POLITECNICO DI TORINO

Dipartimento di Ingegneria Meccanica e Aerospaziale

Master's Degree in Biomedical Engineering

Master's Degree Thesis

Integration and Validation of an Event-driven sEMG-based Embedded Prototype for Real-time Facial Expression Recognition

Supervisors

Prof. Danilo DEMARCHI

Ph.D. Fabio ROSSI

M.Sc. Andrea MONGARDI

Candidate

Luca GIANNANTONI

Torino, March 2023

Abstract

Facial Expression Recognition has demonstrated significant potential in biomedical research area: evaluation of emotional well-being, support in non-verbal communication, control of Human-Machine Interfaces (HMI) and assistance in rehabilitative procedures. While computer vision is currently the dominant approach for facial expression recognition, recent research has shown increasing surface ElectroMyoGraphy (sEMG) use. sEMG is a non-invasive technique to acquire the electrical signals generated by skeletal muscles during contraction by applying non-invasive electrodes on the skin. Several parameters can be extracted from sEMG signals, obtaining accurate and direct measures of muscle activity suitable to digital processing, e.g. machine learning (ML).

This thesis presents an implementation of an embedded prototype for facial expression recognition based on the Averaged Threshold Crossing (ATC) technique applied to facial sEMG signals. ATC is an event-driven feature extraction technique in which an event is generated every time the sEMG signal exceeds a threshold. The ATC value is the average of the TC events over a time window (i.e., 130 ms) and has been proven to be an index for muscle activation level. The event-driven approach significantly reduces the amount of data contained in sEMG signals and the complexity of the processing, thus making it well-suited for power-constrained and real-time applications.

The central processing unit is an Apollo3 Blue Micro-Controller Unit (MCU) with an ARM Cortex-M4 processor. Five acquisition boards acquire and filter the sEMG signals to extract the ATC. A Bluetooth Low Energy (BLE) module integrated into the MCU and a USB-dongle connected to an external device provide BLE connectivity for wireless data transmission. A control software handles all data streams and has a Graphical User Interface (GUI) on its highest layer. The GUI eases user control during the development, training, and testing phases.

Preliminary tests were conducted to establish the expressions to be recognized, the corresponding muscles and the appropriate electrode positioning, integrating the information from previous works. For the dataset creation, 24 healthy subjects were

recruited. Each participant performed a list of 11 gestures for 6 repeated sessions. As an ML classification algorithm, a fully connected Artificial Neural network (ANN) was chosen to fit the MCU limited hardware resources. This is the most straightforward ANN architecture and can be easily implemented with optimized ARM libraries. The prediction process only uses matrix multiplication requiring minimal computational overhead. Different ANNs were trained and validated using k-fold cross-validation. The best performing ANN (1 hidden layer, 42 nodes) was then exported to the MCU for testing. The system was tested on 6 additional healthy subjects reaching 97.4% average accuracy.

During active prediction functioning, power consumption measurements on the MCU showed a 0.582 mA mean current absorption. The average prediction latency of the classifier was measured 0.627 ms. The maximum application latency is 205.627 ms, considering the ATC window contribution (130 ms), the maximum BLE connection interval (75 ms) and the prediction latency (0.627 ms).

The low current absorption, the application latency (< 300 ms) and the classification accuracy shown by this approach meet the requirements for a future realization of a wearable system for real-time facial expression recognition.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Human Machine Interfaces (HMI) have recently emphasized the significance of voice, facial expression, and posture as important interfaces, leading to facial expression recognition and analysis emerging as important areas of biomedical research [1]. Although significant progress has been made, recognizing facial expressions remains challenging due to their subtlety, complexity, and variability. Computer vision-based methods are commonly used but suffer limitations such as dependence on image quality and vulnerability to factors like camera angle, background, and lighting. In addition, most research has focused on surface appearance rather than the underlying muscular changes [2]. On the other hand, physiological signals, including the galvanic skin response, heart rate, electroencephalography, and ElectroMyoGraphy (EMG), have gained increasing attention for emotion and expression recognition in the HMI field. Surface ElectroMyoGraphy (sEMG), which reflects neuromuscular activity, has been found useful for recognizing facial expressions, despite being mainly used for recognizing gestures, sign languages, and movements of upper and lower limbs [3].

This thesis presents an implementation of a prototype for real-time facial expression recognition based on the Averaged Threshold Crossing (ATC) technique applied to facial sEMG signals. The ATC technique is an event-driven feature extraction technique in which an event is generated every time the sEMG signal exceeds a threshold, with the ATC value being the average of the Threshold Crossing (TC) events over a time window (i.e., 130 ms) and proven to be an index for muscle activation level. The event-driven approach significantly reduces the processing complexity, making it suitable for power-constrained and real-time applications [4]. The system's central processing unit is a Micro-Controller Unit (MCU). Five acquisition boards acquire and filter the sEMG signals to extract the ATC. A Bluetooth Low Power (BLE) module integrated into the MCU and a USB dongle connected to an external device provide wireless connectivity for data transmission. A control software handles all data streams and allows easy user control thanks to a Graphical User Interface (GUI).

Chapter 1 provides readers with background information about the muscular system, specifically the facial muscles. Then the sEMG signals acquisition and feature extraction for facial expression recognition are introduced. The chapter also focuses on the ATC technique and the current state of the art in facial expression recognition using sEMG.

Chapter 2 details the system configuration, focusing on the MCU, its firmware, and the five acquisition channels used to extract the ATC signal from sEMG. The chapter also describes how the system can be controlled via a GUI thanks to a wireless communication system based on BLE.

Chapter 3 dives into the process of choosing facial expressions and then focuses on the electrode positioning strategy to acquire sEMG signals and extract the ATC parameter from target muscles. The chapter then defines the acquisition protocol for constructing a dataset for training a Machine Learning (ML) model for multiclass classification and provides information about the dataset.

Chapter 4 outlines the Artificial Neural Network (ANN) algorithm used in this study, the learning algorithm used for offline training, and the process of selecting the most efficient ANN structure for the present application. It also explains how the selected model can be implemented on the MCU system for online predictions.

Chapter 5 describes how the system was tested for real-time usage, presents and discusses the ANN classification results, reports on the MCU measured power consumption in different functional modes, and the time latency of the system.

Finally, chapter 6 provides thesis conclusions and suggests possible future developments.

1.1 Muscular system

The human skeletal system provides support and stability to the body. This stability is maintained through the action of the skeletal muscles, which are controlled by the nervous system. The production of force by the skeletal muscles is known as muscular force, which is critical in ensuring that the skeleton remains stable under various conditions. The intensity of muscular force generated by a particular muscle depends on several factors, including the muscle's structure. In particular, the cross-sectional area of the muscle and the pennation angle play a significant role in determining the amount of force generated during a specific movement. The cross-sectional area refers to the thickness of the muscle and its ability to generate force. The pennation angle refers to the angle at which the muscle fibers are aligned relative to the bone [5].

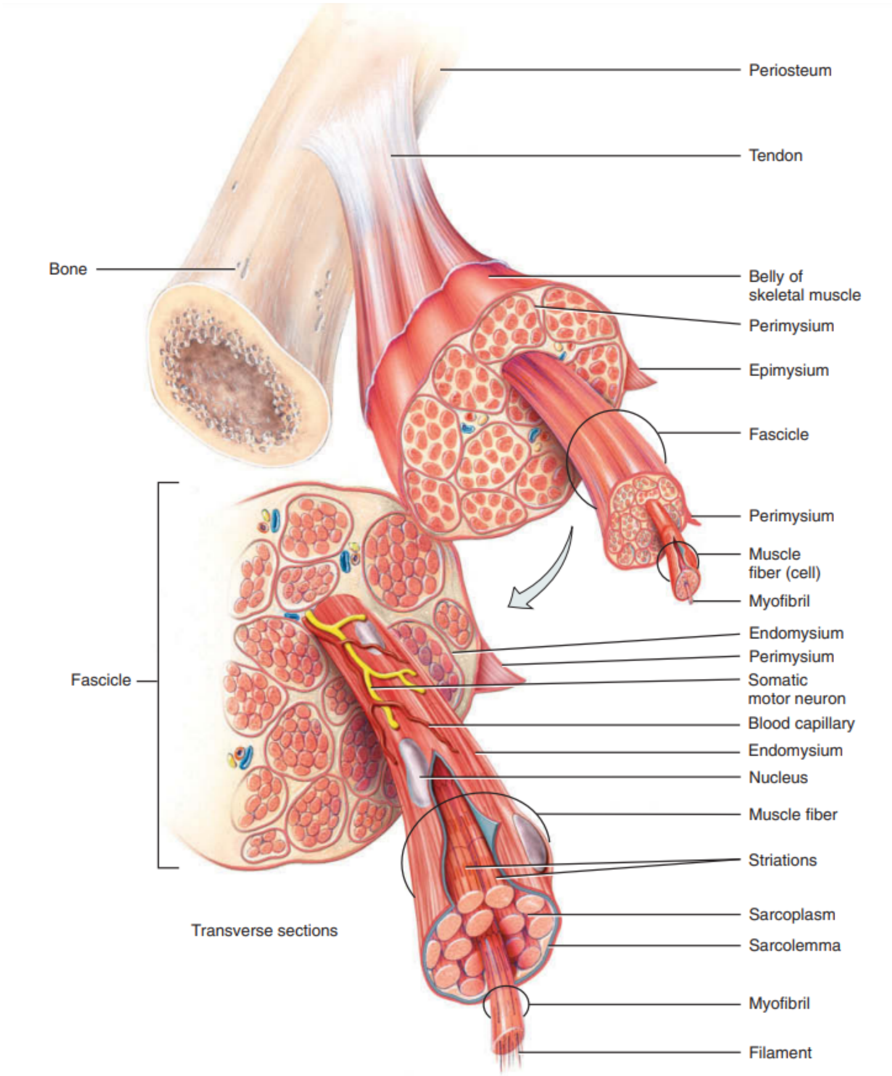


Figure 1.1: Components of skeletal muscles. Transverse section of the muscle with detailed fascicle structure [6].

1.1.1 Skeletal muscle physiology

Skeletal muscles are connected to the skeleton and allow for voluntary movements. Unlike other types of muscles, such as cardiac or smooth muscles, skeletal muscles are under conscious control. Their structure is complex and multi-layered. At the

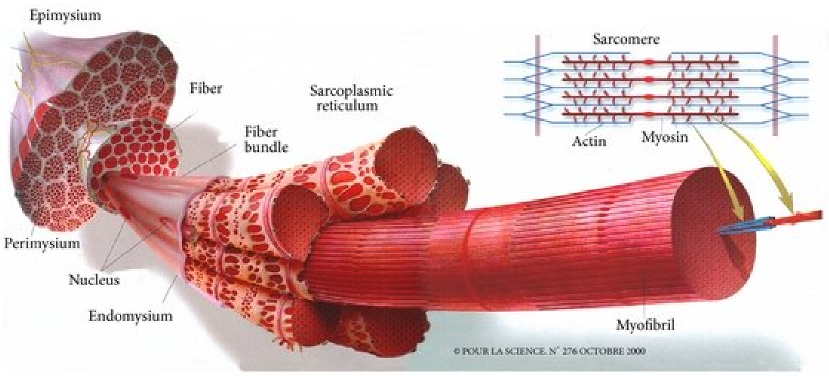


Figure 1.2: Skeletal muscle organization, from the epimysium external layer to the inner microscale components [7].

macroscopic level, these muscles are attached to bones through tendons, which transmit the power generated by the muscles to allow for movement. Various components of the muscle structure are identified, moving from the outside toward the inner part of the muscle, as depicted in Fig. 1.1 and Fig. 1.2. The bundles of muscle fibers, known as muscle fascicles, are safeguarded by the epimysium, a protective connective membrane that attaches the entire muscle to the tendons. Further dividing the fascicles is the perimysium membrane, which encompasses individual muscle fibers. The innermost membrane, or endomysium, safeguards the fibers and contains the necessary nutrients and extracellular fluid for survival. Each fiber contains multinuclear cells, sarcolemma, and sarcoplasm. The sarcoplasm produces energy through the mitochondria, transforming the substances from the circulatory system into Adenosine TriPhosphate (ATP). The number of mitochondria within the fiber depends on its function in the body. The fibers consist of myofibrils, parallel structures composed of sarcomeres arranged serially, and are considered the fundamental functional units of the skeletal muscle. The sarcomeres comprise thick and thin filaments that form overlapping parallel matrices of myosin and actin proteins [7]. These filaments can be organized into five different zones, which are essential to muscle contraction:

- M-line: it is located at the center of the sarcomere and consists of proteins that connect the thick filaments in the middle of the sarcomere. It serves as an anchor point for these filaments and helps maintain the sarcomere's overall structure.

- H-zone: it is located on either side of the M-line and consists of thick filaments only. During muscle contraction, the H-zone becomes shorter as the thick

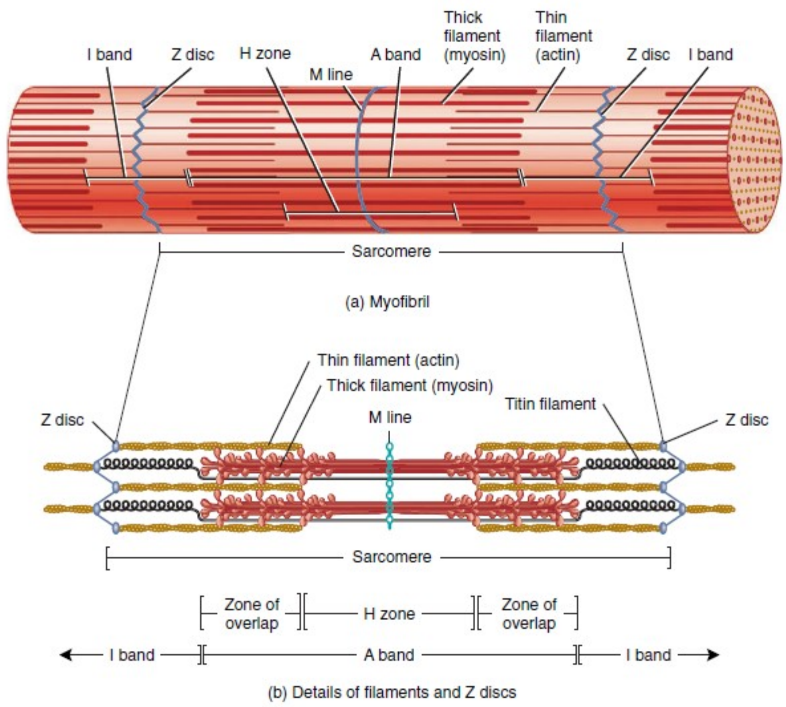


Figure 1.3: Sarcomere: basic functional units of the skeletal muscle. Thick and thin filaments composed of myosin and actin proteins arranged in overlapping parallel matrices and five band zones organization in evidence [6].

filaments slide past the thin filaments.

- A-band: it is located on either side of the H-zone and extends to the ends of the thick filaments. It consists of thick and thin filaments and remains the same length during muscle contraction.
- I-band: it is located on either side of the A-band and consists of only thin filaments. During muscle contraction, the I-band becomes shorter as the thin filaments are pulled toward the center of the sarcomere.
- Z-line: it marks the boundary between adjacent sarcomeres and is composed of proteins that attach to the thin filaments. It helps maintain the filaments' alignment and provides stability to the overall structure of the muscle.

Understanding the different zones of the sarcomere is essential for understanding how muscles contract and generate force. Figure 1.3 presents a detailed representation of the sarcomere structure. By sliding past one another, the thick and thin filaments in the sarcomere cause the muscle to shorten and generate force [6].

1.1.2 Skeletal muscle contraction

Muscular contraction is a complex physiological process involving interaction between the Central Nervous System (CNS), motor neurons, and muscle fibers. The process of muscular contraction can be divided into several stages involving electrical and chemical mechanisms [8]. The CNS manages the control of muscle fiber activation through the transmission of signals along motor neurons. As shown in Fig. 1.4, the α-motoneurons in the spinal cord receive instructions from the CNS and relay activation commands to clusters of muscular fibers using their axons. This collection of muscle fibers innervated by a single motor neuron is known as a Motor Unit (MU).

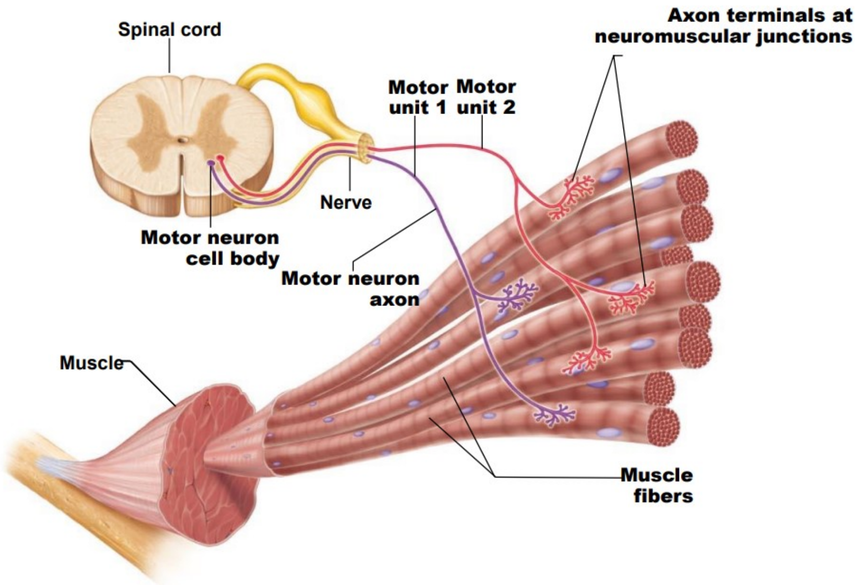


Figure 1.4: Motor Unit anatomy. Each MU consists of a collection of muscle fibers innervated by a single motor neuron [9].

The MU undergoes a cyclical process of polarization and depolarization. Without conduction, the sodium-potassium pump, a specialized protein that regulates ion transport across the axon membrane, maintains the transmembrane differential voltage, also known as the resting potential, at $-70\,\text{mV}$. This protein facilitates the efflux of sodium ions (Na^+) and the influx of potassium ions (K^+). An Action Potential (AP) is generated when a neuron is stimulated. The AP travels along the axon and triggers the opening of sodium channels, causing an influx of Na^+ ions and increasing the membrane potential to $40\,\text{mV}$. As a result, potassium channels are opened, leading to the efflux of K^+ ions and the repolarization of the membrane. The potential is then reduced back to its resting value of $-70\,\text{mV}$. During this process, neurotransmitters stored in the neuron are released and can travel to the muscle fibers. After a brief interval, the protein that enables the transmission blocks the pathway, concluding the process. Upon reaching the terminal end of the axon, the neurotransmitters directly stimulate the fibers innervated by the neuron, which results in an instantaneous contraction. [10].

MUs are organized hierarchically, with smaller motor units with fewer muscle fibers and larger motor units with more muscle fibers. In general, smaller Motor Units are recruited first during muscle contractions, as they have a lower threshold for activation. This means that they require less stimulation to contract and generate force. As the force requirement of the movement increases, larger Motor Units are recruited, leading to more muscle fiber activations and larger overall force output of the muscle. The type of movement being performed also influences MUs recruitment. For example, slow, sustained movements such as holding a weight in a static position may recruit smaller, slower-twitch motor units to maintain the force output over a more extended period. On the other hand, rapid, explosive movements such as jumping or sprinting may require the recruitment of larger, fast-twitch motor units to generate a high-force output quickly [11].

1.1.3 Head and Neck muscles

In this study, the focus is on the electrical signals (described in Sec. 1.2) generated by facial and neck muscles during the execution of specific facial expressions. A good knowledge of these muscles is necessary to plan how to acquire the signals and to understand the results obtained. The muscles involved in jaw movements and facial expressions, illustrated in Fig. 1.5 and Fig. 1.6, are described in the following.

