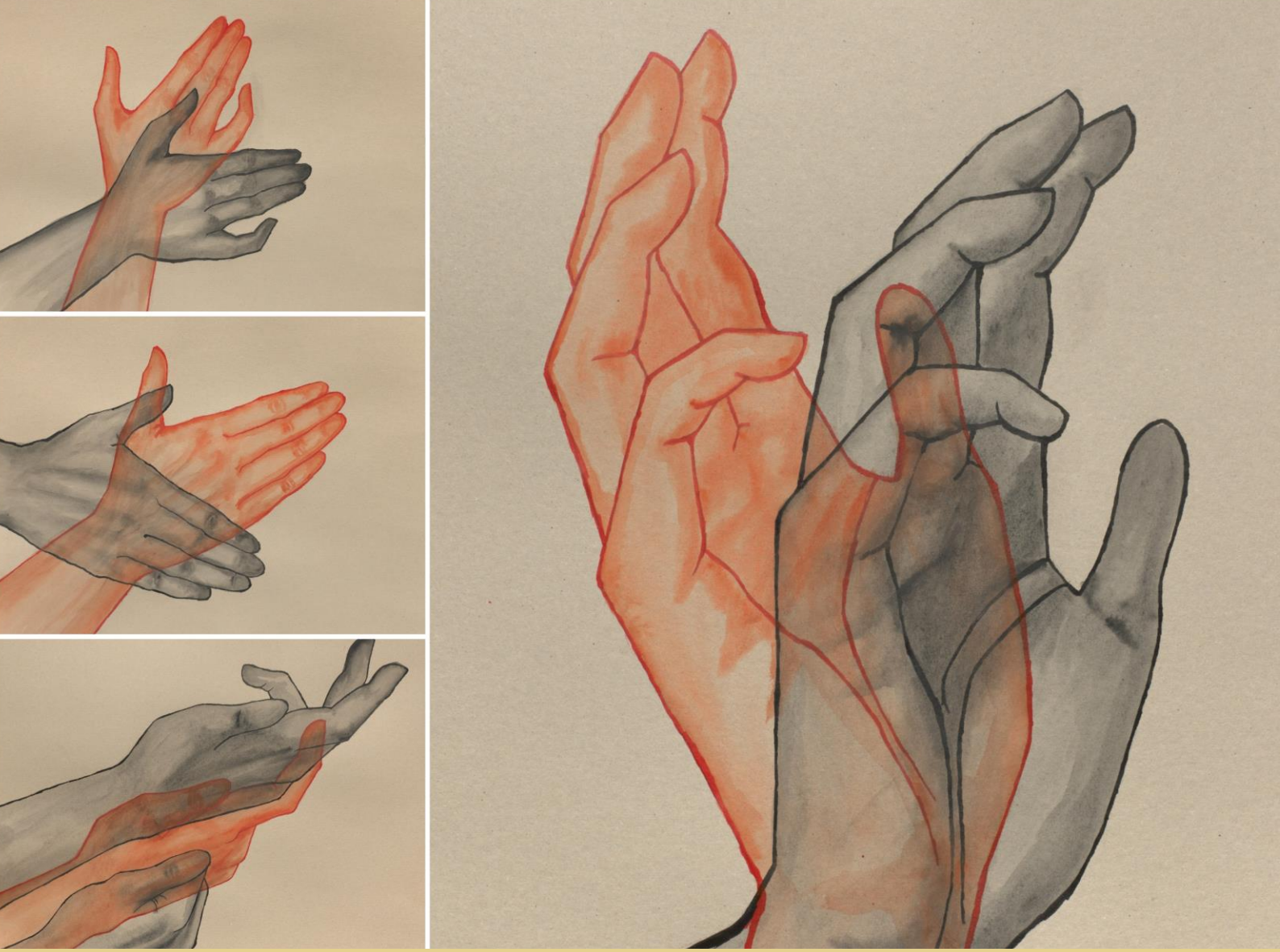

Ana Cláudia Fernandes dos Reis

Extraction of Sign Language Information through Biosignals

Dissertação submetida para a satisfação parcial dos requisitos do grau de Mestre em Engenharia Electrotécnica e de Computadores

Julho, 2017

· U C ·

Universidade de Coimbra

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Extraction of Sign Language Information through Biosignals

Ana Cláudia Fernandes dos Reis

Coimbra, July 2017

Extraction of Sign Language Information through Biosignals

Supervisor:

Prof. Rui Pedro Duarte Cortesão

Jury:

Prof. Fernando Manuel dos Santos Perdigão

Prof. Paulo José Monteiro Peixoto

Dissertation submitted in partial fulfillment for the degree of Master of Science in Electrical and Computer Engineering.

Coimbra, July 2017

Acknowledgements

Em primeiro lugar, gostaria de agradecer ao meu orientador, Professor Rui Pedro Duarte Cortesão, por todo o apoio e ajuda prestados durante o desenvolvimento da minha tese. O seu encorajamento e feedback foram essenciais para a finalização do meu trabalho.

Agradeço também aos meus amigos, Ana Teresa e Carlos, por estarem sempre presentes quando precisava de falar e por serem um grande apoio ao longo destes anos na faculdade, São amigos verdadeiros e sei vão estar na minha vida para sempre. Além disso, não posso deixar de agradecer aos resistentes que perduram do secundário, Teresa, Mariana e Ana José, que tenhamos sempre tempo para os nossos cafés. Não só, queria também agradecer às "cobaias" que disponibilizaram os seus braços para me ajudar, obrigada Hélder Marques e Ricardo Mendes pela vossa paciência.

Quero ainda agradecer aos meus pais, pelo seu apoio incondicional sem o qual nada disto seria possível. Obrigada por acreditarem em mim e me incentivarem ao longo da minha vida, vocês fizeram de mim a pessoa que sou hoje.

Por fim, gostaria de agradecer ao Zé pelo seu apoio e carinho incondicional. Tiveste sempre uma palavra de encorajamento para mim, e apoiaste-me nos meus piores momentos, sempre com um sorriso no rosto. Espero que saibas o quão especial és para mim, e que enquanto quiseres vou estar sempre aqui para ti.

"I hear you say 'Why?' Always 'Why?' You see things and you say 'Why?' But I dream things that never were and I say 'Why not?'"

— George Bernard Shaw, *Back to Methuselah*

Abstract

In recent years, a significant amount of effort has been dedicated to the field of human-computer interaction, particularly regarding the topic of gesture recognition. These developments aim to improve the quality of life of elderly and disabled people, through the creation of a new communication interface between humans and computers. More precisely, these researches explore the electric nature of the human nervous system and attempt to apply it to the control of human-assisting devices. As such, the goal of these systems is to translate hand and arm motions into commands, through the detection, processing and classification of biosignals, namely Electromyography signals (EMG).

This work focuses on Sign Language Recognition (SLR), an intricate and challenging problem which has yielded different approaches in the past. This is due to the inherent complexity of these types of gestures, which can either be static or dynamic, when more than one stage of motion is required. Thus, there are different classification method suitable for these types of signals such as Hidden Markov Models (HMMs), Finite State Machines (FSM), particle filtering and Artificial Neural Networks (ANNs). However, while the use of ANNs is very common in the classification of simple arm/hand movements it's use in sign language recognition is still being explored. In fact, most efforts at SLR are employed with HMMs due to the fact that they are more effective at recognizing signals which vary over time.

Therefore, this study presents a step-by-step development of an EMG pattern recognition system, based on the classification of a discriminatory set of features by an ANN. Moreover, it evaluates the performance of the aforementioned system by applying it to the classification of 10 sign language gestures. As such, this document includes an analysis on the optimal parameters of the network's architecture as well as a study on the influence of the sampling population and the representation of the input data on the system's performance.

Moreover, the obtained classification results support the efficacy of an artificial neural network as a classifying tool for complex sign language gestures. The developed system can classify 10 sign language gestures, from 4 subjects, with a classification accuracy of 95.4%. Not only, this system is also suitable for the classification of simple movements, achieving an accuracy rate of 92.7% for the classification of 10 hand and arm movements, performed by 4 subjects.

Keywords

Sign language recognition, Gesture recognition, Electromyographic signals, Biosignals, Artificial neural network

Resumo

Recentemente, o campo de interação humana-computador tem sido o foco de bastante interesse, particularmente o tópico de reconhecimento gestual. Estas pesquisas procuram melhorar a qualidade de vida de idosos e indivíduos com necessidades especiais, através da criação de uma nova interface de comunicação entre humanos e computadores. Mais especificamente, estes estudos exploram a natureza elétrica do sistema nervoso humano, procurando aplicá-lo ao controlo de aparelhos assistivos. Deste modo, o objetivo destes sistemas passa pela tradução de movimentos simples em comandos através da deteção, processamento e classificação de bio-sinais, nomeadamente sinais electromiográficos (EMG).

Este trabalho foca-se no reconhecimento de língua gestual (SLR), um problema desafiante e complexo com várias abordagens de resolução. Esta multiplicidade de abordagens é causada pela complexidade inerente a estes tipos de gestos, que podem ser classificados como gestos estáticos ou como gestos dinâmicos, onde mais do que uma fase de movimento é diferenciável. Assim, há diferentes tipos de métodos de classificação que podem ser aplicados a estes tipos de sinais, nomeadamente "Hidden Markov Models" (HMMs), máquinas de estado finito, filtragem de partículas e redes neuronais artificiais. Contudo, apesar do uso de redes neuronais artificiais ser bastante comum na classificação de movimentos simples, o seu uso no reconhecimento de língua gestual ainda está a ser explorado. De facto, a maioria dos esforços em SLR é empregue com o uso de HMMs, dada a sua eficácia no reconhecimento de sinais que variam ao longo do tempo.

Esta dissertação tem como objetivo o desenvolvimento de um sistema de reconhecimento de língua gestual, através da medição de sinais EMG. Deste modo, a metodologia proposta baseia-se na extração de um conjunto discriminatório de características ("features") e na sua classificação através de uma rede neuronal artificial.

O método desenvolvido foi aplicado á classificação de 10 sinais gestuais, de maneira a determinar os parâmetros ótimos da arquitetura da rede neuronal. Deste modo, o presente documento inclui um estudo sobre os parâmetros do modelo escolhido e os resultados de classificação correspondentes. Não só, apresenta uma análise sobre a influência de certos fatores na performance do sistema, nomeadamente o tamanho da população de amostra e o conjunto de "features" utilizado na representação.

Por sua vez, os resultados obtidos apoiam o uso de uma rede neuronal artificial como ferramenta de classificação, na área de reconhecimento de língua gestual. Neste caso, o sistema desenvolvido classifica 10 sinais gestuais, provenientes de 4 sujeitos, com uma taxa de precisão de 95.4%. Não só, o sistema também é adequado para a classificação de movimentos simples, alcançando uma precisão de 92.7% na classificação de 10 movimentos efetuados por 4 sujeitos.

Palavras Chave

Reconhecimento de língua gestual, Reconhecimento de gestos, Sinais electromiográficos, Biosinais, Rede neuronal artifical

Contents

List of Figures

List of Tables

List of Acronyms

SLR	Sign Language Recognition
LGP	Língua Gestual Portuguesa
ASL	American Sign Language
EMG	Electromyography
EEG	Electroencephalography
EOG	Electrooculography
ECG	Electrocardiography
ENG	Electroneurography
sEMG	Surface Electromyography
ANN	Artificial Neural Network
HMM	Hidden Markov Model
FSM	Finite State Machine
AR	Autoregressive
IAV	Integral Absolute Value
MAV	Mean Absolute Value
WL	Waveform Length
RMS	Root Mean Square
SSC	Slope Sign Changes
ZC	Zero Crossings

CNS	Central Nervous System
PNS	Peripheral Nervous System
ANS	Autonomic Nervous System
ENS	Enteric Nervous System
ATP	Adenosine Triphosphate
MUAP	Motor Unit Action Potential
TKEO	Teager-Kaiser Energy Operator
SNR	Signal-to-noise Ratio
PCA	Principal Component Analysis
BP	Backpropagation
GN	Gauss-Newton
LM	Levenberg-Marquardt
ACC	Accelerometer

CHAPTER 1

Introduction

Contents

In the last decade, there have been tremendous developments, made both in technology and science, which have led to the proliferation of computerized machines in society. Consequently, the subsequent need to interact with communication and display technologies has sparked an interest in intuitive interfaces that can recognize the user's body movements and translate them into commands. Gesture recognition can be defined as "the recognition of meaningful expressions of motion, involving hands, arms, face, and/or body" [6]. This is a natural and convenient way of interacting, which allows the development of an intelligent and efficient human-computer interface. Gestures can be static, in which the user assumes a certain configuration that does not vary in time, or dynamic when it is possible to perceive different stages of motion. Moreover, gestures are often language and culture specific and therefore ambiguous and incompletely defined [7].

This work focuses on hand gestures, which play an important role in non-verbal communication and have wide-ranging applications such as rehabilitation engineering, gesture based control and Sign Language Recognition (SLR). Furthermore, it centers on the development of a simple sign to speech system that can be used later on to improve the quality of life of deaf or non-vocal persons.

1.1 Motivation

Hand gesture identification is an intricate problem, where a large number of muscles is involved even for a simple hand movement. Therefore, there are different approaches to handle this problem, ranging from statistical modeling, computer vision, image processing and soft-computing [6].

The most common approach to SLR is statistical modeling, specifically Hidden Markov Models (HMMs), which are effective in the recognition of signals that change over time [8]. This variation derives from the inability to exactly replicate a gesture, due to inter-individual differences and subject specific differences, when the same subject performs a gesture differently in separate instances [9]. Nonetheless a soft-computing approach through Artificial Neural Networks (ANNs) can also be used in gesture recognition, especially for large data sets. This approach is commonly used in gesture based control, where the classification regards simple hand and arm movements. As such, its efficiency in the classification of complex sign language gestures is still debatable when compared to other methods such as HMMs [10] [9].

Additionally, there are different sensing technologies that can be used to capture gestures, each one with its own accuracy, resolution and range of motion. Generally, methods of extracting data in SLR can be divided into two groups: sensing devices, where sensors are attached to the user's arm, and computer vision-techniques which don't interfere with the user [6]. Although computer-vision approaches offer the advantage of no interference, they have the downside of being extremely sensitive to the testing environment, more precisely to its lighting, color and background texture [10]. Hence, for this research, data was obtained through Surface Electromyography (sEMG), a technique which measures the electrical current generated by the contraction of muscles and directly represents neuromuscular activity.

1.2 Objectives

The goal of this thesis is to develop a pattern recognition system capable of classifying 10 sign language gestures, with the use of an ANN. Moreover, this research aims to determine the optimal conditions, for which the classification accuracy is maximized, through a study of the network's architecture and the representation of the input data. Also, it focuses on the relation between classification accuracy and data inter-variability, particularly the effect of increasing the sampling population.

