$$

If $\Gamma(\boldsymbol{x}_i) = 1$ the desired movement is grasp, and if $\Gamma(\boldsymbol{x}_i) = -1$ then the pattern corresponds to release. The result of the classification is shown in the screen using an iconic image representing the recognize gesture. Figure 3.10 shows the soft margin SVM classification flow chart.

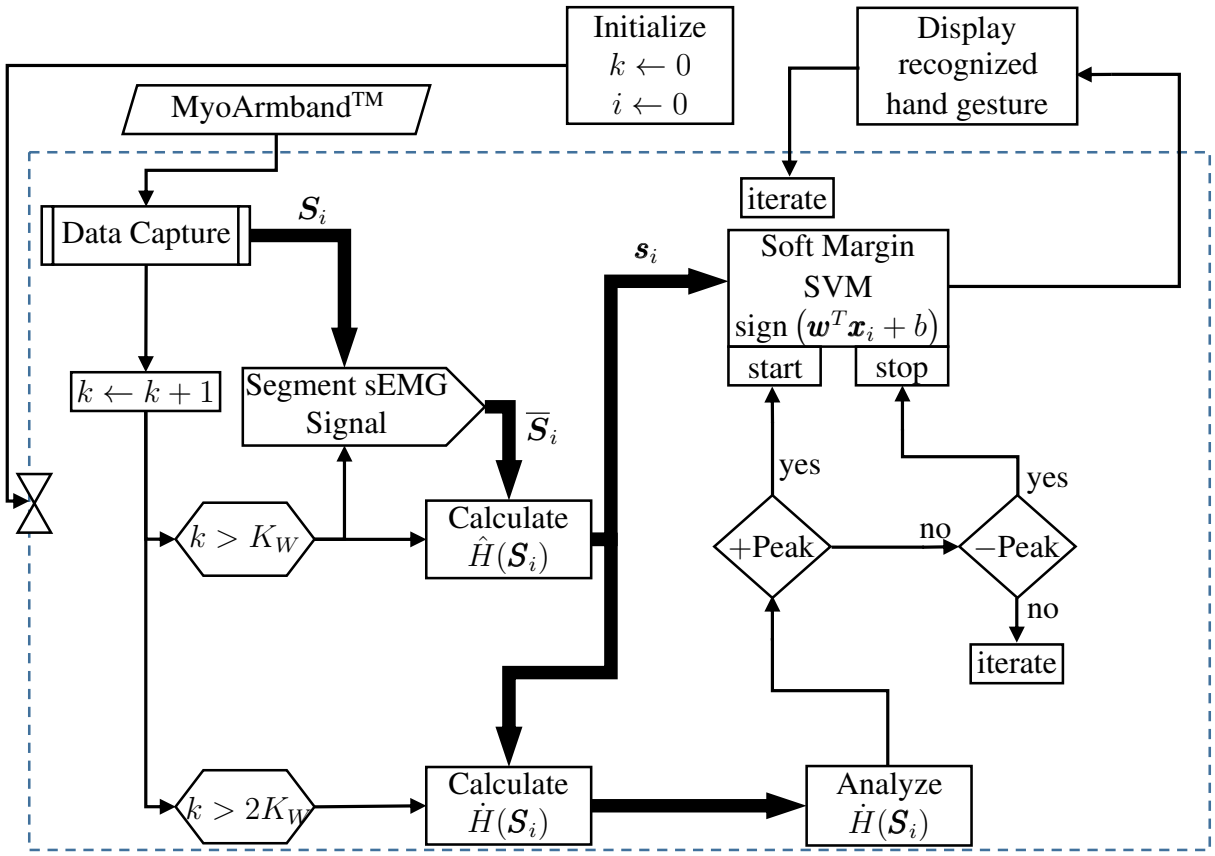


Figure 3.17: Pattern classification capture flow chart.

Then, the result of $\Gamma(\boldsymbol{x}_i)$, is compared with the expected y_i. If the values match, the SVM has correctly classified the pattern $\boldsymbol{x}_i$ As a result in the following table we summarize the rating success for all subjects discriminating into grasp, release and general classification:

Table 3.2: Summary of lag values for all subjects.

Subject	release trials		grasp trials		overall	
	Trials	Success [%]	Trials	Success [%]	Trials	Success [%]
1	20	100	20	100	40	100
2	20	95	20	100	40	97.5
3	20	100	20	90	40	95

3.5 Conclusion

We present a new movement detection method which is based on : (i) the model of sEMG through the analysis of entropy produced during movement, (ii) the identification of movement inception through the entropy flow, (iii) and the recognition of prehension patterns through SVM. The proposed methodology is tested using MyoArmband^{TM} bracelet, performing two experiments, one for patterns capture and one for movement recognition.

As a result, we propose a high efficiency recognition system in real-time. It operates out of laboratory conditions. The system detects movement approximately in 245.9ms after movement inception. The rating success for all subjects is 97.5%, and indicates that the proposed system is able to identify with precision two grasping gestures.These results indicate that more grasping motion can be trained even individual finger movements.

Chapter 4

ProMain- I robotic hand control

4.1 ProMain-I Hand characteristics

The bio-inspired robotic hand prosthesis ProMain-I [1] has been developed, tested and manufactured completely at the LEME laboratory. The hand has three fingers, which are disposed to perform precision grasping, each finger has three joints: MetacarpoPhalangeal (MP), Proximal Interphalangeal (PIP) and Distal Interphalangeal (DIP). All joints have one Degree of Freedom (DoF) to perform flexion and extension.

Hence the Medial (M) and Distal (DP) phalanges are driven by the Proximal Phalanx (PP) motion. The clockwise rotation of the actuator produces flexion, and the opposite rotation produces extension. The rotation angles of the PIP and DIP joints are dependent on the rotation angle of the MP joints, which are directly driven by servomotors. The relation between the angles is $\theta_{j2} = \theta_{j3} = 0.9\theta_{j1}$, where θ_{j1} is the MP joint angle, θ_{j2} is the PIP joint angle and θ_{j3} is the DIP joint angle. The subindex j indicates the finger, with $j = 1$ for the thumb, $j = 2$ for the index and $j = 3$ for the middle finger. Likewise, l_{j1}, l_{j2} and l_{j3} are the lengths of the proximal, medial and distal phalanges. Figure 4.1 shows the lengths and the joint angles for the index finger $(j = 2)$.

Considering that the robotic hand is a under-actuated multi-link articulated chain, a particular control system based on the robotic hand kinematic model. Taking into account that the controller requires sensory feedback, each finger is equipped with a new fingertip smart force sensor, which produces a signal when contact with objects is detected. Description and performed experimental identification of the new smart force sensor are presented in the following sections.

The hand controller is addressed in three main stages: (i) the identification of the finger transfer function, (ii) the position control of the finger, (iii) and the formulation of a hybrid force-position control system. The identification of the finger takes into account the parametric model of the ProMain-I actuator, *i.e.* the XL-320 Dynamixel^{TM}servo-motor, coupled with the

[1] Patent Number: FR1656914

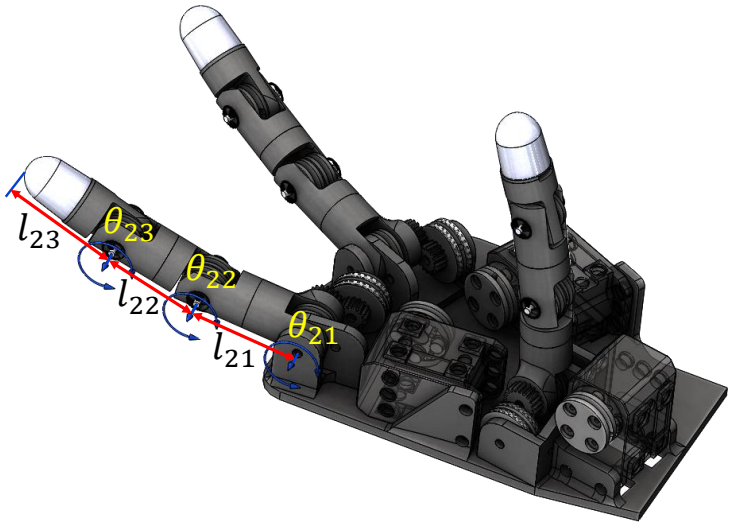


Figure 4.1: ProMain-I Hand with angles θ_{ji} and lengths l_{ji}.

finger mechanism. The position is identified directly using experimental data obtained from the actuator, and the fingertip force transfer function is obtained from a complete measure of the kinematic and fingertip force. The controller and transfer function parameters are estimated using an optimization approach.

4.1.1 Soft fingertip force sensor

The smart force sensor (1B), presented in Figure 4.2, is a new measure system whose main target is the detection of fingertip contact with obstacles or objects during grasping movements of the ProMain-I hand. The sensor (developed, tested and manufactured at the LEME laboratory [2]) has high sensibility detecting contact with objects and offers robust measures of fingertip forces.

The smart force sensor is composed of: (i) a soft cover (40), (ii) a fixation ring (30) for the resistive sensor (12), (iii) a Force Sensor Resistive (FSR^{TM}) transducer[3](12), (iv) bone (20), (v) and two support rings (50). The soft cover (40) is completely manufactured in a flexible material, using a 3D printed mold. The fixation ring (30), the bone (20), and the two support rings (50) are fabricated with rigid materials using a 3D printer. Table 4.1 summarizes characteristics and main properties of selected manufacture materials.

[2]Patent Number: FR1655991

[3]FSR is a Polymer Thick Film (PTF). This sensor varies its resistance depending on how much pressure is being exercised to the sensitive area. In our particular case the sensor used is FSR01 which has an sensing circular area of 5mm

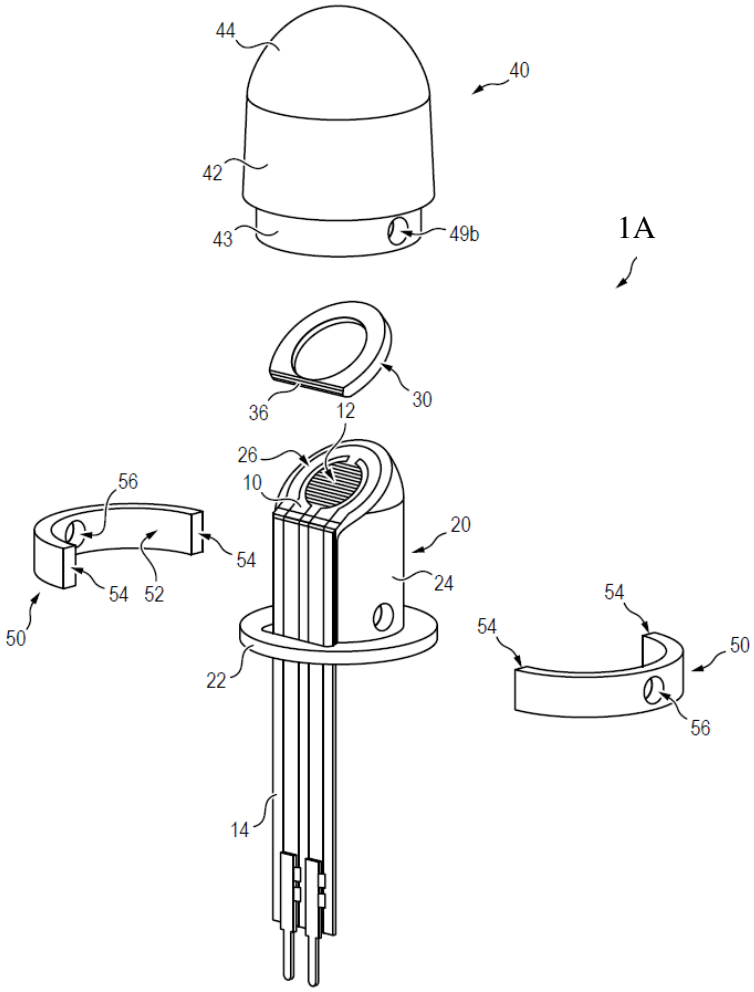


Figure 4.2: New fingertip force sensor detail view.

Device	Type	Example	Main characteristics
Soft cover (40)	Flexible: Hyper-elastic	RTV 3535 Polyaddition	Hardness Shore A = 50 Tensile Strength = 8 MPa Elongation at break = 120%
		RTV 127 Polyacondensation	Hardness Shore A = 22 Tensile Strength = 3MPa Elongation at break = 380%
SRS (30) Bone (20) Supports ring (50)	Rigid: Thermoplastic, Polyester plastic	Polylactic Acid (PLA)	Elastic (Young's Tensile) Modulus = 3.5 GPa Elongation at break = 6%

Table 4.1: Materials of smart sensor force components.

The soft cover (40) has a unique geometry developed to transfer the force that is applied

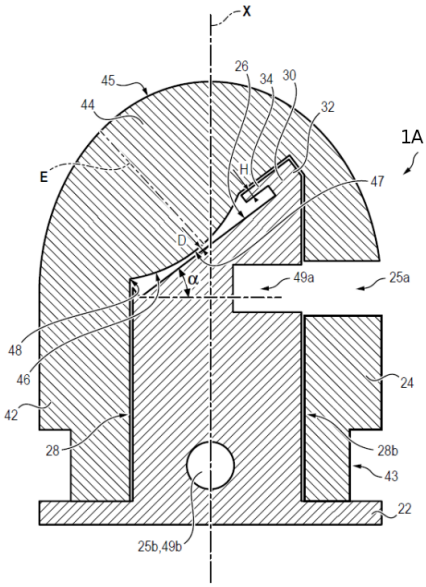


Figure 4.3: New fingertip force sensor: Detail view of new force sensor fingertip.

on the external fingertip surface toward the FSR transducer. The Soft cover (40) geometry is formed by: i) an envelop (44), ii) the elements (42 and 43), iii) and two drillings (49a) and (49b). Likewise, the soft cover (40) has a internal spheric zone (47), as shown in Figure 4.3, which gets in contact with the FRS transducer (12) when an external force is applied over the surface (45). This internal spheric zone (47) has a circular bound, which is defined by the zone (46) and (48).

The soft cover (40) is covering elements (20), (12) and (30) (see Figure 4.3). The support rings (50) pieces (42 and 43) to the bone (20) through two screws placed on the drilling (49a) and (25b). This fixation guarantees that the zones (28) and (28b) are: (i) blocked along the X axis, (ii) and in contact with the element (22). The FSR transducer (12) is placed over the zone (26) and is hold by the areas (34) and (32), of the fixation ring (30). The zone (26), of the bone (20), serves as a support for the force resistive sensor (12).

The operation of the smart force sensor is influenced by four geometrics parameters as follows: (i) the thickness E of the soft cover, which is the distance between active zone (47) and the external surface of soft cover (45) (see Figure 4.3), (ii) the thickness H of the fixation ring (30), (iii) the distance D between active zone (47) and the FSR transducer (12), (iv) and the α angle formed by the bone (20) (see Figure 4.3).

Considering that the target of the ProMain-I hand is precision grasping, we set the geometrics parameters as follows (see [4] for more details about the influence of geometric sensor parameters): E = 6.6mm, H = 0.39mm, D = 0.01mm and $\alpha = 36.79°$. The α value is proposed considering disposition of the fingers in the ProMain-I hand [1], and parameters E, H, and D are set regarding the required sensibility of the robotic hand that is up to 6N.

Experimental characterization of smart force sensor

The contact detection and the measure of fingertip forces is performed using FSR transducer (12) that transforms the intensity of an applied pressure into a variation of electrical resistance. Thus, we designed an op-amp based conditioning circuit to transform the resistance variation into voltage. As a result the force can be measured as a voltage variation. Furthermore, based on the above geometric parameter and using the selected materials (see Table 4.1), we proposed an experiment with the objective to know the correlation between external force and output voltage of force sensor.

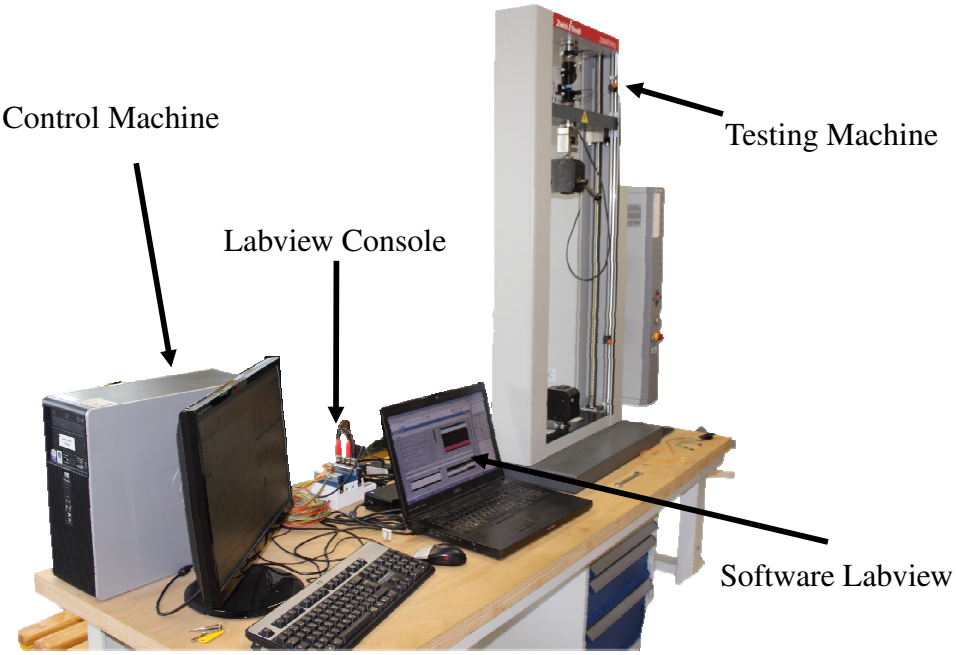
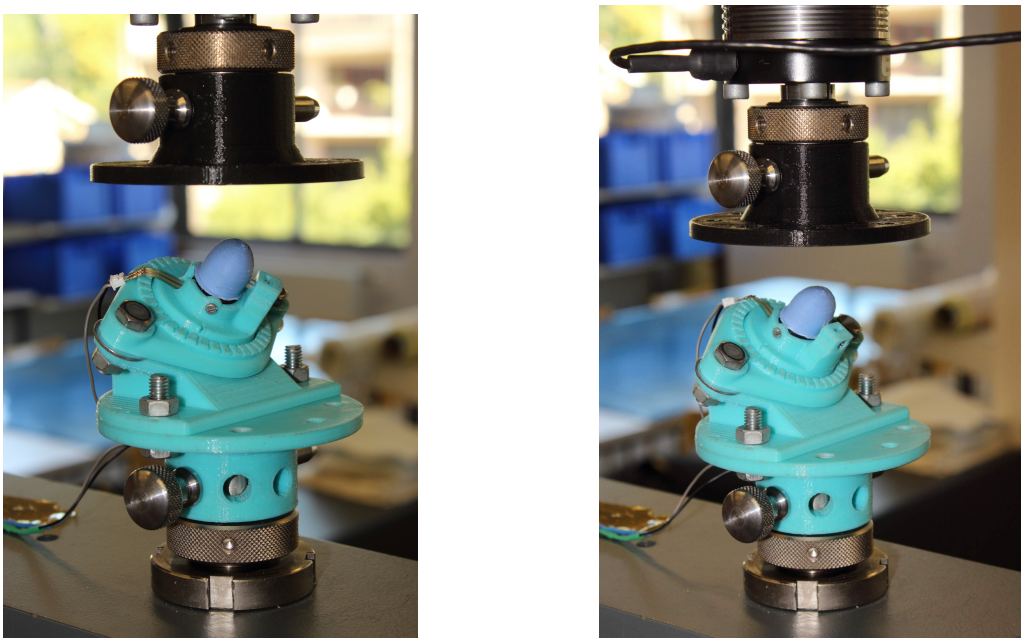


Figure 4.4: Experimental set-up.

The experiment consists in applying a known amount force over the fingertip's soft cover (40). The force is applied perpendicular to the FSR transducer (12) surface, guaranteeing that the force is distributed over FSR (12). Thereafter, the voltage produces by the smart force sensor is captured. The force is applied using a test machine Zwick/Roll^{TM}(see Figure 4.4). The machine is set to apply forces in the range $[0, 18]$ N. Taking into account that it is necessary to identify the influence of the direction of application of the external force, we replaced the clamping jaws of the machine by two supports, one is located in the upper side to apply the force using a horizontal plane and the other one is located lower sider to allow fingertip rotation from $-20°$ to $40°$. Figures 4.5a and 4.5b show the upper support black, and the lower support (blue) that allows finger rotation.