Facial muscles

The muscles accountable for facial expression have a thin and variable structure, often merging with adjacent muscles. Unlike other skeletal muscles, these muscles attach to the skin rather than the skeleton. A concise but not comprehensive overview of the facial muscles involved in creating facial expressions is presented in the following [12, 9]:

- **Orbicularis oculi** is a thin sphincter muscle situated around the eyelids, encompassing the edge of the orbit. Its primary function is to close the eyes, as observed in activities like blinking and squinting.
- **Orbicularis oris** is a multi-layered muscle that surrounds the oral cavity and comprises concentric fiber bands that run in different directions, primarily circular. This muscle allows for the closing and puckering of the lips during actions such as kissing, whistling, and sucking.

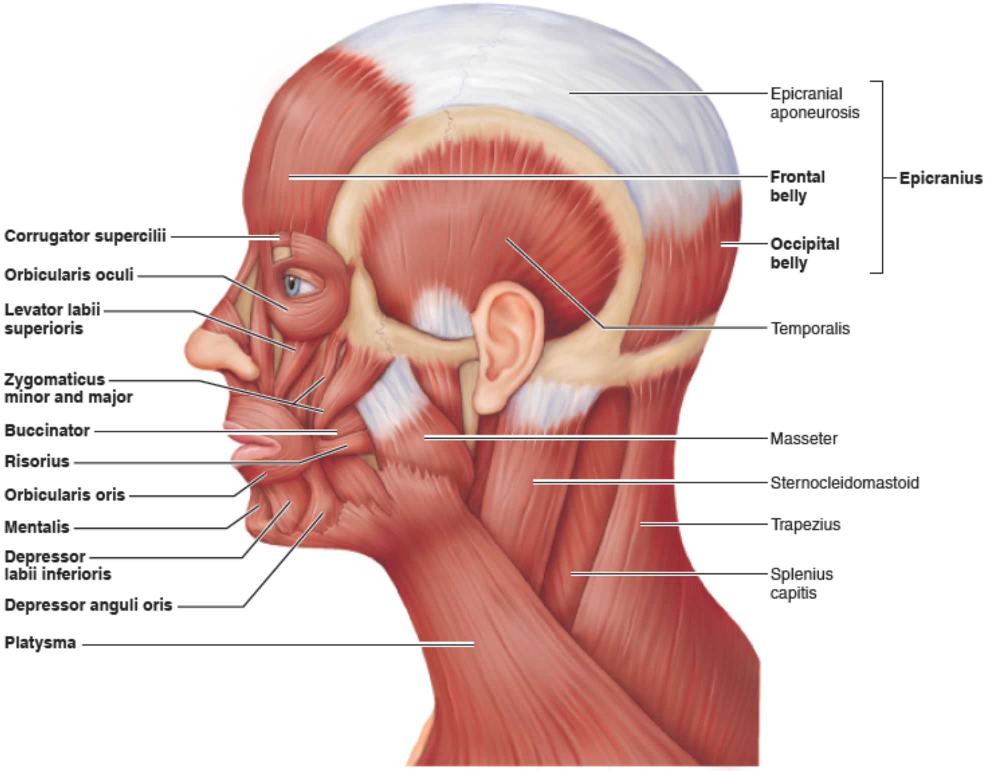


Figure 1.5: Superficial muscles of the head and neck [9].

- **Corrugator supercilii** is a small, pyramidal-shaped facial muscle located at the medial end of the eyebrow. This muscle and the *orbicularis oris* move the eyebrows inferiorly and medially towards the nose and inner corner of the eye, resulting in vertical wrinkles on the forehead.

- **Zygomaticus** includes a pair of muscles that run obliquely from the zygomatic bone to the mouth corner. They are responsible for elevating the corners of the mouth upwards and outwards and contributing to the formation of dynamic wrinkles in the cheek and nasolabial area. These muscles play a significant role in smiling, facial expressions involving the mouth, and in the non-verbal communication of emotions.

- **Epicranius**, also known as the *occipitofrontalis* muscle, is a bipartite muscle consisting of two parts: the frontal belly and the occipital belly. These two parts are joined by the epicranial aponeurosis and work together to alternate the scalp pulling forward and backward. The *epicranius* muscle is responsible for movements such as raising the eyebrows and wrinkling the forehead and plays a role in facial expressions and non-verbal communication.

- **Mentalis** muscle is a small muscle located at the tip of the chin. It is responsible for movements of the inferior lip and activates to push the lower lip forward. This action results in the creation of wrinkles on the chin. The *mentalis* muscle plays a role in facial expressions and non-verbal communication and involves movements such as puckering the lips and protruding the lower jaw.

- **Platysma** is a superficial muscle of the anterior neck region. When activated, it contributes to lowering the jaw and mouth corners, leading to tightening skin in the mandibular region and the anterior neck. This muscle is involved in facial expressions such as smiling, frowning, and talking and is an essential component of the dynamic anatomy of the face and neck.

- **Risorius** is situated in the lower face region, laterally beneath the *zygomaticus*. It can retract the corners of the mouth and is involved in smiling, laughter, and grimacing, but it does not produce wrinkles around the eyes. This helps differentiate the type of smile because unlike the smile resulting from the action of the *zygomaticus*, the *risorius* generated smile does not involve wrinkles around the eyes.

- **Levator labii superioris** is involved in the movement of the upper lip. It originates from the infraorbital ridge and inserts into the upper lip's skin and *orbicularis oris* muscle. The muscle is responsible for elevating and furrowing the upper lip and is involved in various facial expressions, such as smiling, frowning, and sneering.

- **Depressor labii inferioris** is located in the lower region of the face, beneath the lips. It originates from the jaw, extending to the inferior lip. It primarily contributes to lowering the lip. Its action results in pouting and can create wrinkles and folds in the skin around the mouth.

- **Depressor anguli oris** is a facial muscle responsible for the downward movement of the corner of the mouth. It works in opposition to *zygomatic* muscles, which raise the mouth's corners and is activated when making expressions such as a frown or a scowl.

- **Buccinator** muscle, located in the cheek region, is a thin and horizontally oriented muscle responsible for pushing the cheek towards the molars. It helps hold food in the mouth between the molars while chewing and helps in blowing by contracting bilaterally. It is part of the masticatory muscles and helps keep food out of the oral cavity while chewing.

Jaw movements muscles

The following muscles work together for movements such as biting, chewing, jaw opening and closing, and tongue movements [9]:

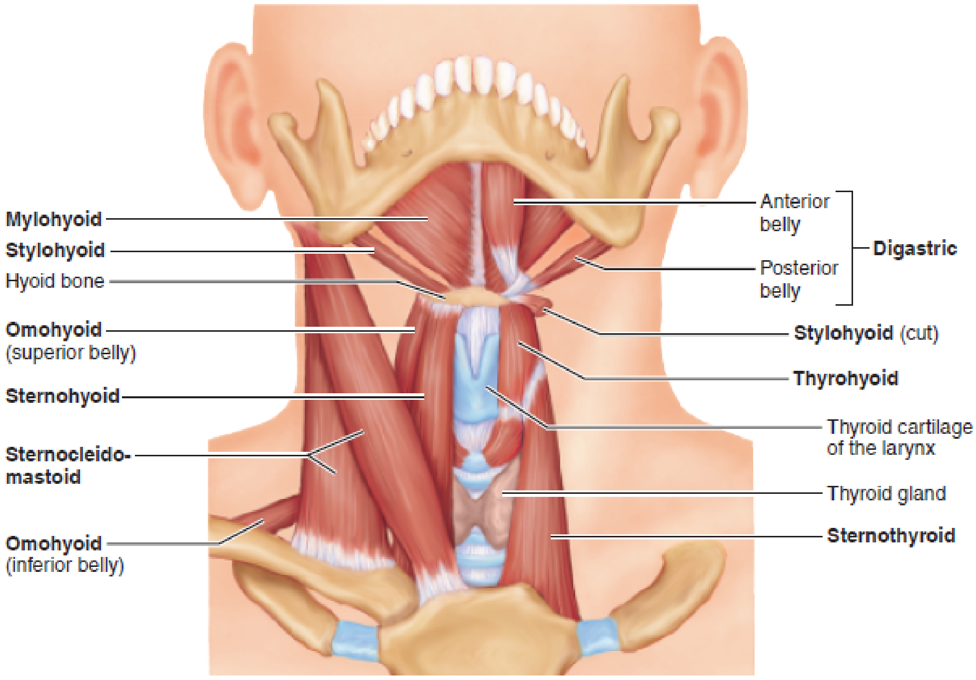


Figure 1.6: Neck muscles, superficial muscles illustrated on the left, deeper muscles on right [9].

- **Temporalis** muscle is located in the head's temporal region. It plays an essential role in the movement of the jaw. It has a unique fan-shaped anatomy composed of horizontal, vertical, and oblique fibers. This allows the muscle to have a versatile function in the movement of the jaw. It is responsible for the mandibular retrusion and elevation, leading to the closure of the jaw. This muscle is a crucial player in the functional anatomy of the masticatory system and significantly impacts oral functions such as chewing, speaking, and swallowing.
- **Masseter** is a paired, strong, thick, rectangular muscle that originates from the zygomatic arch and extends down to the mandibular angle. It consists of a superficial and a deep part. As it is one of the masticatory muscles, its specific functions are elevation and protrusion of the mandible for biting or chewing and providing support to the articular capsule of the temporomandibular joint. The muscle allows the generation of significant biting force, enabling static

and dynamic teeth grinding, such as in clenching or bruxism. This muscle is essential to the masticatory system and significantly impacts oral functions.

- **Digastric** is a paired muscle located in the neck region. It comprises two bellies, the anterior and posterior bellies, connected by an intermediate tendon. Its actions involve elevating the hyoid bone and opening the jaw. The anterior belly elevates the hyoid bone, essential for swallowing and speaking, while the posterior belly opens the jaw by pulling the mandible downwards. The *digastric* muscle is essential in the complex movements required for chewing and swallowing. Its actions are coordinated with those of other jaw muscles to produce the precise movements required to break down food and facilitate the passage of food through the digestive system.

- **Stylohyoid** is a thin, elongated muscle situated in the neck region, extending from the styloid process of the temporal bone to the hyoid bone. Its insertion is located in the body of the hyoid bone. Its main function consists of elevation and retraction of the hyoid bone, which is an essential action for the proper functioning of swallowing and speaking. The stylohyoid muscle is crucial for the stability of the hyoid bone, which is involved in the movement of the tongue and larynx during speech and swallowing. Additionally, the stylohyoid muscle is involved in the vocal tract modulation during speech production, contributing to the precise articulation of sounds.

- **Mylohyoid** is a thin, flat muscle located in the anterior part of the neck, forming the oral cavity floor. It runs from the mandible's mylohyoid line, a ridge on the inner surface of the jawbone, to the hyoid bone, attaching to it and forming a sling-like structure. When it contracts, it elevates the hyoid bone and the tongue. Mylohyoid assists in jaw opening and widening the pharynx during swallowing. Moreover, it assists in the production of various sounds during speech by modifying the tongue's position.

- **Geniohyoid** muscle is a tight, fan-shaped muscle that lies deep into the *mylohyoid* muscle. It runs from the mandible's mental spine to the hyoid bone's anterior surface. The *geniohyoid* primary function is to elevate the hyoid bone during swallowing, speaking, and breathing. In fact, plays a vital role in the swallowing process and contributes to the maintenance of upper airway patency.

1.2 Electromyographic signal

The muscle fiber contraction is elicited by a bioelectric command, as discussed in Sec. 1.1.2. These currents generate an electric potential in the surrounding volume and create a depolarization region in the physiological tissue that propagates from the endplate of the muscle toward the tendons. ElectroMyoGraphy (EMG) is the discipline that measures and studies electric activity generated by the muscles. EMG acquisition is performed in two main modalities:

- **needle EMG** is an invasive recording technique used to detect a few MUs, as the tip of the sensor has a detection volume of $1\,\mathrm{mm}^2$-$2\,\mathrm{mm}^2$. This method is suited for detecting MUs physiology and pathology, evaluating MUs recruitment strategy, and studying muscular denervation as an effect of neurodegenerative disorders. The recorded signal has a range of 0.1 mV-5 mV.
- **surface EMG (sEMG)** is a non-invasive recording technique that acquires muscle electrical activity with superficial electrodes placed on the skin. Because of the larger detection volume than the invasive technique, sEMG is an interfering signal from several MUs. Its amplitude and spectral characteristics strongly depend on location, Inter-Electrode Distance (IED), and size of the electrodes [13]. sEMG is a versatile technique, and it is useful in several cases, including:
  - muscle function assessment to evaluate the activation and strength of a muscle or a group of muscles by measuring the electrical activity.
  - physical therapy and rehabilitation, to monitor progress and ensure proper muscle activation during exercises and to apply biofeedback as a guidance for the patient
  - ergonomics, to assess the muscle activity of workers during repetitive tasks to identify potential sources of injury and improve the ergonomic design.
  - movement disorders, to diagnose conditions such as nerve damage, muscle disorders, and dystonia.
  - neuroprosthetics, to control prosthetic limbs and external actuators by detecting muscle signals from residual limb muscles.

1.2.1 The surface EMG signal

The electric signal recorded by the surface electrodes is a space-time summation of the Action Potentials (AP) generated by each muscle fiber in the MU, referred to as the Motor Unit Action Potential (MUAP). The sEMG signal is an interference signal because it combines the contribution of many different MUs [14]. The number of excited MUs may range from tens to hundreds according to the type of muscle and the contraction intensity [15]. Spatial and temporal recruitment of MUs that generates the EMG is modeled in Fig. 1.7.

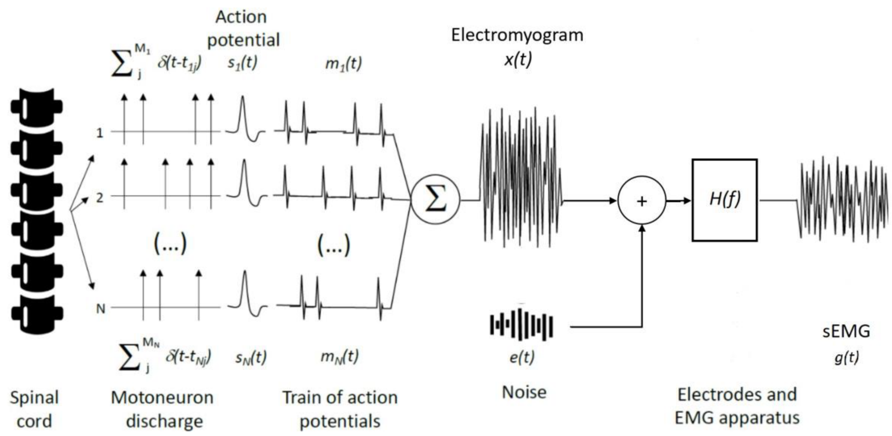


Figure 1.7: Schematic model for the generation of the EMG from MUAPs. $g(t)$ is the recorded signal, $x(t)$ is the signal of interest, $e(t)$ is additive noise, and $H(f)$ is the recording system transfer function [15].

The individual contributions (i.e., APs) can be approximated by combining three simple components, as shown in Fig. 1.8. The first is the depolarization phase, elicited by an above-threshold stimulus. When sodium channels close and potassium channels start to open, a repolarization phase begins, finally leading to a more extended hyperpolarization phase which eventually returns the fiber to its resting potential.

The sEMG signal has an amplitude range of 0.5 mV - 10 mV and frequency band of 0.1 Hz - 500 Hz, but most of the power is found in 20 Hz - 150 Hz range [13].

Because of its interfering nature, sEMG amplitude and spectral characteristics strongly depend on the electrodes' location, IED, and size. The muscle's operating conditions can influence the shape of the depolarization wave. For instance, if the muscle is fatigued, the central peak decreases, and the wave broadens because the repolarization phase becomes slower than in not fatigued muscle [17]. It can be affected by multiple noise sources:

- **motion artifacts**: relative movements between the skin and the electrode can introduce a noise component into sEMG, making the separation of the actual muscle activity from the noise difficult. Motion artifacts usually have a frequency range below 10 Hz, so they can be filtered in a preprocessing phase without affecting important spectral components.

- **crosstalk**: sEMG sensors have a large pick-up volume with respect to intra-muscular EMG, so the signal detected by one pair of electrodes is usually a mixture of contributions from the target muscle and nearby muscles. Crosstalk

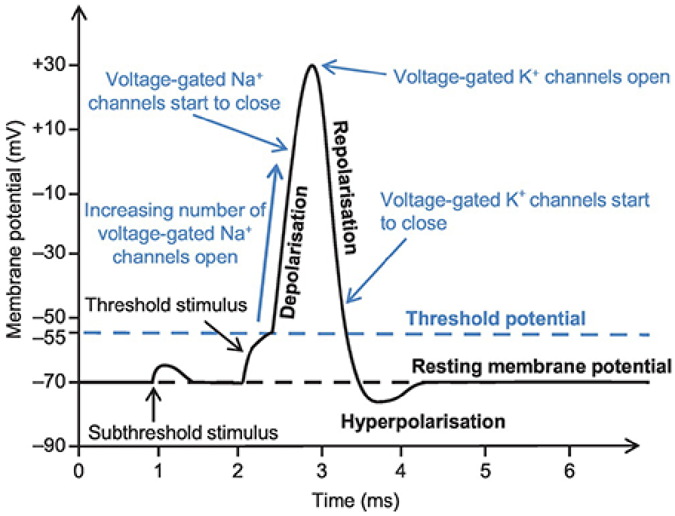


Figure 1.8: Action Potential illustration. Depolarization, Repolarization, and Hyperpolarization phases, with involved physiological mechanisms, are represented [16].

may originate from muscles located either deeper, adjacent, or all around the target muscle. The presence of small and overlapped muscles, like in the face, may increase crosstalk. Separating contributions from target and crosstalk muscles is an open problem and can affect all sEMG applications. Accurate electrode placement can effectively manage crosstalk, although filtering cannot completely eliminate it. However, there is no optimal configuration, depending on the case under investigation [13].

- **muscular fatigue**: it is a decrease in maximal force, power production, contraction velocity, and in general, a performance deterioration. A mechanical manifestation of muscle fatigue is evident only when a subject fails to produce an effort. However, sEMG shows variations since the beginning of a contraction. These variations cause a lowered spectral content, a reduction of the Conduction Velocity (CV), and a falsified amplitude; such variations help investigate the muscular fatigue phenomenon and must be carefully considered in sEMG interpretation.

- **external interference**: they are mostly electromagnetic interference. sEMG signal can be disrupted by other electrical signals, such as those from electrical devices, power lines, and other sources of electromagnetic fields. The power line is the most prominent among the external interferences. The power line introduces a 50 Hz noise component, which overlaps the frequency band of the sEMG signal.

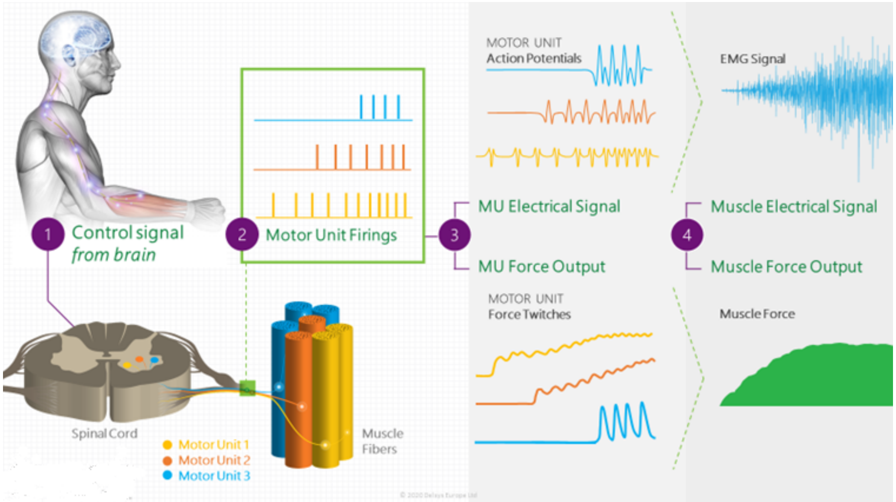


Figure 1.9: Overview of EMG generation: from CNS control signal to muscle force output [18].

1.2.2 sEMG feature extraction

Feature extraction is a technique utilized to extract meaningful and significant information from raw data, such as sEMG signals. Features can be categorized into three primary domains: time-domain, frequency-domain, and time-frequency domain features. However, due to their complexity and high computational cost, time-frequency features are unsuitable for real-time applications and will not be further discussed [19].