1.3 Dissertation Outline

This thesis is organized in 5 chapters, starting with an introductory chapter, where the reader is presented with the aim of the dissertation and a brief review on SLR. The 2nd chapter corresponds to the state of the art, which provides background information regarding sign language, surface electromyography and ANNs.

The following chapter details the methodology of the pattern recognition system, providing information about each element of the aforementioned system. Moreover, section 4 presents the experimental results of the classification of sign language gestures and simples arm/hand movements. Finally, the conclusion of this research is presented on chapter 5 along with some suggestions of improvement for future work.

CHAPTER 2

State of the Art

Contents

2.1 Sign Language and Sign Language Recognition

Sign language is a complex visual language that employs gestures to convey words as an alternative to acoustic sound, commonly used by the hearing impaired community. As with any language, it is composed of words, which follow specific grammatical rules, and are combined to form sentences [7]. Moreover, sign language has three major components: world-level sign vocabulary, finger-spelling and non-manual features which consist of facial expression and body positioning [11].

To the contrary of normal belief, sign language is not universal, each country has its specific sign language with its own roots and history. In Portugal, the majority of the deaf community uses the Portuguese sign language, known as “Língua Gestual Portuguesa" (LGP), whose origins date back to 1823 [12]. Each sign of the LGP can be described by 5 main parameters such as: hand configuration (symbolic information), articulation location (spatial information), hand movement (phatic information), hand orientation and non-manual components (body/facial expression) [13].

Given the importance of sign language in today’s society, there is an ever-growing need for a practical device that can facilitate this type of communication. Therefore, this research aims to develop a robust pattern-recognition system that can accurately identify the sign conveyed by a subject’s hand. In this case, data was extracted through sEMG, a method which has recently gained popularity in the SLR research area [14].

As previously mentioned, data extracting methods can be based on either computer vision techniques or data sensing devices. Computer based recognition systems capture gestures through a fixed set of cameras, which makes this technology very sensible to the properties of the external environment, namely its background texture, color and lighting [8]. Another problem associated with these types of system is the occlusion of parts of the user’s body, particularly the hands as this can lead to missing gestures during tracking [10].

On the other hand, computer based methods do not restrict or interfere with the user’s motion, thereby allowing a more natural interaction with the system, as opposing to sensing devices, which require the connection of cables or the use of a tracking device. Moreover, sensing devices encompass different devices such as magnetic field trackers, data gloves, body suits and bioelectric signals [6].

This research focuses on the bioelectric field, where signals are created by an alteration of electric current, due to a difference of potential across a specialized tissue, organ or cell [14]. Furthermore, these signals can be classified according to the source of measurement: Electroencephalography (EEG), Electrooculography (EOG), Electrocardiography (ECG), Electromyography (EMG) and Electroneurography (ENG) [15]. In the context of the SLR problem, the EMG stands out as an ideal method, since it measures the electrical activity induced by the user's arm/hand muscle thereby allowing the interpretation of the movement. In addition to that, EMG signals also offer better signal to noise ratio, in comparison to other signals such as the EEG or the ENG [14].

Subsequently, the popularity of this method sparked an abundance of studies on the subject of classifying EMG signals. The earliest research dates back from 1975, when Graupe and Cline [6] classified EMG signals to obtain command signals for prosthetic control. This was pioneer work for the time, having achieved 85% on the classification of data from a single channel, using Autoregressive (AR) coefficients as features.
However, this system was not able to identify complex gestures, a goal achieved in 1993 by William Putnam et. al [16]. Their research developed a real time system for pattern recognition of EMG signals, capable of controlling a graphical computing environment, based on AR coefficients. The system used neural networks to identify two gestures, bicep flexion and bicep extension, in order to move a slider in different directions (upward or downward) and achieved a 95% accuracy in classification.

In addition, another important milestone occurred in 2006, when G. Tsenov et al. discovered that the classification performance of hand/finger movements depends considerably on feature extraction [17]. In their research they concluded that identification methods cannot help the accuracy classification if the selected features are not relevant. Therefore, the determination of a complete set of discriminatory features is of great importance to the performance of pattern-recognition systems. Moreover, they were able to obtain 98% accuracy with a Multi-layer Perceptron and the use of the following time-domain features: Mean Absolute Value (MAV), Variance, Waveform Length (WL), Norm, Number of Zero Crossings (ZC), Absolute Maximum, Absolute Minimum, Maximum minus Minimum and Median Value.

Since then, much of the recent research has focused on increasing the number of gestures whilst improving the classification accuracy, by designing new ways of feature representation and by applying different gesture classifying techniques, which will be briefly described in the following paragraphs.

The most common classification methods in SLR are HMMs, Finite State Machines (FSM), particle filtering and ANNs [6] [10]. Firstly, in the FSM approach, gestures are viewed as a set of trajectories, each one corresponding to a group of points in 2-D space. Therefore, a gesture is modeled as an ordered sequence of states, where each state has a set of parameters that specify the spatial-temporal information captured by it [18].

Particle filtering is directed towards real-time estimation of nonlinear non-Gaussian systems, through the approximation of probability densities by weighted samples [6]. This system uses particles, random variables sampled directly from the state space, to represent posterior probability. These filters can also be viewed as a sample-based variant of Bayes filters, as the particle system is propagated recursively according to the Bayesian rule [19].

Another well recognized method is the HMM, which can be defined as "a statistical model capable of modeling spatial-temporal time series" [10]. More precisely, an HMM model is a collection of finite states ruled by transitions, whose outputs depend on the probability distribution of the associated state [20]. Therefore, there are 2 sets of probabilities: the transition probability which gives the probability of undergoing the transition and the output probability, which defines the probability of emitting an output symbol [6]. In this case, HMM based recognition uses multidimensional HHMs to represent a set of gestures, and each trained model is used to evaluate new incoming gestures after the proper training.

Finally, ANNs are a specific type of network, characterized by its ability to learn in a similar way to working biological systems. Thus, an artificial neuron can be seen as computational model inspired by natural neurons, which connects to other computational units forming a complex system with powerful processing capabilities [21]. These systems are example-based learners, that do not require explicit programming and can generalize and associate data [1]. Consequently, they can find acceptable solutions for problems for which they were not explicitly trained which makes them robust to noisy input data. This was the method of choice for classifying data, due to its advantage of learning relationships directly from measured data and capability of generalization.

2.2 Neurons and the EMG acquisition system

This chapter explores the characteristics of electromyography, a common technique in the area of medical robotics, used mainly for controlling prostheses and rehabilitation applications [8]. As stated by J.V Basmajian in his book "Muscles Alive", electromyography can be defined as "an experimental technique concerned with the development, recording

and analysis of myoelectric signals", where the term myoelectric refers to signals caused by physiological variations in muscle fiber membranes [22] More precisely, this technique uses electrodes to measure electrical potentials generated by muscle cells as a byproduct of contraction, which is generated due to the reception of signals.
These measuring devices can be classified as surface electrodes, which offer the advantage of convenience and simplicity during application, or wire/needle electrodes which although being intrusive are the preferred choice when analyzing deeper muscles that cannot be measured superficially [2].

In order to understand the process by which EMG signals are measured, it is necessary to review some principles regarding the human nervous system and the way movements are produced. The nervous system is the body's processing unit, responsible for coordinating movements through the transmission of signals. This system can be divided in two regions: the Central Nervous System (CNS), where information is stored and processed by the brain and spinal cord, and the Peripheral Nervous System (PNS). The PNS is a branched and complex network comprised by the peripheral nerves, the Autonomic Nervous System (ANS), related to the innervation of the blood vessels and internal organs, and the Enteric Nervous System (ENS) which controls the activity of the gut [23].

Moreover, the spinal cord gives origin to a pair of spinal nerves formed by the fusion of nervous segments known as the dorsal and ventral roots. To clarify, the term nerve refers to an enclosed, cable-like bundle of nerve fibers, whereas each nerve fiber consists of an axon and the myelin sheath that encloses it. In addition, there are 31 pairs of spinal nerves, each one made from the combination of nerve fibers which can either be afferent nerve fibers, that carry information from sensory organs to the spinal cord, or efferent nerve fibers, which carry motor information from the spinal cord to muscles and effectors [23].

As a whole, the nervous system is composed by billions of cells, which communicate continuously in a coordinated fashion, yielding enormous processing capacity. These processing units, named neurons, are cells found in nerves that behave in a similar way to a switch, that when in presence of enough stimuli from other neurons outputs a pulse. The cell's nucleus, or soma, is in charge of the accumulation of activating and inhibiting signals, received through the dendrites, branched projections of the soma. Conversely, the axon is responsible for conducting an output impulse from the neuron's cell body, to other neurons, muscles or glands. (Figure 2.1) [14] .

As previously mentioned, the soma accumulates the signals it receives until a certain threshold is exceeded, which activates the nucleus and triggers the transmission of an elec-

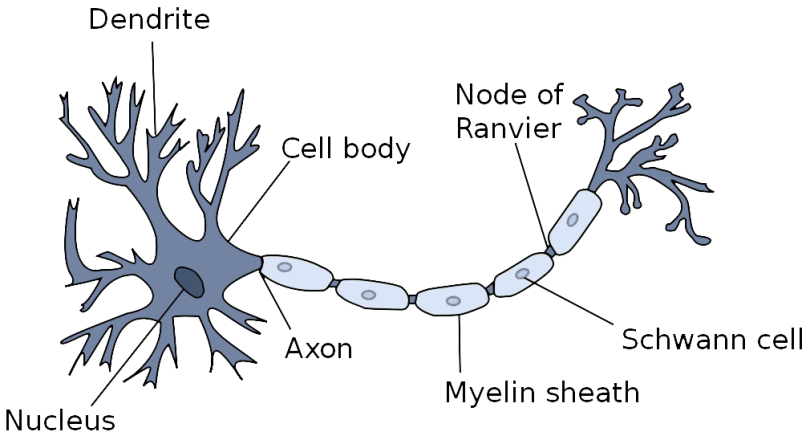

Figure 2.1: Illustration of a biological neuron, from the book "A Brief Introduction to Neural Networks", chapter 2, page 17 [1].

trical pulse to other cells (action potential). Moreover, it is important to mention that above this threshold, each nerve impulse has the same amplitude and duration, which is independent from the intensity of stimulus. This phenomenon is referred to as the law of "all-or-nothing", and states that impulses have a standard duration of 0.5 to 1ms, in the case of mammals [23].

The reception of these signals occurs through the synapses, structures designed for inter-neuron communication, between the axon terminals of a presynaptic cell and the dendrites of postsynaptic cell. In addition, these synapses can be electrical, where neurons are physically connected to one another via gap junctions, or chemical where neurotransmitters are released into a synaptic cleft which separates the transmitter and receiver sides [1]. This communication has a great impact on the neuron's membrane and it leads to a set of phenomenons which generate the EMG signal. Therefore, it is interesting to analyze the behavior of the membrane during a resting period, when no signals are received.

In the stationary phase, each neuron has a difference in electrical charge compared to the environment, called membrane potential. This potential is created by different concentrations of specific ions within and outside the neuron, as shown by Figure 2.2. In this case, the inside of the neuron has a high concentration of negative ions (collectively represented by A^-) and positive potassium ions (K^+). Therefore, the concentration gradient leads to the diffusion of K^+ to the outside of the membrane, whereas the A^- ions remain unmoved due to the impermeability of the membrane to this specific type of ions. Consequently, the inside of the neuron becomes negatively charged, creating an electrical gradient that attracts positive K^+ and sodium (Na^+) ions. It is important to mention that while the concentration

of Na^+ is higher on the outside, the cell's membrane is not permeable to this type of ion and it pours very slowly into the cell.

In addition to this mechanism, generated by the permeability of the membrane to specific ions, ATP proteins transport ions against the direction they want to take. Therefore, Na^+ is pumped out of the cell and K^+ is pumped back in, hence the name "sodium-potassium pump". This pump maintains the concentration gradient for the sodium and potassium, creating a steady state equilibrium and a resting potential of -70mv [1].

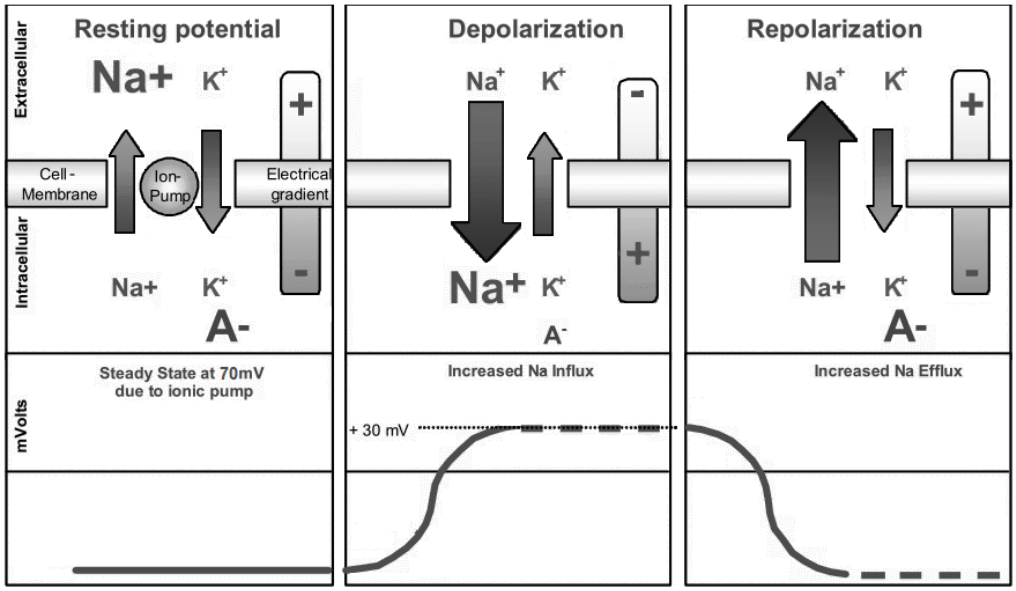

Figure 2.2: Ions and their effect on the membrane's potential, adapted from the book "The ABC of EMG, A Practical Introduction to Kinesiological Electromyography", page 17 [2].

However, when a stimulus reaches a cell, it opens channels which lead to the diffusion of Na^+ to the inside of the cell. Consequently, this slowly increases the intracellular charge until the threshold of -55mv is exceeded and the action potential is initiated. Next, depolarization starts and more potassium channels open, which quickly increases the membrane's potential from -55 to +30 mv, generating an electrical pulse. The following stage is Repolarization, sodium channels close and potassium channels open, and positive ions diffuse to the outside of the membrane, making the interior of the cell negative. Lastly, during Hyper-polarization sodium and potassium channels close, and the membrane is slightly more negative than the resting potential due to the fact that potassium channels close more slowly. After a refractory period, during which the neuron cannot react to new stimuli, the resting state is reestablished [1] [2].