(a) Support fingertip without rotation.

(b) Support fingertip rotation = 40°.

Figure 4.5: Support fingertip.

The fingertip sensor is rotated by regular intervals of 10°, from 0° (see Figure 4.5a) up to 40° (see Figure 4.5b). These rotations allow us to quantify the relationship between external forces and output voltage for different operation conditions. We developed a data capture interface using Labview™that: (i) captures output voltage signal issued from smart fingertip force sensor, (ii) and captures position and force signals obtained from the testing machine. The sampled frequency is 100[Hz] for all signals.

Analysis of experimental results

We performed several trials for each fingertip's rotation value; during each trial, we measure the force applied by the testing machine and in the same time the voltage produced by the force fingertip sensor. Moreover, the experiments was performed considering two flexible materials: (i) Silicone RTV 3535 polyaddition (ii) and Silicone RTV 127 polycondensation.

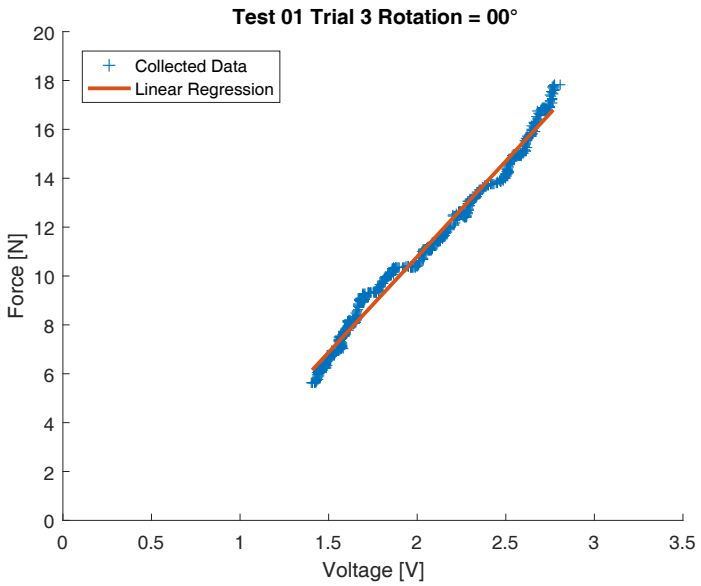
Regarding the experiments performed with the soft cover manufactured using the RTV3535 silicone, we repeated six trials six times when fingertip's rotation value is zero, four times when rotation is 10° and three times in all other cases; the number of trials for the experiment at 0° and 10° are higher than others as they constitute the most frequent operation for the ProMain-I hand [1].

The raw data is shown in Figure 4.6a, in which blue line represents the force applied by the testing machine, and orange line represents the output sensor voltage. The raw data was analyzed to find a experimental expression describing the relationship between the applied external force and the output voltage. On the basis of the obtained results, a first order equation

(computed by linear regression shown in Figure 4.6b) has been used to fit experimental data as shown in equation (4.1) below:

$$y(f_{st}) = mf_{st} + b \tag{4.1}$$

(a) Raw data of force obtained by the testing machine and sensor output voltage.



(b) Raw data experiment and its corresponding linear regression of one trial.

Figure 4.6: Experiment result of one trial.

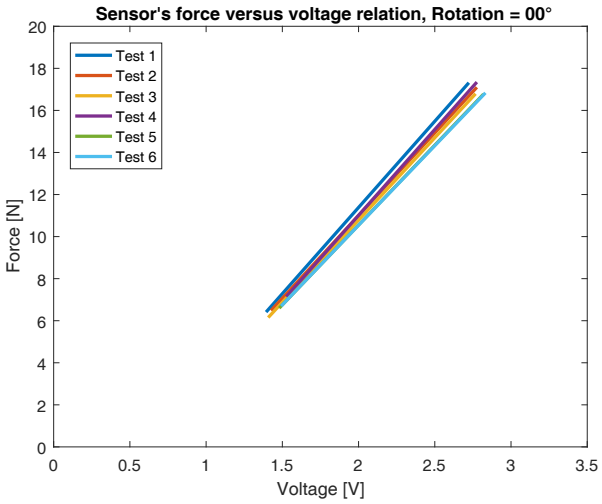
The complete set of experiments considering all trials for different rotations is summarized in Table 4.2. The mean value and the standard deviation for each rotation are presented. Moreover, Figure 4.7 presents the obtained results for all rotations.

Rotation	Trial	Slope *m* [N/V]			y-intercept *b* [N]		
		Value	Mean	std_dev	Value	Mean	std_dev
$0°$	1	8.199	7.867	0.270	-5.014	-4.856	0.271
	2	7.861			-4.723		
	3	7.823			-4.859		
	4	8.162			-5.313		
	5	7.594			-4.640		
	6	7.565			-4.592		
$10°$	1	8.300	7.992	0.305	-4.809	-4.814	0.320
	2	8.152			-4.840		
	3	7.916			-5.197		
	4	7.600			-4.412		
$20°$	1	8.353	9.143	0.695	-4.830	-5.86	0.985
	2	9.414			-5.958		
	3	9.662			-6.793		
$30°$	1	11.313	11.08	0.211	-6.750	-6.462	0.250
	2	11.026			-6.3467		
	3	10.901			-6.2915		
$40°$	1	17.193	16.680	0.717	-13.385	-13.08	0.886
	2	15.861			-13.385		
	3	16.987			-13.778		
$-20°$	1	10.185	10.114	0.087	-7.078	-7.298	0.489
	2	10.016			-6.958		
	3	10.143			-7.86		

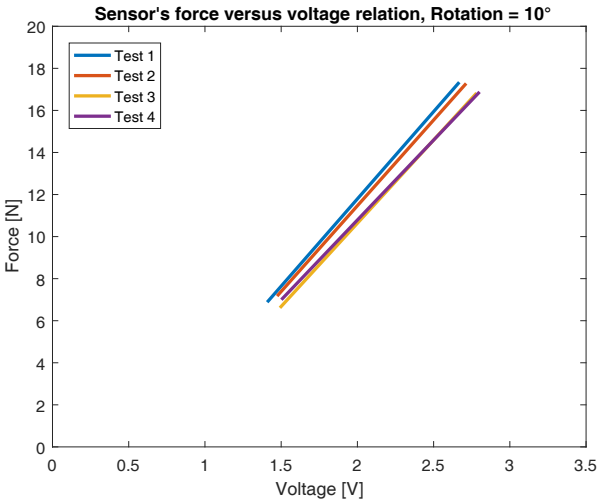
Table 4.2: Experimental results fingertip force sensor.

Experimental results obtained considering all rotations ($0°$, $10°$, $20°$, $30°$, $40°$, and $-20°$) have standard deviation regarding the calculated slope m and the ordinates-intercept b. the lower standard desviation indicates that values m and b are clustered around a mean value that describes the global behavior of all experiments. We found that for fingertip's rotation of $0°$, the slope mean value m is of 7.867[N/V] with standard deviation of the 0.270[N/V] and the mean ordinates-intercept b is -4.856[N] with standard deviation of the 0.271[N].

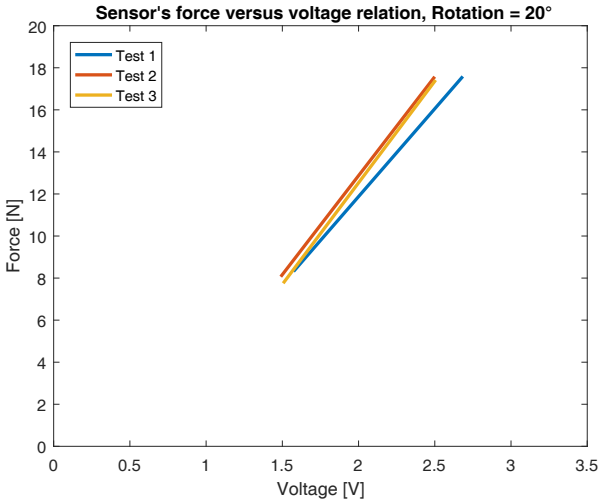
In addition, the slopes values increase when the rotation increases (*i.e.* when the force application plane is not parallel to the transducer surface the slopes values are higher than when the force application plane is parallel to the transducer surface). For instance, the case of $10°$ rotation, the mean slope value increases with respect to the values obtained at $0°$ rotation (see Figures 4.7a and 4.7b). Similarly, the lower limit of the detected force increases, when the force application plane is not parallel to the transducer surface. *e.g.* minimum detected values at $0°$ are about 6N, and at $10°$ are about $7N$.



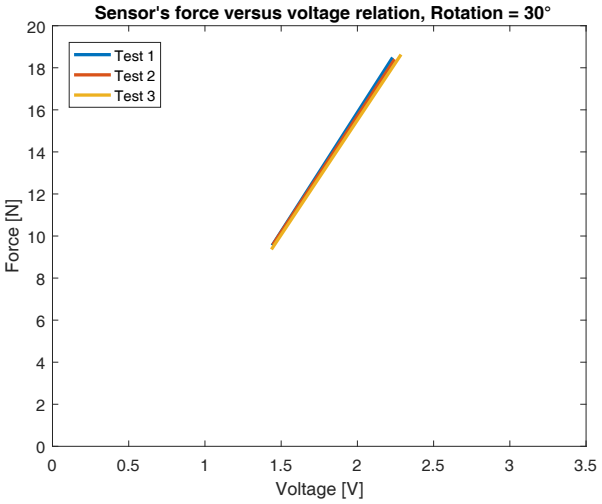
(a) Rotation = 0°.



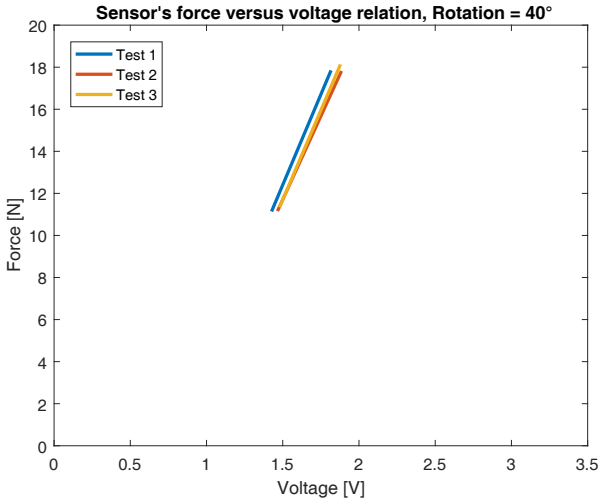
(b) Rotation = 10°.



(c) Rotation = 20°.



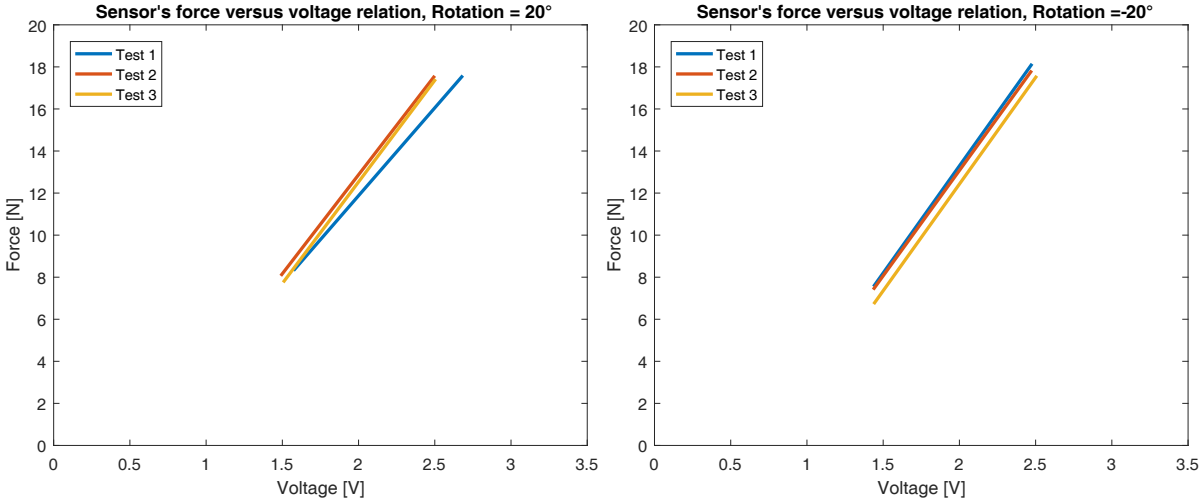
(d) Rotation = 30°.



(e) Rotation = 40°.

Figure 4.7: Experiment results of relation force and voltage, soft cover made of silicone RTV 3535.

Furthermore, comparing the experiment performed at $20°$ rotation with respect to $-20°$ rotation, we found that the fingertip sensor presents a symmetric behavior between positive and negative rotations. During experiments at $20°$ rotation the mean slope was 9.143[N/V] and experiments at $-20°$ rotation the mean slope was 10.114[N/V] (see Figure 4.8 and Table 4.2). Overall, the sensor is able to measure accurately external forces applied from $-20°$ till $20°$ rotation.



(a) Rotation = $20°$.

(b) Rotation = $-20°$.

Figure 4.8: Experiment result of force applied with positive and negative inclination.

On the other hand, experiments performed using RTV 127 silicon, which is more soft with respect to the RTV 3535, seven trial are performed at $0°$ and six for the other rotations; the more elevated number of trials is due the higher sensibility of the sensor due to more soft material usage. Figure 4.9 shows an example of the experiments performed and in this particular case the fingertip has a $0°$ rotation.

Figure 4.10a shows the raw data and the polynomial regression for the experiment performed at $0°$ rotation. The blue line represents the force applied by the testing machine, and the orange line represents the output sensor voltage. Consequently, a mathematical description is proposed aiming to find the relationship between the applied external force and the sensor output voltage (see Figure 4.10b). Due to the observed nonlinear behavior, a third-degree polynomial is considered:

$$y(f_{st}) = a{f_{st}}^3 + b{f_{st}}^2 + cf_{st} + d \tag{4.2}$$

where a, b, c, d are the polynomial's coefficients to be experimentally identified.

The found polynomial's coefficients values are summarized in Table 4.3. Additionally, Figure 4.11 presents the obtained results for all rotations. The standard deviation values, for trial at $0°$, $10°$ and $20°$ rotation values, are below 9.2% for all coefficients. This standard deviation is higher than the found for the first tested soft material, but it is still suitable for the measure of grasping forces. Moreover, the sensitivity of the sensor is better as the measure starts at 3N.

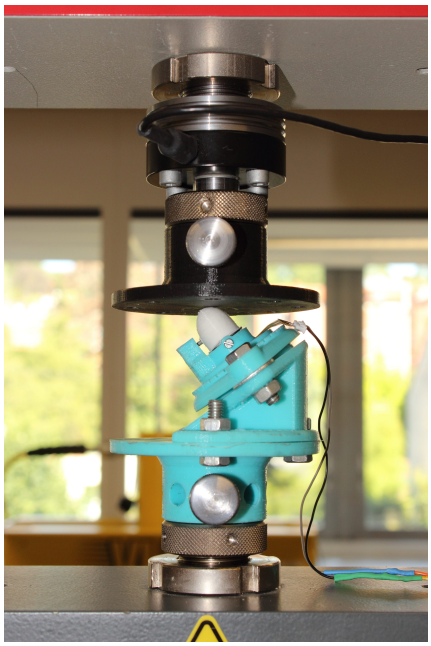
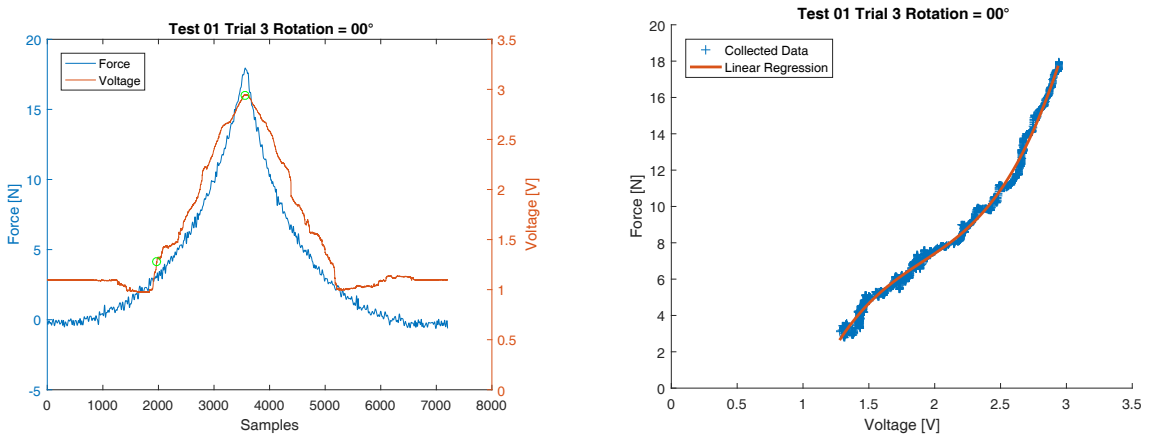


Figure 4.9: Experiments fingertip force sensor made of silicone RTV 127.

The standard deviation at 30° and 40° rotation values are higher than at 20°. Furthermore, we observed that the lower limit of the detected force increases when the force application plane is not in parallel to the transducer surface. For example, the case at 0°, 10° and 20° the minimum value detected is about 3N. The case at 30° the minimum value is about 4N, and finally the case at 40° rotation the minimum value is about 5N.



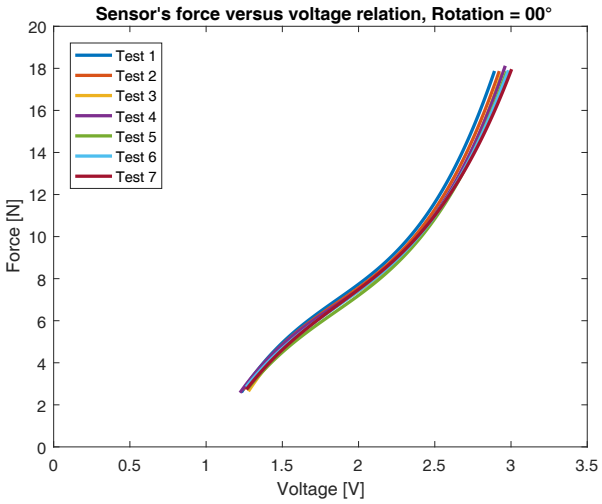
(a) Raw data of force obtained by testing machine and sensor output voltage.

(b) Raw data experiment and its corresponding polynomial regression.

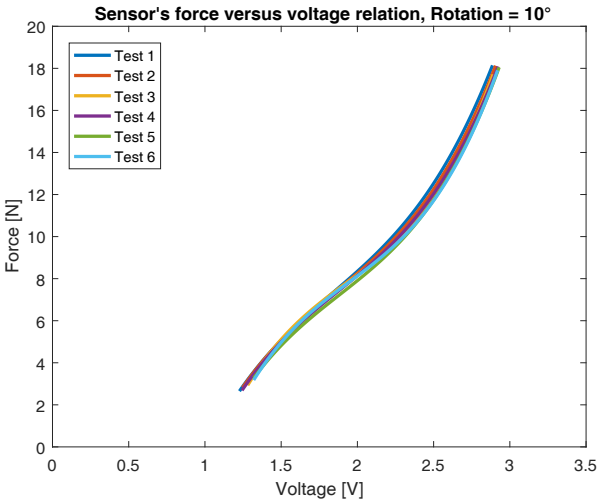
Figure 4.10: Experiment result of one trial force applied.