Time domain features

sEMG time-domain features provide information about the amplitude and timing of muscle activity over time. Some common features of this type are:

- **Mean Absolute Value: MAV** measures the average amplitude of the sEMG signal over a specified time interval. It can be used to provide information about the overall level of muscle activation.

$$MAV = \frac{1}{N}\sum_{n=1}^{N}|x_n| \tag{1.1}$$

- **Mean Absolute Value Slope: MAVS** measures the rate of change of the sEMG signal. It is calculated as the difference between the MAVs of adjacent EMG time windows.

$$MAVS_i = MAV_{i+1} - MAV_i \tag{1.2}$$

- **Root Mean Square: RMS** measures the average power of the sEMG signal over a specified time interval. It can provide information about muscle activation, especially when the mean amplitude is affected by noise or movement artifacts.

$$RMS = \sqrt{\frac{1}{N}\sum_{n=1}^{N} x_n^2} \tag{1.3}$$

- **VARiance:** VAR measures the spread of the sEMG signal over time. It can also provide information about the level of muscle activation and the degree of variability in the signal.

$$VAR = \frac{1}{N-1}\sum_{n=1}^{N} |x_n|^2 \tag{1.4}$$

- **Zero Crossing Rate: ZCR** provides a count of the number of times the sEMG signal crosses zero during a specified time interval. It can provide information about the frequency of muscle activation and identify instances of low-level muscle activation or noise.

$$ZC = \sum_{n=1}^{N-1} f_{ZC}(\cdot) \tag{1.5}$$

$$f_{ZC}(x_N, x_{n+1}) = \begin{cases} 1, & \text{if } x_n \cdot x_{n+1} < 0 \\ 0, & otherwise \end{cases}$$

- **Integral Absolute Value: IAV** provides a measure of the cumulative absolute value of the sEMG signal over a specified time interval. It can be used to provide information about the overall level of muscle activation.

$$IAV = \sum_{n=1}^{N} |x_n| \tag{1.6}$$

- **Waveform Length: WL** provides a measure of the length of the waveform of the sEMG signal over a specified time interval. It can be used to provide information about the level of muscle activation and identify instances of low-level muscle activation or noise.

$$WL = \sum_{n=1}^{N} |x_{n+1} - x_n| \tag{1.7}$$

Frequency domain Features

Frequency-domain features of sEMG signals provide information about energy distribution in the frequency domain. Some commonly used frequency-domain features are:

- **MeaN Frequency: MNF** is the average frequency of the sEMG signal in a given time interval. It is calculated as the weighted average of the sEMG Power Spectral Density (PSD), where the weights are the frequencies of the sEMG signal.

$$MNF = \frac{\sum_{j=1}^{M} f_j P_j}{\sum_{j=1}^{M} P_j} \tag{1.8}$$

- **MeDian Frequency: MDF** is the frequency at which 50% of the total energy of the sEMG signal is contained. It is calculated as the median of the cumulative distribution function of the PSD of the sEMG signal.

$$\sum_{j=1}^{MDF} P_j = \frac{1}{2}\sum_{j=1}^{M} P_j = \sum_{j=MDF}^{M} P_j \tag{1.9}$$

Both mean or median frequency changes can provide insight into energy distribution in the frequency domain, indicating muscle fatigue.

1.2.3 Facial sEMG

Facial sEMG is a technique that measures the electrical activity of the muscles in the face and neck during contractions. This field is rapidly developing and has vast potential applications, as the human face is one of the most critical areas of the body and plays a vital role in health and communication. The analysis of facial muscles is crucial in both verbal and non-verbal communication, mastication, and the expression of emotions. sEMG is an effective method for evaluating mastication function and disorders in the oral system and can aid in studying human speech production and recognition. In stroke patients, facial sEMG can assist in the diagnosis process by revealing swallowing problems in their early stages. Additionally, it could play an essential role in surgical planning, as most methods used to recognize and analyze facial pathologies rely on subjective inference and visual evaluation. Monitoring facial sEMG activity can also detect pain conditions in patients unable to communicate. As sEMG is a promising computer access method for individuals with motor impairments, it is a crucial tool in several Human-Machine Interfaces (HMI) for tetraplegic patients [20].

1.3 The Average Threshold Crossing (ATC)

Looking at the State-of-Art works, expression recognition systems compute multiple sEMG parameters to train Machine Learning models. The feature extraction process is generally made onboard embedded systems or via software on a computer after the complete sEMG information has been recorded and digitalized. However, sEMG sampling and elaboration for feature extraction is a high computational cost process and it could not be a good strategy in power-constrained applications. Trying to relax these computations, a different approach has been proposed, named Averaged Threshold Crossing technique [21]. The main idea is to count how many times the analog sEMG signal, once amplified and filtered, overcomes an externally provided threshold in a given time window, as illustrated in Fig. 1.10.

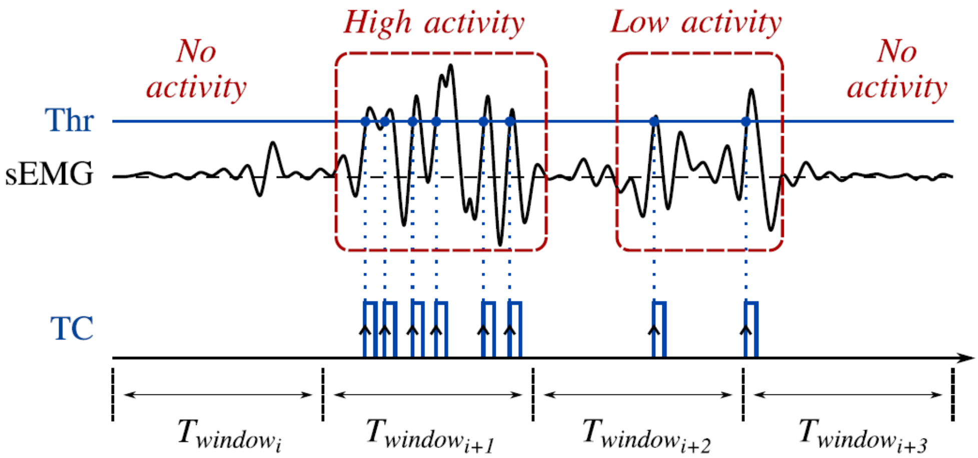


Figure 1.10: Average Threshold Crossing technique illustration [21].

The time window is set to 130 ms as reported in the tests presented in [22], where this value has been proved to be an optimal trade-off between the time resolution of the muscle activation and the discrimination of different levels of generated muscular force. The threshold value is set just above the sEMG signal baseline during muscle rest to detect relevant muscle activation and to be robust against environmental noise and physiological artifacts [23]. This approach uses a voltage comparator to directly allow the feature extraction process on hardware, which eliminates the need for an Analog-to-Digital Converter (ADC) and avoids unnecessary sampling of sEMG, thereby leading to simpler acquisition hardware. Acquired signals are usually sent to an external device for processing purposes and with the ATC technique, the number of packets sent is significantly lower than standard sampling [24]. The limited data throughput necessary to manage ATC transmission significantly reduces power consumption. A drawback of this technique is the partial loss of information contained in the original sEMG, which can not be reconstructed. However, in contrast with sEMG continuous sampling applications, the event-driven approach succeeds in monitoring muscular activation with reduced processing and transmission data load, leveraging on power consumption and operating time [21]. The use of the ATC parameter as a reliable force estimator has been demonstrated to be comparable to the ARV gold standard estimator [4]. The possibility to distinguish different force levels based on the ATC parameter also allows the investigation of Human-Machine Interfaces. Indeed, hand gesture recognition using Machine Learning models based on the ATC parameter has been evaluated and demonstrated to yield promising results [25].

1.4 State of the art

Facial expression recognition based on sEMG is an emerging technology that uses the electrical activity of facial muscles to detect and interpret an individual's facial expressions and emotional expressions. This technology can be used in various contexts, such as evaluating emotional well-being, supporting nonverbal communication, evaluating performance in games and video games, and controlling HMI and external devices. Most of the methods for facial expression recognition are based on computer vision because they are easy and inexpensive and can achieve acceptable accuracy. However, the results are highly affected by light level, camera resolution, view occlusion, head movements, and other external factors [26]. Several facial expression recognition systems based on sEMG have been developed in recent years and have shown to be accurate and reliable. sEMG is widely used in gesture and motion recognition of limbs and has been helpful in facial expression recognition. Compared to computer vision methods, sEMG does not impose limits on the external environment, is less affected by head movements, provides non-visual information, and reflects underlying muscle activity in subtle movements otherwise not detectable. However, there are also challenges in using surface EMG for facial expression recognition. For example, external factors such as head movement or speaking can still influence facial muscle activity, making detecting emotional expressions more challenging. Some people may have difficulty generating sufficient quality EMG signals for facial expression recognition due to health issues or other factors. Additionally, the location, size, and shape of facial muscles can vary between individuals, leading to differences in muscle activity patterns. Factors such as age, gender, and facial expression habits can also affect the patterns of muscle activity. Despite these challenges, sEMG-based facial expression recognition represents a promising technology for analyzing human emotions and supporting nonverbal communication. With further research and development, this technology will likely become increasingly accurate and reliable.

In [27, 28], authors present methods for recognizing voluntary human facial expressions through three bipolar sEMG channels for HMI applications. Depending on the type of features and machine learning algorithms, they found that up to 11 expression control commands can be applied to HMI with over 90% accuracy. In [29], a wavelet denoising protocol is applied to enhance Signal to Noice Ratio (SNR) and improve classification performance.

In [30, 31] capability in phones, phonetic features, and syllables recognition via facial sEMG is assessed. Classification methods, including deep neural networks, are tested for the feasibility of a silent-speech interface based on facial sEMG. Only 70% mean accuracy is provided, showing a possible application. Speech analysis is helpful in disfluency detection, and speech therapy too [20].

In [32], malfunction of swallowing-related muscles is analyzed. Swallowing, or deglutition, is a complex process involving multiple muscle coordination, and muscle malfunctioning can lead to dysphagia. This work presents a classification scheme of swallowing phases based on facial sEMG for non-invasive and standardized support in diagnosis. Eight acquisition channels, nine time-domain features, and Support Vector Machine (SVM) and Artificial Neural Network (ANN) classifiers were used, obtaining over 92% recognition accuracy of oral and pharyngeal phases.

A Human-Machine Interface (HMI) was developed in [33] to aid patients in using a computer through facial movements. Facial movement patterns detected by four EMG channels are classified and mapped to cursor actions to complete drawing and typing tasks. High classification accuracy (98%) is achieved, with a median input rate of 5.9 letters per minute and a median path efficiency of 80%.

In [34], a hand-free control system for an electric wheelchair based on facial sEMG is realized. Simple rules on specific face muscle activations classify motion control commands. In order to face the problem of safe and fine control using a biological signal only, a semi-automatic control based on a laser range scanner is introduced. In [35] is found that user training is a fundamental strategy in prosthetic control and HMI. Giving the user real-time feedback in generating consistent sEMG patterns enhances online accuracy and eases user control.

Other studies [36, 37, 38] report on the validity of the facial sEMG method for measuring pain expression. The activity of several facial muscles is recorded during induced pain throughout a laser system. Groups of muscles as key for pain expression are indicated, particularly around *orbicularis oculi* and *depressor anguli oris*. Also, *corrugator supercilii* is most strongly associated with self-reported pain. At the same time, *orbicularis oculi* and *levator labii superioris* show a statistically significant increase in activation when the pain reaches self-reported pain thresholds. Although the performance of predictions remains modest, and although the technology is not yet ready for use outside of laboratory settings, these are steps towards an objective pain measurement tool that can substitute self-report measures in clinical practice.

Muscles essential for facial movements are characterized in [39, 12]. 48 small electrodes were bilateral and symmetrically applied to record monopolar sEMG signals while performing 29 movements of high clinical relevance. Almost all functionally relevant facial movements are based on a specific interplay of several facial muscles. To reveal the relevance of the findings, the results were compared with other facial muscle examinations accomplished with invasive intramuscular fine wire electrodes. The use of reference data can provide an objective assessment of facial paresis and facilitate monitoring of changes throughout disease progression and treatment. After clinical evaluation, a reduction in the number of electrode positions is planned to simplify the application of the method.

In [40], a wearable device that interprets positive and negative facial expressions is presented. The authors analyzed facial morphology and sEMG signals from different facial locations to select a zone for electrode positioning with low facial mobility, and still high amplitude detected sEMG. Based on this analysis, the device records signals distally on a side of the face. This method cannot identify activity from each facial muscle; facial expressions are regarded as multiple muscles' synergistic action, and pattern-based classification methods are used for expression recognition. The device uses independent component analysis to separate independent sEMG components. The expressions neutral, smiling, biting, frowning, and simultaneous biting and smiling were used to train the ANN to recognize between smiling, frowning, and neither, reaching over 90% accuracy.

In another study [41], a facial expression recognition method based on sEMG from eyebrows activity is proposed. They designed a special headband to record sEMG activity from the *frontalis* and *corrugator supercilii* muscles of 6 participants instructed to produce five facial expressions. Using 6 time domain features as input vectors to a neural network model, an average recognition rate of 96.12% was achieved.

In [42], the researchers aimed to develop a Human-Machine Interface (HMI) for recognizing three facial expressions, namely happiness, anger, and disgust, by recording facial sEMG signals from the *zygomaticus* and *corrugator supercilii* muscles. As a feature extraction method from the two-channel signal, they applied wavelet packet transform, which decomposes the signal in sub-bands based on the wavelet family. Then energy, mean, and standard deviation features are computed. The resulting dataset was used to train an SVM to classify 3 different facial expressions with a resulting accuracy of 91.66% on 12 subjects.

In [43], a preliminary study was conducted to establish the feasibility of facial expression recognition applications based on the ATC feature. The optimal electrode location was studied for proper signal detection as the first step. 21 healthy subjects performed 8 facial expressions during the data collection. The raw sEMG signal was recorded and then processed offline to compute the ATC feature. Several Machine Learning algorithms were tested on the dataset to recognize facial expressions: ANN, SVM, k-Nearest Neighbour (k-NN), and Random Forest (RF). The four classifiers demonstrated similar accuracy, reaching an overall success greater than 60% when recognizing 8 expressions. The average accuracy improved to 75% when two poorly defined expressions were removed from the dataset. When only jaw movements were classified, an average accuracy close to 80% was achieved. These results were the starting point for future developments, aiming for an embedded system implementation for real-time facial expression recognition.

Most of the presented works in facial expression recognition have a similar scheme. sEMG signal is acquired and filtered, then several features are extracted and undergo a feature selection process, and finally used to train machine learning models. This approach usually reaches a high accuracy level in pattern recognition, as the maximum amount of information can be extracted from the original EMG signal. However, the computational load required by the aforementioned approach would not be suitable for real-time and embedded implementation.

This thesis will implement a hardware-based ATC feature extraction approach to develop an sEMG-based embedded prototype for real-time facial expression recognition. The extracted ATC values will serve as inputs to a Machine Learning algorithm, specifically an implementation of a fully-connected Neural Network. This simple and light neural network implementation has been chosen because it is particularly well-suited for embedded systems designed to process a low-complexity signal.

Chapter 2

Hardware and Software Setup

The system developed in this thesis inherited part of its structure from related works, which deal with ATC feature extraction from sEMG signal and gesture recognition[44, 45]. The proposed system, schematized in Fig. 2.1, is composed of the following main blocks:

- A central processing unit, an AmbiqMicro Apollo3 Blue EVB Micro-Controller Unit (MCU), which is an evaluation board with an ARM® Cortex®-M4 processor. It has a wide range of peripherals and various communication interfaces [46].

- Five acquisition channels acquire the sEMG signal and make it suitable to be processed by the MCU. Each acquisition channel has its Analog Front-End (AFE), an analog circuit responsible for signal amplification and filtering, and for the extraction of the ATC feature as well.

- A wireless communication system that uses Bluetooth Low Power (BLE) module equipped on the MCU and a USB-dongle (i.e., nRF52840 [47]) connected to an external device to provide wireless connectivity.

- A PC software featuring a Graphical User Interface (GUI) on the application layer to ease the user to interface with the electronic system and control its functionality through BLE communication protocol.

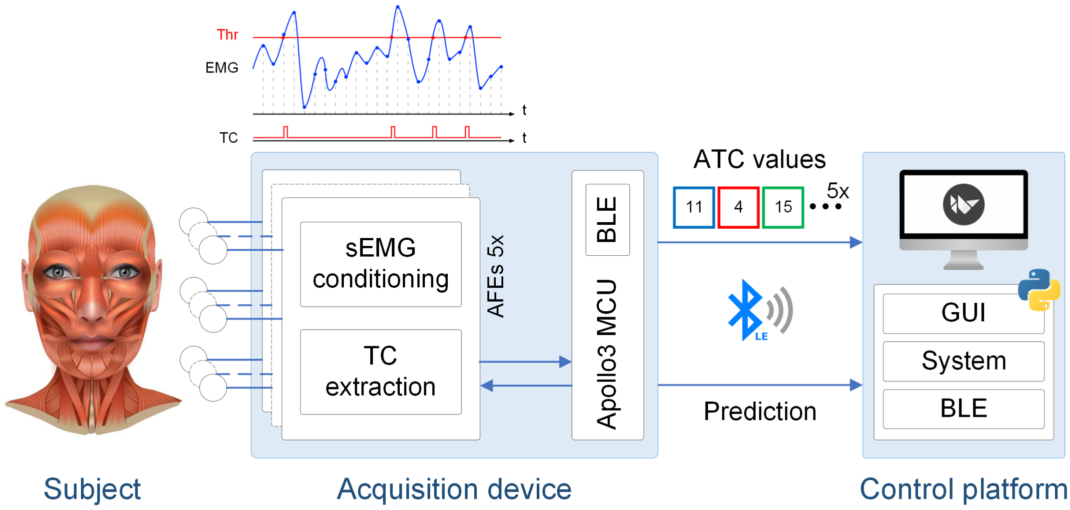


Figure 2.1: System schematic configuration. The acquisition device (middle) comprises an Apollo3 MicroController Unit (MCU) and 5 Analog Front Ends (AFE). The AFEs, under I^2C commands from the MCU, filter the sEMG signals from face muscles and generate the TC signals. The TC output channels are connected as inputs to the MCU General Purpose Input/Output (GPIO) to calculate the ATC and make predictions. The MCU streams the data through its Bluetooth Low Energy (BLE) module. The user receives the data on a control platform using a modular software developed in Python to control the system functionality through a Graphical User Interface (GUI). Image adapted from [48, 49].

2.1 Acquisition Channels

For sEMG recording, circular pre-gelled disposable electrodes (i.e., Ag/AgCl H124SG [50]) with a diameter of 24 mm are applied on face skin. The electrodes are connected to the acquisition boards using shielded wires with a clip head.

One per channel's acquisition boards (shown in Fig. 2.2) are Printed Circuit Boards (PCB), slightly modified from a previous system for low-power event-driven sEMG acquisition[51]. Each single acquisition module is characterized by a weight of 2 g and a body area of 414 mm^2.

The three inputs of the AFE are meant for single-differential signal acquisition. Two inputs are for exploring electrodes placed on the muscle of interest. The third input is connected to a reference electrode placed on a stable skin site with little or no electrical activity (i.e., the mastoid bone). The AFEs, as depicted in Fig 2.4, have the following stages:

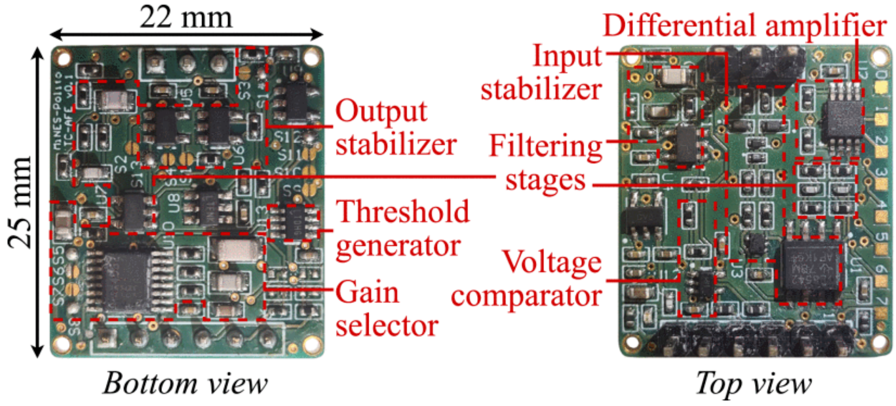


Figure 2.2: AFE for TC feature extraction from sEMG [48].