As shown by Figure 2.3, the polarization cycles come together to form an electrical dipole, which defines the EMG signal. This signal is used to detect neuromuscular activations, triggered by certain movements and postures, through the measurement of action potentials of muscle fiber membranes. This detection requires a pair of bipolar electrodes, which measure two potentials in the muscle tissue each with respect to a reference electrode, positioned on an electrically neutral area. After this process, the two signals are fed to a differential amplifier which amplifies the difference of the two signals, canceling out external interferences, which are equal in phase and amplitude [2].

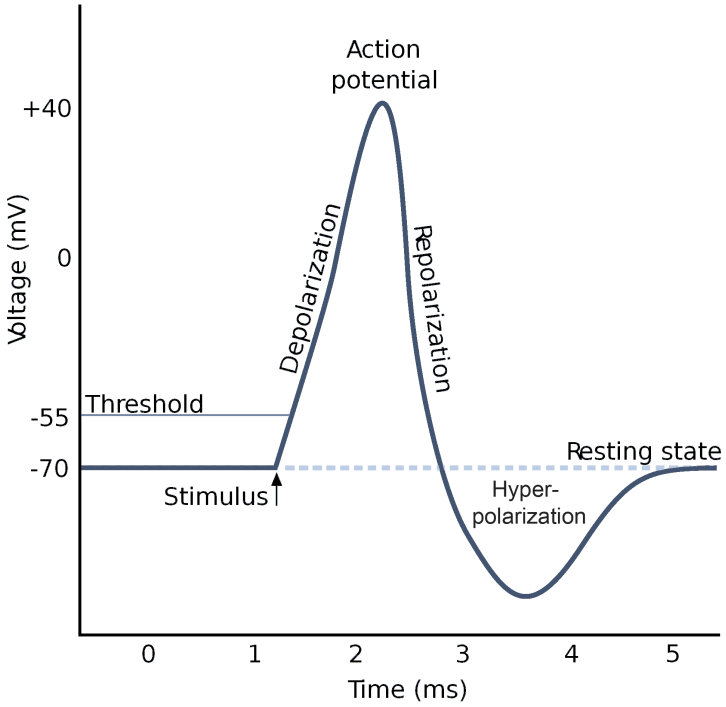

Figure 2.3: Variation of the membrane's potential during activation, adopted from the book "A Brief Introduction to Neural Networks", chapter 2, page 22 [1].

The previous paragraphs study the components of the nervous system, focusing on the neuron and its role in the EMG acquisition system. Thus, the next step is to understand how the nervous system is linked to the motor system, namely how the exchange of nerve impulses controls the body's movements.

The human motor system is comprised by the elements of the central nervous system related to motor control. More specifically, this includes the primary motor cortex, an area of the brain which generates neural impulses to control the execution of movements, and other regions of the cortex by the name of secondary motor cortices (Posterior Parietal Cortex, Pre-Motor Area, and Supplementary Motor Area). Therefore, signals from the primary motor

cortex travel from the body's midline in order to activate skeletal muscles of the opposite side of the body, hence the known phrase "the left hemisphere of the brain controls the right side of the body" [24].

Furthermore, neurons in the primary motor cortex, supplementary motor area and premotor cortex come together to form the fibers of the corticospinal tract, a pathway of the spinal cord which descends through the brainstem. This tract is the main pathway for the control of voluntary movements in humans, and it transports upper motor neurons which are responsible for the activation of muscle cells. These motor neurons travel until they reach the appropriate spinal nerve root, where they synapse directly with lower motor neurons, as shown by Figure 2.4.

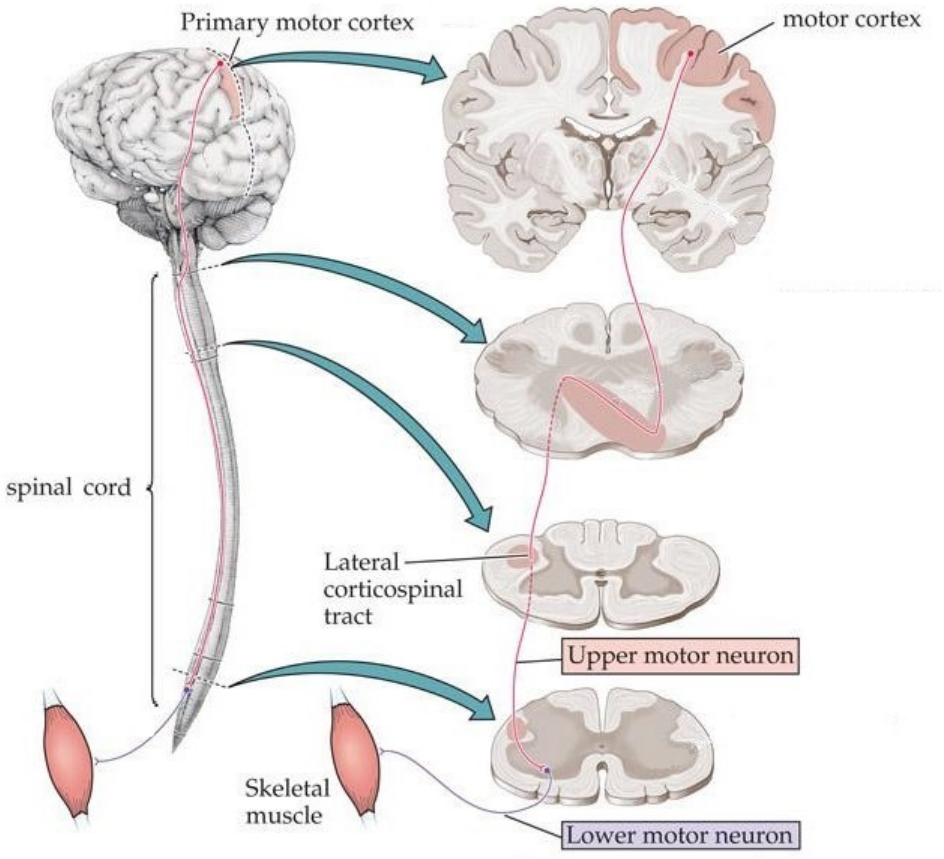

Figure 2.4: Anatomical diagram of the corticospinal tract.

Each lower motor neuron branches out and innervates different skeletal muscle fibers, forming a motor unit. Moreover, while the nerve fiber (neuron's axon) penetrates the muscle belly and stimulates different muscle fibers, each muscle fiber is only innervated by one neuron, which when fired makes all of its muscle fibers contract [4].

2.3 The EMG signal

After clarifying the concept of a motor unit, it is clear that the signal registered by a pair of electrodes does not correspond to a single muscle fiber. This is due to the fact that electrodes can detect the potential of all innervated muscle fibers within a motor unit. Therefore, the electrical signal measured by the electrodes corresponds to a sum of different fiber potentials, named Motor Unit Action Potential (MUAP) which constitutes the fundamental unit of the EMG signal. Thus, the EMG is formed by the electrical superposition of all the MUAPs, of all the active motor units, detectable under the electrode's position [25].

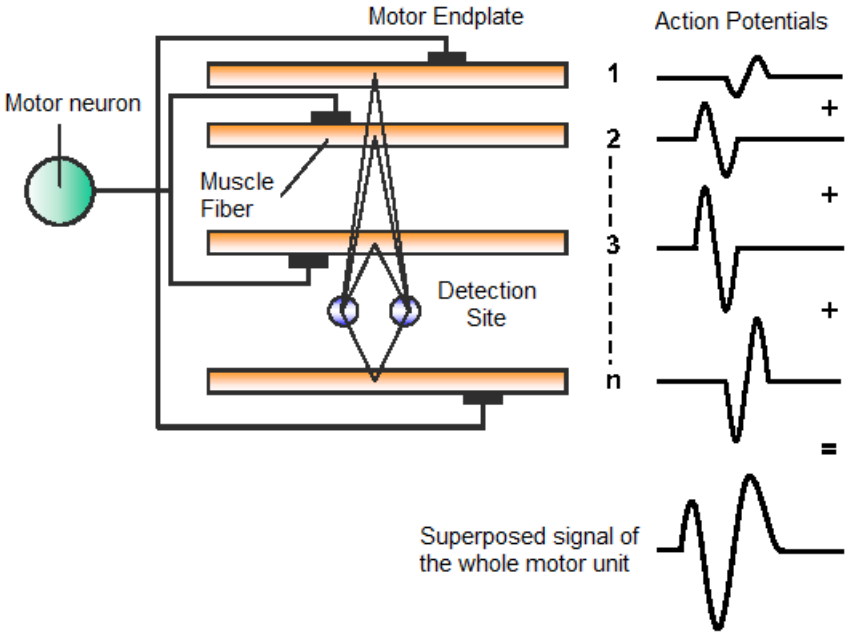

Figure 2.5: Electrical signal composed by a superposition of MUAPS, adapted from the book "The ABC of EMG, A Practical Introduction to Kinesiological Electromyography", pg 9 [2].

The EMG signals obtained directly from the acquisition system are referred to as raw EMG signals due to their non-processed nature, represented in Figure 2.6. These signals are composed of a noise-free baseline, during which the muscle is relaxed, and EMG spikes of non-reproducible shapes, which correspond to contractions.

This type of bioelectrical signal has the disadvantage of being very susceptible to noise, as it is affected by the anatomical and physiological properties of muscles, and by the characteristics of the instrumentation used to detect and measure it. Therefore there are external factors, that influence the quality of this signal such as: the quality of the electrodes and internal amplifier, physiological cross talk due to neighboring muscles, changes in the po-

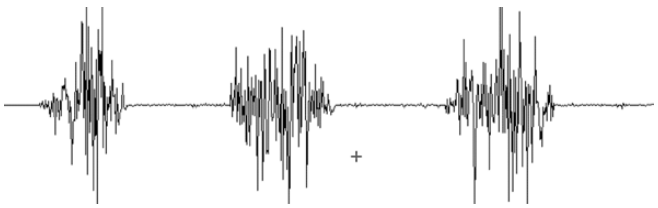

Figure 2.6: Raw EMG recording, retrieved from the book "The ABC of EMG, A Practical Introduction to Kinesiological Electromyography", page 11 [2].

sitioning of the electrodes and electrical noise [2]. In order to improve the quality of the extracted data one should apply electrodes in parallel to the muscle fiber direction, clean the skin where the electrodes will be placed, choose a suitable position for the electrodes and verify their positioning during data acquisition.

2.4 Artificial Neural Networks

For this thesis, the development of a sign language recognition system required a study of the different approaches to this problem, in order to choose a suitable extraction and classification methods. As such, the previous chapter focused on the chosen extraction method and the characteristics of the EMG signal. Thus, the next step is to analyze the classification method: Artificial Neural Networks.

Networks rely on the principle that simple elements can be gathered to produce complex systems. They are composed by a set of nodes, whose connections leads to a global behavior which surpasses the ability of each element. These nodes are computational units that receive and process inputs, in order to obtain an output. Furthermore, the connections between them determine the information flow between nodes, which can either be unidirectional or bidirectional [21].

ANNs are a type of network which models nodes after the biological behavior of a neuron, as shown by Figure 2.7. The artificial node is an abstract representation of a neuron, it receives inputs, which are multiplied by weights, and requires a mathematical function to determine its activation.

As previously mentioned, the study of these types of systems is motivated by their capacity to learn from training samples, and generalize and associate data. The network is trained to obtain desired outputs for specific inputs, by adjusting the weights of the connections between its nodes. More precisely, the higher the weight, the stronger the input multiplied by

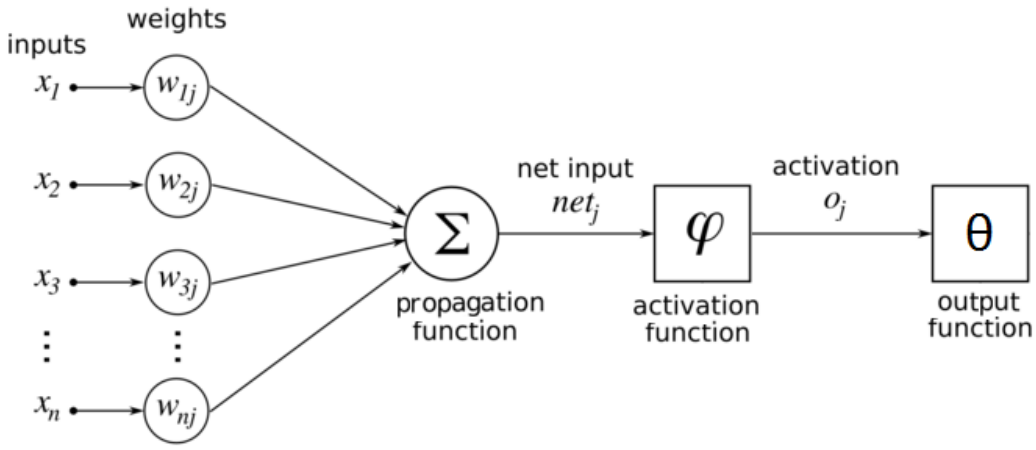

Figure 2.7: Diagram of the basic components of an artificial neural network.

it will be, and in the case of negative weights the more inhibited it will become.
Generally speaking, an artificial network can be reduced to four basic components:

Connection weights

A neural network has a set of neurons N, a set of connections V, between neuron i and j where (i,j) | i,j ∈ N, and a set of connection weights. These weights can be defined as $W : V \rightarrow R$ where w_{ij} represents the connection weight between neuron i and j [1].

Propagation Function

This function receives the outputs of the neurons connected to j, represented by x_{ij}, and transforms them into the network input according to the connecting weights. Generally it is given by the weighted sum of the inputs [21]:

$$net(j) = \sum_{i=1}^{m} w_{ij} x_{ij}, \text{ where m stands for the number of connections.} \tag{2.1}$$

Activation Function

The activation function transforms the network input and the previous activation state (on/off) in a new activation state, according to a specific threshold. This value indicates the point at which a neuron starts firing and marks the maximum gradient value of the activation function. In addition, this function is usually global for all neurons whereas the threshold values are specific to each neuron. The following are simple activation functions [26]:

◇ Step Function – If the input is above a certain threshold the function changes from one value to another, but otherwise remains constant. This function has the downside of being non differentiable at the threshold, which makes backpropagation learning impossible.

◇ Sigmoid Function - This type of function is commonly used in artificial network due to its differentiability. Moreover, two popular sigmoid functions are the Fermi function (2.2), which maps values to the range of (0, 1), and the Hyperbolic tangent (2.3) which maps values to (1, 1).

$$f(x) = \frac{1}{1 + e^{-x}} \qquad (2.2) \qquad\qquad f(x) = \frac{1 + e^{-x}}{1 - e^{-x}} \qquad (2.3)$$

Output Function

According to the activation state, the output function calculates the value transferred to the other neurons connected to j. Generally, this corresponds to the identity function and therefore the output is equal to the activation.

There are different ways to combine these elements into an artificial neural network, which depend of its design or topology. Generally speaking, neural networks can be divided into three types: feed-forward, recurrent or completely linked [1]. The most widely used is the feed-forward topology, where neurons are grouped in: one input layer, one or more hidden processing layers and one output layer. Moreover, each neuron in one layer has direct connections to the neurons of the next layer and no other connections are permitted.