Rotation 0°				
Trial	***a***[N/V]	***b***[N/V]	***c***[N/V]	***d***[N]
1	5.044	-28.069	57.152	-34.642
2	4.710	-26.216	53.599	-32.450
3	5.328	-30.446	62.758	-38.953
4	4.496	-25.053	51.321	-30.934
5	4.015	-22.207	45.990	-28.071
6	3.827	-21.298	44.605	-27.219
7	3.595	-20.247	43.328	-26.948
Mean	4.431	-24.791	51.250	-31.317
std_dev	0.645	3.749	7.157	4.420
Rotation 10°				
1	3.725	-20.549	44.027	-27.306
2	3.910	-21.841	46.653	-28.967
3	5.270	-30.424	63.896	-40.068
4	3.948	-22.234	47.616	-29.731
5	4.231	-23.891	50.659	-31.693
6	5.232	-30.554	64.857	-41.228
Mean	4.386	-24.916	52.951	-33.166
std_dev	0.690	4.447	9.106	5.976
Rotation 20°				
1	5.515	-30.411	62.443	-38.281
2	4.099	-21.713	44.651	-26.398
3	4.579	-25.311	52.994	-32.461
4	3.855	-20.892	44.255	-26.754
5	4.132	-22.551	47.400	-28.764
6	4.360	-24.001	50.257	-30.611
Mean	4.423	-24.146	50.333	-30.545
std_dev	0.589	3.455	6.805	4.433
Rotation 30°				
1	11.245	-56.701	102.930	-58.008
2	12.258	-63.755	118.294	-69.004
3	7.105	-35.309	66.760	-38.335
4	9.164	-46.321	85.897	-49.017
5	9.111	-47.136	89.721	-52.694
6	10.212	-53.062	99.917	-58.432
Mean	9.849	-50.381	93.920	-54.248
std_dev	1.813	9.800	17.498	10.319
Rotation 40°				
1	10.423	-48.192	87.922	-48.902
2	17.302	-79.047	132.953	-71.060
3	17.936	-85.197	146.958	-80.702
4	27.217	-130.517	220.683	-120.240
5	35.721	-172.537	288.059	-155.389
6	13.790	-61.827	104.364	-55.355
Mean	20.398	-96.219	163.490	-88.608
std_dev	9.381	46.704	76.421	41.262

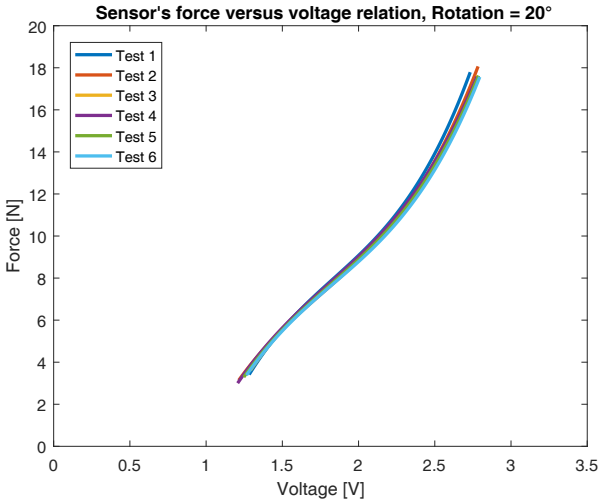
Table 4.3: Experimental results.



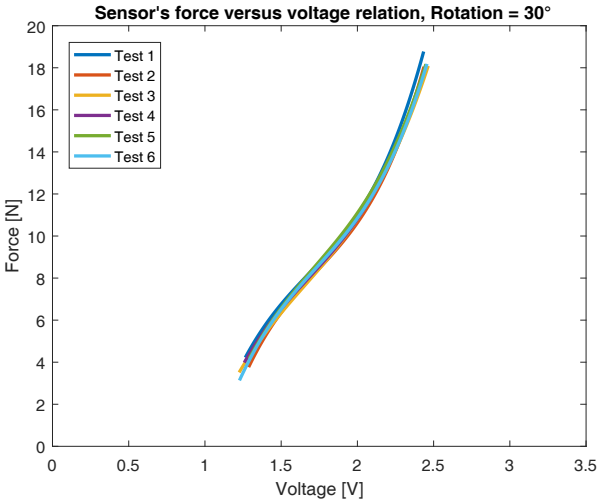
(a) Rotation = 0°.



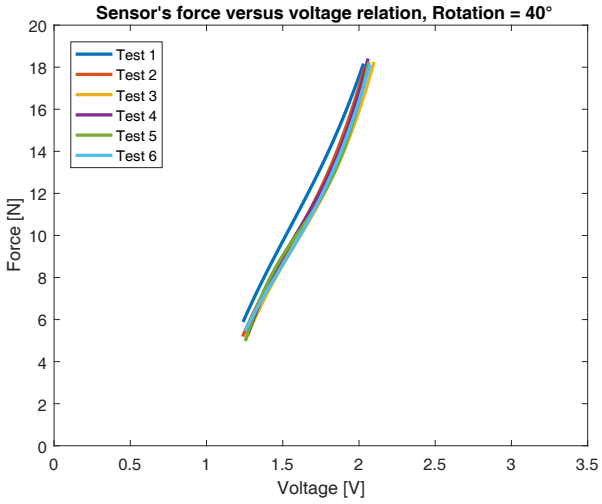
(b) Rotation = 10°.



(c) Rotation = 20°.



(d) Rotation = 30°.



(e) Rotation = 40°.

Figure 4.11: Experiment results of the relation force and voltage, softcover made of silicone RTV 127.

Comparing the results obtained when soft cover (40) are fabricated with RTV3535 and RTV127 silicones, we find out that: (i) standard deviation obtained using RTV3535 are lower ($\sim 1\%$) than obtained during test with RTV127 ($\sim 9\%$), (ii) the operation range of both materials are the same, from $-20°$ until $20°$, (iii) and the lower limits of the detected force using RTV127 are lower ($\sim$ 3N) than considering RTV3535 ($\sim$ 6N). Regarding grasping task, soft cover (40) made of RTV127 is more suitable than RTV3535 for the ProMain-I hand because of its force detection limits. In addition, the standard deviation founded in the experiments results are enough for the ProMain-I hand application.

4.2 Robotic finger parametric model

The parametric modeling of the robotic ProMain-I hand prosthesis is addressed in three stages: (i) formulation of a parametric model of the selected actuator XL-320 Dynamixel™, (ii) experimental identification of actuator's parameter, (iii) and experimental identification of the robotic finger taking into account rotations θ_1, θ_2 and θ_3 (of MP, DIP and PIP joints respectively) and the applied fingertip force f_j during grasping.

Consequently, in the following subsections, three stages to model and identify the robotic finger are proposed. Taking into account that the ProMain-I hand is composed of three fingers with the same architecture, the controllers and the models are applied to each finger of the hand.

4.2.1 Actuator transfer function

Parametric model of the actuator is based on the electric model and mechanical model. The electric model considers the sum of the voltage produced in the circuit, that includes resistance, inductor and internal voltage in the motor. The mechanical model takes into account the sum of torque produced by the elements, like servomotor's rotor and the ball bearing. Moreover, both models are coupled, to find a mathematical model, which relates the rotation produced in the servomotor $\theta(s)$ by the voltage applied $V_{in}(s)$.

The input rotation $\theta_1(s)$, which corresponds to MP joint, is provided by a XL-320 Dynamixel ™ servomotor, which is a robotic smart actuator that integrates a DC motor, a reduction gear-box, drivers, a communication module, a controller, a torque sensor, and a position sensor. Servomotor is represented by the block diagram shown in Figure 4.12, in which the main blocks represent transfer functions of: (i) torque sensor $S_\tau(s)$, (ii) position sensor $S_p(s)$, (iii) proportional–integral–derivative controller $PID(s)$, (iv) and direct current motor $G(s)$.

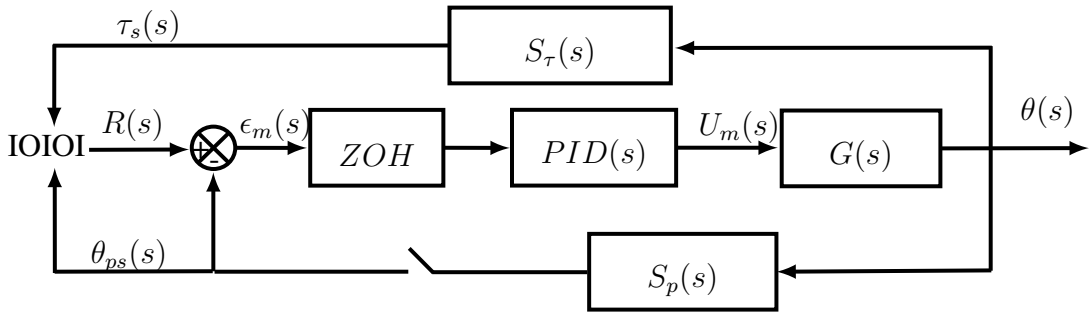


Figure 4.12: Servomotor block diagram.

The transfer function $S_\tau(s)$ represents the measure of magnitude and direction of the torque $\tau_s(s)$, and the transfer function position $S_p(s)$ depicts the measure of input angle $\theta_{ps}(s)$. Both τ_s and $\theta_{ps}(s)$ signals are digitized and sent to the computer through the serial interface. The computer calculates $R(s)$, which corresponds to the desired reference angular value. The difference between $R(s)$ and $\theta_{ps}(s)$ is the error signal $\varepsilon_m(s)$. This error signal is corrected through a $PID(s)$ controller, which output is the so-called control signal represented by $U_m(s)$. Furthermore, $U_m(s)$ is the input signal of the $G(s)$, say actuator input. The PID controller block diagram, shown in Figure 4.13, is composed of a proportional element $K_p s(s)$, a integral element $K_i(s)$ and a derivative element $K_d s^2(s)$. The transfer function G_c of the PID is presented in Equation (4.3) below:

$$G_c(s) = \frac{U_m(s)}{\epsilon_m(s)} = \frac{K_d s^2(s) + K_p s(s) + K_i(s)}{s} \tag{4.3}$$

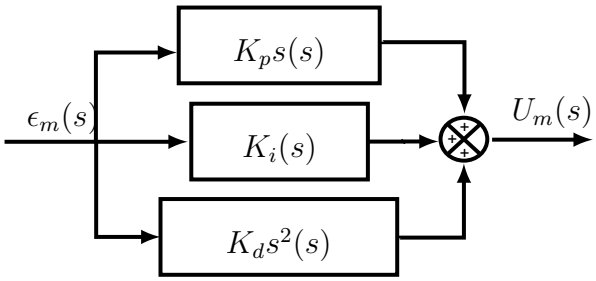


Figure 4.13: Proportional–integral–derivative controller block-diagram.

The transfer function of the direct current motor $G(s)$ is determined through a parametric model, considering electrical and mechanical behavior of the actuator. Figure 4.14 presents a equivalent electromechanical model of a DC actuator. Electrical model considers the following three internal elements: (i) a resistance R_{dcm}, (ii) a inductor L, (iii) and the electrical behavior of the motor M. Considering that these elements are disposed in series, the same current I flows through all the components, thus the Kirchhoff's voltage law is written as follows:

$$V_R + V_L + V_M - V_{in} = 0 \tag{4.4}$$

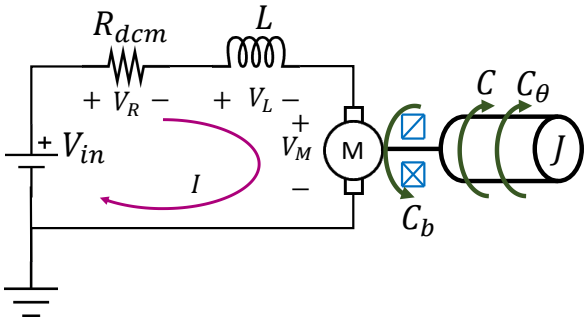


Figure 4.14: Electro-mechanical motor equivalent model.

Furthermore, the electromotive force, which corresponds to the voltage produced by the motor during movement, is modeled as $K_i\dot{\theta}$, where K_i is deduced from the Faraday's Law, considering: (i) the number of turns in the coil, (ii) turns geometry, (iii) rotor geometry, (iv) and magnetic field. As consequence Kirchhoff's voltage law is reformulates as presented in Equation (4.5) below:

$$IR + L\frac{dI}{dt} + K_i\dot{\theta} - V_{in} = 0 \tag{4.5}$$

Mechanical model is based on torque equilibrium considering the following components: (i) inertial reaction torque C, (ii) tthe ball bearings reaction torque C_b, (iii) and the motor torque C_M. ; and it is formulated as:

$$C_b - C_M + C = 0 \tag{4.6}$$

Taking into account that: (i) $C = J_M\ddot{\theta}$, being J_M the motor's moment of inertia and $\ddot{\theta}$ the angular acceleration, (ii) $C_M = K_tI$, being K_t the actuator torque, (iii) and $C_b = b_c\dot{\theta}$, being b_c the damping constant of the ball bearing; equation (4.6) is reformulated as:

$$b_c\dot{\theta} - K_tI + J_M\ddot{\theta} = 0 \tag{4.7}$$

Typically the coefficients K_i and K_t are considered to have the same magnitude, as a result both are substituted by a coefficient K_T. Then applying Laplace's transformation to equations (4.5) and (4.7), the transfer function $G(s)$ represents the relationship between the applied voltage and the corresponding rotation as follows:

$$G(s) = \frac{\theta(s)}{V_{in}} = \frac{K_T}{J_M s^3 L + (b_c L + J_M R)\, s^2 + ({K_T}^2 + b_c R)s} \tag{4.8}$$

The output of torque and position transfer functions $S_\tau(s)$ and $S_p(s)$ correspond exactly to the measured torque and position values. Thus both transfer functions is considered as $S_\tau(s) = 1$ and $S_p(s) = 1$. Likewise, the PID controller is disable by setting the coefficients to $K_p = 1$, $K_i = 0$ and $K_d = 0$. Thus, the block-diagram representation of the actuator (see Figure 4.12) is simplified by the equivalent block-diagram shown in Figure 4.15. This simplified block-diagram is reduced into only one block G_T using the block diagram algebra, resulting in the following transfer function:

$$G_T(s) = \frac{\theta(s)}{R(s)} = \frac{{}^{K_T}/_{J_M L}}{s^3 + \left({}^{b_c L + J_M R}/_{J_M L}\right) s^2 + \left({}^{K_T{}^2 + b_c R}/_{J_M L}\right) s + {}^{K_T}/_{J_M L}} \tag{4.9}$$

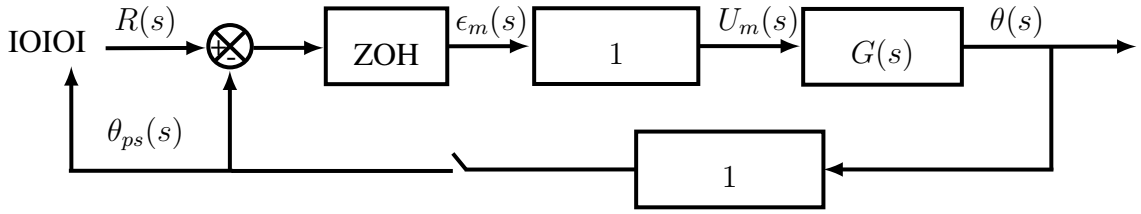


Figure 4.15: Equivalent block-diagram.

4.2.2 Actuator identification

The identification of actuator parameters (required for the governing equation (4.8)) is performed using an experiment that consists in: (i) applying a step input $R(s) = 300°$, (ii) measuring $\theta_m(s)$ of servomotor using the embedded position sensor, (iii) calculating parameters (J_M, K_T, L, R and b_c) to fit the model proposed in equation (4.9) regarding experimental data, (iv) and formulating the actuator transfer function substituting the found parameters in equation (4.8). Figure 4.16 presents the used experimental set-up.

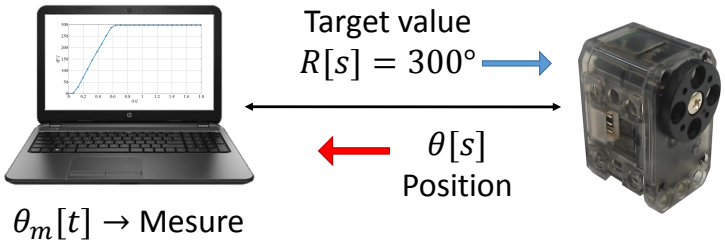


Figure 4.16: Experiment set-up.

The step input $R(s)$ is sent to the servomotor using a software especially developed for this experimentation. The software performs two tasks: (i) send the target value to the actuator, (ii) and read the servomotor's position value. The sent reference value and the obtained actuator position are synchronized and stored for the post-processing analysis.

Experiment results

In order to estimate parameters J_M, K_T, L, R and b_c of the servomotor, we solve equation (4.9) for the angle $\theta(s)$, as a result equation (4.10) is introduced. It is to note that we substitute the angle $\theta(s)$ by $\theta_c(s)$ to highlight that the resulting value is a simulation of the actuator angular position, *i.e.* $\theta_c(s)$ corresponds to the simulated angular position in response to the step input $R(s) = 300/s$.

$$\theta_c(s) = \frac{K_T/J_M L}{s^3 + \left(b_c L + J_M R/J_M L\right) s^2 + \left({K_T}^2 + b_c R/J_M L\right) s + K_T/J_M L} \frac{300}{s} \tag{4.10}$$

Then, applying inverse Laplace transformation to $\theta_c(s)$, we obtain the position response in time as $\theta_c(t) = \mathcal{L}^{-1}\{\theta_m(s)\}$. The simulated position value with respect to the time depends on the parameters J_M, K_T, L, R and b_c, thus it can be formulated as $\theta_c(t, J_M, L, R, b_c, K_T)$. The simulated value $\theta_c(t, J_M, L, R, b_c, K_T)$ is compared with the experimentally measured rotation $\theta_m(t)$ to compute the mean square error as follows:

$$\delta = \frac{1}{P} \sum_t \left(\theta_c(t, J_M, L, R, b_c, K_T) - \theta_m(t)\right)^2 \tag{4.11}$$

Consequently, the parameters J_M, K_T, L, R and b_c are calculated to fit experimental data minimizing mean square error introduced in equation (4.11). Thus, the optimization problem is formulated as:

$$\begin{aligned} &\underset{J_M, K_T, L, R, b_c}{\text{argmin}} \quad \delta = \frac{1}{P} \sum_t \left(\theta_c(t, J_M, L, R, b_c, K_T) - \theta_m(t)\right)^2 \\ &\text{subject to} \quad J_M > 0, K_T > 0, L > 0, R > 0, b_c > 0 \end{aligned}$$

The problem is solved using the minimization algorithm of constrained nonlinear multivariable functions of the MATLAB® optimization toolbox, which uses a sequential quadratic programming method. The obtained results are presented in Figure 4.17, in which the orange points represent experimental data collected from the servomotors' position sensor θ_m in degrees, and the blue line corresponds to the approximate value θ_c also in degrees. The yellow line in Figure 4.17 represents the error signal along the discrete time KT[s], which is calculated finding the absolute difference between θ_m and θ_c.

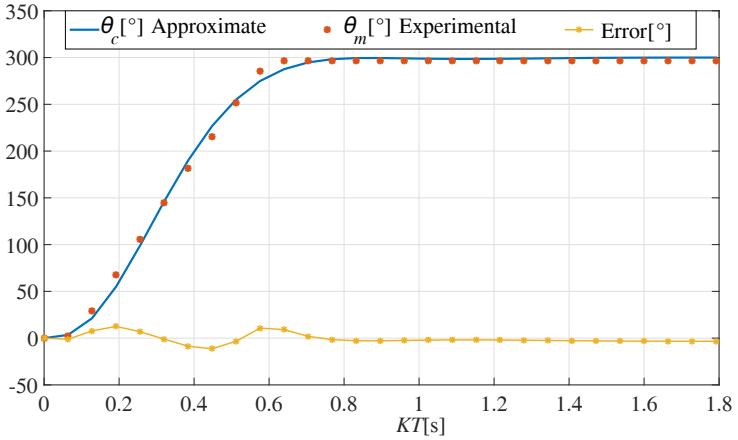


Figure 4.17: Experimental results of the motor transfer function identification.

Table 4.4: Experimental parameters: XL-320 Servomotor.

Parameters	Symbol	Value
Rotor inertia	J_M	0.0483
Inductance	L	0.0164
Resistance	R	0.2168
Ball bearing	b_c	0.0931
Motor constant	K_T	0.2677

Furthermore, substituting the parameters J_M, K_T, L, R and b_c (reported in Table 4.4 as a result of the optimization problem) in equation (4.8) we obtain the $G(s)$ transfer function that corresponds to the actuator as follows:

$$G(s) = \frac{0.26}{0.0004\,s^3 + 0.0093\,s^2 + 0.0865s} \tag{4.12}$$

Actuator control system

In this section, the tuning of proposed PID controller for the system $G(s)$ is introduced, the control system seeks to drives the position during grasping. The methodology to tune up the controller consists in: (i) defining control conditions, (ii) establishing the desire response, (iii) and finding the controller constants. The control conditions are taken into account to define the operational boundaries of the control system. These conditions are defined regarding the object to grasp and the prehension functionality of the robotic hand.