Overvoltage protection

Overvoltage protection helps prevent circuitry damage from voltage spikes, transients, and out of the allowed range voltage inputs. It ensures the safe and reliable operation of the circuitry.

Impedance decoupling

Impedance decoupling is made with TLV8542 integrated circuit. This component decouples the input from the following differential high-pass filter.

Differential High Pass Filter (HPF)

The movement artifact components are removed from the signal with differential second-order HPF. This filtering stage with no active components has a 33.86 Hz cut-off-frequency.

Differential amplifier

INstrumentation Amplifier (INA333) then amplifies the differential signal. This component is well suited for biological signals because of its stable gain and high common-mode rejection ratio. The gain, ranging from 1 to 10000, is adjusted via an external resistor. The gain provided in these boards is 500 V/V, which a gain selector can further increase.

Low Pass Filter (LPF)

This LPF attenuates the low-frequency components and apply negative feedback to the reference voltage of the differential amplifier. The cut-off frequency was adjusted to 70 Hz to suppress power-line interference.

Gain Selector

The gain selector component allows changing the gain of the board. The different gains are set according to external resistors connected to the component. In these boards the possible gains are $\times 1$, $\times 2$, $\times 3$, $\times 5$, $\times 6$.

Low Pass Filter

A second low-pass filter attenuates high-frequency components above the sEMG spectrum. This stage is a second-order LPF with a cut-off frequency of 397.40 Hz.

Voltage Comparator

The voltage comparator detects Threshold Crossing (TC) events in the processed signal by comparing it with an adjustable threshold. When the signal crosses the threshold, the comparator switches its output from a low to a high digital state. A 30 mV hysteresis is set to ensure stable switching between high and low digital states. The threshold level is adjusted using a Digital-to-Analog Converter (DAC) component, which operates under I^2C commands from the Micro-Controller Unit.

Voltage Regulator

The voltage regulator changes the 3.3 V voltage supply from the Apollo3 to a 2.5 V range for all the components on the board.

Output Signals

There are two output signals from the boards:

- the sEMG signal can be obtained from the output of the last LPF stage, just before the voltage comparator stage. It can be useful to verify the proper functioning of the board.
- the TC signal is produced as the output of the voltage comparator. The TC output channels are connected as inputs to the microcontroller's General Purpose Input/Output (GPIO) to raise interrupt events every TC count.

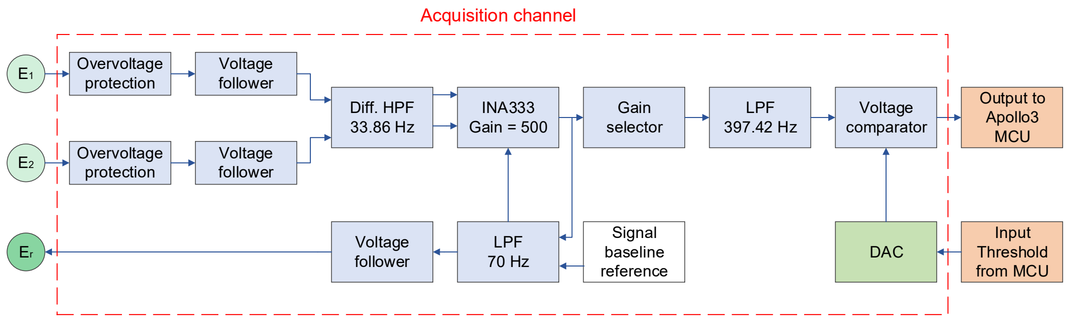


Figure 2.3: Structure of the conditioning stages of the acquisition boards.

2.2 Apollo3 Blue Board

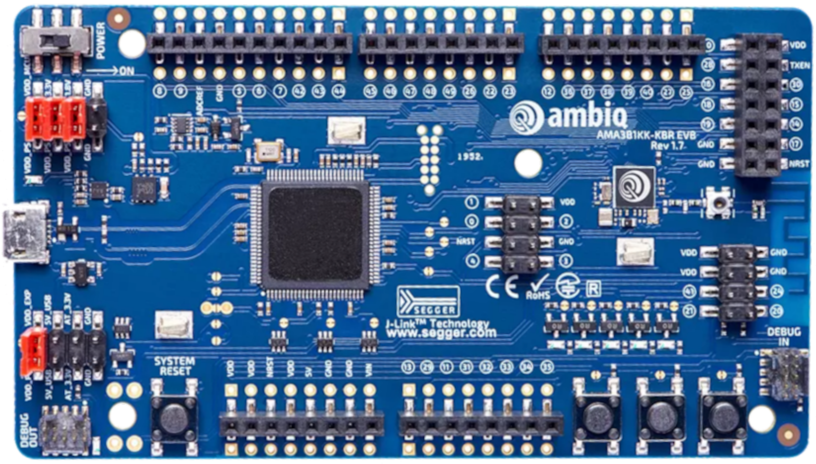


Figure 2.4: The AmbiqMicro Apollo3 Blue evaluation board [46].

The system's core is the AmbiqMicro Apollo3 Blue EVB, ultra-low power and highly integrated microcontroller platform, which is in charge of all the computation needed for the system to work and supplies power to the external AFE. The Apollo3 Blue EVB is equipped with an ARM Cortex-M4 processor, a 32-bit microprocessor developed to address digital signal control markets that demand an efficient, easy-to-use blend of control and signal processing capabilities [52]. The Apollo3 main characteristics are summarized in the Table 2.1. Among other technical characteristics, low power consumption is one of the strengths of this microcontroller.

For this application, the main clock frequency chosen is 24 MHz, while the low frequency 32.768 kHz crystal is used for the time window implementation needed for the ATC.

Table 2.1: Apollo3 Blue features and specifications.

Max clock frequency	48 MHz 96 MHz turboSPOT Mode
MCU	32 bit-ARM® Cortex®-M4 with FPU
Flash/SRAM	1 MB/384 kB
MCU min current supply	6 µA/MHz at 3.3 V
V_{DD}	1.755 V-3.63 V
I/O	(6×) I^2C/SPI (2×) UARTS
Connectivity	Integrated Bluetooth Low Energy module

2.2.1 Overview of Firmware Implementation

The MCU firmware was developed utilizing the ARM Development Studio IDE v2020.1 due to its compatibility with the ARM product family. Several ARM libraries are accessible as ARM DS support packages. In particular, the Digital Signal Processing (DSP) library, which is an ARM native suite of standard signal processing functions for use on Cortex-M-based devices, was utilized for the application. The library was explicitly used to provide helpful support in vectorized implementations of Basic Math Functions, Matrix Calculations, and Statistical

Functions[53]. In addition, AmbiqMicro provides support packages with high-level functions, allowing the developer to perform complex tasks by interacting with every system component in a more user-friendly way. The firmware inherited its structure and functionalities from related projects featuring systems for acquiring, processing, and transmitting muscular information [21]. The firmware handles signal acquisition (in ATC mode) and Bluetooth transmission simultaneously, thanks to FreeRTOS custom implementation. FreeRTOS is a popular open-source Real-Time Operating System often used in embedded systems and microcontrollers. It provides a range of features and services, such as multitasking, task prioritization, memory management, and communication protocols, which make it ideal for use in systems that require real-time operation and low latency [54]. A Bluetooth Low Energy server was integrated on the board to facilitate data exchange and user control through a client/server communication with a central device [45]. Specifically for this thesis, a laptop with support software is the central communication device and the board act as a peripheral node. BLE-related functionalities were implemented in another off-the-shelf firmware module, Cordio Bluetooth. Cordio Bluetooth is a low-power, high-performance Bluetooth solution developed by ARM, designed to enable the development of Bluetooth-enabled devices, such as wearables and sensors. It provides a complete Bluetooth stack that supports BLE features and a range of Application Programming Interfaces (APIs) to enable easy integration with microcontrollers. When combined with FreeRTOS, Cordio Bluetooth can provide a complete Bluetooth solution for microcontroller-based IoT devices. The Cordio Bluetooth stack can be integrated with FreeRTOS to enable Bluetooth connectivity and communication. At the same time, the FreeRTOS scheduler can manage the various tasks required for Bluetooth communication, such as data acquisition, processing, and transmission, while providing real-time functionality.

The program is organized in modules based on the operations to execute and on the active microcontroller components. This part of the code, in some case specifically customized for the specific application, accomplish various functions:

- Initial configuration: Universal Asynchronous Receiver-Transmitter (UART) print interface is enabled for debugging purposes. Depending on the environmental constants, certain functions are invoked, such as setting the main clock to 48 MHz for high frequency or enabling low power mode when operating on 24 MHz.

- LEDs Configuration: Board LEDs are configured, enabled for usage, and then switched off.

- Timer Configuration: a timer is configured for ATC window requirements. The external low-frequency clock is used for power management, running at a frequency 32.768 kHz to generate 130 ms windows. When latency performance measurement is required, a timer is set with frequency 6 MHz, a quarter of the frequency of the main clock.

- GPIO configuration: the five pins connected with the AFEs are enabled as input to trigger an interrupt as a TC event comes from an acquisition board. A specific interrupt function is assigned in the interrupt service table for each configured pin. The functions count the TC event on the corresponding channel for these particular pins. Interrupts from AFEs can concurrently be received and managed by the system.

- Inter-Integrated Circuit (I^2C) interface: the I^2C interface is configured to communicate with each DAC on the AFEs. The device's unique addresses are assigned and the pins for the Serial CLock (SCL) line and the Serial Data (SDA) line are configured. According to this information, then, the available boards are detected.

- ANN variables initialization: the matrices needed for machine learning related operations are initialized and paired with weight values from the trained model.

- DAC threshold initialization: DAC threshold values are set to 0 via I^2C as the initial configuration.

- Functional Task: after RTOS and BLE related initial configurations, a *Radio-Task* for BLE communication handling and a *Predict* task for classification purposes are created and immediately suspended. The tasks remain suspended until an interrupt from the BLE service or user command enables to resume of the functional task.

- Run Task Loop: once system configuration is completed, the system enters a while loop, and the board is set in a *Deep Sleep* mode. Only when a command from the user enables the ATC acquisition mode, the Machine Learning mode, or another system's routine, the MCU is awakened for the time necessary to execute all the needed computations. Then the MCU returns to Deep sleep mode until an interrupt is generated, either from time window expiration or from a TC event.

2.3 Wireless communication system

The BLE stack provides a robust data transfer protocol to the system, granting quick user accessibility without compromising future wearable applications. The BLE server implemented on the MCU allows connecting the system with any external device, e.g., a smartphone or laptop. An nRF52840 USB dongle (in Fig.

2.5) facilitates the BLE connectivity of the MCU to the laptop running the control software. The purpose of this external component is to establish a consistent wireless protocol stack, thereby ensuring optimal application performance.

Figure 2.5: nRF52840 Bluetooth USB dongle by Nordic Semiconductors [47].

Thanks to the BLE module, the user can operate the device and receive all notifications of the parameter (i.e., board status and errors, ATC values, ML predictions). ATC event-driven technique requires a low data rate (w.r.t. sEMG fixed sampling). Therefore the connection parameters are chosen in low-power optics: with the 130 ms ATC window, a 56 ms-75 ms connection interval is set, still granting data transfer without to much delay. The BLE characteristics available on the server are divided into two groups: a command and status group, used to control the board and obtain feedback; and data values group, which contains values acquired and device configurations. The BLE server is structured similarly to the service introduced in [21] and redefined in [45], and has the following available characteristics:

- *Command* initiates the execution of a specific routine based on the written value (e.g., board activation, gain selection, notification enabling, ATC threshold calibration);
- *Status*: this value defines the current operating status of the board and can be read or notified to the user;
- *Available*: when read, the operating state of each AFE board is returned; it can be written to update the boards status to unavailable, available, and active;
- *Gain*: the board AFEs gain multiplier can be obtained by reading this value, and it can be written to set the gain manually for one or more groups of boards.
- *ATC*: it is read-only characteristic with the latest computed ATC data from up to nine acquisition boards, the actual duration of the last ATC window (measured using a 6 MHz timer of the MCU), and the packet number (i.e.,

sequentially updated number for each ATC window); if notification is enabled, the values update every ATC window;

- *Threshold*: it can be read to obtain the threshold values set for each board or written to set threshold values manually from 0 V to 2.5 V;
- *Prediction*: the notification request provides the value of the latest predicted class and the relative packet number as in ATC.

2.4 Graphical User Interface

This thesis work focused on integrating a Graphical User Interface (GUI) in a software for controlling the system from a computer to ease its usage in the training and testing acquisition phases. The software is implemented in Python and inherited its layered structure from other projects[45, 21, 48]. The code follows a bottom-up structure comprising three layers:

- in the first layer there is the BLE module, which is responsible for establishing and managing the connection between the Apollo3 board and the external device. This module utilizes server functionality to enable data communication and handles input and output packets.
- The second layer is the *Ap3 sEMG* system module, which is specific to this system. This module translates user commands into executable actions and processes incoming packets to extract meaningful data.
- The third and topmost layer is the application layer, specifically created for this thesis work. An open-source Python library, Kivy [55], has been used to develop the user interface application. Kivi is designed to be cross-platform and provides a range of features useful to support various input devices, such as touchscreens, mouse, and keyboards, and a wide range of user interface widgets, such as buttons, text inputs, and dropdown menus. Although the cross-platform philosophy, for this thesis work, the application has been tested only on a Windows environment.

The communication of data from the bottom to the top layer in the system relies on a queue-based approach. Specifically, each system layer includes one or more queues to facilitate data transmission based on the specific type of information that needs to be transferred. For instance, the middle layer contains specific queues for managing ATC and ML data, board status, and error data. This data communication structure facilitates the organization and management of the data types requested by user actions. By utilizing queues to transmit data between the layers, the system is better equipped to handle efficiently various types of data streamlined. This approach also helps to ensure that each layer of the system can effectively process the data it receives and that the relevant data is passed up to the application layer for user interaction. From the user interface point of view, the top-down functions are launched through method callbacks. It means that the software has specific methods invoked when a particular event occurs, such as user inputs through buttons and spinners or system events.

Control panel

The Control Panel of the GUI (reported in Fig. 2.6) contains all the user commands needed to operate the system:

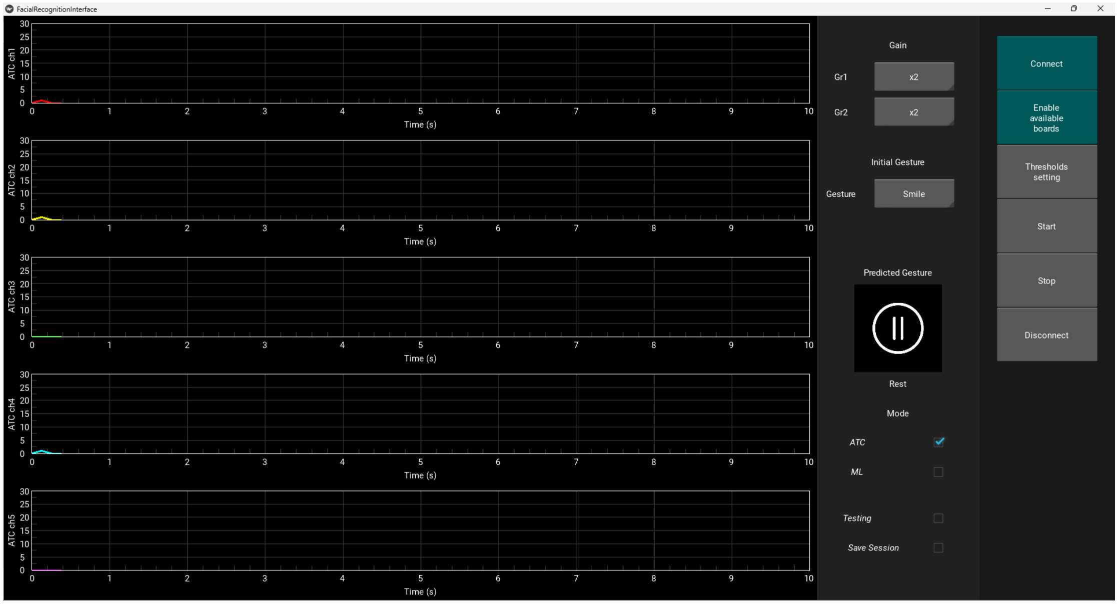


Figure 2.6: GUI Control Panel: on the left side, one per channel graph plot ATC received data; on the right side buttons and spinner for system control.

- Connect: the connection button initiates the scan operation to search the system's advertising packets. The status notifications are enabled when the connection is established and the board's availability status is checked.
- Enable boards: the enable button transits the command to enable the available boards if correctly connected and ready for acquisition.
- Threshold setting: this button launches the ATC thresholds calibration. This operation is performed on each board and requires a variable time from some seconds to 20 s. During this operation, the monitored subject has to be still and relaxed to allow proper threshold calculation.

- Start/Stop: these commands initiate and terminate the system's data stream, while the data obtained from the queues are plotted on the ATC graphs.
- Mode Selection: using this checkbox, the user can select ATC data values acquisition or decide to start Machine Learning (ML) mode for expression prediction. ML mode enables the ANN prediction routines implemented on the Apollo3 when the acquisition starts. If ML mode is enabled, a picture in the panel shows the predicted expression.
- Disconnect: disconnection button terminates the Bluetooth link before closing the GUI.
- Gain: gain selection spinners are used to adjust the multiplier factor of the AFEs. Gains are set for groups of boards and not for a singular board. The available multiplier values are: $\times 1$, $\times 2$, $\times 3$, $\times 5$, $\times 6$. The default value is $\times 1$
- Testing mode: when the testing mode checkbox is enabled, a second panel is created for acquisition session purposes. The testing panel will be illustrated in the next subsection.
- Initial expression selection: the spinner enables the user to select the initial expression for the acquisition protocol.
- Save: the checkbox enables the saving mode. The data streamed from the system are stored in a txt file. Timestamps, ATC values, and packet numbers are saved for each time window. Depending on the selected mode, expression labels, and predicted labels are saved if testing mode, ML mode, or both are enabled.

Testing panel

The software is equipped with a testing mode, which is helpful as guidance for the subjects during the acquisitions for dataset construction and for the testing phase. When the mode is selected, a second panel (shown in Fig. 2.7) is created to guide the subject in performing the expressions, as described in the training protocol in section 3.3.1. The panel displays a graphical representation of the expression to be executed and a timer. At the same time, a side image shows the next expression that the subject will have to perform after the timer has expired. The testing mode also labels the acquired data according to the performed expressions. For each ATC acquisition window, a label is added to the saved data identifying the expression being performed at that moment, with the timing dictated by the testing window. Whenever the subject is asked to change expression, the label information is communicated via a multiprocessing queue to the control panel, which takes care of data saving. In addition to the timing dictated by the testing window, the labeling process had to consider the possibility of response delays in executing facial expressions. Therefore, for each set of ATC values, a norm was calculated and a threshold was imposed on this norm. Values with a norm greater than the threshold were identified as the class being executed at that moment; values with a norm equal to or less than the threshold were identified as Rest.