On the other hand, recurrent networks allow neurons to influence themselves by any connections and therefore do not have explicit input or output neurons. In addition, this recurrence can be direct, where neurons connect to themselves, indirect, where neurons use indirect forwards connections to influence themselves, or lateral where each neuron inhibits the neurons of his layer in order to strengthen himself. Lastly, completely linked networks allow symmetric connections between all neurons, with the exception of direct recurrences.

CHAPTER 3

Methodology

Contents

The goal of this work is to develop a sign language recognition system, which can withstand the noisy nature of the EMG acquisition system and deliver an accurate reading of a sign language gesture, made by an untrained user (with no previous knowledge of sign language). To do so, this thesis builds upon the work of two previous dissertations, developed by students of the same faculty, whose main objective is motion classification of EMG acquired data [24, 4].

The pattern recognition system was developed in two stages. The first step required developing a motion classification system, based on the knowledge gained by the previous dissertations. During this initial learning phase, the system was applied to the classification of 6 simple movements, used in one of the aforementioned thesis [4]. This was done in order to facilitate the familiarization with the software, as simple motions are more easy to measure and interpret. As such, the results of this classification helped to establish the different steps of the methodology, which was developed by taking into account the efficiency achieved in the previous researches [24, 4].

The second and final step involved fitting the developed system for sign language classification and acquiring new data to test its performance.

Therefore, Figure 3.1 represents the different elements of the proposed pattern recognition system, which will be described in the following sub-chapters.

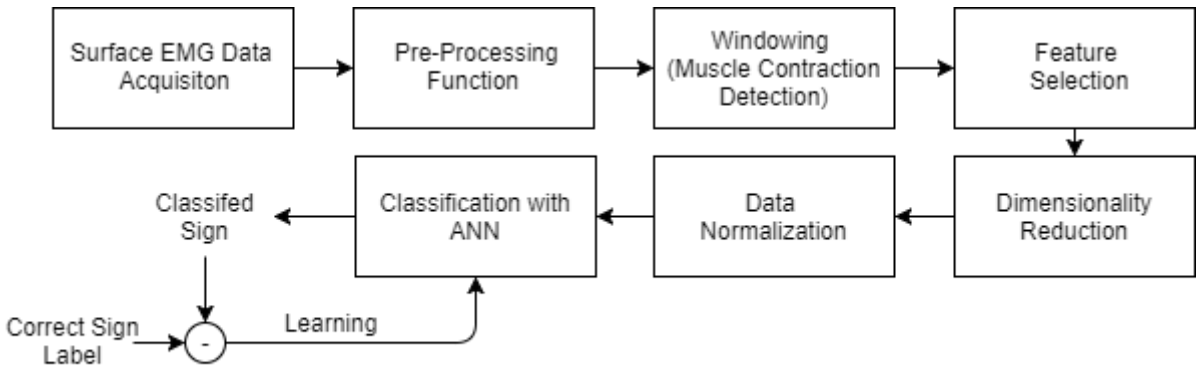

Figure 3.1: Block diagram of the EMG pattern recognition system.

3.1 Data Extraction

Data was acquired with BITitalino Plugged, from PLUX®, and the OpenSignals (r)evolution software. In addition, the first 4 analog input channels of the device were connected to 4 EMG sensors, whose specifications are represented by Table 3.1.

Vcc	Bandwidth	Sensor Gain	Nº of bits
3.3V	10 - 400Hz	1000	10

Table 3.1: Specifications of the EMG sensors, retrieved from the electromyography sensor data sheet [5].

As recommended by the OpenSignal's Revolution manual [27], a sampling rate suitable for processing EMG signals was selected. Therefore, after amplification and analog filtering, data is sampled to a rate of 1000HZ and transferred by Bluetooth to a computer, paired with the bitalino device. Simultaneously, the Open Signals software shows a real time visualization of the multichannel EMG signals. In addition, at the end of the recording session the extracted data is saved to a text file which is fed to Matlab R2014a, where it is processed and classified.

This text file shows the acquisition parameters (sampling frequency, sampling resolution and time/date), the device (mac address and device type) and the extracted data, which follows a specific structure:

- ◇ The first column indicates the sequential number of the sample.
- ◇ The second to fifth column correspond to digital channels.
- ◇ The sixth to ninth column indicates the amplitude values of the EMG signal, where each column represents one of the 4 available channels.

Therefore, one EMG measurement corresponds to a vector of 4 analog signals, one from each channel, where each signal has a resolution of 10 bits.

3.2 Data Pre-processing

The first step of processing the extracted digital data is converting it to analog, according to Equation 3.1, fron the EMG sensor's data sheet [5]:

$$EMG(V) = \frac{(\frac{ADC}{2^n} - \frac{1}{2}) \times Vcc}{G_{EMG}} \tag{3.1}$$

where Vcc stands for the operating voltage in volts, ADC refers to the digital value to be converted, n is the number of bits of the channel and G_{Emg} corresponds to the amplifier's gain. These values are converted to microvolts and their direct current component is removed, by subtracting each signal by its mean, with the use of the Matlab function *detrend*.

Generally, the following step is to obtain the linear envelope of the signal, which requires full wave rectification and some sort of low pass filtering within the 5-100 Hz range [9, 28, 29]. In this case, the low pass filtering was achieved by a 2nd order Butterworth filter, with a cutoff frequency equal to 22Hz, applied in both forward and backward directions in order to minimize the phase shift phenomenon [2, 29]. This was achieved with the use of the Matlab function *filtfilt*, which filters the data in the forward direction, reverses the filtered sequence and filters it again. Consequently, this double filtration requires the adjustement of the cutoff frequency by 25%, in order to obtain the desired cutoff fequency of 22Hz (fco):

$$fco_{adj} = fco \times 1.25 = 27.5 \tag{3.2}$$

Therefore, the amplitude response of the filter is represented in Figure 4.2.

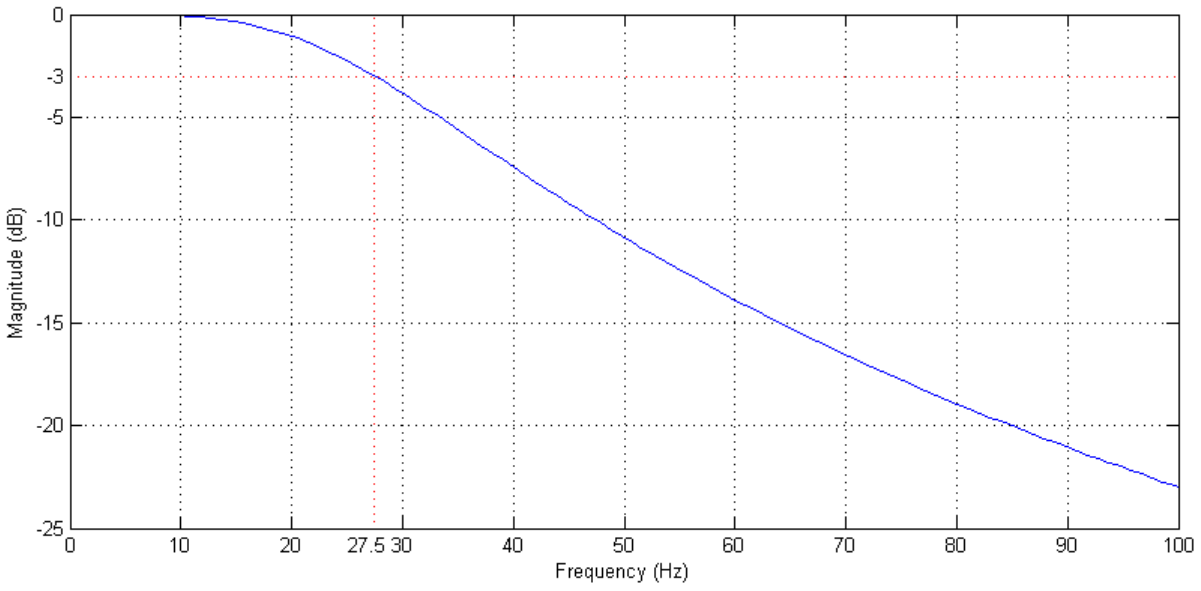

Figure 3.2: Amplitude response of the 2nd order Butterworth filter.

However, it is important to mention that this step exists purely as an auxiliary visualization tool, which when activated displays the linear envelope of the acquired signals to the user, for post-processing analysis. This is because at a practical level such processing generated less informative EMG signals which consequently decreased the accuracy of data classification. This alteration of methodology occurred during the classification phase, as

opposed to the choice to avoid notch filters, which was made initially. Moreover, this was due to the advice found in the book "The ABC of EMG", which stated that notch filtering would destroy too much signal information.

3.3 Muscle Contraction Detection

The next step is to conduct signal windowing, in order to select the signal intervals with relevant information, namely the activity spikes generated during muscle contraction. This requires an accurate identification of the onset time of the EMG burst, in order to correctly determine the amount of relevant information transmitted to the classifier. Moreover, this step was applied to all four raw signals because the potential flow might reach different muscles at different times, leading to different windows of contraction.

The simplest approach to this problem is the single threshold method, where muscle contraction is determined when the amplitude of the signal exceeds a fixed threshold [2]. However, this method often leads to false positives and as such it requires the addition of a second threshold to improve the detection's sensitivity [4]. Therefore, the double threshold methods detects signal onset when a predefined number of samples (2nd threshold) consecutively exceeds the first fixed threshold. In addition, the first threshold can be defined by: $T = \mu + h \times \sigma$, where μ and σ refer to the mean and standart deviation of the baseline of the signal, during a period of inactivity, and h is a preset variable which defines the level of the threshold.

Besides the threshold-based methods, there are other popular algorithms for onset detection such as visual determination, where individuals set the boundaries of the EMG burst with the aid of a Matlab graphical user interface, and approximated generalized likelihood, an advanced statistical method where the raw EMG is presumed to be a Gaussian white noise signal, filtered by tissues or electrodes [29].

In this case, the chosen method was a combination of the double threshold method and the Teager-Kaiser energy operator (TKEO). This operator measures instantaneous energy changes of signals, composed of a single time varying frequency, and it was proposed by Li et al. (2007) in order to improve the signal-to-noise ratio (SNR) and EMG onset detection accuracy [28].

The decision to combine these two methods was based on a research by Solnik et al.(2008), who reported that TKEO improves all three onset detection methods when TKEO conditioning is applied beforehand. Therefore, the discrete form of the Teager's energy operator

can be defined by Equation 3.3:

$$\psi[x(n)] = x^2 - x(n+1)x(n-1) \tag{3.3}$$

where x refers to the sample vector and n is the sequence index. Moreover, this equation shows how the calculated energy is derived from the instantaneous amplitude and instantaneous frequency of the signal. Thus, this operator improves the ability to analyze muscle activity, as it incorporates main characteristics of muscle contractions, namely rapid fluctuations in a signal's amplitude and frequency [29].

According to the research made by Solnik et al [29], the TKEO operator is applied to the digital EMG signal and the envelope of the TKEO signal is used to detect muscle contraction, according to the double threshold method. More precisely, this envelope is calculated in the same way as the one previously described during pre-processing, the signal is rectified and then double filtered by a 2nd order Butterworth filter.

After the onset point is determined, it is necessary to choose the window size and delimitate its boundaries. In order to measure the potential action at 1KHz sampling rate, the window size was set to 1000 samples, as a bigger size lead to no improvement on classification accuracy. In addition, the boundaries of the window were determined experimentally, and the window was set to begin 10 samples before the onset point.

In this case, there is no need for offset detection as each EMG segment has a fixed size throughout the entirety of the measuring procedures. This choice was made in the interest of decreasing the inter-individual variability of gestures, as the inability to replicate gestures with the same intensity, namely the amount of applied force, can often lead to different ending points for muscle contraction.

3.4 Feature Selection

This research fits in the realm of statistical pattern recognition, a science which aims to find, learn and recognize patterns in complex data, where each pattern is represented in terms of d features, and it is viewed as a point in a d-dimensional space.

These features must be chosen in a way that allows pattern vectors of different categories, to occupy disjoint regions of the d-dimensional feature space [30], meaning that given a set of training samples, the model should be able to establish decision boundaries who correctly separate patterns from different classes.

Therefore, the following step is mapping the large EMG vector to a lower dimension vector according to a set of discriminatory features, through a process named feature selection.

Generally, features regarding EMG analysis can be divided into three main groups: time domain, frequency domain and time-frequency domain features [31]. For this thesis, only the first two groups were considered due to the fact that time-frequency domain features such as Fourier transform, discrete wavelet transform and wavelet packet transform, require much more complicated processing.

Furthermore, the majority of the selected features are time-domain features, a popular choice in the field of EMG classification due to the fact they provide high classification accuracies at a low computational cost [4, 31]. In addition, this research also analyses autoregressive model coefficients (AR), a frequency-domain feature, as it has yielded good classification results in previous researches [8, 16, 24]. As such, the selected features are as follows:

3.4.1 AR Coefficients

These coefficients are used to characterize an auto-regressive model, a prediction model which describes each sample of the EMG signal as a linear combination of the previous samples x_{k-i}, plus a white residual noise e_k [31]. It can be mathematically defined as:

$$x_k = \sum_{i=1}^{p} a_i x_{k-i} + e_k \tag{3.4}$$

where a_i represents the auto-regressive coefficients and p is the AR model order.

There are a number of different techniques for computing AR coefficients which approximately give the same coefficients. For this research, the chosen algorithm is the Burg method due to its computational efficiency and stability [4].

Consider a discrete signal x_n, with n $\in [0, N]$ where N corresponds to the length of the signal. This signal can be approximated by k coefficients (a_n) through a forward linear prediction: a weighted sum of the previous k known values (y_n), and a backward linear prediction: a weighted sum of the next k known values (z_n), defined by Equations 3.5 and 3.6 respectively [4].

$$y_n = -\sum_{i=1}^{k} a_i x_{n-i} \tag{3.5}$$

$$z_n = -\sum_{i=1}^{k} a_i x_{n+i} \tag{3.6}$$

The Burg method minimizes the sum of the forward and backward prediction errors, defined by Equations 3.7 and 3.8, while satisfying the Levinson-Durbin recursion [1]. Therefore, instead of minimizing the forward or backwards prediction error individually with a "covariance-like" error method, this method minimizes the sum of these errors and finds a more stable model.

$$F_k = \sum_{n=k}^{N} (x_n - y_n)^2 \tag{3.7}$$

$$B_k = \sum_{n=k}^{N} (x_n - z_n)^2 \tag{3.8}$$

Finally, this function was implemented with the use of Matlab's function AR, after selecting an adequate order for the AR model. Moreover, this parameter should be carefully chosen as it greatly impacts the fidelity of the reconstructed signal and the computational cost of the algorithm. According to the research done by André Ferreira in his master thesis [4], the model's order was set to 7, a compromise between the quality of the reconstructed data and the number of components of the feature.