Regarding the object to grasp three considerations[4] are proposed: (i) the object is placed (with respect to the hand) in a position suitable for grasping, (ii) the object has the appropriate size, (iii) and the distance to reach the object and the size object are unknown. Regarding the functionality of the robotic hand, the only consideration is that the movement would be stopped when the fingertip force[5] reaches a value suitable to grasp the object in a steady way.

Based on the experimental observations carried out with the ProMain-I finger prototype[6], we find out that the finger response can be approximated using a second order behavior. Thus, we define the desired response using the transfer function introduced in equation (4.13) below:

$$\frac{\theta_d(s)}{R(s)} = \frac{K\omega_n^2}{s^2 + 2\zeta_d\omega_n s + \omega_n^2} \tag{4.13}$$

where ζ_d is the damping ratio, ω_n is the undamped natural frequency, K is a constant to handle the output amplitude, and $R(s)$ is a reference input signal.

The damping response of a second order system is influenced by parameter ζ_d. So that, four types of damping responses are defined: (i) the Undamped step response obtained when

[4]These considerations are done considering that a user manages the prosthesis to place it in grasping position.
[5]The fingertip force is measured using the sensor described in section 4.1.1.
[6]These experiments are detailed in section 4.2.3.

$\zeta_d = 0$, (ii) the under-damped step response achieved when $0 < \zeta_d < 1$, (iii) the critically damped step response reached when $\zeta_d = 1$, and (iv) the over-damped step response obtained when $\zeta_d > 1$. Figure 4.18 shows the step response obtained changing the damping ratio ζ_d; the reference signal is a step signal $R(s) = 300°/s$.

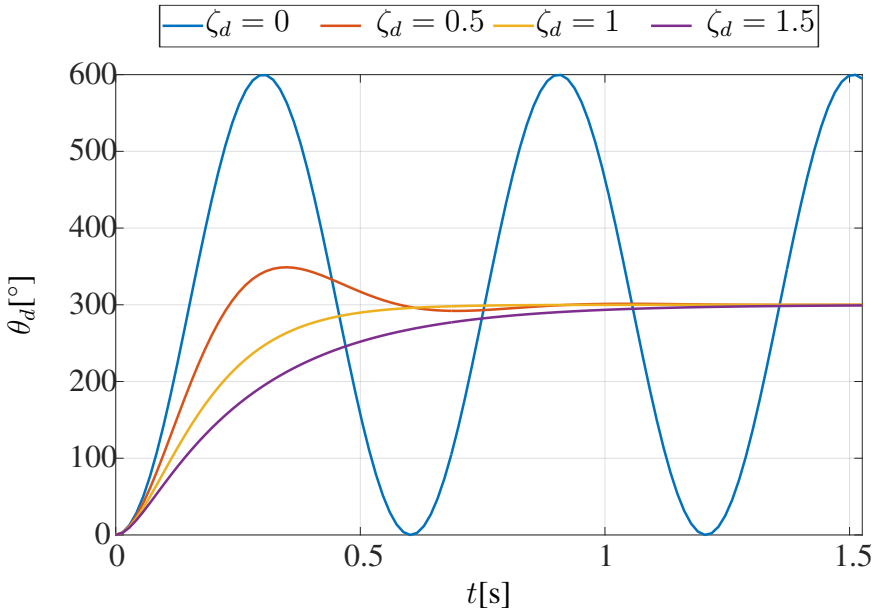


Figure 4.18: The influence of ζ_d.

The time response of a second order system has a strong relationship with undamped natural frequency ω_n. The higher ω_n value, the faster step responses, the influence is plotted in the Figure4.19, having the same reference signal $R(s) = 300°/s$.

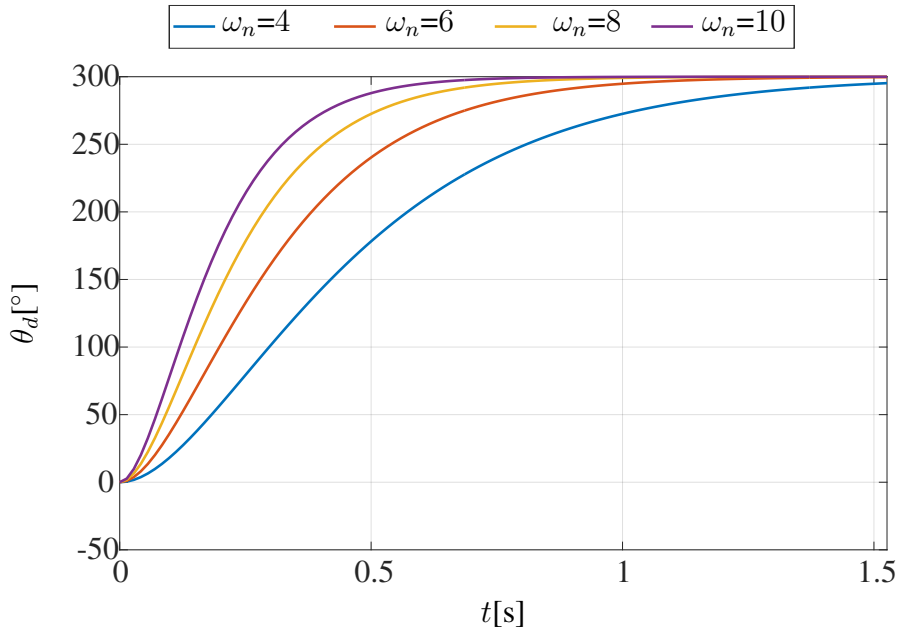


Figure 4.19: The influence of ω_n.

The force applied by humans during precision grasping presents a critically damped behavior and a settling time of $t_s = 0.244s$ [73]. Therefore, we choose the damping ratio $\zeta_d = 1$ to follow the critically damped behavior, and the undamped natural frequency $\omega_n = 4/ts$ to replicate the human settling time. The architecture of the controller $G_c(s)$ is the same presented in equation (4.3). Thus, kp, Ki, and Kd are calculated to fit the system to the human grasping conditions. The block-diagram presented in Figure Figure 4.20 shows the complete feedback control loop.

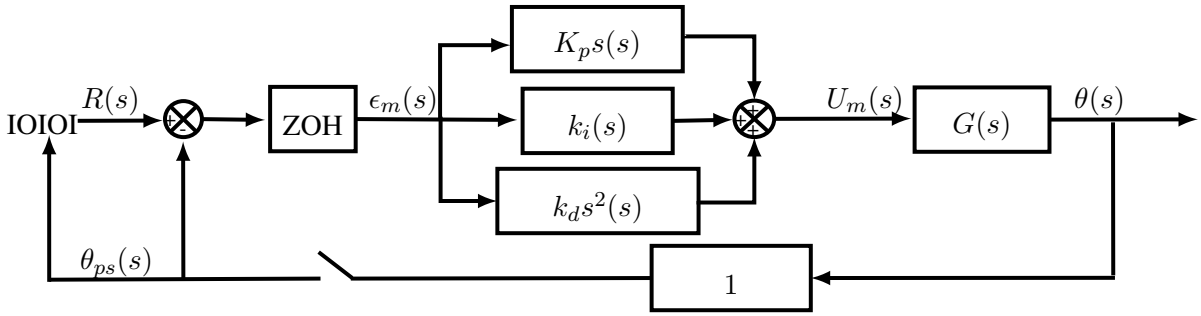


Figure 4.20: Block-diagram scheme of the ProMain-I hand.

The system is reduced to a transfer function $G_T(s) = {}^{\theta(s)}/_{R(s)}$, which is useful to compute the system output as function of the controller constants kp, Ki, and Kd. For that purpose, the controller is noted as $G_c(s)$ transfer function, and the block-diagram is reduced as presented in Figure 4.21.

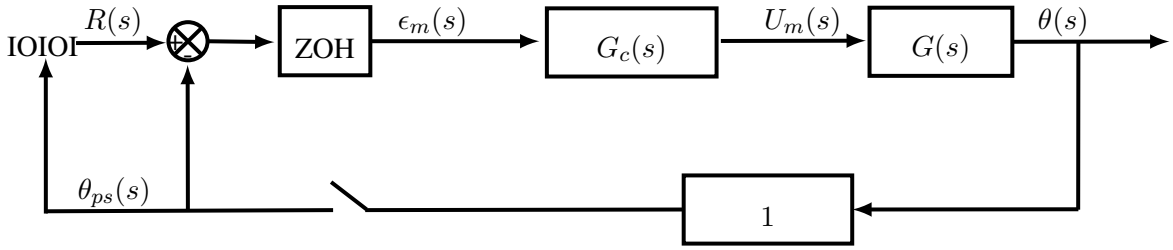


Figure 4.21: Equivalent block-diagram scheme of the ProMain-I hand.

The equivalent transfer function $G_T(s)$ of the reduced system is computed using equation (4.14). Furthermore, the step input corresponding to reference value $R(s)$ is formulated as presented in equation (4.15) in which the constant K corresponds to the desired actuator rotation and must be in the range $[0°, 300°]$ respecting the actuator limits.

$$G_T(s) = \frac{\theta(s)}{R(s)} = \frac{G_c(s)G(s)}{1 + G_c(s)G(s)} \tag{4.14}$$

$$R(s) = \frac{K}{s} \tag{4.15}$$

Therefore, operating equation (4.14) with the dc motor transfer function $G(s)$ (equation (4.3)) and controller $G_c(s)$ (equation (4.12)), we obtain the transfer function $G_T(s)$, which is operated to calculate the output position $\theta(s)$ as follows:

$$\theta(s) = \frac{2600(kd\,s^2 + kp\,s + ki)}{4.0\,s^4 + 93.0\,s^3 + (865.0 + 2600.0\,kd)\,s^2 + 2600.0\,kp\,s + 2600.0\,ki}\frac{K}{s} \tag{4.16}$$

Then, applying inverse Laplace transformation to $\theta(s)$, we obtain the output position in time domain as $\theta_c(t, k_p, k_i, k_d) = \mathcal{L}^{-1}\{\theta(s)\}$. Furthermore, we use the notation $\theta_c(t, k_p, k_i, k_d)$ to highlight that the resulting value is a simulation of the controlled angular position (*i.e.* $\theta_c(t, k_p, k_i, k_d)$ corresponds to a simulation of the controlled angular position in response to the step input $R(s) = K/s$).

Similarly, we calculate the desired output position, calculating the inverse Laplace transformation to $\theta_d(s)$. As a result, we find $\theta_d(t)$that corresponds to the desired output position in time domain. Subsequently, mean square error δ_{PID} between the simulated $\theta_c(t, k_p, k_i, k_d)$ and the desired $\theta_d(t)$ output signals is computed as follows:

$$\delta_{PID} = \frac{1}{P}\sum_t (\theta_c(t, k_p, k_i, k_d) - \theta_d(t))^2 \tag{4.17}$$

Thus, the controller constants are calculated using the following optimization problem:

$$\begin{aligned} &\underset{k_p,k_i,k_d}{\text{argmin}} \quad \delta_{PID} = \frac{1}{P}\sum_t (\theta_{t,k_p,k_i,k_d} - \theta_d(t))^2 \\ &\text{subject to} \quad k_p > 0, k_i > 0, k_d > 0 \end{aligned}$$

The problem is solved using the minimization algorithm of constrained nonlinear multivariable functions of the MATLAB® optimization toolbox, which uses a sequential quadratic programming method. Figure 4.22 presents the step response of the controlled system, the blue line is the simulated response of the controlled system, the yellow dashed line is the desired behavior and the red markers is the experimental measure of the controlled actuator. The obtained values of the PID constants are presented in Table 4.5.

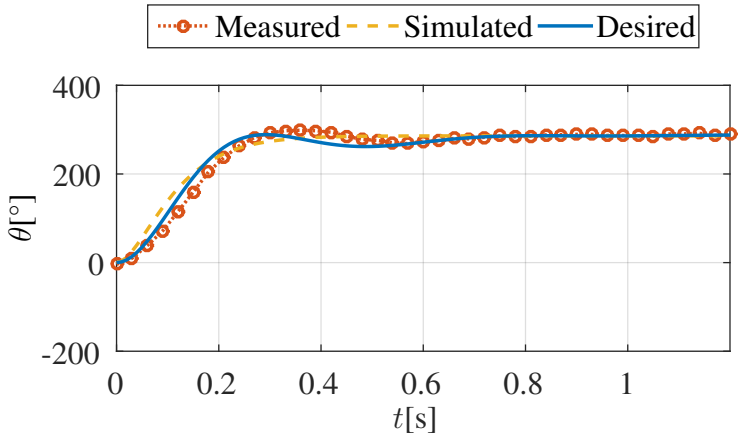


Figure 4.22: Step responses of controlled servomotor.

Table 4.5: Parameters servomotor control.

Parameters	Symbol	Value
Proportional constant	k_p	0.8566
Integral constant	k_i	0.0191
Derivative constant	k_d	0.0439

4.2.3 Robotic finger identification

The ProMain-I hand has three fingers: thumb, index, and middle. Each finger has the same architecture, number of phalanges and joints. Likewise, joints are notated by subscripts i (*i.e.* $i = 1$ represents MP joint, $i = 2$ PIP joint and $i = 3$ DIP joint). The kinematic model of the ProMain-I hand [6] uses a subindex j to distinguish fingers, however, for the sake of simplicity and considering that all fingers have the same behavior, we do not use this subindex[7] j here.

Each finger of the ProMain-I hand is controlled only by one servomotor that is mechanically coupled with the MP joint. The PIP and DIP joints of the finger are driven by a tendon-driven mechanism that guarantee a angular relation between joints given by $\theta_2 = \theta_3 = 0.9\theta_1$, in which θ_1, θ_2, and θ_3, are joint angles of MP, PIP and DIP joints respectively. Moreover, the force applied by fingertip is only present when the finger is in contact with an object or an obstacle. Thus, the identification of one finger aims to find a set of transfer functions to model the relation between the input angle $\theta(s)$ produced by the actuator and: (i) the joints rotation θ_1, θ_2, θ_3 (ii) and the force $f_j(s)$ produced during grasping task. Consequently, we define the following transfer functions:

$$k_{\theta_1} = \frac{\theta_1(s)}{\theta(s)} = 1 \tag{4.18}$$

$$k_{\theta_2} = \frac{\theta_1(s)}{\theta(s)} = 0.9 \tag{4.19}$$

$$k_{\theta_3} = \frac{\theta_1(s)}{\theta(s)} = 0.9 \tag{4.20}$$

$$G_f(s) = \frac{f_j(s)}{\theta(s) f_c(s)} \tag{4.21}$$

The first three represent MP, PIP and DIP joints rotation, and are based on the fixed mechanical rotation relations. The last one represents the relation between the fingertip force $f_j(s)$, the input angle $\theta(s)$, and the object contact modeled as $f_c(s)$. Figure 4.23 shows the block-diagram that represent the finger model.

[7] The proposed models and the identification methodology are used for all fingers, the subindex j is not used only to simplify notation

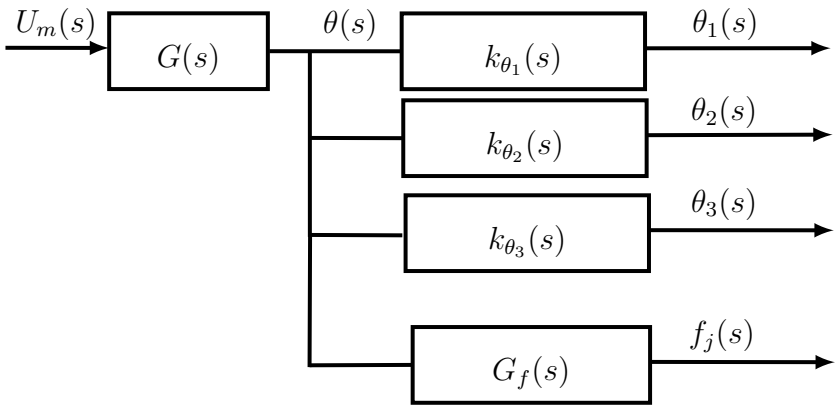


Figure 4.23: Block-diagram of joint angles and fingertip force.

4.2.3.1 Experimental identification of $G_f(s)$

The force $f_j(s)$ produced during grasping tasks is influenced by input angle, and the position of the object. Thus, an experiment to measure finger position and force is required.

Therefore, we designed a test platform in which a finger prototype is equipped with white markers placed over joints that are tracked using a high-performance 4 megapixel CCD camera Prosilica GE-2040. Figure 4.24 shows the experimental setup. A force sensor [8] is placed in a support located in the fingertip trajectory, so that, during the movement the fingertip gets in contact with the force sensor and the applied force is measured. A software to drive the finger and synchronize measured data has been developed. The software controls the finger to produce finger flexion until the finger gets in contact with the force sensor, and finally repositioning the finger. Several trials have been performed. Figure 4.25 illustrates the force measure procedure.

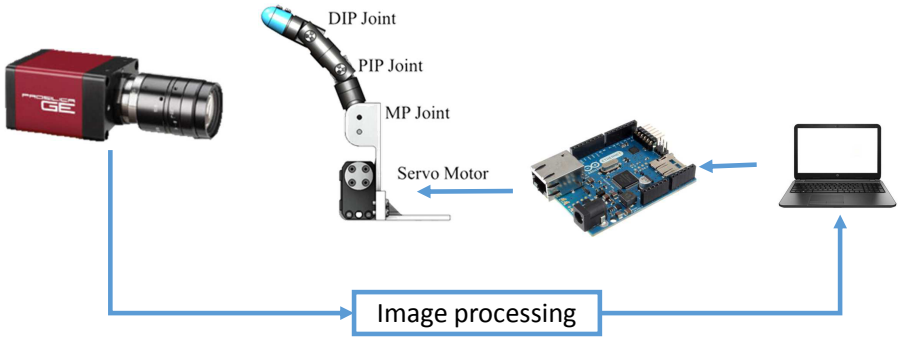


Figure 4.24: Set-up tracking position of the robotic finger articulations.

[8] The sensor is a FSR transducer Flexiforce®

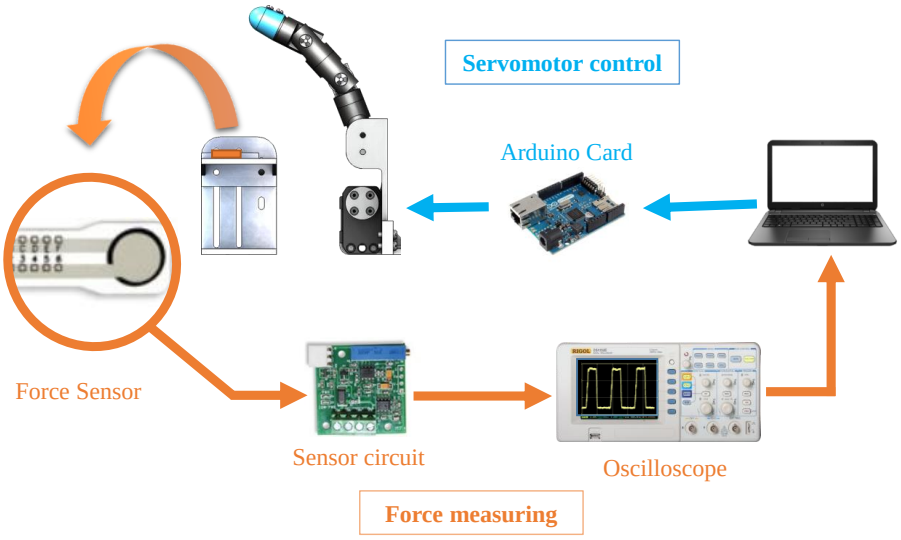
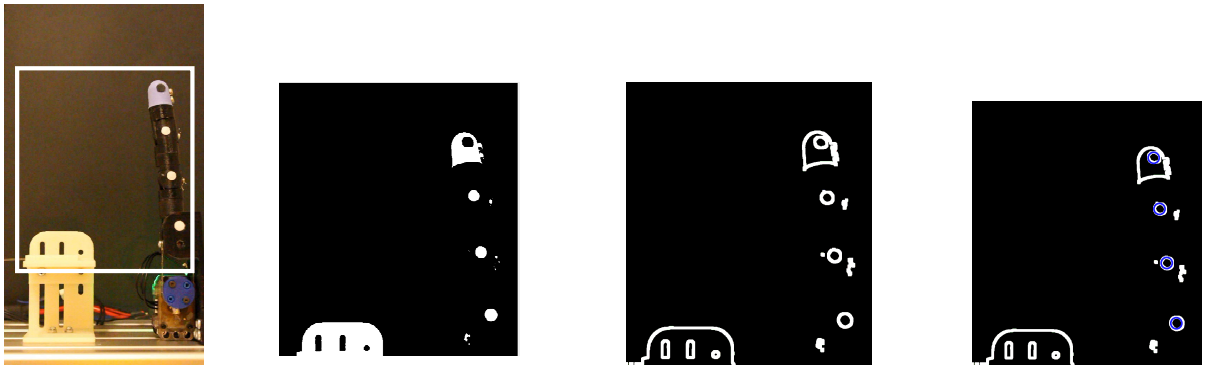


Figure 4.25: Set-up fingertip force.