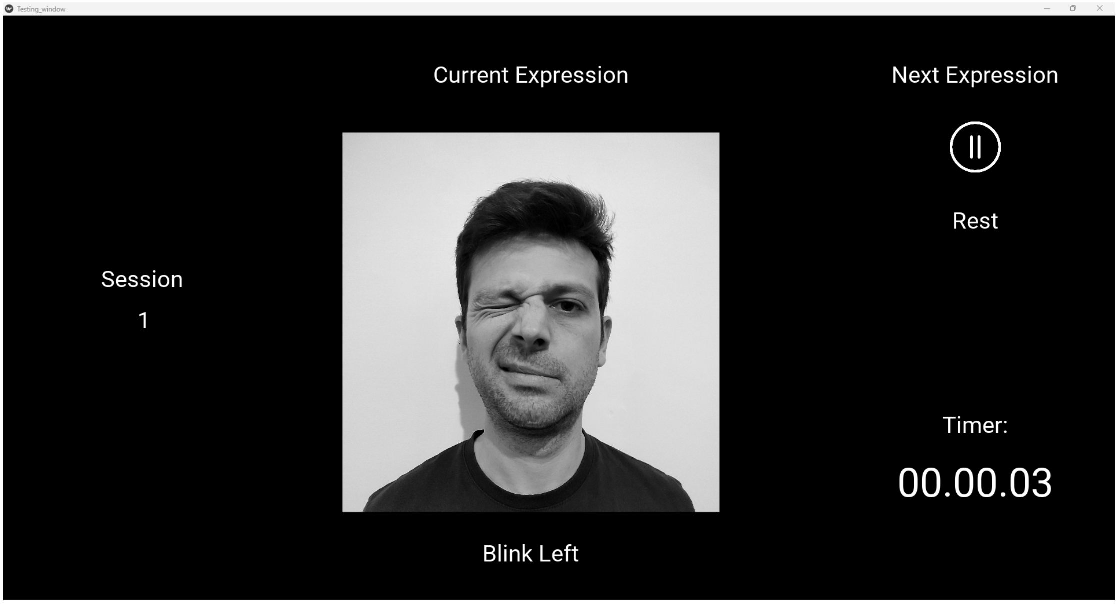


Figure 2.7: GUI Testing Panel: the central image shows the current expression to perform. A timer indicates the remaining time before switching to the next expression, previewed on the top right.

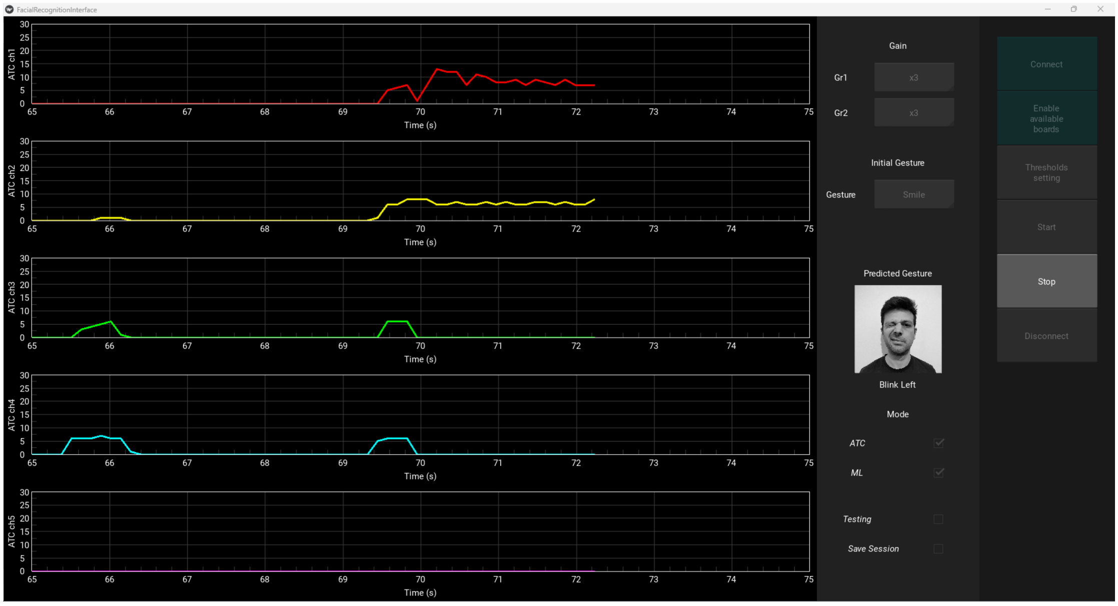


Figure 2.8: GUI Control Panel during combined Machine Learning and ATC mode functioning. ATC is plotted on the left graph, while the resulting predicted class is displayed in an image box on the right side of the panel.

Chapter 3

Data acquisition for Facial Expression Recognition

In sEMG acquisition, effective electrode positioning is critical for capturing high-quality EMG signals and characterizing facial expressions accurately [56]. Based on feasibility work in a previous thesis [43], a study was conducted to determine which expressions to perform and which were the best electrode positioning to obtain satisfactory values.

3.1 Facial Expressions

The expressions to be performed have been selected starting from applications in literature works described in Sec. 1.4. Knowing that the system has 5 acquisition channels and that not all the face muscles can be monitored, the key step was understanding the association between expressions and involved muscles. In [12], an atlas of facial expressions and related muscle activations was developed. Using high-resolution sEMG, they created a map of the activation of 10 facial muscles related to 29 different facial muscle tasks. To reveal the relevance of the findings, the results were compared with other facial muscle examinations accomplished with invasive intramuscular fine wire electrodes. However, it should be remembered that facial muscles act more as a whole than as single actuators. The activation map for the expressions of interest are shown in Fig. 3.1, 3.2, 3.3, 3.4, 3.5, 3.7, 3.8, all adapted from [12].

Smiling

Voluntary smiling consists in pulling the corners of the mouth upwards and backward and involves the contraction of several muscles. The main muscles involved are *zygomatic*, *orbicularis oris*, *mentalis*, *depressor anguli oris*, *depressor labii*.

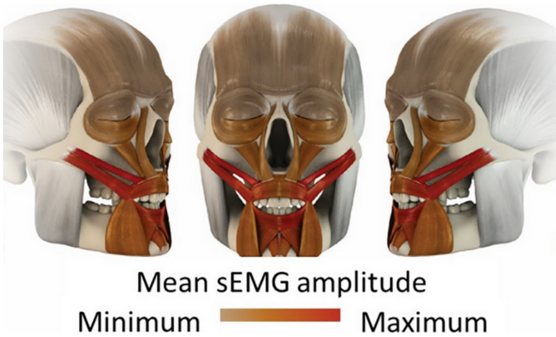
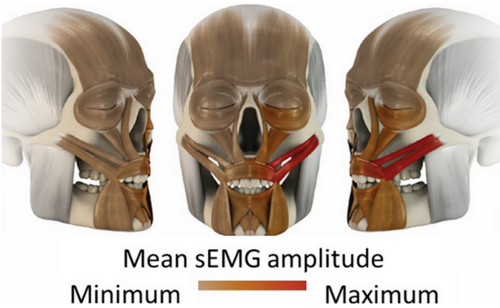


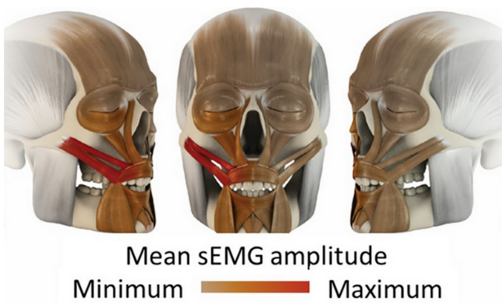
Figure 3.1: sEMG activation map for 10 face muscles in smiling[12].

Smiling with a side of the face

Smiling with a side of the face consists in pulling just one of the corners of the mouth upwards and backward. The main muscles involved are the same as in *Smiling* but only from one side of the face.



(a) Left smile.



(b) Right smile.

Figure 3.2: sEMG activation map for 10 face muscles in smiling with a side of the face[12].

Clenching teeth

Clenching of teeth is primarily controlled by the muscles of mastication. Principal involved muscles are *masseter* and *temporalis*. The first is the main muscle responsible for closing the jaw and grinding the food during chewing. *Temporalis* helps in closing the jaw, as well as retracting it. These muscles work together to produce a clenching of the jaw and teeth. However, other muscles such as the medial and lateral *pterygoid*, *digastric*, and *stylohyoid* can also play a role in jaw movement and tooth clenching.

Opening Jaw

The opening of the jaw for pronouncing vowel sound *a* is primarily controlled by the muscles that oppose the muscles of mastication. These muscles include lateral *pterygoid*; *digastric* with its two heads, that help in depressing the jaw; *mylohyoid* located in the mouth's floor, which helps in elevating the jaw and tongue. Unfortunately, the mentioned muscles are not reported in Fig. 3.3 because they were not monitored in [12].

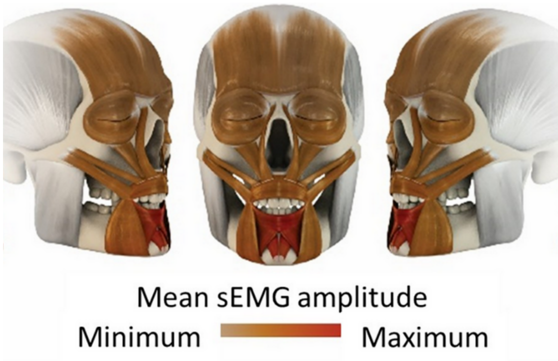


Figure 3.3: sEMG activation map for 10 face muscles in jaw opening and pronouncing the vowel *a* [12].

Raise eyebrows

Raising the eyebrows and wrinkling the forehead is controlled by *frontalis* group muscles and the *orbicularis oculi* muscle. *Frontalis muscle* is located in the forehead and is responsible for raising the eyebrows and producing wrinkles in the forehead. *Orbicularis oculi*, that encircles the eye usually helps in closing the eyelids, as well as raising the eyebrows.

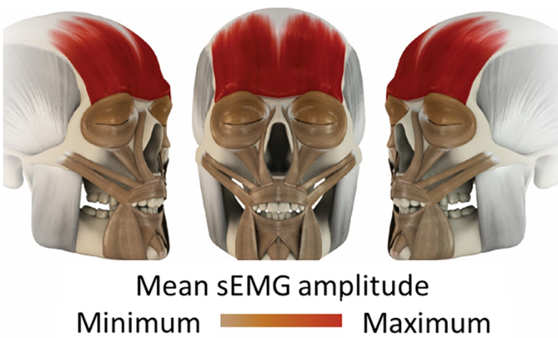


Figure 3.4: sEMG activation map for 10 face muscles in raising the eyebrows [12].

Frowning

The contraction of the eyebrows for frowning is primarily controlled by the *orbicularis oculi* muscle. Other muscles such as the *corrugator supercilii*, located in the forehead, also play a role in frowning. Together, these muscles contract and pull the eyebrows together and produce vertical wrinkles, allowing for expressions of sadness, anger, or other emotions.

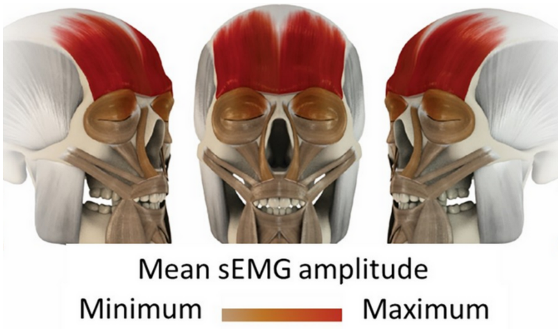


Figure 3.5: sEMG activation map for 10 face muscles in frowning (i.e., contracting the eyebrows) [12].

Closing eyes

Closing the eye forcefully is primarily controlled by the *orbicularis oculi* muscle and *frontalis* muscles. In addition, other muscles, such as the *retractor bulbi* and the *levator palpebrae superioris* from the upper eyelid, can also play a role in squeezing the eyes shut.

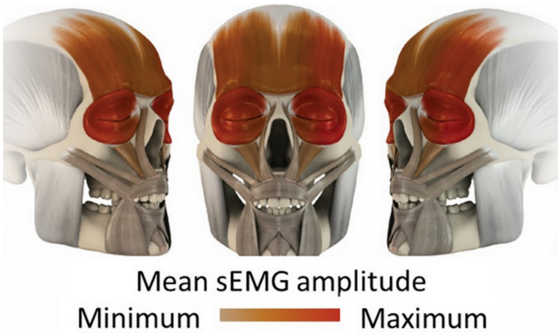


Figure 3.6: sEMG activation map for 10 face muscles in closing the eyelids forcefully[12].

Pronouncing vowel "u"

The *orbicularis oris* muscle, as well as the *buccinator* and *risorius* muscles, shape the lips for the pronunciation of the *u* sound. Temporalis and masseter muscles help to stabilize the jaw and provide the necessary force for speech production. In this case, the expression is mute, as the pronunciation of *u* is an indication to standardize the expression.

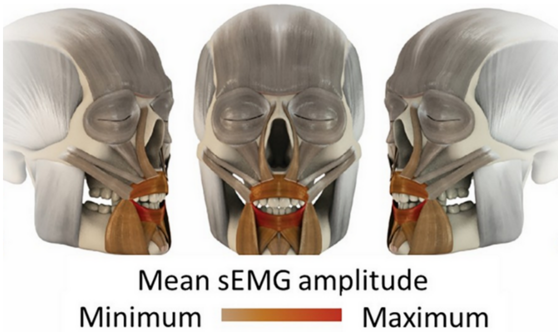
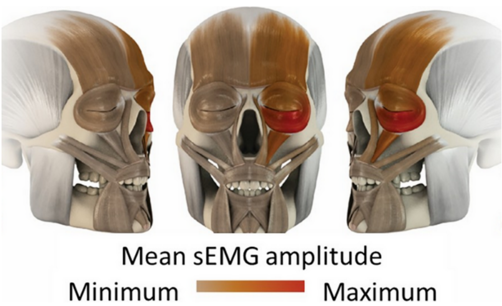


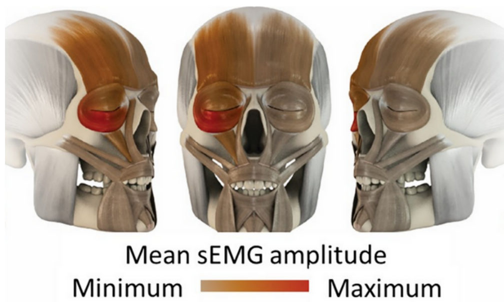
Figure 3.7: sEMG activation map for 10 face muscles in pursing the lips to pronounce the vowel *u* [12].

Eye Blink

Eye blinking, as in closing eyes expression, involves *frontalis* and *oribilaris oculi* muscles on one side of the face. When the blink is forced or exaggerated, *zygomaticus*, which is located in the cheek, and *levator labii superioris* can contract to raise the cheek.



(a) Left blink.



(b) Right blink.

Figure 3.8: sEMG activation map for 10 face muscles in blinking (i.e., closing forcefully one eye at at time) [12].

3.2 Electrodes Placement

The choice of muscles to monitor and the corresponding facial expressions performed are closely linked, as detailed in sec. 3.1. Starting from the positioning used in [43], two slightly different configurations have been considered. The muscles monitored muscles were *zygomaticus*, *corrugator supercilii*, *frontalis*, *masseter*, *dygastric*; in the alternative configuration mentalis muscle would substitute *dygastric*. As explained in sec. 1.1.3, *dygastric* muscle is mainly involved in jaw movements, and *mentalis* instead participates in various facial expressions. A preliminary investigation on this matter revealed that, while monitoring the *mentalis* muscle could add information related to facial expressions in general and especially in *u* pronunciation, the proper signal from *dygastric* in a steady jaw opening expression would be lost. Furthermore, the small accessible area on the chin for H124SG electrode size (24 mm diameter, in Fig. 3.9) would cause unstable contact.

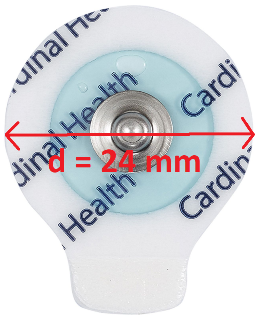


Figure 3.9: Kendall™ H124SG, foam pre-gelled Electrodes [50].

At this point, the muscles to monitor were defined: *zygomaticus*, *corrugator supercilii*, *temporalis*, *masseter*, *dygastric*. The electrode positioning is a critical aspect because of the very nature of facial muscles: they are small, interdigitated and overlapped; moreover, their approximate location and ability to use them vary from person to person[57]. Differential electrode positions (depicted in Fig. 3.10) are defined following indications from [58] and [59], in which a facial anatomical landmarks approach is suggested:

- **Masseter**: the first electrode is along the line extending from the gonion (i.e., the inferior point on the angle of the mandible) to the exocanthion (i.e., outer corner of the eye), 2 cm from the gonion; the second is placed superior and slightly medial. The suggested inter-electrodic distance (IED) is 1 cm. Masseter location can be easily determined by palpation while clenching teeth.
- **Zygomaticus**: one electrode is placed midway along the line joining the cheilion (i.e., the corner of the mouth) and the preauricular depression (i.e.,

where the ear auricle meets the cheek); the second electrode is inferior and medial to the first on the same line. IED of 1 cm is suggested.

- **Corrugator Supercilii**: the first electrode is directly above the eyebrow on a vertical line that traverses the endocanthion (i.e., the inner corner of the eye); the second electrode is lateral to the first on the border of the eyebrow. IED of 1 cm is suggested.
- **Temporalis**: the electrodes should be placed along a vertical line extending from the bony prominence formed by the zygomatic process of the frontal bone. The electrodes must be placed along the muscle fiber direction, just above the upper edge of the zygomatic arch. Clenching the teeth makes it possible to palpate the anterior temporalis and verify the correct positioning.
- **Frontalis**: In some subjects, the scalp boundary limits the electrodes' placement over the temporalis muscle. For this reason, the electrodes were moved toward the anterior portion of the face, on the lateral frontalis muscle. The electrodes are along a line extending from the center of the eye with an approximate inclination of 30° w.r.t. the vertical. In this configuration, in addition to the *frontalis* muscle, the electrical activity of a relatively vast muscle as *temporalis* is still detectable.
- **Dygastric**: with a lifted head, electrodes are positioned at the base of the mandible in the medial submental region. There are other muscles in this region, as described in section 1.1.3, so the recorded activity in this region comes from the activity of several suprahyoid muscles combined.
- **Reference electrode**: placed on the right side mastoid process, the bony prominence behind the ear. When the electrodes are repositioned on the left side of the face (as described in Section 3), the reference electrode remains on the right side mastoid process.

In a preliminary study phase to test the association between muscle activation, facial expressions, and electrode placement, trial tests were performed on a reduced sample of five subjects. The electrodes were applied to the right and left hemiface of the subjects, and they were asked to perform each facial expression in both configurations. The data collected with the system described in Chapter 2 were processed in Matlab® to analyze the associated TC events. Figures 3.11 and 3.12 show the data related to the preliminary acquisitions made on the two sides of the face. The data are represented using a parallel coordinates graph, which allows multiple variables to be visualized simultaneously along parallel axes. Each variable, one per acquisition channel, is represented on a vertical axis. The continuous line passing along each axis represents the 50_{th} percentile of the values taken by the corresponding variable. The shaded area represents the values between the 25_{th}

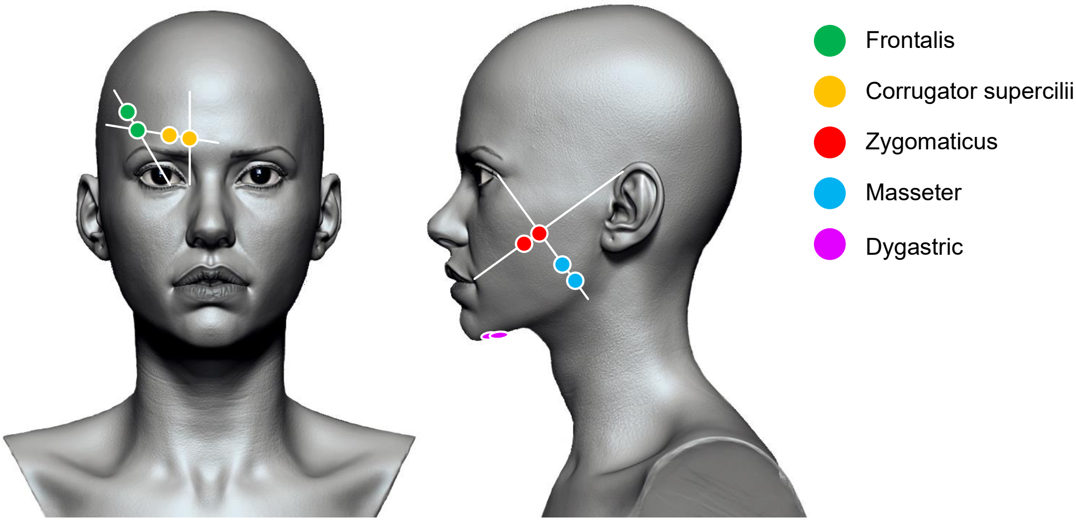


Figure 3.10: Electrodes placement for sEMG recording from selected facial muscles.

and 75_{th} percentiles instead. From images 3.11 and 3.12, it is possible to observe that each of the considered facial expressions shows a characteristic profile of activations, which suggests the possibility of using a classification algorithm for expression recognition based on the TC values. It can be noted that expressions involving only the contralateral muscles to the side of electrode placement, as expected, generate null or almost null activity. In an attempt to discriminate the activity of the two sides of the face (*Left* and *Right Blink*, and *Left* and *Right Smile*) while maintaining the same number of electrodes for data acquisition, a mixed arrangement of electrodes was simulated between the two sides of the face.