3.4.2 Hjorth Parameters

Introduced by Bo Hjorth in 1970, the Hjorth parameters are indicators of statistical properties based on time domain properties [4]. These consist of the following: an "Activity" parameter (Eq. 3.9), which measures the variance of the amplitude of the signal, a "Mobility" parameter (Eq. 3.10), which indicates the dominant frequency and a "Complexity" (Eq. 3.11) parameter, that represents the change in frequency, by quantifying the signal's deviation from the sine shape. These values can be defined by the following equations:

$$A = \sigma_0^2 \tag{3.9}$$

$$M = \frac{\sigma_1}{\sigma_0} \tag{3.10}$$

$$C = \frac{\sigma_2 \sigma_0}{\sigma_1^2} \tag{3.11}$$

[1]The Levinson-Durbin is a specific procedure commonly used in linear algebra to calculate the solution of an equation involving a diagonal-constant matrix [32]

where σ_0 is the standard deviation of the signal contained in the EMG segment and σ_1, σ_2 correspond to the standard deviation of the first and second derivative of the aforementioned signal, respectively.

3.4.3 Integral Absolute Value

The Integral Absolute Value (IAV) is defined as the summation of the absolute values of the EMG segment [31], that can be defined by the following equation:

$$IAV = \sum_{i=1}^{N} |x_i| \tag{3.12}$$

where x_i corresponds to the amplitude of sample i and N represents the length of the EMG segment.

3.4.4 Mean Absolute Value

The Mean Absolute Value (MAV) is one of the most popular features used in EMG signal analysis, it is defined by the mean absolute value of the signal contained in the data segment [31]. This value can be given by the following equation:

$$MAV = \frac{1}{N} \sum_{i=1}^{N} |x_i| \tag{3.13}$$

where x_i represents the ith sample of the segment and N is the signal's length.

3.4.5 Root Mean Square

The Root mean square (RMS) is the square root of the arithmetic mean of the signal's squared values [31]. In addition, the mathematical definition of this feature can be given by:

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^{N} |x_i^2|} \tag{3.14}$$

where x_i is the ith sample of the segment and N corresponds to the signal's length.

3.4.6 Skewness

This feature measures the asymmetry of the data around the sample mean, thus if the data is perfectly symmetric then the skewness of the normal distribution is zero [4]. Moreover,

this value was calculated with the aid of the Matlab function *skewness*, according to the following equations:

$$s = \frac{E(x-\mu)^3}{\sigma^3} = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^3}{\left(\sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2}\right)^3} \tag{3.15}$$

where $E(t)$ represents the expected value of the quantity t, x corresponds to the EMG segment and μ, σ and N are its mean, standard deviation and length, respectively.

3.4.7 Slope Sign Changes

The Slope Sign Changes (SSC) is defined by the number of times the slope of the EMG waveform changes sign, within an analysis window. This feature reduces noise-induced counts trough the use of a threshold value according to the following equations [33]:

$$SSC = \sum_{i=2}^{N-1} f\Big[(x_i - x_{i-1}) \times (x_i - x_{i+1})\Big] \tag{3.16}$$

$$f(t) = \begin{cases} 1, & \text{if } t \geq \text{ threshold} \\ 0, & \text{otherwise} \end{cases} \tag{3.17}$$

where x_i corresponds to the nth sample of the EMG segment, N is the signal's length and $f(t)$ is the auxiliary function used to screen noise induced slope changes.

3.4.8 Waveform Length

The Waveform Length (WL) measures the complexity of the EMG signal, and is defined by the cumulative length of the EMG waveform during the time segment [4]. It can be calculated using Equation 3.18.

$$WL = \sum_{i=1}^{N-1} |(x_{i+1} - x_i)| \tag{3.18}$$

where x_i corresponds to the ith sample of the EMG segment and N represents the signal's length.

3.4.9 Zero Crossings

This feature is defined as the number of times the amplitude values of the EMG signal cross the zero amplitude level [33]. Furthermore, in order to avoid low voltage fluctuations or background noises, a threshold condition is implemented according to the following equations:

$$ZC = \sum_{i=1}^{N-1} g\Big[(x_i \times x_{i+1})\Big] f\Big[|x_i - x_{i+1}|\Big] \tag{3.19}$$

$$g(t) = \begin{cases} 1, & \text{if } t < 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.20}$$

$$f(t) = \begin{cases} 1, & \text{if } t \geq \text{ threshold} \\ 0, & \text{otherwise} \end{cases} \tag{3.21}$$

where x_i corresponds to the nth sample of the EMG segment, N is the signal's length, $f(t)$ is the auxiliary function that enforces the threshold value and $g(t)$ is the function used to detect slope sign changes.

3.5 Dimensionality Reduction

Taking into account all of the features previously described, one can conclude that the size of the system's input vector will be very high. Furthermore, each EMG segment will result in: 7 time-domain features composed by 1 element, 1 time-domain feature formed by 3 elements (Hjorth Parameters) and one frequency-domain feature formed by 7 elements (AR coefficients). Consequently, this leads to an overly complex model, with an input vector of 68 features (17×4 EMG channels) and complicated decision boundaries.

Consequently, this level of complexity has a negative impact on the model's ability to provide good generalization, even if it might lead to a perfect classification of training samples [30]. This is explained by the "peaking phenomenon", who states that adding features may degrade the classifier's performance if the number of training samples, used to design the classifier, is small compared to the number of features [34].

Moreover, the exact relationship between the probability of misclassification, the number of training samples and the number of features is very difficult to determine. Thus, the general guideline is that one should use at least ten times as many training samples per class

as the number of features: $n_c/d > 10$, where n_c is the number of training samples per class and d is the number of features [34].

Therefore, the minimum number of training samples per class is 680 samples, a very high amount considering the small scale of this research. Hence, it is is necessary to perform dimensionality reduction in order to establish a subspace of dimensionality m, of the original feature space of dimensionality d, where $m < d$.

This can be done by different methods such as: principal component analysis, factor analysis, linear discriminant analysis, among others. In addition, it is important to mention that these functions achieve dimensionality reduction by creating new features, based on transformations or combinations of the original feature set, instead of selecting a smaller subset of features.

For this thesis, the chosen method was Principal Component Analysis (PCA), an algorithm which calculates the dimension-reduced matrix Y, by determining the m largest eigenvectors of the $d \times d$ covariance matrix of the n d-dimensional data [4]. Thus, this relationship can be expressed by the following equation:

$$Y = XH \tag{3.22}$$

where X is the $n \times d$ original data matrix, Y is the desired $n \times m$ reduced matrix and H is the $d \times m$ matrix whose columns correspond to the eigenvectors.

Finally, the linear transformation matrix (H) was constructed by taking the m highest eigenvectors (as columns), of a list of eigenvectors calculated by the Matlab function *pca*, and by multiplying it by the original data matrix (X).

3.6 Data Normalization

Before classification, the full data set should be scaled so that each feature vector has the same unit variance and zero mean. Therefore, the data is standardized to fit the range of [0,1],through the use of the following equation:

$$\sum_{i=1}^{m} \frac{X_i - \mu_i}{\sigma_i} \tag{3.23}$$

where m is the reduced number of features, X_i corresponds to the components of feature i, and μ_i and σ_i are the mean and standard deviation of X_i. This step helps to ensure a uniform learning procedure and to improve the classification accuracy, while shortening the training time [4] [30].

3.7 Classification with Neural Network

After the previous processing steps, the resulting input vector will have the following format:

$$InputVector = \begin{bmatrix} f_{(1,1)} & f_{(1,2)} & f_{(1,3)} & \cdots & f_{(1,m)} \\ f_{(2,1)} & f_{(2,2)} & f_{(2,3)} & \cdots & f_{(2,m)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f_{(n,1)} & f_{(n,2)} & f_{(n,3)} & \cdots & f_{(n,m)} \end{bmatrix} \tag{3.24}$$

where each line of the matrix corresponds to an m-dimensional representation of an EMG signal and n is the total number of measured EMG signals.

As previously mentioned, the aim of this thesis is to develop a sign language recognition system based on previously established work [24, 4]. As such, the system was developed in two main stages. First it was applied to the classification of six simple arm movements and a methodology for motion recognition was established. Then, its was applied to the classification of 10 sign language gestures, and its parameters were tuned accordingly.

3.7.1 Selected Movements

In order to test the system aptitude for discriminating simple EMG signals, the following six movements were chosen:

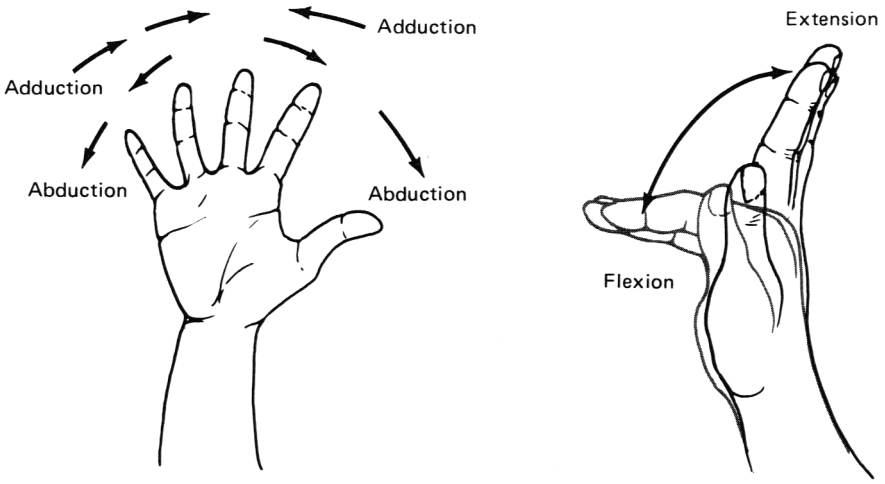

Figure 3.3: Hand abduction and adduction, wrist flexion and extension. Retrieved from the book "Kinesiology: Scientific Basis of Human Motion" [3], chapter 6.

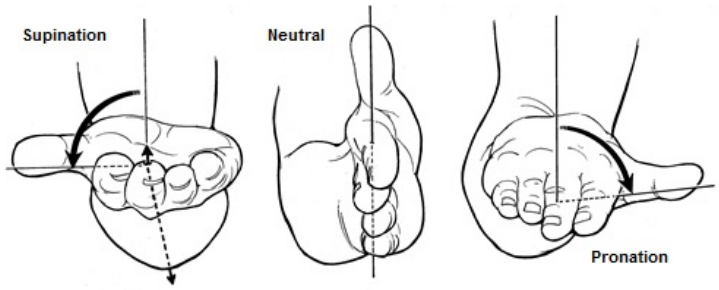

Figure 3.4: Hand supination and pronation, adapted from the website www.gustrength.com

Furthermore, the following muscles were chosen for sensor monitoring:

Muscle	Function	Nº of Sensor
Extensor Digiti Minimi	Wrist and litle finger extension.	1
Extensor Digitorum	Phalanges and wrist extension.	2
Flexor Carpi Radialis	Wrist flexion, forearm over arm flexion.	3
Flexor Carpi Ulmaris	Wrist flexion, forearm over arm flexion, hand adduction.	4

Table 3.2: Targeted muscles during movement classification.

In order to measure the desired muscles the placement of the sensors follows a specific layout, depicted in Figure 3.5. It is important to mention that each movement was performed from a neutral position, with the left arm flexed at 90 degrees by the side of the trunk [24].

Figure 3.5: Electrode placement during movement classification, where the 5th sensor corresponds to the reference electrode. [4].

3.7.2 Selected Sign Language Gestures

For this thesis, static and dynamic gestures were selected from the LGP to test the system's ability to classify both simple and complex gestures, where different stages of movement are distinguishable.

In order to do so, the following gestures were selected: 4 static gestures, representative of cardinal numbers (Figure 3.6), and 6 dynamic gestures regarding word-level vocabulary (Figure 3.7). Furthermore, the dynamic gestures were chosen in pairs according to their similarity in movement, in order to test the systems ability to distinguishing similar patterns.

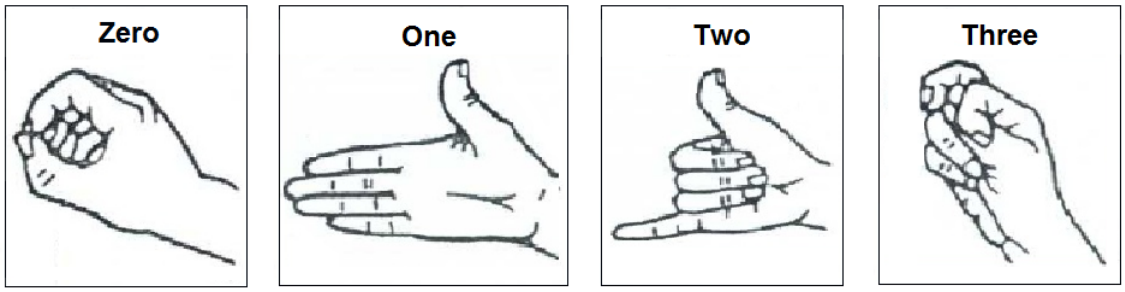

Figure 3.6: Static sign language gestures chosen for classification.

Figure 3.7: Dynamic sign language gestures chosen for classification.

These gestures were performed from the same starting position as the one previously described, with the same configuration of sensors except for sensor nº4. This electrode

needed to be placed in another area due to the combination of the sensor's location and its cable length, as it limited the range of motion demanded by specific dynamic gestures and eventually lead to the breakage of the aforementioned cable. As such, the sensors were placed in a different layout, described in Figure 3.8.

Figure 3.8: Electrode placement during gesture classification, where the 5th sensor corresponds to the reference electrode.

3.7.3 Neural Network's Parameters

In order to maximize the system's performance during classification, different types of *feed-forward* artificial neural networks were tested. More precisely, this was done by varying certain parameters, which will be described shortly, until an optimal architecture was found. These types of networks are based on the *backpropagation* (BP) algorithm, a theorem used to minimize the error between the output data and the target data [26]. This algorithm requires a *supervised learning*, thus the network was provided with a set of inputs and the expected corresponding outputs. Therefore, the artificial neurons propagate their signals "forward" and the errors (difference between actual and expected results) are propagated backwards [21]. As such, the analyzed parameters were the following:

Number of hidden neurons

While the number of input and output neurons are determined by the input vector's size and the number of categories, the number of hidden neurons is variable and does not follow

any observable rule [30]. Furthermore, an increase in the number of hidden units does not necessarily mean an increase in classification accuracy. Therefore, a high number of hidden neurons may lead to a very small training error at the cost of an unacceptably high test error, caused by overfitting.

Data Division

The training procedure requires the division of the extracted data into three sets, namely a training set, a validation set and a testing set. This allows for an objective evaluation, as the network performance is analyzed with a test set for which it was not trained before. Therefore, the training data is used to train the network, the validation data monitors the network's performance and the testing set evaluates the predictive ability of the trained network [1].