Detection of joints position and angles

The images obtained from the camera are analysed with the aim of find the white markers positions that correspond to the joints position. The image processing follows these steps: (i) transformation image into a gray scale as shown in Figure 4.26b, (ii) detection of image edges as shown in Figure 4.26c, (iii) and find the position of white markers in the image apply Hough transform, see Figure 4.26d. Figure 4.27 shows a flexion cycle during four samples $t[s]$.



(a) Original. (b) First step. (c) Intermediate step. (d) Final step.

Figure 4.26: Image processing to track link.

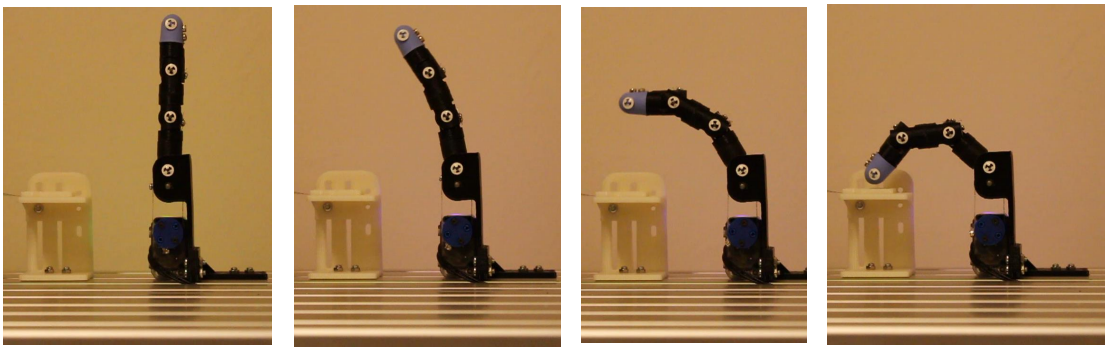


Figure 4.27: Position of the robotic finger articulations during flexion.

Once the position of each market is recognized, following three vectors linking joints are defined: (i) vector $\vec{r}_1$ between the MP and PIP joints, (ii) vector $\vec{r}_2$ between the PIP and DIP joints, (iii) and vector $\vec{r}_3$ between the DIP joint and fingertip. These vectors are used to calculate rotation angles θ_i as:

$$\theta_i = \arccos\left(\frac{\vec{r}_i \cdot \vec{r}_{i-1}}{\| \vec{r}_i \| \| \vec{r}_{i-1} \|}\right) \tag{4.22}$$

The first angle θ_1 is calculated with respect to a reference positive vertical unitary vector $\vec{r}_0 = \{0, 1, 0\}$. As a result, we found that the relation between angle variation in all joints are constant in time. The experiment is repetitive and corresponds to the mechanical design. Furthermore, MP joint motion range is $[0°, 50°]$, PIP and DIP joints motion range are from $[0°, 45°]$, as is shown in Figure 4.28.

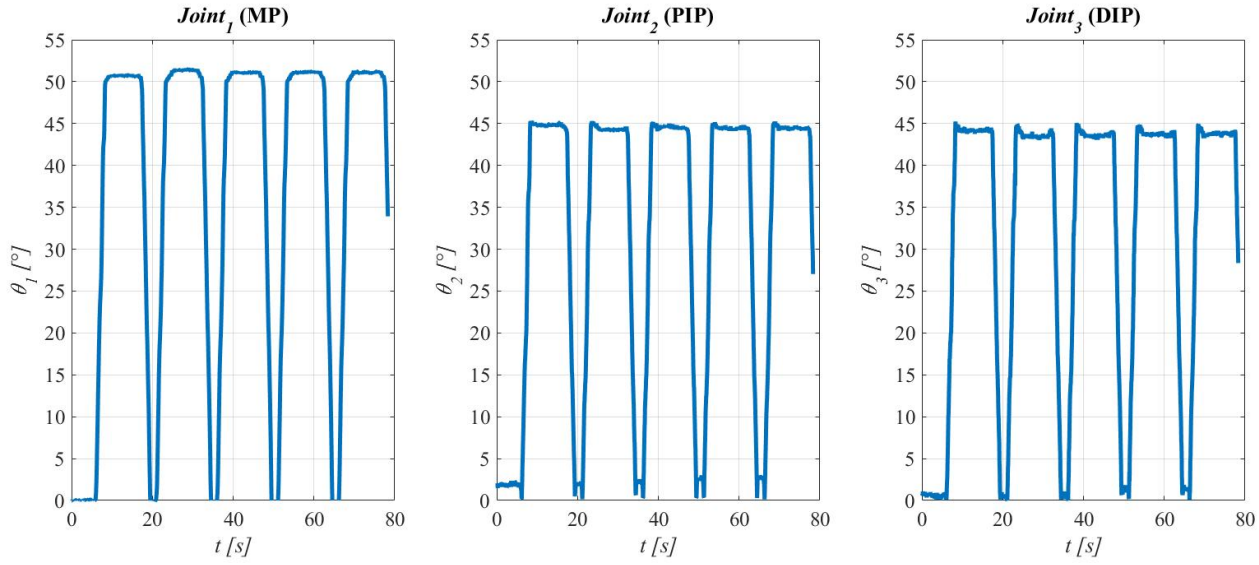


Figure 4.28: Set-up tracking position of the robotic finger articulations.

Identification of fingertip force

In order to identify the force as a non-parametric[9] transfer function $G_f(s)$, we analyze the measured angle θ_i and fingertip force $f_j(s)$ performed during the experiments. The measured fingertip force presents an overshoot as is shown in Figure 4.29, wherein blue lines correspond to the measured force $f_j(t)$[N] and orange lines tag the period whereby fingertip and the sensor are in contact.

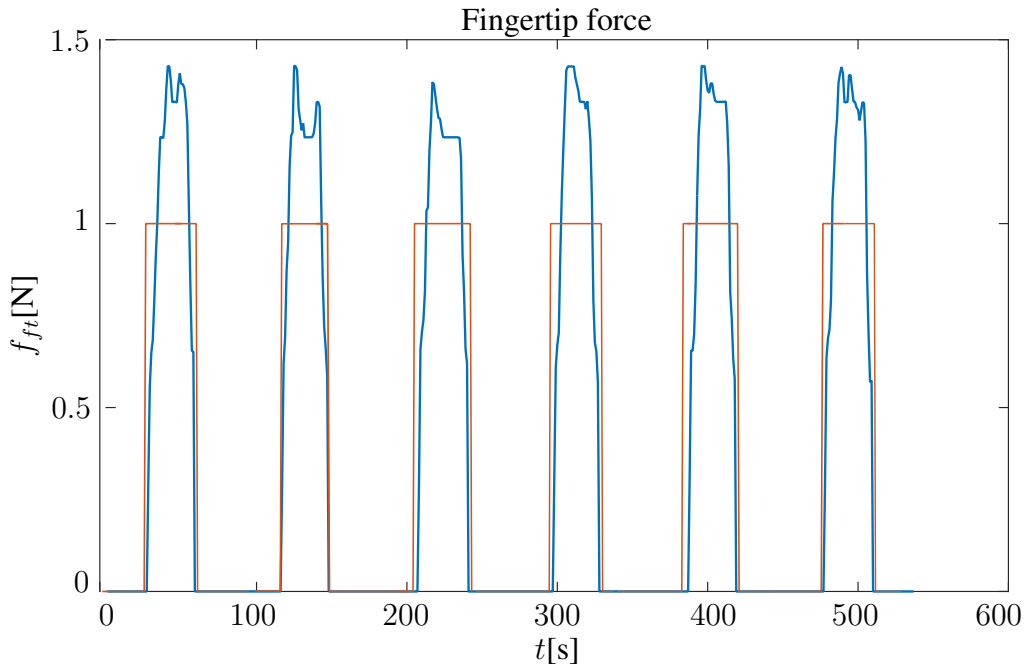


Figure 4.29: Experimental results fingertip force.

The fingertip force is modeled as transfer function $G_f(s)$ (see Equation (4.21)) in which: the output is the fingertip force $f_j(s)$ and the inputs are the angle $\theta_1(s)$ and the contact f_c. The contact f_c is considered as an input. Moreover, the input of the transfer function $G_f(s)$ is simplified to the angle increment $\Delta\theta(s)$ as:

$$\Delta\theta(t) = \begin{cases} 0 & \text{if } \nexists f_c \\ \theta(t) - \theta(t_c) & \text{if } \exists f_c \end{cases} \tag{4.23}$$

where t_c is the contact occurrence time.

Consequently, transfer function $G_f(s)$ is formulated as a second order dynamic system considering: an fingertip force gain k_{op}, a natural undamped frequency ω_{n_f}, and a damping coefficient $< 0\zeta_f < 1$. and $G_f(s)$ is formulated as:

$$G_f(s) = \frac{f_j(s)}{\Delta\theta(s)} = \frac{k_{op}\omega_{n_f}}{s^2 + 2\zeta_f\omega_{n_f}s + \omega_{n_f}^2} \tag{4.24}$$

[9]The non-parametric estimation of dynamic system response uses time-domain correlation analysis of experimental to generate a model in the form of a transfer function.

The coefficients k_{op}, ω_{n_f}, and ζ_f of the transfer function $G_f(s)$ are approximated fitting the experimental measures $f_{jm}(t)$ (shown in Figure 4.29) and the approximated f_{jc} issued from equation (4.24). Therefore, we propose an objective function based on the mean square error between the experimental results an the simulated force as follows:

$$\delta_{f_{iden}} = \frac{1}{L}\sum_{i=0}^{L}(f_c - f_d(t))^2 + \frac{K_p}{L}\sum_{i=t_{ss}}^{L}(f_c - f_d(t))^2 \tag{4.25}$$

where, L represents the total duration of the experiment, t_{ss} is steady state begin time, and K_p is a constant used to emphasize the importance of steady state response.

The value K_p is set to zero is no special attention is presented to the steady state value, and can be incremented to better fit steady state response the transfer function. Thus, the optimization problem is formulated as:

$$\begin{aligned} \underset{\zeta_f,\omega_{n_f},k_{op}}{\text{argmin}} \quad & \frac{1}{L}\sum_{i=0}^{L}(f_c - f_d(t))^2 + \frac{K_p}{L}\sum_{i=t_{ss}}^{L}(f_c - f_d(t))^2 \\ \text{subject to} \quad & 0 < \zeta_f < 1 \\ & 0 < \omega_{n_f} < 40 \\ & 0 < k_{op} < 30 \end{aligned}$$

The numeric problem is solved using fmincon Matlab®function, as shown in Figure 4.30 wherein: (i) experimentally measured force f_d is represented by blue crosses, (ii) yellow line depicts the contact, (iii) simulated force is represented by purple line, and (iv) $\Delta\theta(t)$ is represented by the orange line. The precision in steady state is increased setting $K_p = 100$.

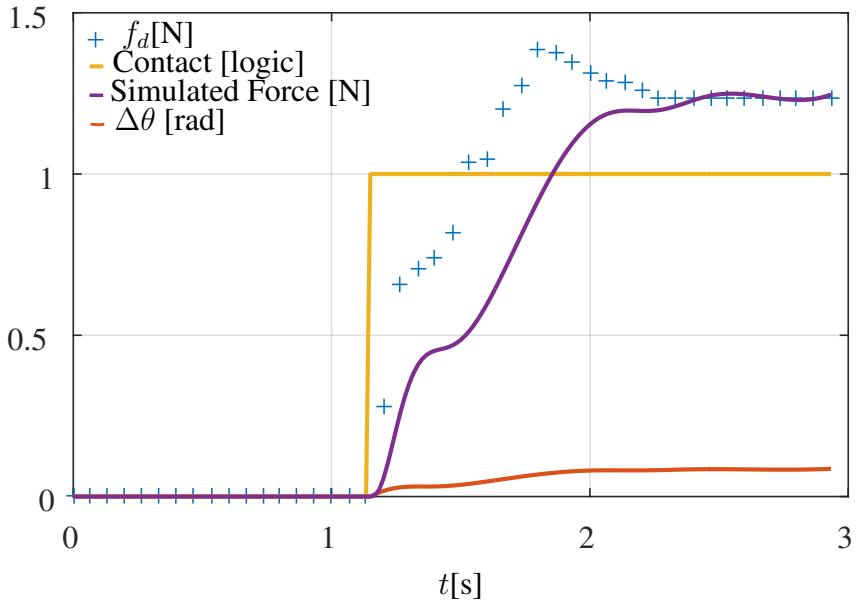


Figure 4.30: Identifications results.

Table 4.6 shows the results of the no-parametric identification of $G_f(s)$. Furthermore, the definitive transfer function $G_f(s)$ is formulated as follows:

$$G_f(s) = \frac{f_j(s)}{\Delta\theta(s)} = \frac{13.2669}{s^2 + 26.5338s + 217.2671} \tag{4.26}$$

Constants	Symbol	Values
Gain	k_{op}	30.0
Damping ratio	ζ	0.9
Natural frequency	w_{n_f}	14.741

Table 4.6: Results for $G_f(s)$ transfer function.

4.3 Optimized finger controller

Once all the transfer function are defined, we proposed the complete control scheme. We pointed out that: (i) force reference is denoted by $R_f(s)$, (ii) error force is denoted by $\epsilon_f(s)$, (iii) force controller $G_{cf}(s)$ produces an output $\Delta\theta$, which is the angle increment required to regulate force during graping, (iv) and the angle feedback is performed with $U\theta_1(s)$ as it matches the actuator angle value, the complete control system is presented in Figure 4.31.

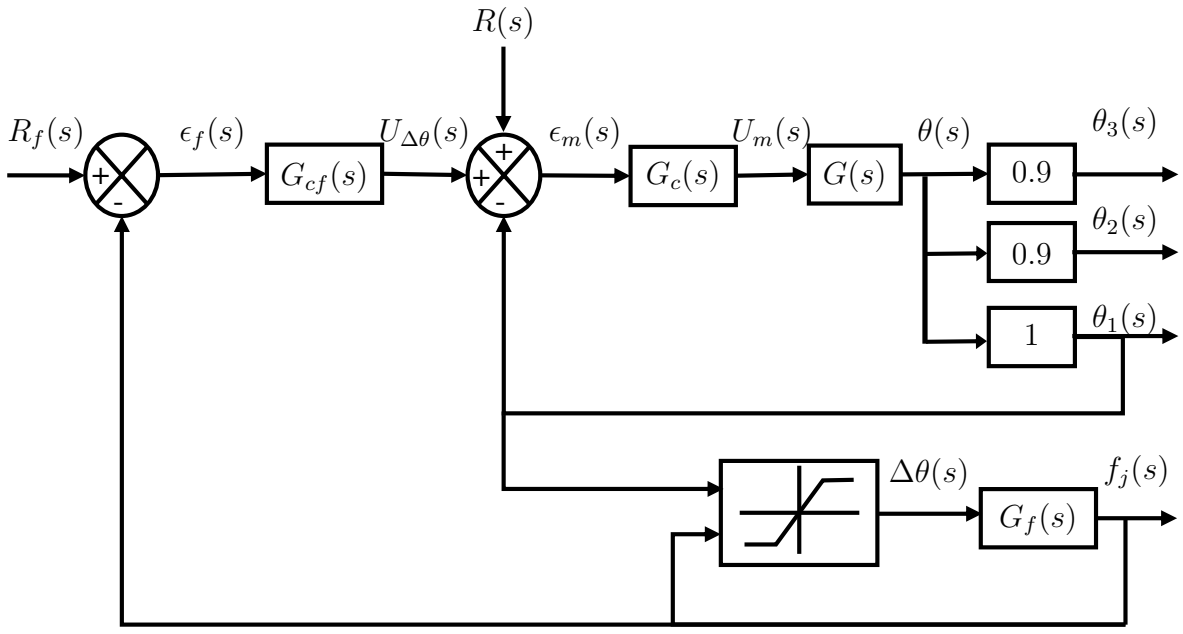


Figure 4.31: Block-diagram of control Promain-I robotic hand control.

The control variables are $\theta(s)$ and $f_j(s)$, as consequence, we formulated a force-position hybrid control. The controller consists of two closed loops, one using $G_{cf}(s)$ and other using

$G_c(s)$, in which the controller signals are $U\Delta\theta(s)$ and $U_m(s)$, respectively. The input signal of $G_{cf}(s)$ is $\epsilon_f(s)$, which is the force error value and is issued from the difference between force reference value $R_f(s)$ and the fingertip applied force $f_j(s)$. In other words, force error value is the difference between the desired and the actual forces.

The error $\epsilon_m(s)$ results from the difference between of $\theta_1(s)$ and the sum of $U\Delta\theta(s)$ signal and $R(s)$. Notably, it is also the difference between the actual θ angle value and commanded angle after controller's correction, considering not only the position ideal value but also the force ideal value.

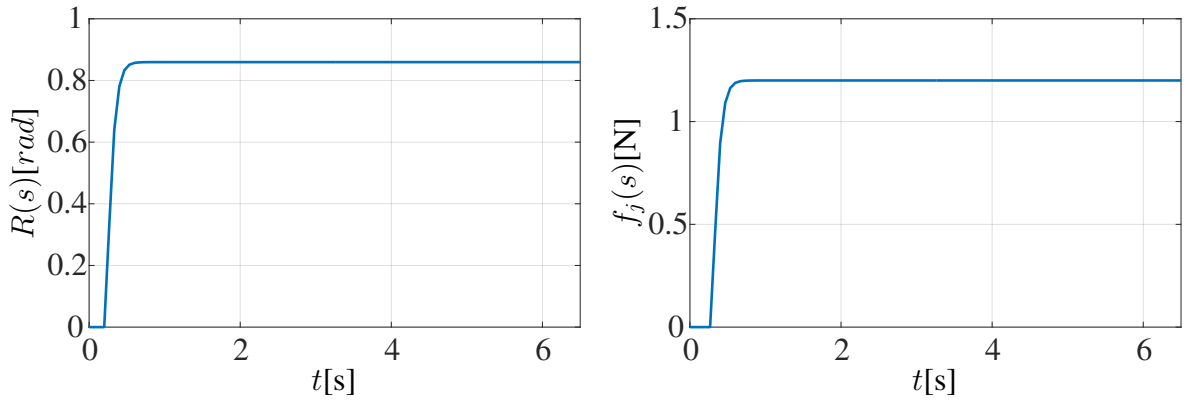
In addition, the internal closed-loop controls the $\theta(s)$ values thanks to the controller $G_c(s)$, which provides the control signal $U_m(s)$ to the $G(s)$ plant in oder to generate a corrective input. The loop is feeding with the $\theta_1(s)$ value, with the aim to (i) predict how the $G(s)$ plant will function in future updates, (ii) and make corrections in the present state based on previous evaluations.

The external closed-loop controls the $f_j(s)$ values, through the $G_{cf}(s)$ controller. The output of the controller is the $U\Delta\theta(s)$ signal. This signal causes a perturbation on the internal closed-loop providing a corrective input signal that also affects the θ_1 variable. This configuration based on feedback feeding improves the overall system performance.

In order to established the controller for the robotic finger, we followed these methodology: (i) defining of the motion and force specifications, (ii) modeling all the elements which are involved in the system, such as $G(s)$, $G_f(s)$, $k_{\theta_1}(s)$, $k_{\theta_2}(s)$, and $k_{\theta_3}(s)$, which are presented in subsections 4.2.1, 4.2.2, and 4.2.3, (iii) proposing the controller structure, and (iv) simulating the overall system.

Motion and force specifications

The first step toward the control of robotic finger consist in establishing the desired response of the robotic finger. According to global control architecture presented Figure 4.31 as block-diagram presented previously, it is required to establish two ideal responses: one for position (angle), and one force. In both cases, the damping ratios are $\zeta_{R(s)} = \zeta_{R_f(s)} = 1$ to guarantee a critical response, and the natural frequency of the system is set $w_{n_{R(s)}} = w_{n_{R_f(s)}} = 20$, which is higher than human grasping requirements [73]. Figure 4.32 presents the waveform of the desired outputs signals.



(a) Ideal $\theta(s)$ response.