Fig. 3.13 shows an example of data resulting from the substitution of electrodes placed on the *zygomatic* muscle and the *corrugator supercilii*, with their respective contralateral electrodes, in the right-handed acquisition configuration. Muscle activity profiles do not change for symmetric expressions in the mixed configuration. It is interesting to observe that for one-sided expressions, this electrode placement configuration generates different activity profiles between left-face and right-face expressions. These observations led to selecting the right and left blinks and right and left smiles among the performed expressions, which would be difficult to distinguish from their respective two-sided versions with an electrode placement arrangement on only one side of the face.

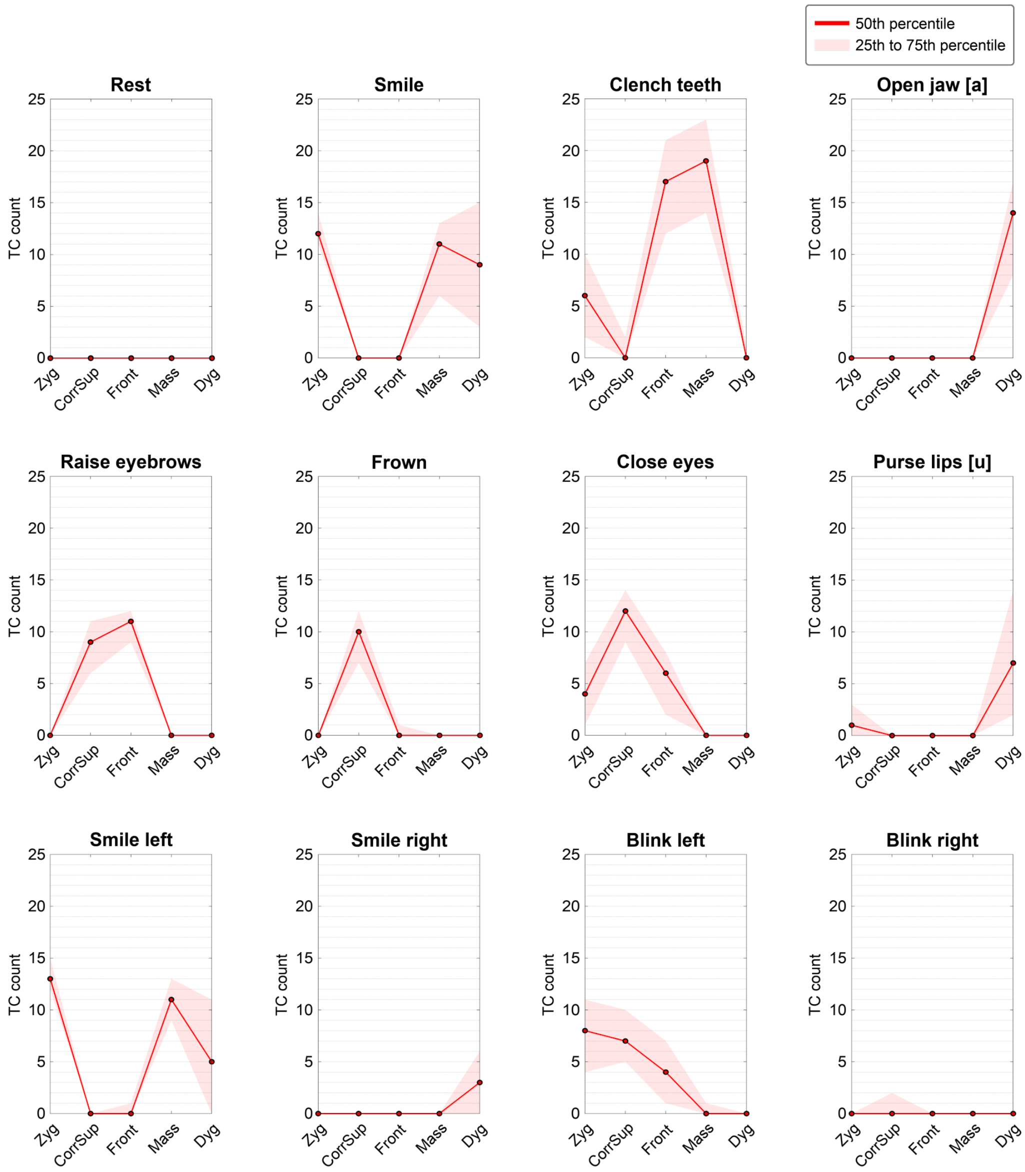


Figure 3.11: Parallel plot graphs from left side preliminary acquisitions. Each subplot represents an expression. Target muscles on x-axis are: *Zygomatic* (Zyg), Corrugator Supercilii (CorrSup), *Frontalis* (Front), *Masseter* (Mass), and *Dygastric* (Dyg). TC count is on y-axis. The shaded area represents $25th$ to $75th$ percentile of the data. The continuous red line is the $50th$ percentile of the data.

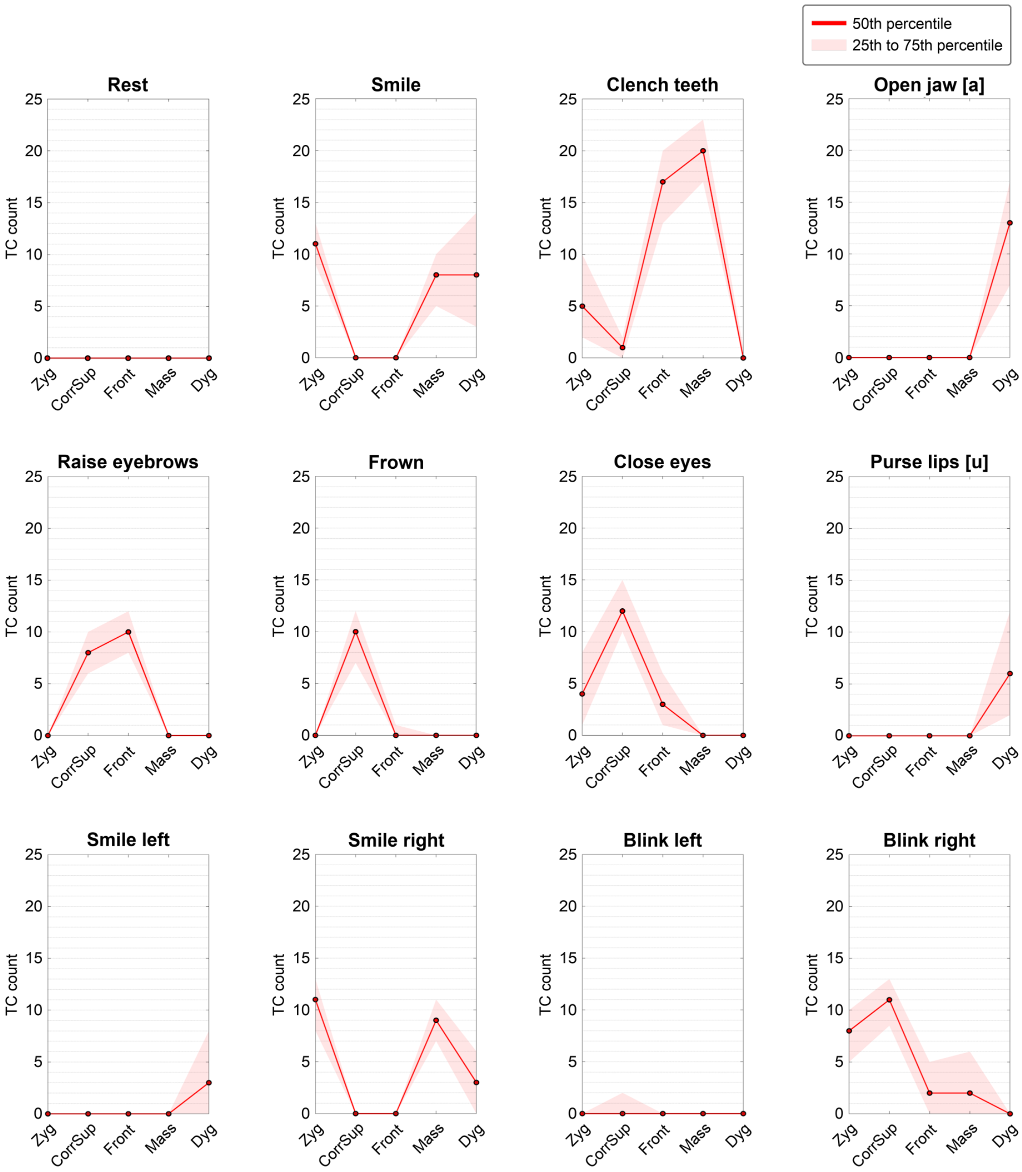


Figure 3.12: Parallel plot graphs from right side preliminary acquisitions. Each subplot represents an expression. Target muscles on x-axis are: *Zygomatic* (Zyg), Corrugator Supercilii (CorrSup), *Frontalis* (Front), *Masseter* (Mass), and *Dygastric* (Dyg). TC count is on y-axis. The shaded area represents 25*th* to 75*th* percentile of the data. The continuous red line is the 50*th* percentile of the data.

3.3 Experimental Protocol

Following preliminary analyses, an in vivo experimentation was initiated to construct a dataset that would be useful for training and testing machine learning models for

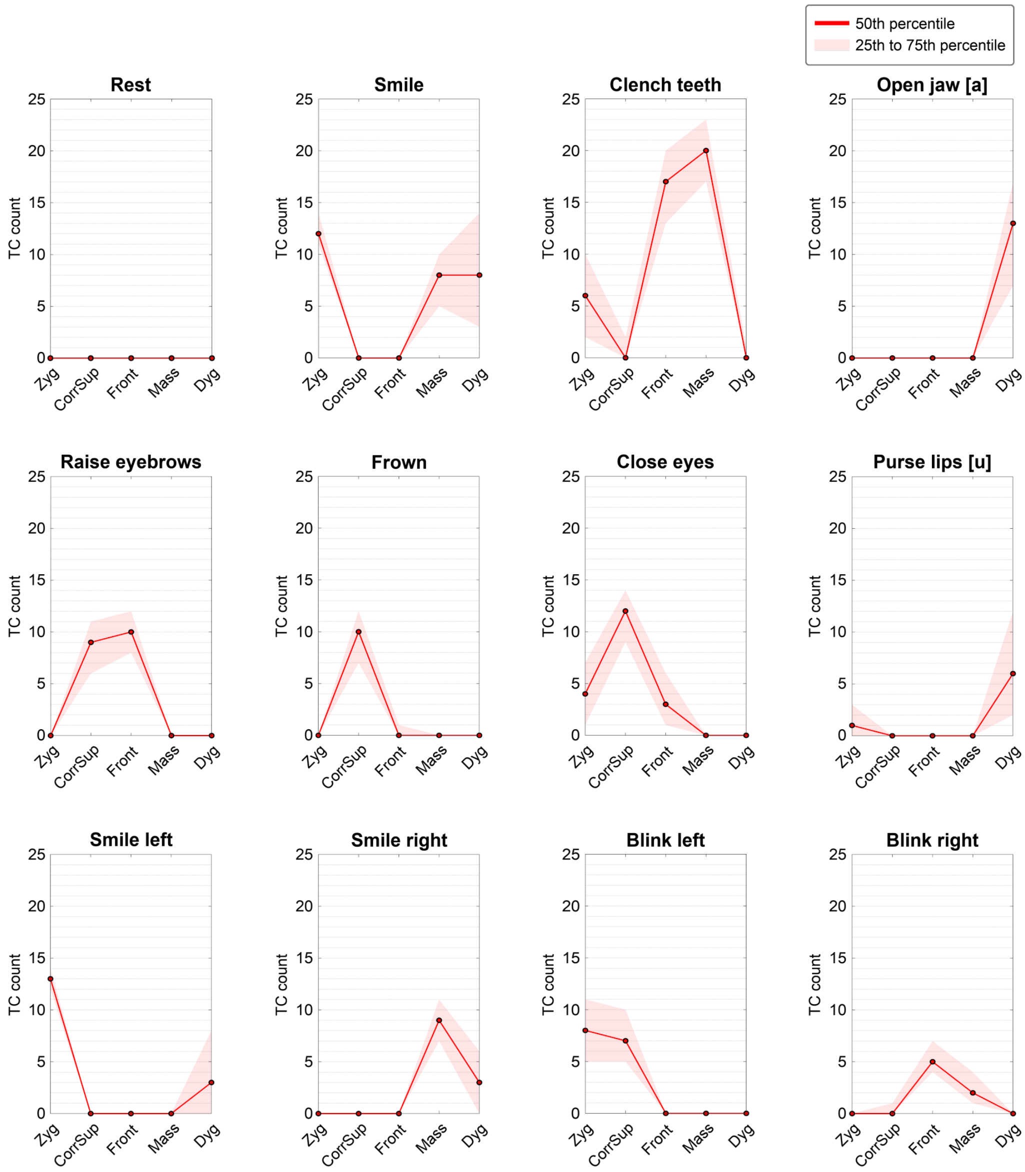
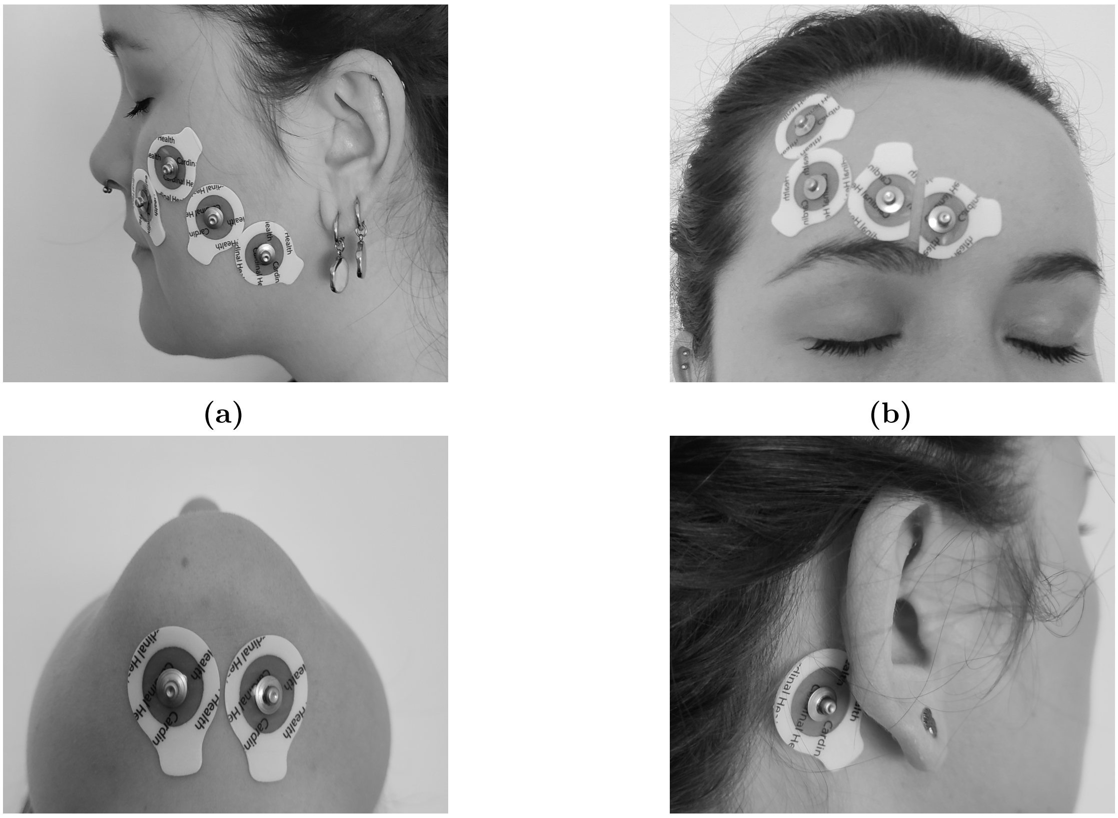


Figure 3.13: Parallel plot graph in mixed configuration: *Zygomaticus* (Zyg) and *Corrugator supercilii* muscles from left configuration; *Frontalis*, *Masseter*, and *Dygastric* muscles from right configuration.

facial expression recognition. The experimental protocol, which adhered to current regulations for scientific research involving healthy human volunteers and was promoted by Università degli Studi di Torino, has been approved by the University Bioethics Committee (Protocol Number 510188). Thirty healthy subjects, 12 males and 18 females, were recruited for the experimentation. Each subject was informed of the nature of the experimentation and the associated risks. An information module containing all the necessary details of the experiment was provided. After an interview, each subject approved and signed an informed consent under the bioethics committee guidelines. The right to privacy was ensured by assigning a progressive number, a unique identifier for each subject throughout the study. During the experiment, each subject was seated on a chair in an upright and relaxed position. Each electrode was placed as described in section 3.2. However, in some cases, deviating from the prescribed positioning was necessary to adapt to the subject's facial features and the variability of the resulting signal. Fig. 3.14 shows an example of electrode placement on a volunteer subject.



(a) **(b)** **(c)** **(d)**

Figure 3.14: Example of electrode placement on a subject: (a)*masseter* and *zygomatic*; (b)*Corrugator supercilii* and *Frontalis*; (c)*Dygastric*; (d)Reference electrode.

A brief initial calibration phase was performed to assess proper electrode placement and verify muscle activations across all channels, during which the subject was asked to perform each expression. Some electrodes were positioned slightly differently after visually inspecting the generated signals if necessary. In addition, during this phase, the proper gain value of the acquisition boards for the specific subject is assessed. The selected gain represents the minimum level required to obtain a satisfactory activation signal across all channels. Although higher gain values may enable the detection of saddle movements, they also carry a greater risk of capturing noisy signals. Once the calibration phase was completed, the acquisition could begin.

3.3.1 Training Protocol

During the execution of the protocol (schematized in Fig. 3.15), the subject is guided by the Graphical User Interface described in Sec. 2.4. The subject is seated in front of the designated testing screen. The supervisor operates the Control Panel on a secondary screen to prevent potential influences from the resulting signals. With the subject in a resting position, the threshold calibration routine is launched. Subsequently, the supervisor initiates the acquisition process and the testing session on the designated screen with the start command. After a 10 s initial rest period, the subject is guided to execute the on-screen actions as follows:

1. The subject executes and holds the on-screen expression for 7 s.
2. A 3 s rest period is observed;
3. Points 1 and 2 are repeated three times for the same expression;
4. An additional 2 s rest period is observed proceeding to the next expression, resulting in a total 5 s rest before a expression change;
5. If there are still expressions remaining to be executed, the flow restarts from point 1;
6. After all 11 movements have been executed, a session is completed, and the subject rests for 2 min;
7. The flow restart from point 1 until three sessions are completed.

Once the three sessions are completed, the described process, starting from proper electrode placement and initial calibration, is repeated on the other side of the face.