Thus, besides the general data sectioning ratio (80% training data, 10% validation data and 10% test data), other ratios were tested in order to analyze the classifier's performance in different scenarios.

Training algorithm

In order to obtain the desired outputs, the ANN adjusts the connection weights between its neurons during a process commonly refereed to as training or learning. This behavior can be implemented by different algorithms: modifications of the BP algorithm which aim to overcome some of its disadvantages, namely its slow convergence [35].

Furthermore, it is difficult to pinpoint a specific algorithm which works best for a certain problem, as multiple factors such as the network's size, the training goal and the computation complexity influence its performance [26]. Therefore, in order to find the optimal method for the problem at hand, different training algorithms were tested with the artificial neural network:

- **Scaled conjugate gradient BP (*trainscg*)**

From an optimization point of view, learning in a neural network is equivalent to minimizing a global error function. Therefore, the goal of these algorithms is to minimize a function by adjusting several thousands of the network's weights.

This can be done by a scaled conjugate gradient method, which differs from the standard gradient method due to the fact that it does not perform a line search at the end of each

iteration. A line search involves several calculations of either the global error function or its derivative, which consequently increases the complexity of the algorithm [36]. This technique bypasses these calculations and it is used as the default learning algorithm in pattern recognition neural networks, by Matlab's neural network toolbox. Thus, this method was tested as a possible contender the network's learning algorithm.

◇ **Conjugate gradient BP with Powell-Beale restarts (*traincgb*)**

Normally, for all conjugate gradient algorithms the search direction is periodically reset to the negative of the gradient. In addition, this occurs when the number of iterations is equal to the number of network parameters (weights and biases). However, by searching along the steepest descent direction the immediate reduction in the objective function is usually smaller than it would be without this restart. Thus, there are other reset methods that can improve the efficiency of training, namely the Powell-Beale algorithm [37].

This technique uses the Powell-Beale algorithm by restarting when the orthogonality between the current gradient and the previous gradient is small, without restricting the search direction to the steepest descent direction. In this case, this algorithm was tested due to its use as a training algorithm, in the master thesis developed by Luis Marques [24].

◇ **Levenberg-Marquardt BP (*trainlm*)**

This algorithm is a combination of the steepest descent method and the Gauss-Newton (GN) method, developed by Kenneth Levenberg and Donald Marquardt. Moreover, it performs a combined training process: near an area with a complex curvature, evaluated by second-order derivatives of the error function, the LM switches to the steepest descent algorithm. Then, when the curvature is suitable for quadratic approximation the algorithm becomes the GN method, which speeds up the convergence significantly [35].

Therefore, this method is strongly recommended as a first-choice supervised algorithm for the training of small and medium-sized neural networks, due to its stability and speed of convergence.

◇ **Bayesian regularization BP (*trainbr*)**

While the LM method was especially developed for faster convergence in BP algorithms, the Bayesian Regularization algorithm aims to improve the generalization capacity of the

model. In order to do so, this method includes a term with the residual sum of the squared weights in the minimization function, thus it minimizes a linear combination of the squared errors and the squared weights. In addition, it modifies this combination to obtain good prediction abilities at the end of the training [38]. Besides, it has a better performance than conventional methods in terms of overfitting, which makes it a suitable candidate for testing.

CHAPTER 4

Results Analysis

Contents

While previous researches have stated the efficiency of neural networks in the classification of simple motions [24, 17, 39], the application of these networks to sign language is still debatable when compared to other mechanisms of classification [10]. This can be attributed to the complexity of these types of signals, as sign language gestures are often produced by dynamic gestures where different stages of motion can be differentiated.

Therefore, this thesis approaches the classification of both simple arm motions and complex sign gestures through the use of an ANN. Hence, the pattern recognition system uses a 2-layered artificial network with a tangent-sigmoid activation function and a linear output function to classify EMG patterns offline.

4.1 Classification of Gestures

In order to evaluate the system's ability to classify complex gestures, data was acquired from 4 able bodied subjects with no previous knowledge of sign language. In addition, each subject was submitted to an initial learning session due to their inexperience and lack of consistency during gesturing. Thus, after learning the proper form each person was asked to perform 30 repetitions of 10 sign language gestures.

Furthermore, after some data acquisition sessions this number was increased to 35 ± 5 due to breakages in the sensor's cables. These cables introduced high levels of noise and flat-lining signals, making some samples unfit for classification. Consequently, although the ANN is known for its robustness to noise, such an introduction may have had a negative impact on the system's performance.

This potential loss in performance is due to the contraction detection mechanism, a processing step which precedes the neural network's classification and is highly susceptible to noise. Therefore, in order to partially solve this problem, when background noise overshadows muscle contraction and the onset contraction point is not detected, the EMG samples are automatically discarded.

4.1.1 Model selection

As previously mentioned, this research aims to find the optimal architecture for sign language classification. Therefore, different ANNs were tested in order to determine which parameters result in the highest classification accuracy. The study focused on influential parameters such as the training function of the ANN, number of hidden neurons and number

of features used to represent the input data. par In order to achieve a valid conclusion and select an optimal model, every ANN was run under the same conditions. More specifically, the randomness associated to their execution was fixed through the use of the Matlab function *rng*. This was necessary in order to enforce the same random weights initialization for all ANNs and to make sure that their performance was analyzed on equal groups of data (test data).

Moreover, the performance of each ANN was measured through the classification accuracy of the test data, according to the following equation:

$$\eta(\%) = \frac{\text{Number of correct labels}}{\text{Total number of labels}} \times 100 \tag{4.1}$$

The goal of this process is to obtain a final model with good generalization abilities, capable of predicting correct outputs for previously unseen data. In order to do so, one must avoid overfitting the ANN to the training data as this results in a model with no practical application, whose use is limited to memorizing samples [40].

Due to the small size of the test data it is important to take into consideration the effect of tuning the network's parameters around 10% of its total data. Although this may result in high accuracy results, they are not a true reflection of the networks performance. More specifically, while certain parameters can increase the classification accuracy of a small set of test data, the same isn't true for other sets. Therefore, in order to avoid these pitfalls a 10-fold cross-validation technique was used to determine the classification accuracy of each ANN. This techniques is commonly used when the size of the test data is limited, to ensure a stable and confident estimate of the model's performance [41].

In this case, the classification accuracy of each architecture was calculate with an adapted form of the 10-fold cross-validation algorithm, represented by Algorithm 1. The dataset was randomly divided into 10 subsets of approximately the same size ($T_1 ... T_{10}$) and the network was tested on each subset of data. More precisely, the network was trained 10 times and in each iteration, one subset was chosen for testing (G_{te}), another for validation (G_v) and the remaining subsets formed the training set (G_{tr}).

As a consequence of this algorithm, the proportion of the training, testing and validation subsets is strictly restricted by the number of folds, which in this case was set to 10. However, while each training had a sectioning ratio of 80% training, 10% test and 10% validation, the network's architecture was tested with the entirety of the data, allowing for a better estimation of its performance.

Algorithm 1 10-fold cross-validation

Divide the data in 10 disjoint subsets of the same size: $T = \{T_1, T_2.....T_{10}\}$

Create an extra subset to account for the validation data: $T_{11} \leftarrow T_1$

for $i = 1...10$ **do**

Form the testing, validation and training sets:

$G_{te} \leftarrow T_i, \quad G_{va} \leftarrow T_{i+1}, \quad G_{tr} \leftarrow T \backslash \{G_{te}, G_{va}\}$

Train the model using G_{tr} and G_{va}, and assess its performance with G_{te}:

$Acc(i) \leftarrow \text{Model}(G_{te})$

end for

Calculate the performance by averaging the previously obtained test accuracies:

$$Total\ Accuracy = \frac{\sum_{i=1}^{10} Acc_i}{10}$$

To confirm the necessity of this method, the accuracy was determined with and without 10-fold cross-validation, and the following results were obtained:

Number of Neurons										10-fold
10	20	30	40	50	60	70	80	90	100	Cross-validation
77.60	89.60	89.60	92.00	**99.20**	92.80	91.20	92.00	95.20	92.80	No
73.62	90.08	90.35	94.24	93.98	93.80	93.98	93.62	**94.95**	94.95	Yes

Table 4.1: Test accuracy for 10 sign language gestures, with a feedforward network trained by trainbr, a 80/10/10 sectioning and 20 features.

According to the previous table, without 10-fold cross-validation, the results suggest that it is possible to obtain a test accuracy of 99.2% with 50 hidden neurons, 20 features and the use of the training function *trainbr*. However, this percentage is not a true reflection of the networks performance and it would lead to a wrong choice of optimal parameters. Consequently, when this optimal model is applied to the classification of a different set of test data, with the same size, the classification accuracy drops to 93.6%. As expected, this value is closer to the one provided by 10-fold cross validation, therefore this technique will be used to calculate all the test accuracies from this point on.

Tables 4.2 to 4.5 represent the test classification results for 1250 data samples, in function of the number of hidden neurons, the type of training algorithm and the number of features used to represent the input data. Each accuracy was calculated with the adapted 10-fold validation algorithm, and each fold classified a test set of 125 samples.

Number of Neurons																			Nº of
10	20	30	40	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	Features
73.62	90.08	90.35	94.24	93.98	93.80	93.97	93.62	**94.95**	94.95	93.97	93.98	93.98	94.24	94.24	94.24	94.06	94.33	94.51	94.59	20
72.99	89.29	92.47	94.15	94.51	94.07	93.71	94.77	93.80	95.04	93.80	94.33	94.51	95.04	94.68	94.77	95.13	93.97	94.86	**95.40**	18
72.64	87.16	91.15	92.83	94.07	94.24	94.06	93.54	93.09	94.07	94.25	93.89	94.07	94.06	93.62	92.92	94.07	93.71	**94.60**	93.80	16
70.96	86.81	91.94	92.48	92.29	93.71	93.62	92.65	93.36	93.53	93.00	93.27	92.56	93.44	**93.89**	93.71	93.44	93.53	92.47	93.54	14
65.63	85.21	87.87	90.43	91.32	91.85	91.94	92.03	92.12	91.85	91.85	92.03	91.23	91.94	**92.29**	92.03	91.76	91.76	92.03	91.94	12
64.83	79.81	86.45	87.60	88.39	89.38	90.79	89.19	90.08	88.66	89.91	**90.88**	89.37	89.64	89.99	90.61	90.35	90.34	89.55	89.64	10

Table 4.2: Test accuracy for 10 sign language gestures, with a feedforward network trained by trainbr and a 80/10/10 sectioning.

Number of Neurons																			Nº of
10	20	30	40	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	Features
71.21	86.28	91.23	91.06	93.54	93.44	93.62	94.24	93.18	94.33	93.80	94.24	93.71	**94.86**	94.24	93.71	93.09	93.18	94.06	93.98	20
72.64	87.51	91.23	92.29	92.29	94.15	**94.42**	93.89	94.42	94.15	93.80	93.53	94.33	94.16	93.80	93.63	94.33	93.62	92.91	93.89	18
69.99	85.20	89.72	91.50	92.92	94.06	93.36	93.98	93.98	93.18	93.80	93.00	92.74	**94.24**	93.98	92.56	92.47	92.92	93.71	93.27	16
69.10	85.29	88.39	90.34	91.41	92.20	92.82	93.00	91.85	**93.71**	93.09	93.18	92.30	92.74	91.94	92.30	91.41	92.21	91.85	91.77	14
63.70	81.84	86.53	87.24	90.34	91.23	91.23	90.88	91.94	91.86	91.94	91.41	90.70	**92.30**	91.06	90.97	90.70	90.71	90.17	90.97	12
62.00	77.77	82.37	86.63	86.89	89.02	87.61	**89.73**	88.57	89.46	89.55	89.02	87.95	88.75	89.64	88.93	89.02	88.84	89.02	89.02	10

Table 4.3: Test accuracy for 10 sign language gestures, with a feedforward network trained by trainlm and a 80/10/10 sectioning.

Number of Neurons																			Nº of
10	20	30	40	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	Features
67.76	83.88	88.22	88.31	89.73	91.14	91.49	92.47	**92.82**	91.06	90.26	91.41	90.43	92.20	91.41	90.88	91.59	90.17	92.03	90.87	20
66.01	83.44	87.33	90.17	90.87	90.43	89.99	90.97	90.43	91.41	90.97	90.70	90.35	91.59	**92.74**	89.46	91.15	91.50	91.23	91.67	18
61.83	82.37	85.29	87.60	89.73	89.64	90.44	90.97	90.61	90.97	89.46	**91.23**	89.46	90.96	90.44	88.84	88.93	90.09	89.99	90.43	16
58.55	79.80	83.17	84.23	85.66	86.90	87.78	88.05	87.96	87.69	88.58	**89.20**	87.69	87.51	88.32	88.23	88.57	89.10	88.67	88.22	14
54.37	73.95	81.85	82.55	83.36	84.95	86.28	85.65	86.54	86.19	87.51	85.92	86.71	86.90	87.51	85.57	88.05	85.75	87.43	**88.49**	12
49.71	68.63	73.68	78.75	78.40	81.40	81.85	82.03	83.35	84.06	82.37	84.41	83.62	**85.38**	84.05	84.95	84.86	84.23	84.95	85.21	10

Table 4.4: Test accuracy for 10 sign language gestures, with a feedforward network trained by traincgb and a 80/10/10 sectioning.

Number of Neurons																			Nº of
10	20	30	40	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	Features
66.28	82.55	89.91	85.83	85.92	89.02	**90.61**	89.20	89.99	89.02	89.46	89.99	88.75	89.90	89.82	89.55	89.81	89.47	89.02	89.64	20
62.11	84.77	82.91	90.18	88.57	**90.44**	89.81	89.64	89.10	89.47	87.96	88.75	89.28	89.29	87.87	88.84	88.75	89.02	90.44	89.90	18
61.65	82.04	86.80	87.52	86.28	86.80	**88.93**	85.38	86.45	86.72	84.68	85.57	86.72	88.75	88.31	85.74	84.07	87.96	84.67	88.31	16
57.93	78.74	83.88	82.81	85.47	84.23	84.33	81.93	84.69	86.27	84.94	85.57	84.86	**87.25**	85.66	86.01	86.63	86.98	85.83	86.89	14
52.43	72.18	82.30	77.85	75.40	78.93	81.40	82.90	82.47	82.46	82.91	81.94	81.85	83.62	85.31	83.27	84.25	83.62	**85.13**	83.79	12
50.94	70.14	71.74	74.77	75.11	76.97	79.72	77.51	78.83	79.36	79.10	79.99	79.54	76.08	80.42	81.50	80.61	79.27	**83.00**	80.17	10

Table 4.5: Test accuracy for 10 sign language gestures, with a feedforward network trained by trainscg and a 80/10/10 sectioning.

The following graphic represents the optimal accuracy in function of the training algorithm and number of features. Each optimal accuracy was determined by the highest test accuracy from each line of the previous tables. In addition, when equal accuracies were detected, a preference was given to the smallest number of hidden neurons in order to decrease the complexity of the system.