(b) Ideal $f_j(s)$ response.

Figure 4.32: $\theta(s)$ and $f_j(s)$ desired response, represents by blue lines.

Controller structure

The hybrid force position controller is composed of two control loops, each loop is composed of a PID controller. The position controller (internal loop) is the one formulated inequation (4.3). The force controller is designed considering first-order filter on derivative term as follows:

$$G_{cf}(s) = \frac{U\Delta\theta(s)}{\epsilon_f(s)} = \frac{(K_d + t_f)s^2(s) + K_p s(s) + K_i(s)}{s} \tag{4.27}$$

The controller tunning procedure is the same as in previous sections and can be sumarized as follows: (i) the global system is reduced to a Multiple-Input Multiple-Output (MIMO) transfer function; the inputs are reference values of force $R_f(s)$ and position $R(s)$ and the outputs are finger rotation $\theta(s)$ and applied force $f_j(s)$, (ii) the reduced (MIMO) transfer function is simulated to obtain calculated values of $\theta_s(s)$ and $f_{sj}(s)$, (iii) the values $\theta_s(s)$ and $f_{sj}(s)$ are compared with desired responses to obtain objective function corresponding to the sum of position δ_p and force δ_f mean square errors, (iv) and the controller constant are calculated solving a optimization problem that minimize the computed mean square error.

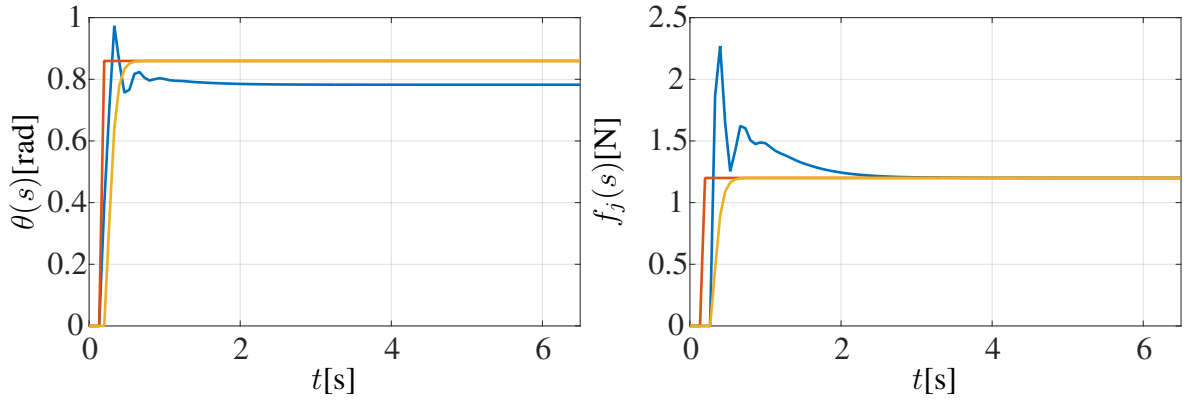
The optimization problem that minimizes the mean square error regarding force is multiplied by a constant K_p. The sum $K_f\delta_f + \delta_p$ defines the position controller constants (k_{pp}, t_f, k_{ip}, and k_{dp}) and force controllers constants (k_{pf}, k_{if}, and k_{df}) which are constrained to guarantee positive constant values. The constant K_f is used the enhance the importance of the force control over the position control, which is important considering that during grasping the object size is unknown and the finger can stop the motion when the fingertip force is detected even before reaching the goal position. The following equation (4.28) presents the optimization problem:

$$\begin{aligned} &\underset{k_{pf},k_{if},k_{df},t_f,k_{pp},k_{ip},k_{dp}}{\text{argmin}} && K_f\delta_f + \delta_p \\ &\text{subject to} && k_{pf} > 0, k_{if} > 0, k_{df} > 0, k_{pp} > 0, k_{ip} > 0, k_{dp} > 0, t_f > 0 \end{aligned} \tag{4.28}$$

Simulating the overall system

In the following the simulation of the complete hybrid control is presented. The obtained results are based on the experimental analysis performed with the finger prototype. The transfer function is transformed into difference equation using the discrete-time differential operator D introduced in section 3.3.2.2. In our case, the parameter μ is set to 1, assuring that the derivative is causal (*i.e* it is calculated based on the past and current values). Consequently, the discrete-time differential operator used is the proposed in equation (3.11).

Figures 4.33a and 4.33b present the simulation of the closed-loop response before launching our hybrid force position controller. The blue lines correspond to the force $f_j(s)$ and angular position $\theta(s)$ responses and yellow lines are the desired responses. Clearly, the finger motion is stopped approximately at 0.8 rad when the finger gets in contact with the object, and the force presents an overshoot after contact of approximately 180% with respect to the desired value. This overshoot implies the appliance of the double of the desired force over the object during grasping.

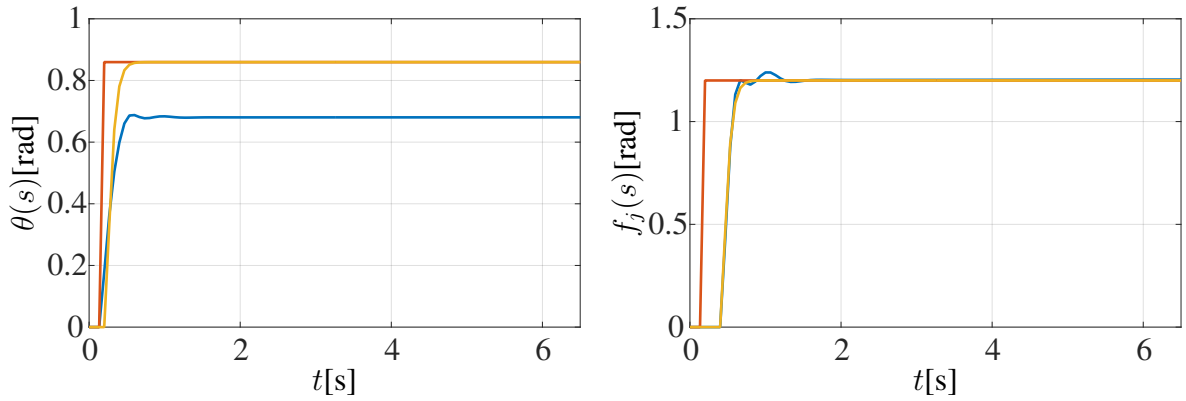


(a) Non-controlled closed loop for angular position response $\theta(s)$.

(b) Non-controlled closed loop for fingertip force response $f_j(s)$.

Figure 4.33: Non-controlled closed loop force-position responses.

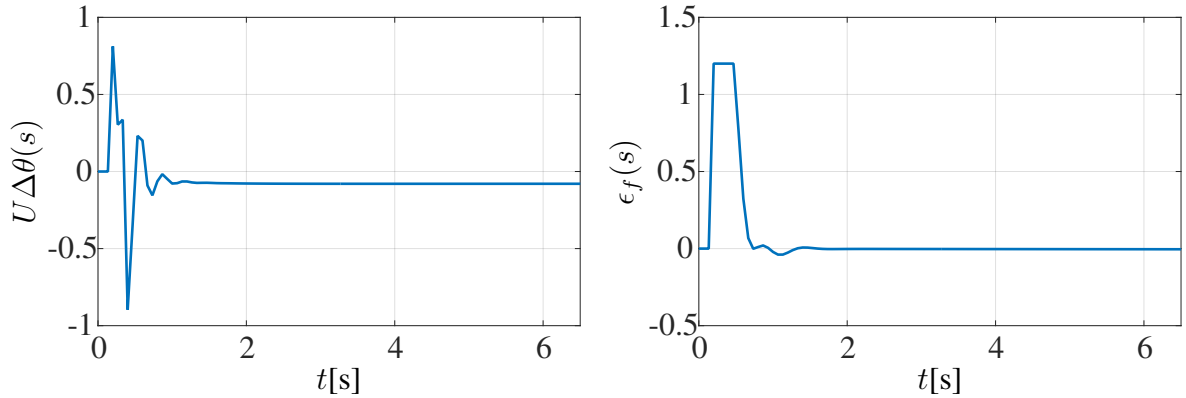
The force and position controllers, described in equations (4.3) and (4.27) that solve the optimization problem introduced in equation (4.28), are launched to control force-position responses of the finger. The controlled force-position responses are presented in figures 4.34a and 4.34b, wherein blue lines correspond to the force $f_j(s)$ and angular position $\theta(s)$ responses and yellow lines are the desired responses. Clearly, the position error in steady state is about two times higher than the non-controlled position error, but the force overshoot is considerably reduced guaranteeing a low steady state force error.



(a) Controlled angular position response$\theta(s)$. (b) Controlled fingertip force response $f_j(s)$.

Figure 4.34: Controlled closed loop force-position responses.

Regarding the control signal of the force controller, it is interesting to note that in steady state it has a negative value as shown in Figure 4.35a. This negative value of $U\Delta\theta(s)$ corresponds to the value subtracted to the reference position value in steady state (see Figure 4.31) that increases the error position but produces a lower overshoot guaranteeing a force steady state error that tends asymptotically towards zero as shown in Figure 4.35b.



(a) Control signal $U\Delta\theta(s)$. (b) Force error signal $\epsilon_f(s)$.

Figure 4.35: Force error and control signals during closed loop hybrid control.

4.4 ProMain-I Hand control

The hybrid force-position controller designed and tuned in previous sections is implemented using a software interface developed using c++. The interface allows a user to calibrate the fingertip force sensors and to normalize force values. Furthermore, the software allows to control each finger individually and to perform open-close hand gestures allowing the prehension of different objects.

The reference values $R_f(s)$ and $R(s)$, are defined according the desired movement introduced by the user; the sensor values are captured using an analog to digital converter imple-

mented in an Arduino card (integrated in the robot forearm); the force and position errors are calculated at each iteration, and then transmitted to the robot through a serial bus together with the force control signal computed using equation (4.27). As a result, we achieve the grasping gestures shown in Figures 4.37 to 4.42, 4.47 and 4.48. during grasping the force reference is set to hold the object in steady way and is controlled to stop motion when the object is grasped.

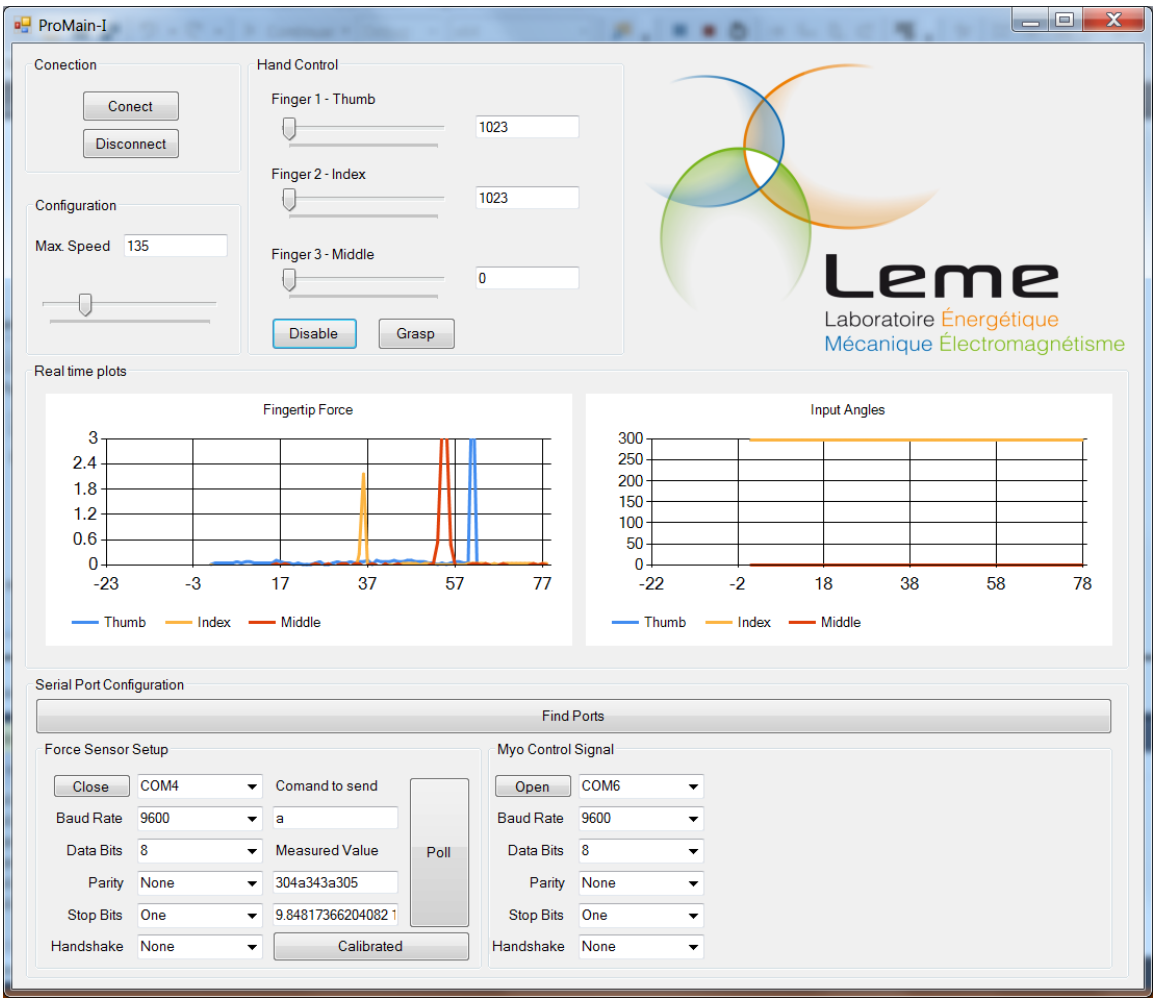


Figure 4.36: Software interface of the controller.

Figure 4.37: Grasping gesture 1 (a).

Figure 4.38: Grasping gesture 1 (b).

Figure 4.39: Grasping gesture 2 (a).

Figure 4.40: Grasping gesture 2 (b).

Figure 4.41: Grasping gesture 3.

Figure 4.42: Grasping gesture 4.

Figure 4.43: Grasping gesture 5 (a).

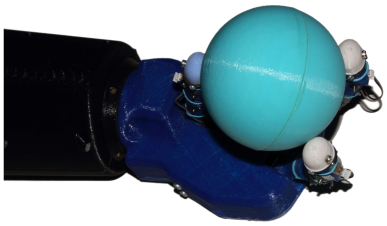


Figure 4.44: Grasping gesture 5 (b).

Figure 4.45: Grasping gesture 6.

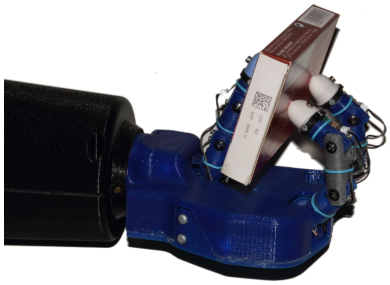


Figure 4.46: Grasping gesture 7.

Figure 4.47: Grasping gesture 8.

Figure 4.48: Grasping gesture 9.

4.5 Conclusion

We present the ProMain-I hand characteristics, and the new smart force sensor, which has been designed and manufactured (in the LEME laboratory) to implemented the feedback force loop.

Several experiments are carried out to characterize the new sensor, we find the relation between the external applied force on the smart sensor and the obtained sensor voltage, these experiments are performed using a testing machine. Additionally, we test two type of soft materials, RTV3535 and RTV127.

As a result, we find that the lower limit of the detected force using RTV3535 is ≈ 5N and using RTV127 is ≈ 3N. As a consequence, the selected soft material is RTV127 due to its more suitable sensibility.

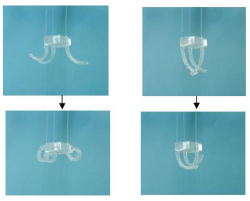
Subsequently, we introduce the robotic finger parametric model to characterize the finger in force and position. Then, we design a optimized finger controller. Finally we propose a force position-hybrid controller. As a result, we achieved to perform grasping gestures setting the force reference to hold the object in steady way.

Chapter 5

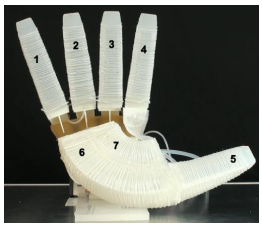
ProMain- II Soft Hand

5.1 Robotic Soft Hands

Soft robotics [18] is a new research field, that considers : (i) the use of soft materials such as elastic, flexible or deformable bodies, (ii) the increment of degrees of freedom with respect to conventional materials, (iii) the use of unconventional materials, like shape memory alloys, thermoplastic polymer or electro-active polymers, (iv) and the innovation regarding mechanism, which is the case of the under-actuated hands, for instance the Laval hands [74].



(a) Soft Hand based on flexible micro-actuator [75].



(b) RBO Hand [17].

Figure 5.1: Soft hands (part A.)

Wakimoto et al. [75] developed a soft hand based on a flexible micro-actuator, as shown in Figure 5.1a. It is composed of fiber-reinforced rubber structure with multi air chambers. The working principle is to inject air in rubber structures chambers, producing bending motion.

Considering the uses of pneumatics continuum actuators, RBO hand [17] was designed using a pneuflex, as shown in Figure 5.1b, that has a high compliance with grasped objects. The main advantages of this hand are: (i) robust to impact, (ii) and safe.

ROBIOSS hand is a fully-actuated and endowed with 16 DoF, as is shown in Figure **??**. The hand has four fingers, each one with four DoF actuated by four DC actuators using a driving mechanism with an elastic transformation[76].

Pisa–IIT SoftHand [16, 77] is an under-actuated hand, as illustrated in Figure 5.2a that is built on the motor control principle of synergies [78]. This concept allows grasping objects with

a single motor. The mechanism consists in a tendon routing and elastic bands at the joints. The hand is simple, adaptable and robust. These characteristics enable to grasp several number of objects.

(a) ROBIOSS Hand [76] (b) Pisa–IIT SoftHand [16, 77].

Figure 5.2: Soft hands (part B.).

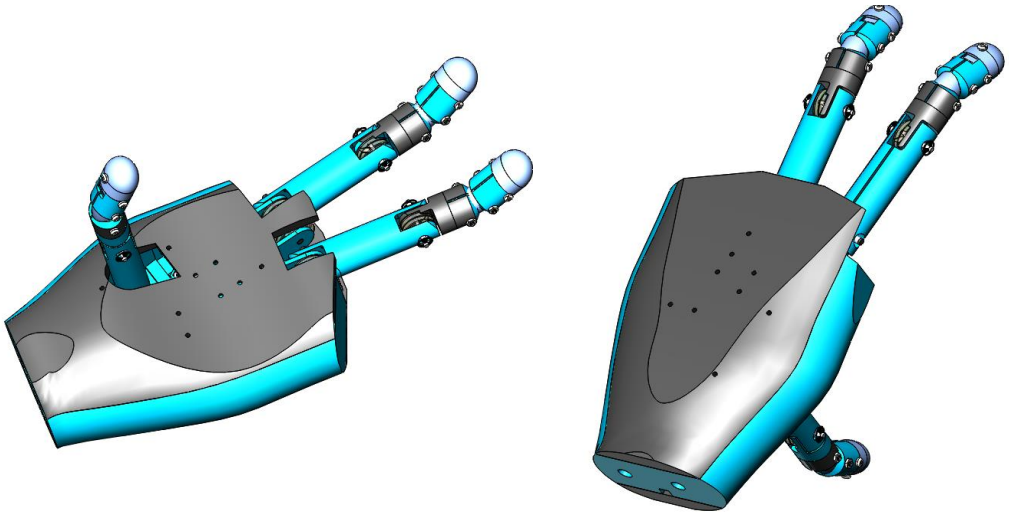
The synergy concept is mainly defined as [79]: (i) postural synergies, which are related to the description of the complete act of grasping using a small number of hand gestures, (ii) kinematic synergies, which studies joint angular velocities using a small number of linear combinations ok kinematic chain parameters extracted from the non-limited hand movements. (iii) and dynamic synergies that are the correlation between joint torques during precision grip movements.

Summarizing, the main advantages of the soft robotics hand are: (i) compliant dexterous manipulation, (ii) safe interaction, (iii) low mechanical impedance, (iv) robust to impact, (v) high flexibility and adaptability, (vi) and low cost.