3.4 Dataset composition

The data acquired during the acquisition campaign were analyzed before they could be used for classification. As supervised machine learning method was chosen for classification (see Sec. 4), the data were labeled during the acquisitions directly by the software (see Sec. 2.4). The support software segmented the signals from

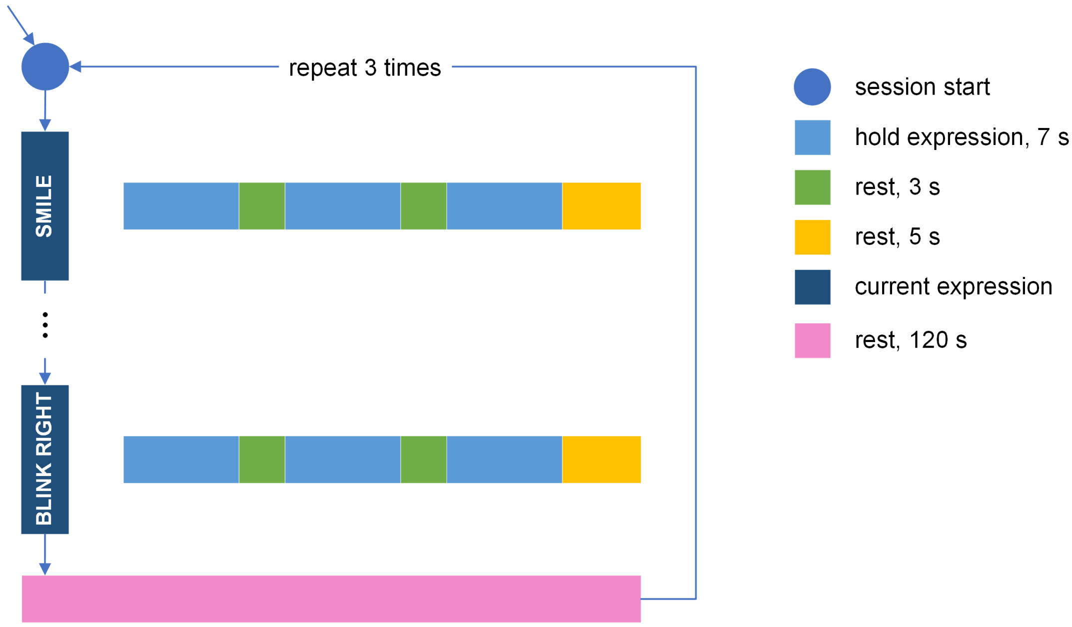


Figure 3.15: Schematic representation of the acquisition protocol for facial expression recognition, where the subject performs on-screen expressions guided by a GUI. The protocol includes three sessions with 11 expressions, each repeated three times, with rest periods in between, followed by a resting period of 2 min before repeating the process on the other side of the face.

each acquisition session with the timings detailed in Sec. 3.3, resulting in 32 s sections for each of the 11 considered expressions. Each section consists of three consecutive 7 s repetitions of the same expression, the 3 s rest period in between, and an additional 5 s rest in the section beginning. During the labeling process, also a threshold was imposed on the norm of the ATC values of each time window. The norm is calculated as in Eq 3.1, where x_i is the TC value for the i_{th} channel. According to Eq. 3.2, where Th is the chosen threshold, values with a norm greater than the threshold were identified as the class being executed at that moment; values with a norm equal to or less than the threshold were identified as Rest state.

$$N = \sqrt{x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2} \tag{3.1}$$

$$Class = \begin{cases} ActualClass, & \text{if } N > Th \\ Rest, & \text{if } N \leq Th \end{cases} \tag{3.2}$$

A $Th = 3$ was found appropriate in correcting labeling errors in cases of response delays from the subjects and still granting weakest expressions (i.e., Purse lips) to be labeled differently from rest in most cases.

The data were uploaded in Matlab® software to be elaborated. For each subject, data from the three consecutive sessions were concatenated and plotted on a parallel plot graph. This process was made separately for right and left acquisition sessions. Part of the data can be deleted if generated from problems that arose during the acquisition. In fact, data related to some expressions were excluded from the dataset for some subjects. expressions data deleted:

- Right Blink: 7 subjects.
- Left Blink: 2 subjects.
- Right Smile: 1 subject.
- Clenching teeth: 1 subject.

This choice was based on two reasons: the subject did not execute that expression, or the subject performed a poor execution because of a limited ability to contract the target muscles selectively. Removing the data was necessary not to let them negatively affect the classifier's performance. A summary of the obtained dataset is pictured in 3.16 for the acquisitions on the right side and 3.17 for the acquisitions on the left side.

The dataset has to undergo a last modification before being used to train the classifier. As explained in Sec. 3.2, it was decided to use a mixed configuration of left and right side electrodes to try a classification of one-sided expressions. The data from the *Corrugator supercilii* (Channel 2) and the *Zygomatic* muscles (Channel 3) of the left side dataset were substituted to the same channels of the right side dataset. The result of the operation is shown in figure 3.18. This dataset was then used to train a Machine Learning algorithm (see Chapter 4).

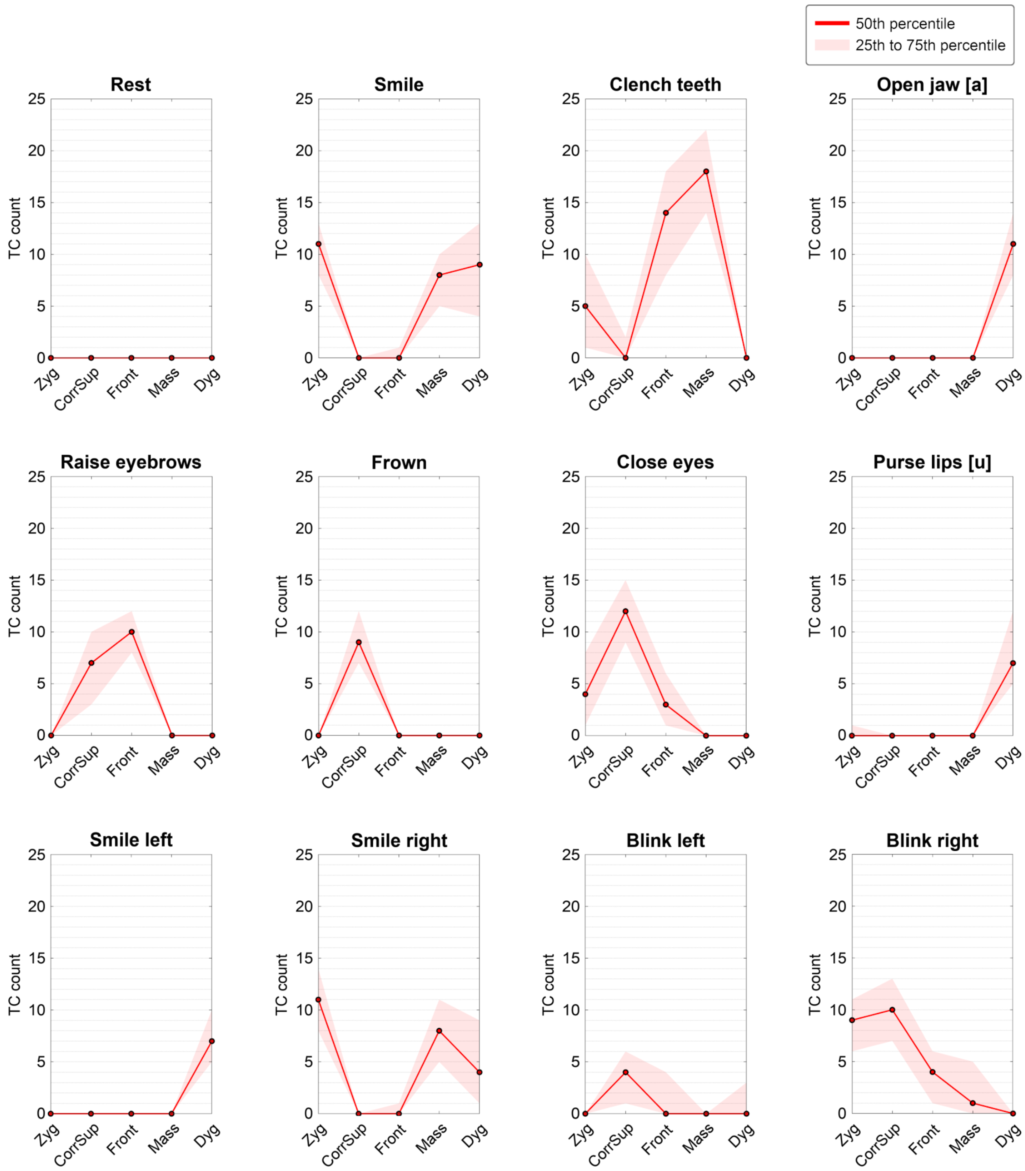


Figure 3.16: Parallel plot graphs of right side acquisition training dataset. Each subplot represents an expression. Target muscles are on x-axis. TC count is on y-axis. The shaded area represents 25th to 75th percentile of the data. The continuous red line is the 50th percentile of the data.

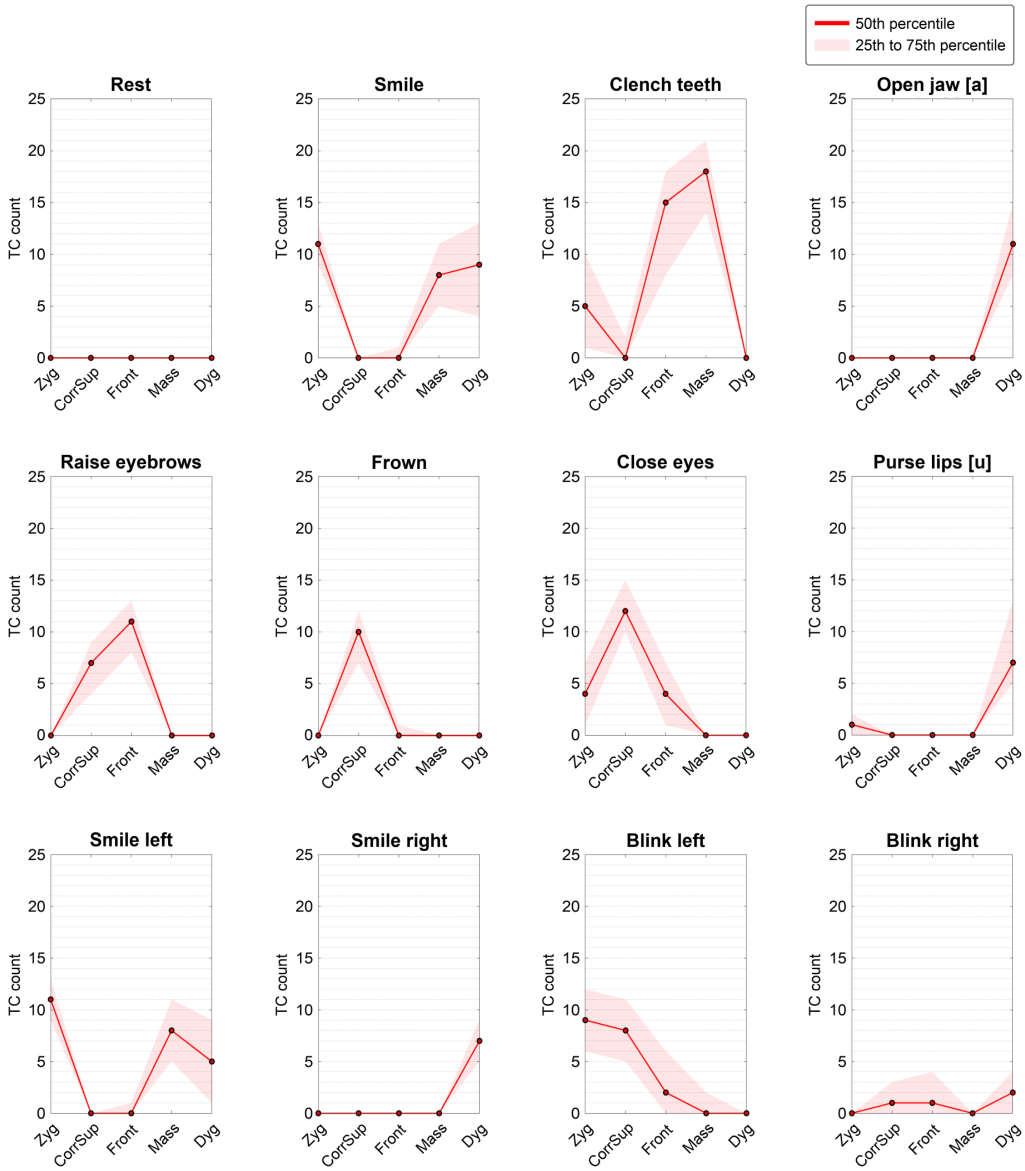


Figure 3.17: Parallel plot graphs of left side acquisition dataset. Each subplot represents an expression. Target muscles are on x-axis. TC count is on y-axis. The shaded area represents $25th$ to $75th$ percentile of the data. The continuous red line is the $50th$ percentile of the data.

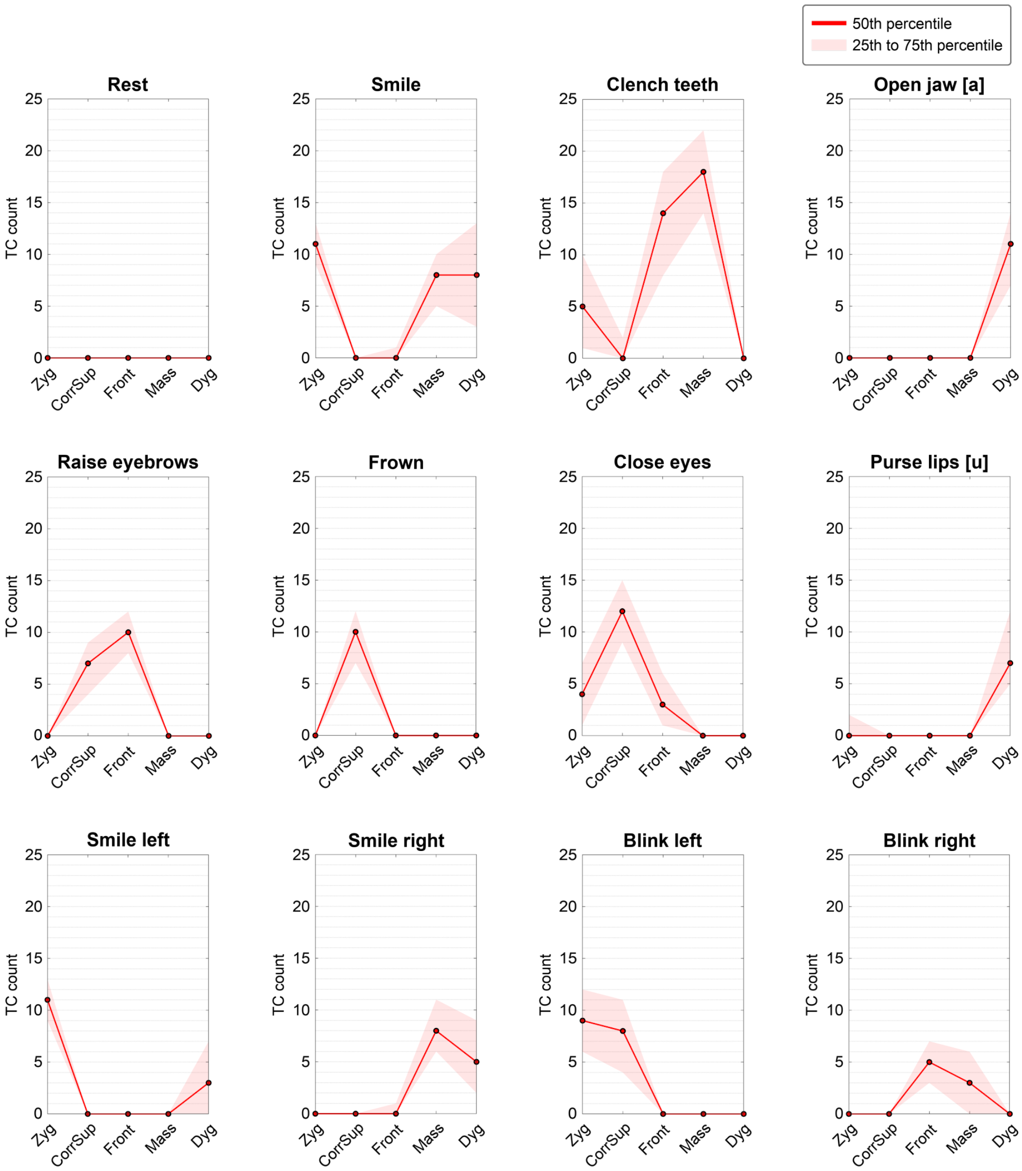


Figure 3.18: Parallel plot graph of the mixed configuration dataset. *Zygomatic* (Zyg) and *Corrugator Supercilii* (CorrSup) channels are from the left side dataset. *Frontalis* (Front), *Masseter* (Mass) and *Dygastric* (Dyg) channels are from the right side dataset.

Chapter 4

Facial expression recognition algorithm

Supervised machine learning is a type of machine learning in which an algorithm is trained to make predictions or decisions based on labeled data. In supervised learning, the algorithm receives input data and corresponding output actual classes called labels or targets. It learns to make predictions or decisions by mapping the inputs to the outputs. The goal is to generalize the learned mapping to accurately predict new, unseen data.

4.1 Artificial Neural Network

Artificial Neural Network (ANN) is a supervised learning algorithm inspired by the structure and functioning of biological neural networks, such as the human brain. An ANN consists of many interconnected processing nodes, or neurons, organized into layers. The input layer receives the input data, which is then processed through one or more hidden layers, and the output layer produces the final prediction or decision. The connections between neurons are weighted and adjusted during training to minimize the difference between the predicted and actual outputs. Given enough training data and computational resources, ANN can learn to approximate any function. ANN has been used successfully in many applications, including image and speech recognition, natural language processing, predictive modeling, and pattern recognition.

The choice of supervised learning model depended on the model size requirements, latency, memory usage, computational effort, and power consumption. The classifier architecture chosen for this thesis work is a fully connected ANN. A fully connected

ANN, also known as a dense ANN, is an artificial neural network in which all the neurons in each layer are connected to all the neurons in the previous and subsequent layers. So, the inputs of each neuron in a layer are the outputs of all the neurons in the previous layer. The outputs of each neuron in a layer are the inputs of all the neurons in the next layer. The layers of a fully connected neural network typically include an input layer, one or more hidden layers, and an output layer. The input layer receives the input data, which is processed through the hidden layers, and the output layer produces the final prediction or decision. Each neuron in the hidden layers applies a nonlinear transformation to the weighted sum of its inputs. It typically uses an activation function such as the sigmoid or ReLU functions to introduce nonlinearity and enable the network to learn complex, nonlinear relationships between the inputs and outputs [60]. A fully connected ANN is the most straightforward possible architecture, as the only mathematical operation needed for the forward propagation prediction is matrix multiplication. As mentioned in section 2.2.1, Cortex-M processor-based devices can use matrix multiplication operations through the CMSIS-DSP package. Therefore, this model is highly suitable to fit the MCU's limited hardware resources.

A drawback of this type of network is that as the number of neurons and layers increases, so does the number of parameters to be learned, making the network more complex and harder to train. Therefore, the most expensive task is training the model, which has been done offline on Matlab® environment, while the real-time prediction has been made on the Apollo3 MCU.

4.2 ANN algorithm

A custom routine for ANN training implementation has been coded in Matlab® environment as studied in Machine Learning online course by Stanford University [61]. An ANN's training can be divided into Forward Propagation and Backpropagation.

1. Forward Propagation: computing the network output for a given input.
   (a) Input Layer: The TC input data plus a bias unit are fed to the input layer.
   (b) Hidden Layers: The input is then processed through one or more hidden layers by multiplying the input with the layer's weight matrix.
   (c) Activation Function: The resulting output from every hidden layer is passed through a non-linear activation function, a Rectified Linear Unit (ReLU) function. The ReLU function applies a non-linear transformation to the input data by returning the maximum of zero and the input value. As depicted in Fig 4.1, it sets negative input values to zero and leaves

positive input values unchanged.

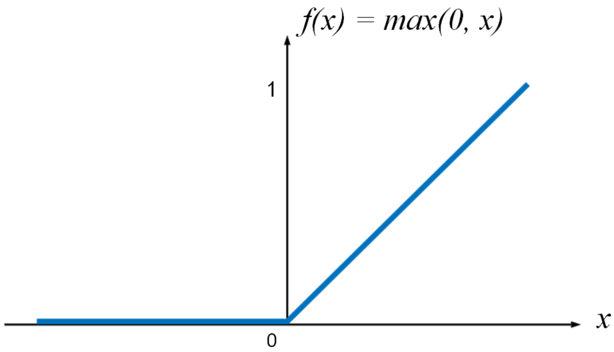


Figure 4.1: ReLU activation function: the function selects the maximum value among 0 and x.