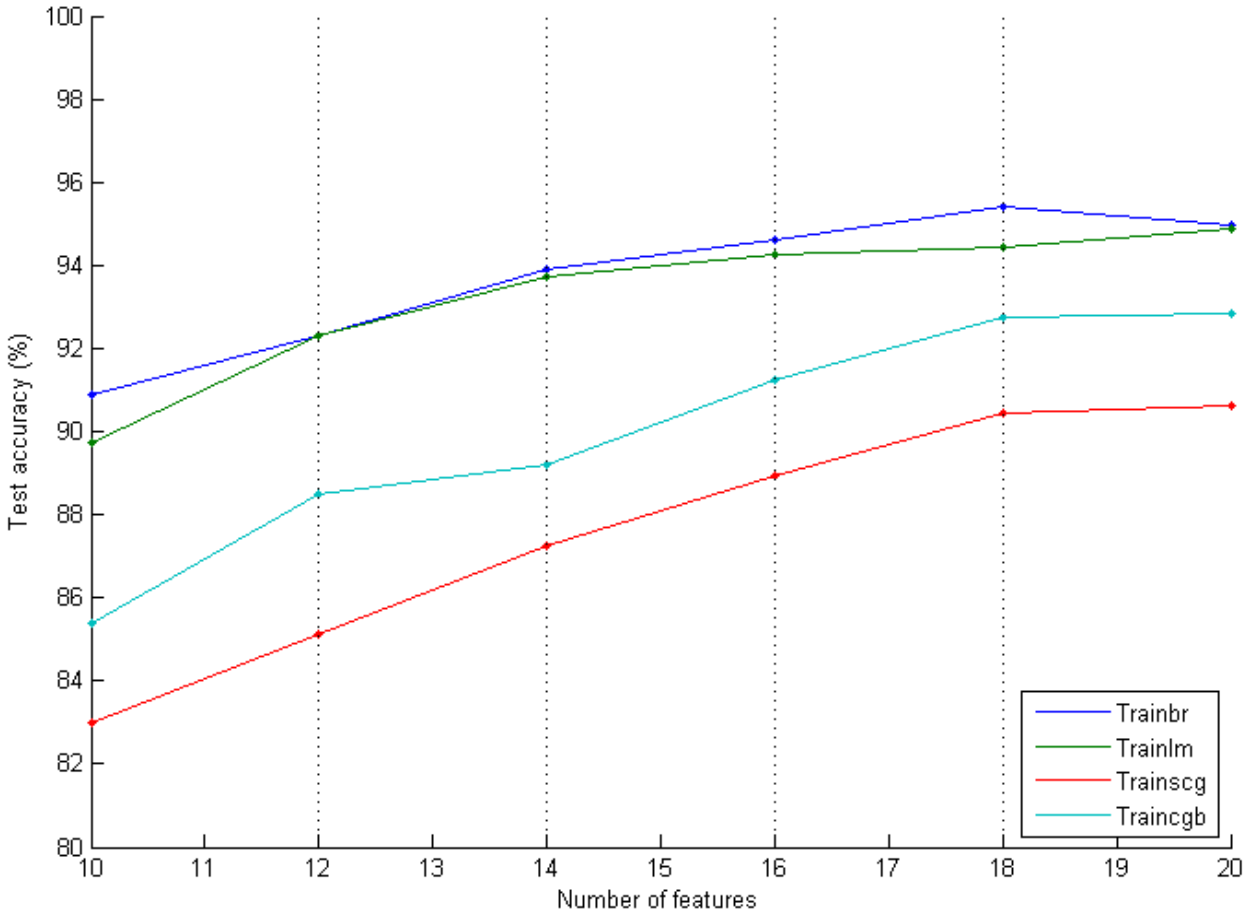

Figure 4.1: Optimal classification accuracy for 10 sign language gesture, in function of the training algorithm and the number of features used to represent the data.

As expected, the training algorithms which lead to higher accuracies are the Levenberg-Marquardt BP algorithm (*trainlm*) and the Bayesian regularization BP algorithm (*trainbr*). Furthermore, the influence of the number of features on the performance of these techniques is very small when compared to the scaled conjugate gradient (*trainscg*) and conjugate gradient with Powell-Beale restarts (*traincgb*) algorithms. While these algorithms achieve lower accuracies, their performance strongly benefits from an increase in the number of features.

Although the use of *trainlm* and *trainbr* results in similar performances, the Bayesian technique achieves slightly higher classification accuracies. This method causes the network to have smaller weights and biases, which forces it's response to be smoother and less likely to overfit. Hence, as this advantage is not offered by the Levenberg-Marquardt technique, Bayesian regularization was chosen as the training method.

As shown by the previous graph, the performance of the system reaches its peak with a data representation composed by 18 features. After this value, the optimal accuracies obtained with the different training algorithms either remain constant or slightly decrease. This behavior is commonly known as the "peaking phenomenon", previously mentioned on Chapter 3.4, and it occurs when the addition new features degrades the system's performance.

Thus, according to the previous results, the number of reduced features chosen to represent the input data was 18. After selecting the two main parameters of the network's architecture, namely the training function and the size of the input data, the next step involved choosing the number of hidden neurons.

Furthermore, it is important to mention that this number was limited to maximum of 200 hidden neurons, in order to keep the network's complexity at a minimum. This is due to the fact that a high number of neurons results in a loss of generalization abilities, and a poor performance in the presence of new inputs patterns. In addition, a potential real-time implementation of the developed pattern recognition system would only benefit from a smaller architecture.

Therefore, taking into account the set of previously selected parameters, the chosen number of hidden neurons was 200. This number resulted in an architecture, which achieved the highest classification accuracy out of all the experimental results (95.4%).

4.1.2 Evaluation of the proposed method

After the determination of the model's parameters, the next step is to evaluate the recognition accuracy of the proposed system. This can be achieved by a brief study of the confusion matrix, a specific table layout where each row represents the instances in a predicted class and each column represents the actual class. Hence, Figure 4.2 represents the confusion matrix, obtained for the classification of 10 sign language gestures.

For this research, the performance of the proposed system was measured by the overall classification accuracy of the 10 tests sets, determined by an adapted 10-fold cross validation

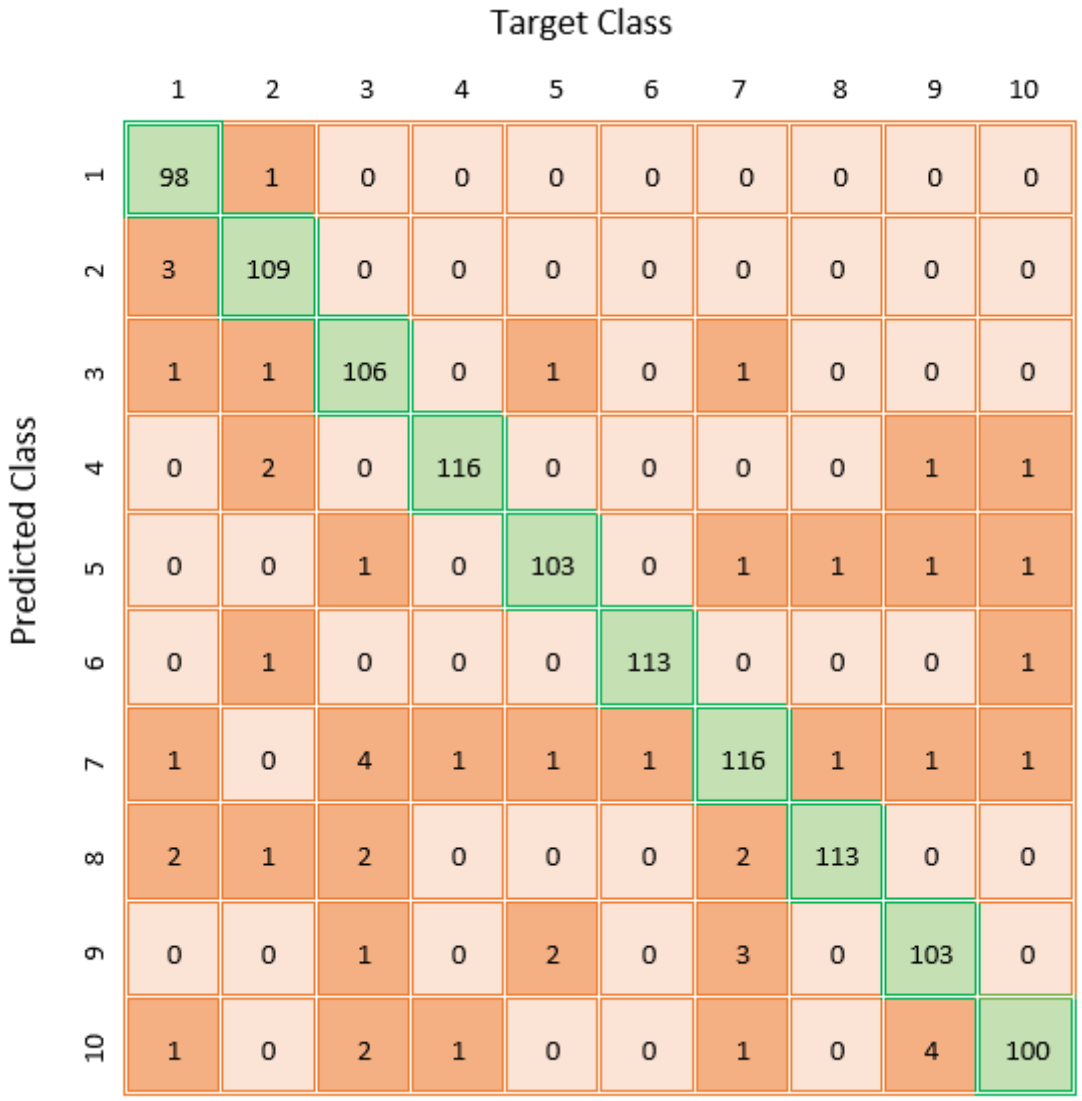

Figure 4.2: Confusion matrix for the classification of 10 sign language gestures. Gestures are numbered in the following order: (1)Yes (2)No (3)Hello (4)Goodbye (5)Thank You (6)Attention (7)Zero (8)One (9)Two (10)Three

algorithm. This accuracy is determined by the values in the diagonal of the confusion matrix, which correspond to correct predictions. Therefore, it can be calculated by the following equation:

$$\eta(\%) = \frac{\text{Number of correct labels}}{\text{Total number of labels}} \times 100 = 95.39\%$$

Moreover, not only does this matrix allow the calculation of the classification accuracy of each gesture, but it also makes it possible to identify specific confusions between any two classes of gestures. However, in this case the misclassification of gestures does not follow a specific pattern, suggestive of similarities between any pairs of gestures.

Table 4.6 represents the classification accuracy of each sign language gestures, where each accuracy is determined by the proportion of correct classifications of the target class:

Yes	No	Hello	Goodbye	Thank you	Attention	Zero	One	Two	Three
92.45%	94.78%	91.38%	98.30%	96.26%	99.12%	93.55%	98.26%	93.63%	96.15%

Table 4.6: Classification accuracy of each sign language gesture, for a sampling population composed by 4 subjects.

As expected, certain gestures are better discriminated than others. For instance, the gestures more prone to misclassification were "Yes" and "Hello". In addition, the highest classification rates were achieved by the signs "Attention" and "Goodbye", a surprising behavior considering they both belong to the class of dynamic gestures.

In order to gain a further understanding on the recognition of these gestures, the developed system was trained and tested for each one of the 4 subjects. The aim of this reduction was to study the effects of inter-individual differences (differences between subjects) in the overall system performance, in order to assess its scalability. Therefore, the following table presents the average of the 4 classification accuracies, obtained at an individual level:

Yes	No	Hello	Goodbye	Thank you	Attention	Zero	One	Two	Three
95.89%	95.92%	98.44%	98.61%	97.45%	97.82%	97.28%	98.6%	97.73%	91.33%

Table 4.7: Average of the individual classifications, in function of the sign language gesture.

In addition, Figure **??** represents the graph bar diagrams formed by the classification accuracies of each gesture, obtained with different sampling populations.

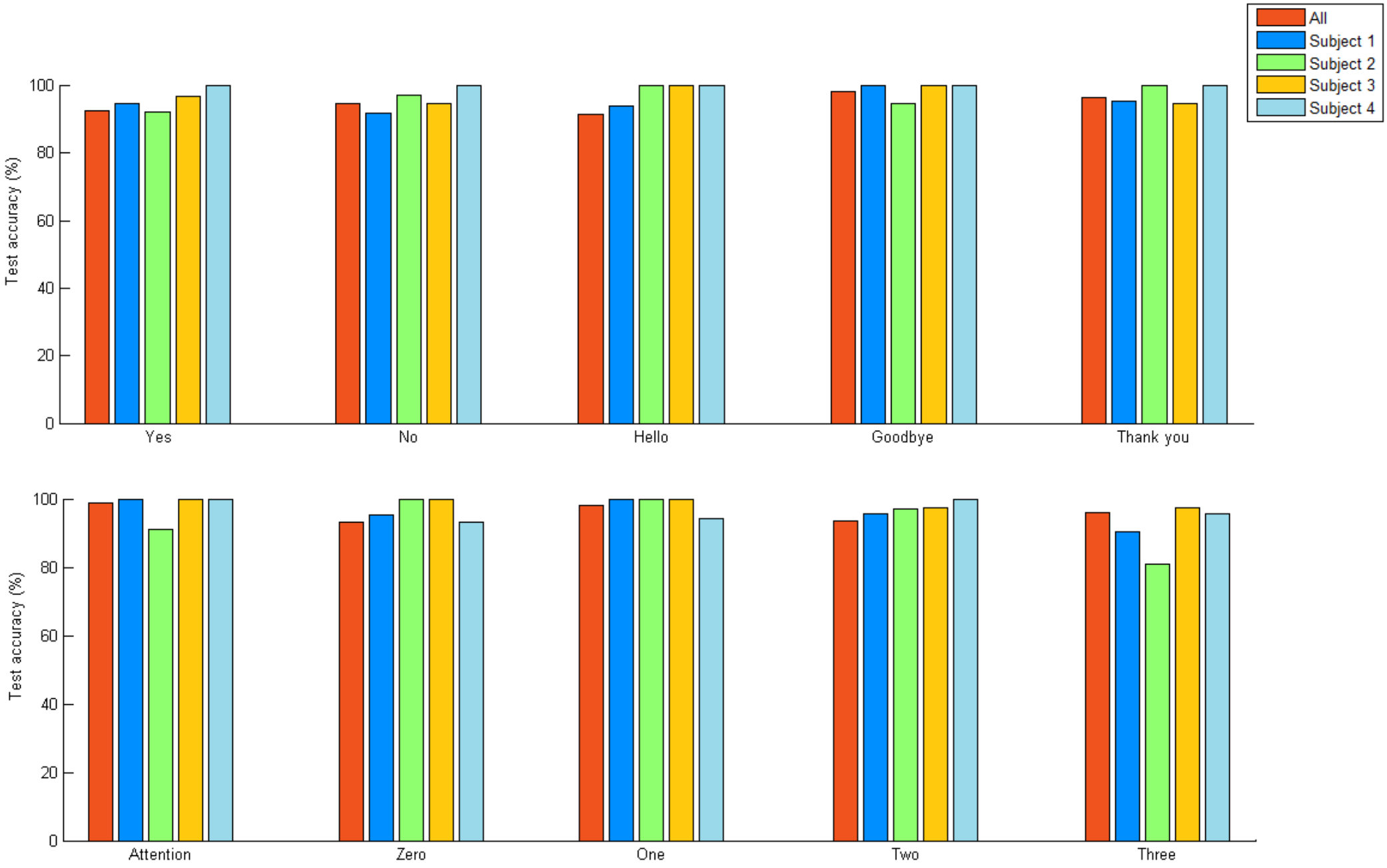

Figure 4.3: Bar graph for the classification of 10 sign language gestures.