5.2 ProMain-II Hand characteristics

The soft robotic hand ProMain-II[1] [5, 73], has been designed and developed completely in the LEME laboratory. It is created with the aim to increases the compliant of the hand, considering flexible materials in the structure. It has three fingers disposed to carry out precision grasping. Each finger has tree joints: MP, PIP and DIP. MP and PIP joints are driven by tendons and each one has one DoF. MP is actuated by a servomotor XL-320 Dynamixel^{TM}, and DIP joint is dependent on the rotation angle of the MP joint. PIP joint is a flexible link, which is made of elastic material. It has a several number of DoF due to their flexible body.

[1]The hand mechanism includes a smart material, particularly a shape memory alloy wire. We can controller the stiffness of the wire, influencing the grasping task.



(a) Anterior view Hand.

(b) Posterior view Hand.

Figure 5.3: ProMain-II Soft Hand.

5.2.1 Soft robotic finger

The soft link (1B) consist of three elements: (i) supports (10) that have elements (10a) and (10b), (ii) support (20), that have elements (20a) and (20b), (iii) and soft body (30), as shown in Figure 5.4. The elements (10) and (20) are made of rigid material like PolyLactic Acid (PLA), the soft body is made of flexible material like silicone RTV 3535, as illustrated in Figure 5.5. The rigid parts are manufactured employing a 3D Printer. The soft body is fabricated through a mold in two parts.

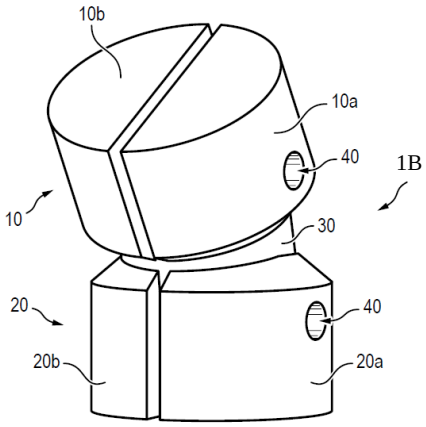


Figure 5.4: Assembling of soft link.

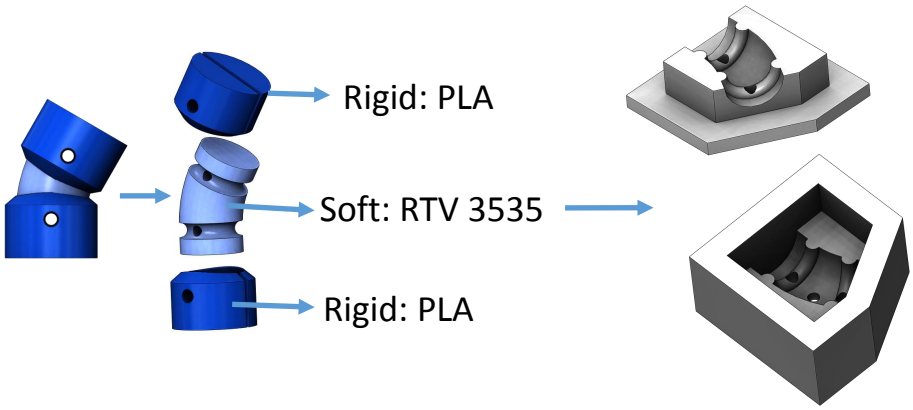


Figure 5.5: Soft joint.

The elements (10a), (10b), (20a) and (20b) have the same shape and size; [2], the detail view is illustrated in Figure 5.6. It has an cavity (11) and a ribbing (12), both elements work together to fix the soft body (30). Furthermore It has a drilling (40) that keeps in place elements (40) and (30). The soft body (30) is detailed in Figure 5.7a. It has three sections (32), (36) and (34). The section (32) is in contact with the section (11) of the elements (10a) and (10b). Similarly element (34) is also in contact with the section (11) of the elements (20a) and (20b). The section (33) is in contact with support (10a) and (10b), likewise the section (35) with supports (20a) and (20b), as shown in Figure 5.7a and 5.7b.

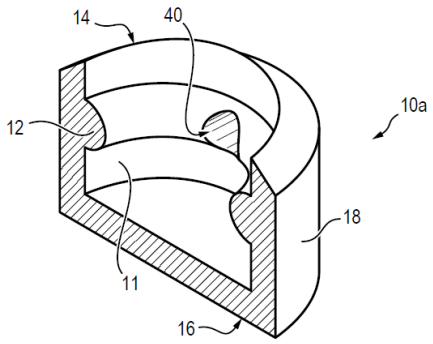
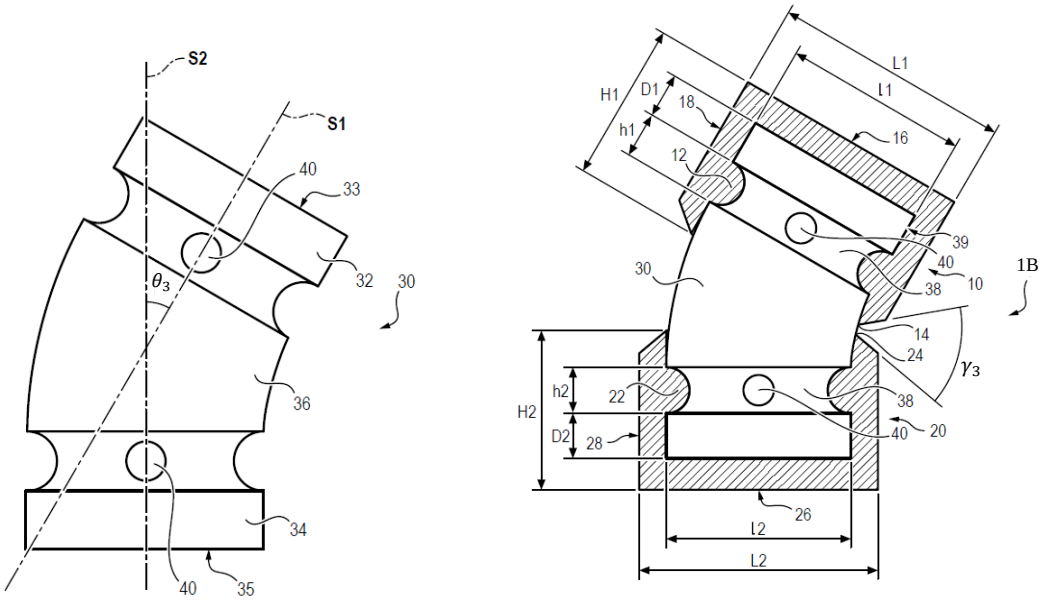


Figure 5.6: Detail view Support(10a).

[2]The elements could have others dimension depend on the specific application



(a) Soft body elements (30).

(b) Assembling soft body (30) and supports (10) and (20).

Figure 5.7: Soft link (1B).

The angle θ_3 is formed by the axes **S1** and **S2**. In addition, the section (36) has two arches, which are influenced by the angle θ_3. The γ_3 is formed by the element (14) and (24) as is illustrated in Figure 5.7b [3]. γ_3 angle limits the torsion of soft body (30), in both direction positive and negative. Positive torsion appears when γ_3 increases its value. It occurs because the elements (10) and (20) are: (i) located opposite each other, (ii) and made of rigid material. Moreover, the performance of the soft link are mainly influence by: (i) soft material selected, and (ii) θ_3 and γ_3 angles.

The soft link is placed in the robotic finger that influences, the manipulation objects are also influence. Thus, the hand compliance level is influences by: (i) the hardness of flexible material, [4] allowing PIP joint additional rotations in all directions, (ii) and the γ_3 and the θ_3 angles values depends on the grasped object.

Experiment: Tracking position

The proposed experiment seeks to measure the position of each joint. Tracking position is similarly carried out as in Section 4.2.3.1. Thus, a platform is designed and manufactured. The platform supports a servomotor and a soft robotic finger. The soft robotic finger has two rigid joints: MP and DIP, and one soft link PIP. The servomotor is controlled through the software implemented using an Arduino Card. The software performs several repetitions of flexion and extension, at the same speed and torque. Moreover, a high-performance 4 megapixel CCD

[3] The geometry is completely described in the patent document [2]

[4] In this case the soft link is made of RTV 3535

camera Prosilica GE-2040 is selected with the aim to track circular markers placed on the finger joints and the fingertip, as shown in Figure 5.8.

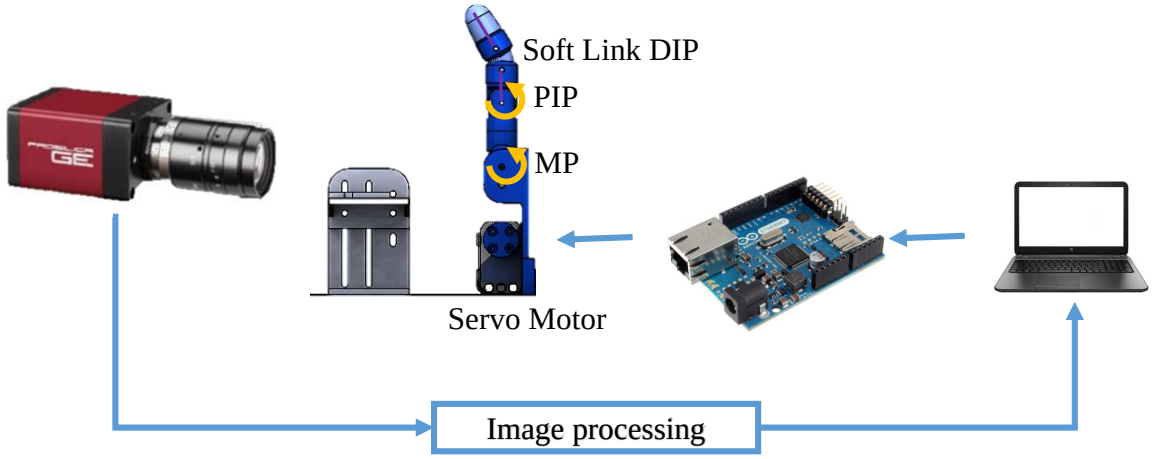
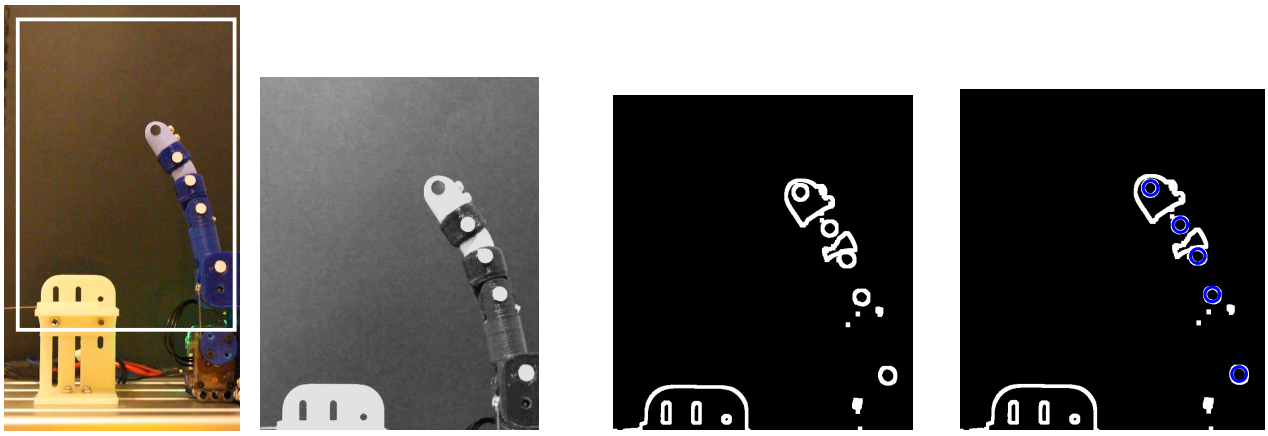


Figure 5.8: Set-up tracking position of the soft robotic finger articulations.



(a) Original. (b) First step. (c) Intermediate step. (d) Final step.

Figure 5.9: Image processing to track links.

Once the desired images are obtained, a methodology is performed to process the images with the aim to recognize markers. The image processing follows these three steps: (i) transforming in gray scale digital images (see Figure 5.9b), (ii) detecting edges (see Figure 5.9c), (iii) applying Hough transform to find marker in the image (see Figure 5.9d).

Then, it is calculated for each finger segment, and consequently, the angle formed by these segments during flexion and extension Figure 5.8 illustrates MP and PIP joints. The α_3 angle that corresponds to the bending articulation (DIP), is described in of Figure 5.7a.

Results

Several cycles of flexion and extension are performed. Figure 5.10 shows the sequence of the flexion. Once the position of each marker is recognized, following three vectors linking joints are defined: (i) vector $\vec{r}_1$ between the MP and PIP joints, (ii) vector $\vec{r}_2$ between the PIP and 20a element of soft DIP joints, (iii) and vector $\vec{r}_3$ between the 10a element of soft DIP joints and fingertip. These vectors are used to calculate rotation angles θ_i as in equation 4.22.

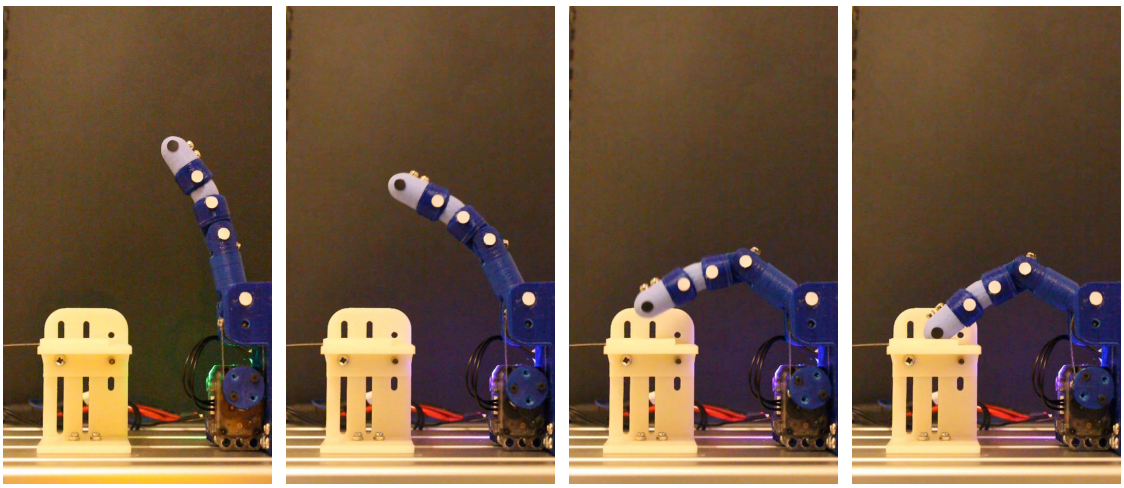


Figure 5.10: Position of the robotic finger articulations during flexion.

The first angle θ_1 is calculated with respect to a reference positive vertical unitary vector $\vec{r}_0 = \{0, 1, 0\}$. As a result, we found that the relations between angle variation in all joints are constant in time. The experiment is repetitive and corresponds to the mechanical design. Furthermore, MP joint motion range is $[18°, 68°]$, PIP joints motion range is from $[2°, 56°]$, DIP soft link is represented as a $link_3 bending(DIP)$ and its range motion is from $[18°, 4°]$, as shown in Figure 5.11.

This results shown that the α of $link_3 bending(DIP)$ decreases when the fingertip gets in contact with the obstacle. As consequence, α angle linking bending depends on the object grasped that is unknown. These results support the idea that the use of a soft link increases the compliance of ProMain-II hand. Moreover the compliant level could be modified considering: (i) selection of the constitutive element of soft body (30); (ii) the soft link geometry, for instance θ_3 and γ.

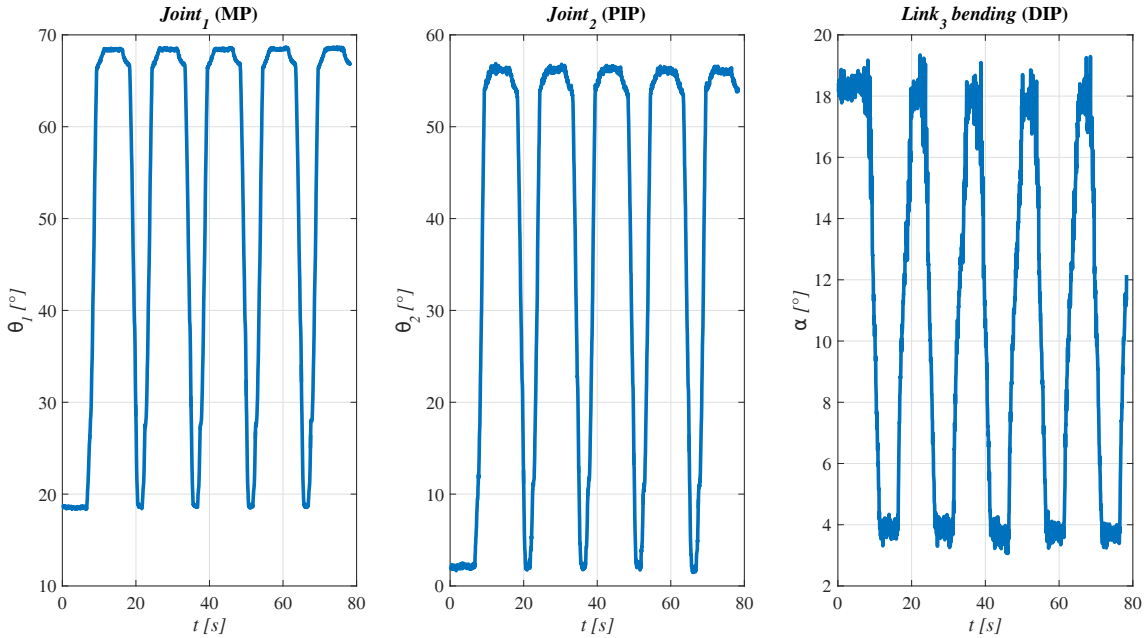


Figure 5.11: Angles performed during flexion/extension soft robotic finger.

5.2.2 Conclusion

We present the ProMain-II hand in which we describe a new soft link. We carried out an experiment for assessing the behavior of the soft link when the finger gets in contact with the obstacle. As a result, we found that the soft link performs bending from $18°$ till $4°$ duringthe experiments. This preliminar analysis allow us to identify: (i) a reduction in the overshoot of the PIP joints[3] when the finger gets in contact with objects, (ii) the finger has a self adaptability according to the object shape, (iii) and we identify a soft behavior represented by the bending of the soft link during contact. As a consequence, the level of compliance of our robotic finger increases using the new soft link.

Chapter 6

Conclusion and Perspectives

6.1 Conclusion

One of the target of the present research is to asses the feature behavior during upper limb motion, we have developed a novel approach that allows to established relation between the feature of sEMG and the kinematic motion. Furthermore, we propose the criterion to chose the best set of features calculating the maximum distance between the nearest values of each couple of feature. We have analyzed the performance of five features mean absolute value, mean value, harmonic, entropy and mean frequency. This study has identified that entropy and the mean frequency features show the best fitness among mean absolute value, mean value, and harmonic mean. Moreover, the maximum feature values are found in the range from 1m/s to 2m/s. In contrast, the features variations at a higher speeds are not significant. It is highly recommended to selected features with a low computational cost which is a relevant consideration in real time applications, such as the control of a prosthesis hand based on sEMG.
We have proposed propose a methodology to identify upper limb motion based on patterns that come from sEMG produced during motion. The produced scheme consists of: (i) extracting features from sEMG produced during motion, (ii) normalizing features, (iii) performing linear or non-linear SVM with σ and C optimized values. As a result the classification scheme has a high accuracy. Moreover, the proposed model can be used out of laboratory conditions with healthy and amputee patients.

Wearable technology are one of the most important advance to human machine interface. We selected a MyoArmband™ device in order to test in real time: (i) the novel model to detect the inception of movements based on sEMG through the entropy and entropy flow measurement, to find automatically the movement inception (ii) and the grasping gesture classification scheme using entropy feature and Support Vector Machine. Furthermore, we developed a new interface human-machine in real time out of laboratory condition which is able to identify with 97.5% accuracy two hand gestures, using a wearable device. The results of this methodology proves that the idea of considering only sEMG signals produced during motion is a right choice.

We present the new ProMain-I Hand. The hand has a new tactile smart sensor which allows

to know the force applied by fingertips. Moreover, we present the characterization of the sensor in different operational conditions, establishing the relation between applied force and measured force. Then, we propose the following models: (i) a parametric model for the ProMain hand servomotor, (ii) a experimental model for the finger position and finger force, (iii) and a hybrid model to control fingers position and fingertips force. As consequence, the synergy among the proposed model allows prosthesis hand to perform grasping gestures in steady way.

Finally, the new Soft hand ProMain-II is introduced describing its characteristics and particularly its new soft link. Moreover, several experiments have been carried out in order to assess the performance of the soft link when it gets in contact with objects. The preliminary tests show that the hand has the habitability to mold itself to the grasped objects.

6.2 Perspectives

Further research could also be conducted to validate the proposed methods in amputated patients even when they have lower level of sEMG produced in the remained muscle, with or without phantom sensation. Moreover, integrating into the present classification scheme other input sensors such as inertial measurements units could be an interesting approach to improve the way in which amputated and healthy patients interact with computers. On the other hand, an interesting approach regarding pattern classification is related to the increment of the number of recognized gesture with an automatic classification process, which leads to the possibility of individual finger movements recognition that is still a challenge.

Additionally, the new soft hand ProMain-II, which is composed of Shape Memory Alloy (SMA) wires tendons, requires the development of a control strategy including the control of the smart materials, particularly SMA. This control enables the tendon stiffness adaptation allowing an improvement in the force and position behavior of the finger. The new control strategy must take into account: (i) position control considering perturbations issued from the stiffness changes, (ii) force control considering perturbations issued from the stiffness changes, (iii) and also a feedback of the stiffness to control wires. This new control strategy will be integrated into the overall system. Additionally, exploring the use other types of smart materials such as IPMC, in soft robotic hands as driven mechanism is an other exciting approach.

Furthermore, we plan to modify the stiffness of the soft link, handling the structure of soft materials through 3D printing techniques. The inner material structure can be modified setting printing parameters, *e.g.* infill patterns, which implies having different stiffness in the same soft link. Consequently, the bending can be restricted in the desired direction or directions. This provides a framework for the exploration of hybrid stiffness bodies in the soft robotics field.

The prototypes of the ProMain prostheses must be evaluated in amputated patients to test the acceptability, the medical and the psychological implications. This study must be developed in an interdisciplinary context and the result are valuable to improve the prosthesis providing and suitable rehabilitation device.

Bibliography

[1] A. Rubiano, J. L. Ramírez, L. Gallimard, O. Polit, and N. Jouandeau. Chaîne articulée comprenant un unique actionneur et ensemble de chaînes articulées associées, FR1656914, Jul. 2016.

[2] A. Rubiano, J. L. Ramírez, L. Gallimard, O. Polit, and N. Jouandeau. Chaîne articulée pour une prothése robotique comprenant une jonction souple, FR1656673, Jun. 2016.

[3] J. L. Ramirez, A. Rubiano, N. Jouandeau, L. Gallimard, and O. Polit. *New Trends in Medical and Service Robots*, volume 39 of *ISSN: 978-3-319-30673-5*, chapter Morphological optimization of prosthesis' finger for precision grasping, pages 1–15. Springer, Nantes, FR, Jul. 2016.

[4] A. Rubiano, J. L. Ramírez, L. Gallimard, O. Polit, and N. Jouandeau. Capteur tactile capable de retranscrire le toucher, FR1655991, Jun. 2016.

[5] J. L. Ramirez, A. Rubiano, N. Jouandeau, Gallimard L., and Polit O. *Requirements for artificial muscles to design robotic fingers*, chapter Requirements for artificial muscles to design robotic fingers. Springer, 2015.

[6] J.L. Ramirez, A. Rubiano, N. Jouandeau, M.N. El Korso, L. Gallimard, and O. Polit. Hybrid kinematic model applied to the under-actuated robotic hand prosthesis ProMain-I and experimental evaluation. In *Proceedings of 14th IEEE/RAS-EMBS International Conference in reabilitation robotics (ICORR)*, pages 301–306, Singapore, SGP, Aug. 2015. IEEE.

[7] A. Rubiano, J. L. Ramirez, N. Jouandeau, M. N. El Korso, L. Gallimard, and O. Polit. Human upper limb motion estimation based on electromyography signals. In *DeMEASS Design, Modelling and Experiments of Advanced Structures and Systems*, Radebeul, Germany, 2015.

[8] A. Rubiano, J. L. Ramirez, M. N. El Korso, N. Jouandeau, P. Le Borgne, S. Longet, L. Gallimard, and O. Polit. Analysis of surface electromyography signals for controlling soft robotic hand prosthesis. In *Journées Nationales de la Recherche en Robotique (JNRR)*, 2015.

[9] J. L. Ramirez, A. Rubiano, L. Gallimard, O. Polit, N. Jouandeau, S. Longet, and P. Le Borgne. Kinematic and dynamic of underactuated soft robotic hand prosthesis. In *Journées Nationales de la Recherche en Robotique (JNRR)*, 2015.

[10] A. Rubiano, J. L. Ramirez, M. N. El Korso, N. Jouandeau, L. Gallimard, and O. Polit. Features selection and classification to estimate elbow movements. In *Journal of Physics: Conference Series*, number 657, ISSN: 1742-6596, pages 1–6. JPCS, 2015.

[11] A. Rubiano, J. L. Ramirez, M. N. El Korso, N. Jouandeau, L. Gallimard, and O. Polit. Elbow flexion and extension identification using surface electromyography signals. In *proceedings of 23th European Signal Processing Conference (EUSIPCO)*, pages 644–648, Nice, Sep 2015.

[12] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau. Emg feature evaluation for improving myoelectric pattern recognition robustness. *Expert Systems with Applications*, 40(12):4832–4840, 2013.

[13] Steeper. *bebionic Technical Manual*. 3619 Paesanos Parkway, Suite 200, San Antonio TX 78231 USA, 2 edition, 2015.

[14] Touch Bionics. *i-limb^{TM}ultra Clinician Manual*. 35 Hampden Road Mansfield MA 02048 USA, 2 edition, Mar 2013.

[15] Ottobock. *The Michelangelo®Hand in Practice*. Max-Näder-Straße 15, 37115 Duderstadt/Germany, 2012.

[16] M. Bonilla, E. Farnioli, C. Piazza, M. Catalano, G. Grioli, M. Garabini, M. Gabiccini, and A. Bicchi. Grasping with Soft Hands. In *proceedings of International Conference on Humanoid Robots 2014 IEEE-RAS*, pages 581–587, Madrid, Spain, Nov 2014. doi: 10.1109/HUMANOIDS.2014.7041421.

[17] R. Deimel and O. Brock. A novel type of compliant, underactuated robotic hand for dexterous grasping. In *proceedings of Robotics: Science and Systems*, pages 1–9, Berkeley, USA, Jul 2014.

[18] S. G. Nurzaman, F. Iida, C. Laschi, A Ishiguro, and R. Wood. Soft robotics [tc spotlight]. *IEEE Robotics Automation Magazine*, 20(3):24–95, Sept 2013. ISSN 1070-9932. doi: 10.1109/MRA.2013.2279342.

[19] M. A. Oskoei and H. Hu. Support vector machine-based classification scheme for myoelectric control applied to upper limb. *IEEE Transactions on Biomedical Engineering,*, 55 (8):1956–1965, 2008.

[20] M. B. I. Reaz, M. S. Hussain, and F. Mohd-Yasin. Techniques of emg signal analysis: detection, processing, classification and applications. *Biological procedures online*, 8(1): 11–35, Mar. 2006.

[21] A. Alkan and M. Günay. Identification of emg signals using discriminant analysis and svm classifier. *Expert Systems with Applications*, 39(1):44–47, 2012.

[22] C. J. De Luca. Surface electromyography: Detection and recording. *DelSys Incorporated*, 10:2011, 2002.

[23] M. Micera, J. Carpaneto, and Raspopovic S. Control of Hand Prostheses Using Peripheral Information. *IEEE Reviews in Biomedical Engineering*, 3:48–68, 2010. ISSN 1937-3333.

[24] E. C. Orosco, N. M. Lopez, and F. di Sciascio. Bispectrum-based features classification for myoelectric control. *Biomedical Signal Processing and Control*, 8(2):153–168, Sep. 2013.

[25] J. Chu, I. Moon, Y. Lee, S. Kim, and M. Mun. A Supervised Feature-Projection-Based Real-Time EMG Pattern Recognition for Multifunction Myoelectric Hand Control. *IEEE/ASME Transactions on Mechatronics*, 12(3):282–290, Jun. 2007.

[26] Ganesh R Naik, Dinesh Kant Kumar, et al. Twin SVM for Gesture Classification Using the Surface Electromyogram. *IEEE Transactions on Information Technology in Biomedicine*, 14(2):301–308, Mar. 2010.

[27] P. Kaufmann, K. Englehart, and M. Platzner. Fluctuating emg signals: Investigating Llong-term Effects of Pattern Matching Algorithms. In *32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 6357–6360. IEEE, Aug. 2010.

[28] K. S. Kim, H. H. Choi, C. S. Moon, and C. W. Mun. Comparison of k-nearest neighbor, quadratic discriminant and linear discriminant analysis in classification of electromyogram signals based on the wrist-motion directions. *Current Applied Physics*, 11(3):740–745, nov 2011.

[29] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau. Feature extraction of the first difference of EMG time series for EMG pattern recognition. *Computer methods and programs in biomedicine*, 117(2):247–256, Jun. 2014.

[30] G. R. Naik, D. K. Kumar, and S. Arjunan. Use of sEMG in identification of low level muscle activities: Features based on ICA and Fractal dimension. In *31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 364–367. IEEE, Sep. 2009.

[31] F. V. G. Tenore, A. Ramos, A. Fahmy, S. Acharya, R. Etienne-Cummings, and N. V. Thakor. Decoding of Individuated Finger Movements Using Surface Electromyography. *IEEE Transactions on Biomedical Engineering*, 56(5):1427–1434, May. 2009.

[32] C. Castellini, E. Gruppioni, A. Davalli, and G. Sandini. Fine detection of grasp force and posture by amputees via surface electromyography. *Journal of Physiology - Paris*, 103(3): 255–262, 2009.

[33] A. H. Al-Timemy, G. Bugmann, J. Escudero, and N. Outram. Classification of Finger Movements for the Dexterous Hand Prosthesis Control with Surface Electromyography. *IEEE Journal of Biomedical and Health Informatics*, 17(3):608–618, May. 2013.

[34] N. Wang, Y. Chen, and X. Zhang. The recognition of multi-finger prehensile postures using LDA. *Biomedical Signal Processing and Control*, 8(6):706 – 712, 2013. ISSN 1746-8094.

[35] H. Kawasaki, M. Kayukawa, H. Sakaeda, and T. Mouri. Learning System for Myoelectric Prosthetic Hand Control by Forearm Amputees. In *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, pages 899–904, Aug. 2014.

[36] M. Zecca, S. Micera, M.C. Carrozza, and P. Dario. Control of multifunctional prosthetic hands by processing the electromyographic signal. *Critical Reviews™ in Biomedical Engineering*, 30(4-6), 2002.

[37] C. L. Pulliam, J. M. Lambrecht, and R. F. Kirsch. User-in-the-loop Continuous and Proportional Control of a Virtual Prosthesis in a Posture Matching Task. In *34th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3557–3559. IEEE, Aug. 2012.

[38] Z. Arief, I. A. Sulistijono, and R. A. Ardiansyah. Comparison of five time series EMG features extractions using Myo Armband. In *2015 International Electronics Symposium (IES)*, pages 11–14, Sep. 2015.

[39] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau. A feasibility study on the use of anthropometric variables to make muscle-computer interface more practical. *Engineering Applications of Artificial Intelligence*, 26(7):1681–1688, 2013.

[40] D. T. Maclsaac, P. A. Parker, R. N. Scott, K. B. Englehart, and C. Duffley. Influences of dynamic factors on myoelectric parameters. *IEEE Engineering in Medicine and Biology Magazine*, 20(6):82–89, Nov. 2001. ISSN 0739-5175. doi: 10.1109/51.982279.

[41] H. Dong-Mei, Y. Yang, and W. Zheng. Measurement system for surface electromyogram and handgrip force based on labview. In *World Congress on Medical Physics and Biomedical Engineering*, pages 67–70. Springer, Sep. 2009.

[42] R. Merletti and P. A. Parker. *Electromyography: physiology, engineering, and non-invasive applications*, volume 11 of *ISBN: 978-0-471-67580-8*. John Wiley & Sons, 2004.

[43] A. Phinyomark, S. Thongpanja, F. Quaine, Y. Laurillau, C. Limsakul, and P. Phukpattaranont. Optimal emg amplitude detectors for musclecomputer interface. In *2013 10th International Conference on Electrical Engineering / Electronics, Computer, Telecommunications and Information Technology (ECTI-CON),*, pages 1–6, May. 2013. doi: 10.1109/ECTICon.2013.6559485.

[44] Y. Velchev, S. Radeva, S. Sokolov, and D. Radev. Automated estimation of human emotion from EEG using statistical features and SVM. In *prodeedings of the 2016 Digital Media Industry Academic Forum (DMIAF)*, pages 40–42, Szntorini, Greece, Jul 2016. doi: 10.1109/DMIAF.2016.7574899.

[45] A. Apatean, A. Rogozan, and A. Bensrhair. Svm-based obstacle classification in visible and infrared images. In *proceedings of the 2009 17th European Signal Processing Conference*, pages 293–297, Glasgow, Scotland, Aug 2009.

[46] P. Staroniewicz and W. Majewski. SVM based text-dependent speaker identification for large set of voices. In *proceedings of the 2004 12th European Signal Processing Conference*, pages 333–336, Vienna, Austria, Sept 2004.

[47] M. Yoshikawa, M. Mikawa, and K. Tanaka. A myoelectric Interface for Robotic Hand Control Using Support Vector Machine. In *proccedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2723–2728, San Diego, USA, Oct 2007.

[48] M. Yoshikawa, Y. Taguchi, N. Kawashima, Y. Matsumoto, and T. Ogasawara. Hand Motion Recognition Using Hybrid Sensors Consisting of EMG Sensors and Optical Distance Sensors. In *proceedings 2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*, pages 144–149, Paris, France, Sept 2012.

[49] M. Tavakolan, Z. G. Xiao, J. Webb, and C. Menon. Emg processing for classification of hand gestures and regression of wrist torque. In *proceedings of 2012 4th IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 1770–1775, Roma, Italy, Jun 2012. doi: 10.1109/BioRob.2012.6290677.

[50] D. Yang, J. Zhao, Y. Gu, L. Jiang, and H. Liu. Emg Pattern Recognition and Grasping Force Estimation: Improvement to the Myocontrol of Multi-dof Prosthetic Hands. In *proceeding of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 516–521. St. Louis, USA, Oct 2009.

[51] M. A. Soltanmoradi, V. Azimirad, and M. Hajibabazadeh. Detecting finger movement through classification of electromyography signals for use in control of robots. In *preceedings of the Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), 2014*, pages 791–794, Tehran, Iran, Oct 2014.

[52] Y. Cai, J. Shi, J. Zhong, F. Wang, and Y. Hu. Characterization of Surface EMG with Cumulative Residual Eentropy. In *proceedings of the IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC)*, pages 55–58, Hong Kong, China, Aug 2012.

[53] S. J. Russell and P. Norvig. *Artificial intelligence: a modern approach*. ISSN: 978-0-13-604259-4. Prentice hall, 2010.

[54] O. Bousquet, U. von Luxburg, and G. Rätsch. *Advanced Lectures on Machine Learning*. ISBN: 978-3-540-23122-6. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[55] E. Yom-Tov. *An Introduction to Pattern Classification*, chapter 1, pages 1–20. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003. ISBN 978-3-540-28650-9.

[56] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, Dec 1943. ISSN 1522-9602.

[57] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, Feb 1995. ISSN 1573-0565.

[58] B. E. Boser, I. M. Guyon, and V. N Vapnik. A training algorithm for optimal margin classifiers. In *proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152, New York, USA, 1992.

[59] O. Güler. On the convergence of the proximal point algorithm for convex minimization. *SIAM Journal on Control and Optimization*, 29(2):403–419, Mar 1991.

[60] Thalmic Labs. Gesture control armband, Mar 2016. URL https://www.myo.com/.

[61] G. Pomboza-Junez and J. A. Holgado-Terriza. Control of home devices based on hand gestures. In *Consumer Electronics - Berlin (ICCE-Berlin), 2015 IEEE 5th International Conference on*, pages 510–514, Sep. 2015.

[62] C. Munroe, Y. Meng, H. Yanco, and M. Begum. Augmented reality eyeglasses for promoting home-based rehabilitation for children with cerebral palsy. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 565–565, Mar. 2016.

[63] I. Phelan, M. Arden, C. Garcia, and C. Roast. Exploring virtual reality and prosthetic training. In *IEEE Virtual Reality (VR) 2015*, pages 353–354, Mar. 2015.

[64] M. Sathiyanarayanan and S. Rajan. MYO armband for physiotherapy healthcare: A case study using gesture recognition application. In *8th International Conference on Communication Systems and Networks (COMSNETS2016)*, pages 1–6, Jan. 2016.

[65] R. Tapu, B. Mocanu, and E. Tapu. A survey on wearable devices used to assist the visual impaired user navigation in outdoor environments. In *2014 11th International Symposium on Electronics and Telecommunications (ISETC)*, pages 1–4, Nov. 2014.

[66] O. D. Lara and M. A. Labrador. A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys Tutorials*, 15(3):1192–1209, Third quarter 2013. ISSN 1553-877X.

[67] Fitbit flex ™. *Wireles Activity + Sleep Wristband*. Fitbit ®, 2016.

[68] K. M. Diaz, D. J. Krupka, M. J. Chang, J. Peacock, Y. Ma, J. Goldsmith, J. E. Schwartz, and K. W. Davidson. Fitbit ®: An accurate and reliable device for wireless physical activity tracking. *International Journal of Cardiology*, 185:138–140, Mar. 2015.

[69] T. Feix, I. M. Bullock, and A. M. Dollar. Analysis of Human Grasping Behavior: Correlating Tasks, Objects and Grasps. *IEEE Transactions on Haptics*, 7(4):430–441, October 2014. ISSN 1939-1412.

[70] C. J. De Luca. Physiology and mathematics of myoelectric signals. *IEEE Transactions on Biomedical Engineering*, BME-26(6):313–325, jun 1979. ISSN 0018-9294. doi: 10.1109/TBME.1979.326534.

[71] S. H. Nawab, S. Chang, and C. J. De Luca. High-yield decomposition of surface EMG signals. *Clinical Neurophysiology*, 121(10):1602–1615, 2010. ISSN 1388-2457.

[72] T. M. Cover and J. A. Thomas. *Elements of information theory*. ISBN: 13 978-0-471-24195-9. John Wiley & Sons, 2 edition, 2006.

[73] J.L. Ramirez, A. Rubiano, N. Jouandeau, L. Gallimard, and O. Polit. *Smart Structures and Materials*, chapter Artificial muscles design methodology applied to robotic fingers, pages 1–17. Springer, 2015.

[74] Lionel Birglen, Thierry Laliberté, and Clément Gosselin. *Design and Control of the Laval Underactuated Hands*, pages 171–207. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. ISBN 978-3-540-77459-4.

[75] S. Wakimoto, K. Ogura, K. Suzumori, and Y. Nishioka. Miniature soft hand with curling rubber pneumatic actuators. In *proceedings of IEEE International Conference on Robotics and Automation (ICRA '09)*, pages 556–561, Kobe, Japan, May 2009.

[76] H. Mnyusiwalla, P. Vulliez, J. P. Gazeau, and S. Zeghloul. A new dexterous hand based on bio-inspired finger design for inside-hand manipulation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(6):809–817, jun 2016. ISSN 2168-2216. doi: 10.1109/TSMC.2015.2468678.

[77] A. Ajoudani, S. B. Godfrey, M. G. Catalano, M. Bianchi, G. Grioli, N. G. Tsagarakis, and A. Bicchi. Exploring Teleimpedance and Tactile Feedback for intuitive Control of the Pisa/IIT SoftHand. *IEEE Transactions on Haptics*, 7:203 – 215, 2014.

[78] S.B. Godfrey, A. Ajoudani, M. Catalano, G. Grioli, and A. Bicchi. A synergy-driven approach to a myoelectric hand. In *Rehabilitation Robotics (ICORR), 2013 IEEE International Conference on*, pages 1–6. IEEE, 2013.

[79] R. Vinjamuri, M. Sun, C.C. Chang, H. N. Lee, R. J. Sclabassi, and Z. H. Mao. Temporal postural synergies of the hand in rapid grasping tasks. *IEEE Transactions on Information Technology in Biomedicine*, 14(4):986–994, July 2010. ISSN 1089-7771.