According to the results, the raise in the number of samples per class is not enough to compensate the increase of data variability, introduced by a higher number of subjects. Consequently, for the majority of the sign language gestures, the classification accuracy with 4 individuals is lower than the average classification with individual subjects.
As such, the only gestures that benefited with the increase in population where:

- ◇ **Attention** - The classification with 4 individuals is very high, even when compared to the highest classifications achieved at an individual level. More precisely, the misclassification error introduced by subject three (91.27%) is reduced, when the network is trained with more samples of this specific class. This means that the "Attention" sign is less prone to inter-subject differences, which can be explained by the fact that it is based on a strong pointing movement which is easy to replicate.

- ◇ **Three** - In this case, the accuracy achieved with 4 subjects is higher than the average individual classification. Therefore, one can conclude that the classification of samples from subject 1 and 2 has considerably improved due to the increase in population. This gesture is characterized by a configuration which resembles a closed fist, which might explain its low inter-individual variability.

Furthermore, the gestures which were not affected by the sampling population where:

- ◇ **Goodbye** - This gesture renders high classification accuracies for a majority of the sampling populations. In addition, the increase in population has no impact on the classification of samples from subject 2 and therefore the average individual classification is quite similar to the one achieved with 4 subjects.

- ◇ **One** - The same behavior is demonstrated by this gesture, where the classification for 4 subjects is lowered by the introduction of gestures concerning a specific subject.

Finally, the gesture which was most affected by inter-individual variability was the sign for **Hello**. As shown by the bar graphs, the accuracy achieved with 4 subjects is lower than any classification obtained at an individual level. This suggests that further increases in population might lead to a decrease in the ability to discriminate this type of gesture. However, this effect might be counter-acted by an increase in the number of training samples of this type of gesture.

4.2 Classification of movements

Although the development of the aforementioned pattern recognition system was directed towards the recognition of sign language gestures, this tool was also applied to the classification of arm and hand movements in an initial stage. Thus, an interesting way to evaluate the range of the developed signal recognition system is to apply it to the classification of movements. It is important to note that while this classification was a useful tool for building the methodology on which this system is based, several parameters were altered to allow the recognition of sign language gestures, namely the size of the sampling window, the thresholds of muscle contraction detection, among others.

For the first stage of data acquisition 4 able bodied subjects were asked to perform 42 repetitions of 6 simple movements. However, due to the presence of high amounts of noise, several EMG samples were discarded and each subject was required to perform additional measurements. Therefore, the language recognition system was applied to 1000 valid samples and the following classification accuracies were obtained:

Number of Neurons									
10	20	30	40	50	60	70	80	90	100
83.00%	90.33%	90.78%	90.11%	92.67%	91.67%	92.44%	91.33%	91.44%	90.67%

Number of Neurons									
110	120	130	140	150	160	170	180	190	200
91.44%	92.11%	91.78%	92.44%	91.56%	91.56%	91.33%	91.67%	92.33%	91.22%

Table 4.8: Test accuracy for 6 movements, with a feedforward network trained by trainbr, a 80/10/10 sectioning and an 18-feature data representation.

The previous results suggest that the pattern recognition system is not necessarily limited to the classification of sign gestures. Moreover, when the right number of hidden neurons is selected, this system can classify movements with an accuracy rate of 92.7%. Due to the simplicity of these signals and the reduction in the number of classes (from 10 to 6), this value might seem low when compared to the one achieved in sign gesture classification (95.4%). This can be attributed to damaged extraction material, namely the sensor cables, which introduced a considerable amount of noise in the input data. Consequently, at the end of the movement's measuring sessions, up to 3 EMG sensor cables had been replaced.

CHAPTER 5

Conclusion

Contents

On the basis of the results of this research, it can be concluded that the proposed hand gesture recognition system is capable of correctly classifying a range of sign language gestures, which encompasses both static and dynamic signals. Moreover, the data supports the use of an ANN as a classification method for sign language recognition. This method has provided a classification accuracy of 95.4% for 10 sign language signals, produced by four different subjects. These results were based on a small subset of gestures and as such the data does not fairly reflect the reliability of this system as a real world application. However, as long as the training procedure is extended the system should be applicable to a wider range of vocabulary.

In this case, the system performs similarly for static and dynamic gestures, a behavior which was not expected as the temporal relations contained in dynamic signals makes these signals more difficult to classify. On the other hand, the system's performance is highly influenced by the repeatability of a gesture, namely if it can be replicated in a similar manner by different individuals, in separate instances. Consequently, when the size of the population increases the classification accuracy of specific gestures decreases considerably. In the worst case, the classification accuracy of the signal "Hello" decreases 7.1% when the population size increases from 1 to 4 subjects. This behavior is due to inter-individual differences, which can result of kinematic differences, the gesture's trajectory, or kinetic differences, when subjects exert different amounts of force. Additionally, there are other sources of subject specific variability such as the amount of subcutaneous fat, muscle fiber composition, amount of hair, among others [9]. Due to these factors, the system's performance is hindered when the sampling population increases, however this effect might be counter-acted with a corresponding increase in the number of training samples per subject.

While the classification of specific gestures is hampered by an increase in the sampling population, other gestures benefit from the consequent increase in training samples. As such, the scalability of the system can not be determined at such a small scale, further studies are required to determine the effects of a diverse population on the system's discrimination abilities. In this case, it is important to compensate these individual differences in order to provide a consistent classification rate independently of the user.

Another aspect of the developed system is its versatility. Although the aforementioned system was developed towards SLR it is also suitable for motion classification. According to the experimental data, the system achieved an accuracy rate of 92.7% for the classification of 6 arm/hand movements, with a simple readjustment of the networks size (number of

hidden neurons) to account for the reduction in input data and classes. This accuracy might seem low, taking into consideration the reduction in the number of classes (10 sign language gestures to 6 movements), and in the complexity of the input signals. This can be attributed to the physical condition of the EMG sensor cables, which introduced high amounts of noise during the first stage of data acquisition, concerning simple movements. Moreover, during data extraction some cables had to be replaced, which introduced a high degree of variability in the arm and hand movements data. This problem was fixed for the second stage of data acquisition, concerning sign language gestures.

5.1 Future Work

The biggest limitation of the developed system is its isolated nature, namely the fact that each gesture needs to be fed separately to be recognized. True human gestures are continuous and as such a continuous recognition system is highly desirable in communication practices between the deaf community. Therefore, the next step of development is to extend the scope of the proposed system, and to create a practical continuous recognition system, useful to the hearing impaired community.

Another point of improvement concerns the system's limited range of vocabulary and small sampling population. Further data extracting sessions are required, in order to increase the number of recognizable sign language gestures, and to determine the system's scalability. In this case, the biggest challenge will be the reproducibility of the EMG measurements, which can cause a big drop in the system's performance. A possible solution is an approach based on combined sensing, namely the combination of Accelerometer (ACC) and EMG sensors. Several studies indicate that this approach can improve the performance of hand gesture recognition significantly, and as such this is a technique that could be explored in future researches [8].

Bibliography

[1] D. Kriesel, *A Brief Introduction to Neural Networks*, zeta2 ed., 2007. [Online]. Available: http://www.dkriesel.com/science/neural_networks

[2] P. Konrad, "The abc of emg," *A practical introduction to kinesiological electromyography*, vol. 1, 2006.

[3] N. P. Hamilton, *Kinesiology: Scientific basis of human motion (12nd Edition)*. Brown & Benchmark, 2011.

[4] A. Filipe Silva Ferreira, "Robot control based on biosignals," Master's thesis, FCTUC, Coimbra, Portugal, 2014.

[5] *Electromyography (EMG) Sensor Data Sheet*, PLUX®, Lisbon, Portugal, October 2015. [Online]. Available: http://www.bitalino.com/datasheets/REVOLUTION_EMG_Sensor_Datasheet.pdf

[6] S. Mitra and T. Acharya, "Gesture Recognition: A Survey," *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 3, May 2007. [Online]. Available: http://dx.doi.org/10.1109/tsmcc.2007.893280

[7] E. V. LGP, "Introdução à língua gestual portuguesa," http://www.lgpescolavirtual.pt.

[8] X. Zhang, X. Chen, Y. Li, V. Lantz, K. Wang, and J. Yang, "A framework for hand gesture recognition based on accelerometer and emg sensors," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 41, no. 6, 2011.

[9] A.-A. Samadani and D. Kulic, "Hand gesture recognition based on surface electromyography," in *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*. IEEE, 2014, pp. 4196–4199.

[10] M. Ebrahim and N. Tahi, "Review in sign language recognition systems," in *2012 IEEE Symposium on Computers and Informatics (ISCI)*.

[11] R. Bowden, A. Zisserman, D. Windridge, T. Kadir, and M. Brady, "Vision based interpretation of natural sign languages," in *3rd International Conference on Computer Vision Systems*, Graz. Austria, April 2003.

[12] A. P. de Surdos, "Informação-língua gestual," http://www.apsurdos.org.pt.

[13] C. Miranda, H. Freitas, P. Duarte, and M. Manuel, *Jornal da Erebas, Escola de Referência para a Educação Bilingue de Alunos Surdos*, vol. 4, December 2016.

[14] R. Ahsan, M. Ibrahimy, and K. Othman, "EMG Signal Classification for Human Computer Interaction: A Review," *European Journal of Scientific Research*, vol. 33, no. 3, 2009.

[15] J. Bronzino, *The Biomedical Engineering Handbook*, 3rd ed., ser. Electrical Engineering Handbook. Taylor and Francis, 2006.

[16] W. Putnam and R. B. Knapp, "Real-time computer control using pattern recognition of the electromyogram," in *Engineering in Medicine and Biology Society, 1993. Proceedings of the 15th Annual International Conference of the IEEE.* IEEE, 1993, pp. 1236–1237.

[17] G. Tsenov, A. Zeghbib, F. Palis, N. Shoylev, and V. Mladenov, "Neural networks for online classification of hand and finger movements using surface emg signals," in *Neural Network Applications in Electrical Engineering, 2006. NEUREL 2006. 8th Seminar on*. IEEE, 2006, pp. 167–171.

[18] P. Hong, M. Turk, and T. S. Huang, "Constructing finite state machines for fast gesture recognition," in *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, vol. 3. IEEE, 2000, pp. 691–694.

[19] Z. Chen *et al.*, "Bayesian filtering: From kalman filters to particle filters, and beyond," *Statistics*, vol. 182, no. 1, pp. 1–69, 2003.

[20] J. Yang and Y. Xu, "Hidden markov model for gesture recognition," DTIC Document, Tech. Rep., 1994.

[21] C. Gershenson, "Artificial neural networks for beginners," *arXiv preprint cs/0308031*, 2003.

[22] J. V. Basmajian and C. J. De Luca, *Muscles alive: their functions revealed by electromyography.* Williams & Wilkins, 1985.

[23] G. Pocock and C. Richards, *Human Physiology-The Basis of Medicine*, 3rd ed. Oxford: Oxford University Press, 2006.

[24] L. Guilherme Nunes Marques, "Decoding arm movements through electromyography," Master's thesis, FCTUC, Coimbra, Portugal, 2012.

[25] C. De Luca, "Encyclopedia of medical devices and instrumentation," Boston, Massachusetts, pp. 98–109, 2006.

[26] L. Wang, "Task load modelling for lte baseband signal processing with artificial neural network approach," Master's thesis, KTH Electrical Engineering, Stockholm, Sweden, 2014.

[27] *OpenSignals (r)evolution User Manual*, PLUX®, Lisbon, Portugal, April 2015. [Online]. Available: http://www.http://bitalino.com/downloads/int-releases/OpenSignals_(r)evolution_Manual.pdf

[28] W. Rose, "Electromyogram analysis," *Mathematics and Signal Processing for Biomechanics. University of Delaware. Retrieved July*, vol. 5, 2014.

[29] S. Solnik, P. Rider, K. Steinweg, P. DeVita, and T. Hortobágyi, "Teager–kaiser energy operator signal conditioning improves emg onset detection," *European journal of applied physiology*, vol. 110, no. 3, pp. 489–498, 2010.

[30] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2nd Edition)*. Wiley-Interscience, November 2000.

[31] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for emg signal classification," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7420–7431, 2012.

[32] K. Vos, "A fast implementation of burg's method," *OPUS codec*, 2013.

[33] D. Tkach, H. Huang, and T. A. Kuiken, "Study of stability of time-domain features for electromyographic pattern recognition," *Journal of neuroengineering and rehabilitation*, vol. 7, no. 1, p. 21, 2010.

[34] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 4–37, Jan. 2000. [Online]. Available: http://dx.doi.org/10.1109/34.824819

[35] B. M. Wilamowski and J. D. Irwin, *Intelligent Systems*, 2nd ed. CRC, 2011.

[36] M. F. Møller, "A scaled conjugate gradient algorithm for fast supervised learning," *Neural networks*, vol. 6, no. 4, pp. 525–533, 1993.

[37] M. J. D. Powell, "Restart procedures for the conjugate gradient method," *Mathematical programming*, vol. 12, no. 1, pp. 241–254, 1977.

[38] M. Kayri, "Predictive abilities of bayesian regularization and levenberg–marquardt algorithms in artificial neural networks: A comparative empirical study on social data," *Mathematical and Computational Applications*, vol. 21, no. 2, p. 20, 2016.

[39] M. Arvetti, G. Gini, and M. Folgheraiter, "Classification of emg signals through wavelet analysis and neural networks for controlling an active hand prosthesis," in *Rehabilitation Robotics, 2007. ICORR 2007. IEEE 10th International Conference on.* IEEE, 2007, pp. 531–536.

[40] Z. Reitermanov, "Data splitting," in *WDS Proceedings of Contributed Papers*, vol. 10, 2010, pp. 31–36.

[41] K. L. Priddy and P. E. Keller, *Artificial neural networks: an introduction.* SPIE press, 2005, vol. 68.