(d) Output Layer: Matrix multiplications between the output of the activation function and the weights of the subsequent layer are repeated until the last hidden layer is reached. The output of the last hidden layer is then multiplied by the weight matrix of the output layer and passed through another activation function, such as the sigmoid function show in Fig. 4.2, to produce the final output of the network. For binary classification problems, the output can be interpreted as the probability of the input belonging to the positive class.

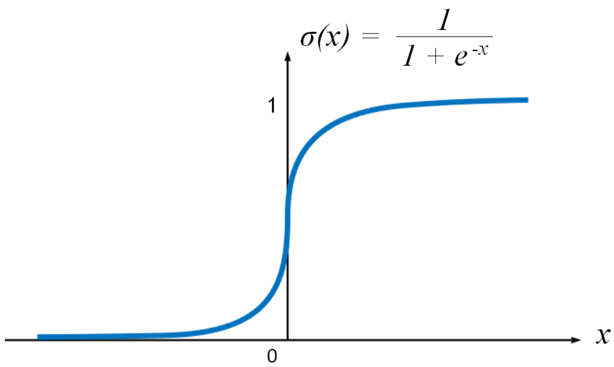


Figure 4.2: Sigmoid activation function: x values are mapped in a (0,1).

(e) Cost Function: Once the output has been computed, a cost function measures the difference between the predicted output and the true output.

In this case, a cross-entropy loss function was used. The cross-entropy loss function measures the difference between the predicted probability distribution and the true label's probability distribution. The true label for a given example is represented as a one-hot vector y, where $y_i = 1$ if the true label is class i and $y_i = 0$ otherwise. The predicted probability distribution is represented as a vector p, where p_i is the predicted probability for class i. Then the cross entropy loss function is calculated as in Eq. 4.1:

$$L_{CE} = \sum_{i=1}^{N_{classes}} -(y_i \log p_i + (1 - y_i) \log(1 - p_i)) \tag{4.1}$$

Intuitively, the cross-entropy loss measures how well the predicted probability distribution matches the true probability distribution. If the predicted probabilities match the true probabilities exactly, the cross-entropy loss is 0. However, if the predicted probabilities are far from the true probabilities, the cross-entropy loss will be high. During training, the goal is to minimize the cross-entropy loss using an optimization algorithm. By minimizing the cross-entropy loss, the neural network learns to predict the correct class with a higher probability. In addition, a regularization term λ can be added to the cost function. Regularization adds a penalty term to the cost function, which encourages the network to have smaller weights that minimize the error on the training data and result in a simpler, more general model. Regularization can help prevent the network from becoming too complex and overfitting the training data, which can cause the network to perform poorly on new, unseen data [61].

2. Backpropagation: is the process of propagating the error back from the output layer to the input layer to update the network weights using an optimization algorithm, such as gradient descent, to improve its accuracy. Backpropagation is used to calculate the gradient of the cost function with respect to the network weights and consequently adjust the weights. The gradient indicates the direction in which the weights should be updated to minimize the cost function. By repeatedly updating the weights in the opposite direction of the gradient, the neural network learns to approximate the target function and minimize the difference between its predicted output and the true output.

4.3 Offline training

Each ANN is trained and validated using the k-fold cross-validation method during the training process. In k-fold cross-validation, the dataset is randomly split up into k groups. One of the groups is used as the Cross-Validation set (CV set) and the rest as the Training Set (Train set). The model is trained on the Training set and scored on the validation set. Then the process is repeated until each unique group has been used as the Cross-Validation set. In addition, the process can be repeated r times (repetitions) with different random partitions of the dataset.

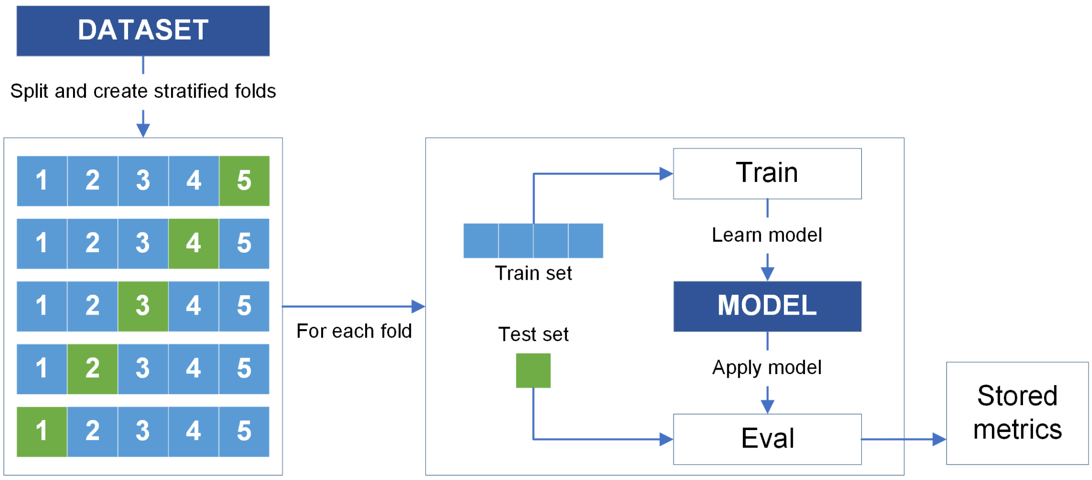


Figure 4.3: k-fold Cross-Validation process. The dataset is split in k folds. For each fold, the model is trained on the Train set and tested on a Cross-Validation set (here Test set). The evaluation metrics for each tested model are stored for further evaluation.

4.3.1 Evaluation metrics

Evaluation metrics for each k-fold trained model were stored to compare the performance of different models and select the best one. The metrics are calculated for each class separately, considering the following values:

- True Positive (TP) as the number of samples correctly classified as positive.
- True Negative (TN) as the number of samples correctly classified as negative.
- False Positive (FP) as to the number of samples incorrectly classified as positive.
- False Negative (FN) as to the number of samples incorrectly classified as negative.

The evaluation metrics are:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.2}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4.4}$$

$$F1 - score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \tag{4.5}$$

4.3.2 Training routine

A custom training routine was implemented in Matlab® to train and validate different ANN models to evaluate different structures and parameters. The main code steps are summarized here:

1. On top of the code number of hidden layers, the number of nodes and λ parameter are picked from their respective vectors. These variables define the structure and regularization term for the ANN.
2. Dataset matrix is loaded separately for each subject and $k \times r$ Train set and CV set pair are created as explained at the beginning of section 4.3.
3. In each Train set and CV set pair, classes are balanced with an undersampling method to ensure that each class has an approximately equal number of samples. They are then shuffled to prevent any inherent order in the data from affecting the model's performance. No standardization was applied because the data are evenly distributed in a range of $0 - 25$ $\mathrm{TC}_{\mathrm{events}}$.
4. The number of training iterations is set and the training procedure detailed in Sec. 4.2 starts. The number of iterations refers to the number of times the algorithm updates the model parameters. Each iteration consists of a forward propagation and a backpropagation pass based on all the training data.
5. Once the training algorithm ends, the computational error of the network is evaluated on the CV set and all the evaluation metrics are stored for further analysis detailed in Sec. 4.3.3.
6. The routine restart from point 4 until all the ANN possible structures are trained and validated using all the Train set and CV set pairs.

4.3.3 Artificial Neural Network model selection

Initially, trials were conducted to determine the optimal network structure. Forty-eight neural networks were trained and validated, with variations in the number of hidden layers, the number of nodes per layer, and the regularization term. The number of hidden layers ranged from 1 to 2, the number of nodes per layer ranged from 30 to 48, and regularization terms within 0.001 to 0.3 were used. A limited number of iterations (i.e., 100) were chosen for this preliminary phase to minimize training time. Table 4.1 shows the results of this training phase. The best network structure was selected based on the highest average F1-score metric on the CV set. The complete network structure (schematized in Fig. 4.4) consists of 1 input layer with 6 nodes (i.e., 5 inputs + 1 bias unit), 1hidden layer with 42 nodes, and 1 output layer with 12 nodes (i.e., 11 facial expressions class + rest class).

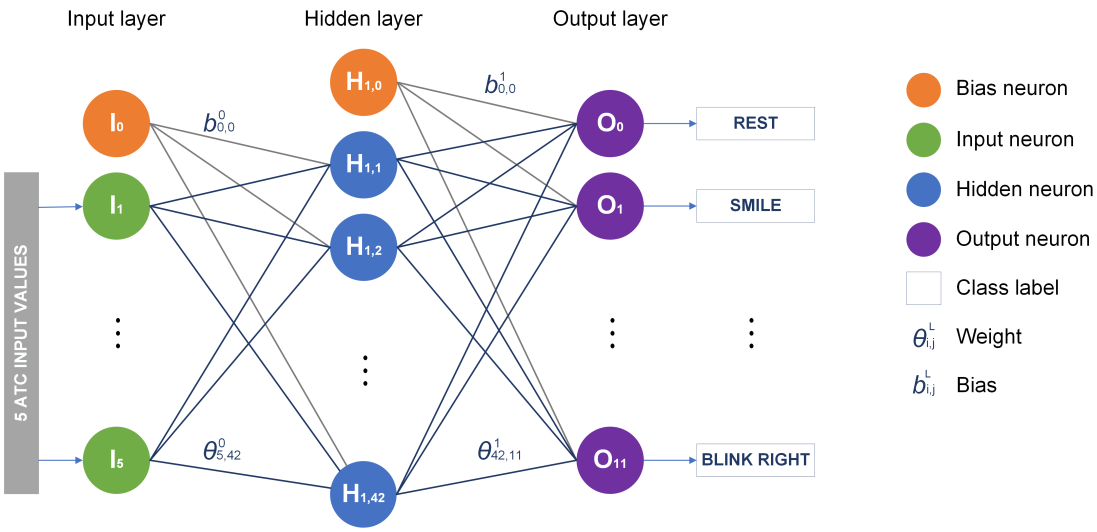


Figure 4.4: Architecture of the selected ANN with 5 ATC inputs plus a bias input, 42 hidden nodes plus a bias node, and 12 classes output nodes.

Once the network structure was selected, an additional training phase was conducted, with the number of iterations increased to 400. Extensive training increase training time but helps fit the data better. Nine models were trained with variations in the regularization term within the range of 0.001 to 10.

Figure 4.5 illustrates the k-fold Cross-Validation metrics for the chosen model, delineating the overall performance of the selected ANN model across various

Table 4.1: Preliminary training (100 iterations) for ANN model selection. Cross-Validation set averaged metrics.

Layers	Nodes	λ	Accuracy %	Precision %	Recall %	F1-score %
1	42	0.1	96.78	79.32	78.17	77.98
1	42	0.003	96.77	79.39	78.21	77.96
1	48	0.1	96.76	79.36	78.11	77.91
1	30	0.01	96.78	79.35	78.16	77.90
1	36	0.03	96.77	79.30	78.14	77.90
1	30	0.03	96.76	79.37	78.11	77.88
1	48	0.01	96.76	79.36	78.07	77.88
1	48	0.001	96.75	79.32	78.09	77.87
1	36	0.003	96.77	79.30	78.04	77.86
1	42	0.3	96.77	79.32	78.11	77.84

Table 4.2: Extensive training (400 iterations) for ANN selected model (1 hidden layer with 42 Nodes) with different regularization terms. Cross-Validation set averaged metrics.

Layers	Nodes	λ	Accuracy %	Precision %	Recall %	F1-score %
1	42	0.001	96.89	80.63	79.70	79.23
1	42	0.03	96.88	80.60	79.66	79.16
1	42	0.1	96.88	80.59	79.59	79.11
1	42	0.003	96.88	80.60	79.61	79.10
1	42	0.01	96.87	80.51	79.56	79.09
1	42	1	96.89	80.56	79.59	79.08
1	42	0.3	96.87	80.49	79.52	79.02
1	42	3	96.86	80.48	79.45	79.02
1	42	10	96.87	80.49	79.52	79.00

classes by representing them individually.

The k-fold cross-validation method generated a set of weights for each fold and repetition in the validation process. Therefore, after selecting the network based on performance averaged across the entire fold, a single set of weights was selected to create the final ANN to the MCU for predictions during the testing phase. Table 4.3 shows the averaged performance of the model for each fold. The choice of the final ANN was, once again, based on the F1-score.

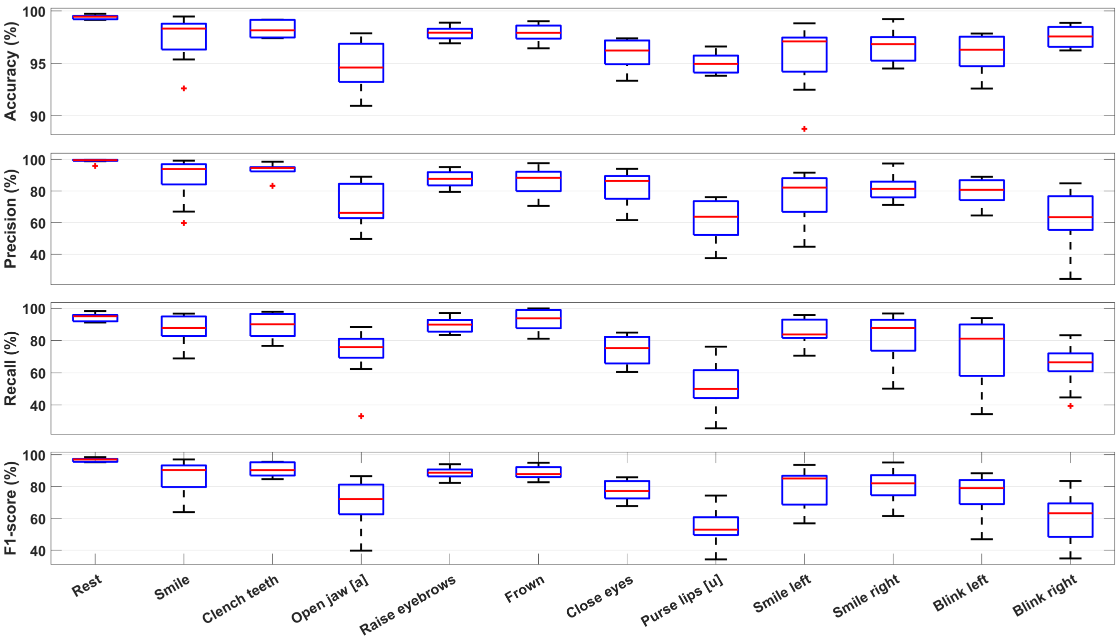


Figure 4.5: Boxplot representation of the k-fold Cross-Validation metrics for the selected ANN across classes. In each box, the central red line represents the median value, while the lower and upper edges of the box represent the 25_{th} and 75_{th} percentiles. The whiskers extend up to the most extreme data points that are not considered outliers, and the outliers are individually plotted using the + symbol.

Table 4.3: Detailed metrics for ANN selected model (1 Layer - 42 Nodes - 0.001 λ) for each of the 5 Folds and 2 repetitions. Cross-Validation set averaged metrics.

Repetition	k-fold	Accuracy %	Precision %	Recall %	F1-score %
1	1	97.93	88.33	87.07	87.33
1	3	97.67	84.95	83.91	84.04
2	3	97.53	84.17	84.06	83.69
1	2	97.53	83.83	83.63	83.41
2	5	97.40	84.14	82.67	83.06
2	2	97.43	82.54	83.29	82.38
1	5	96.80	80.55	77.39	77.32
2	1	96.17	82.54	74.98	74.87
2	4	95.83	72.90	73.19	71.03
1	4	94.64	67.94	66.80	65.21

4.4 Real-time prediction

The weights parameters of the selected ANN are transferred to the Apollo3 MCU firmware. As explained in Sec. 2.2.1, every time the ATC window (i.e., 130 ms expires, the MCU exits the deep sleep mode. The TC values counted by the GPIO interrupts from the acquisition boards are transformed into inputs for the ANN. The forward propagation routine (detailed in Sec. 4.2) transforms the given input to the predicted class output. The only difference in the forward propagation for real-time prediction w.r.t the offline training routine is in the last layer. The output layer of the MCU Neural Network does not include the sigmoid activation function. The sigmoid function was necessary for output normalization for cross-entropy loss function computation for training purposes. The *argmax* function from the CMSIS-DSP library is used instead to obtain the index of the output node with the highest value (i.e., predicted class). As detailed in Sec. 2.3 and 2.4, the predicted class, along with the ATC values, if requested, and the relative packet numbers are sent to the external device via BLE and stored for later analysis.

Chapter 5

Experimental results

After training the ANN model for real-time usage, the prototype was tested in terms of prediction accuracy, latency of the classifier, and power consumption. The results from the testing phase will be detailed in the following sections.

5.1 Online testing phase

For the testing phase, an acquisition campaign involved 6 additional subjects not included in the training phase. The testing protocol followed the same scheme as the training phase, detailed in Sec. 3.3. The only difference was that the electrodes were positioned in the mixed configuration to match the training dataset (see Sec. 3.4). The *zygomaticus* and *corrugator supercilii* channels were positioned on the left hemiface, while the *frontalis*, *masseter*, and *dygastric* related channel on the right hemiface. The single electrode placement over the target muscle followed the procedure described in Sec. 3.2. As a consequence, 3 acquisition sessions for each subject were performed. The subjects were instructed in the facial expressions execution. When the subjects were ready, the acquisition started requesting ATC and ML concurrent notifications from the GUI, as explained in Sec. 2.4. The subject was guided through the sessions using a secondary application screen, and the prediction results were not shown to avoid expression adjustments.

5.1.1 ANN classification results

The data resulting from the testing phase were imported in Matlab® for classification metrics evaluation. The class expressions *predicted* by the system were compared to the *true* labels. The *true* labels were created by the software with the same procedure described in Sec. 3.4, and the same rest norm applied in the training phase was used. The *predicted* labels are the corresponding class predictions the system gives, as described in Sec. 4.4. To balance the testing set classes, only a portion of the signal acquired during the resting phases was considered. Following the testing protocol, the time spent in a session performing each movement was 21 s. Therefore, 2 s from each of the rest states between different expressions were considered because the rest signals in between expressions repetition tended to be more affected by the effects of the subject's reaction time. The predicted values from all the testing subjects were added together and reported in a confusion matrix in Fig. 5.1. The resulting classification metrics are shown in Fig. 5.1.

Confusion matrix

True Class \ Predicted Class	Rest	Smile	Clench teeth	Open jaw [a]	Raise eyebrows	Frown	Close eyes	Purse lips [u]	Smile left	Smile right	Blink left	Blink right
Rest	3070	1	0	12	0	2	0	6	0	7	0	13
Smile	15	2468	13	25	0	0	0	83	45	12	2	1
Clench teeth	4	5	2601	0	1	0	8	0	2	28	0	12
Open jaw [a]	11	27	2	2515	9	0	0	74	54	23	0	14
Raise eyebrows	50	0	0	0	2562	66	10	0	0	0	1	31
Frown	55	0	0	0	156	2363	165	2	0	0	9	2
Close eyes	143	1	13	0	29	52	2103	1	2	0	336	28
Purse lips [u]	451	0	0	758	2	1	0	1175	32	5	1	52
Smile left	80	27	1	22	0	0	1	108	2510	2	41	6
Smile right	116	6	29	58	0	0	0	31	0	2482	0	16
Blink left	93	0	2	0	0	13	227	1	63	0	2313	0
Blink right	64	0	199	35	455	2	196	8	0	76	1	1530

Evaluation metrics (%)

	Accuracy	**Precision**	**Recall**	**F1-score**
Rest	96,56	73,94	98,68	84,54
Smile	99,19	97,36	92,64	94,94
Clench teeth	99,02	90,94	97,75	94,22
Open jaw [a]	96,56	73,43	92,16	81,74
Raise eyebrows	97,52	79,71	94,19	86,35
Frown	98,39	94,56	85,86	90,00
Close eyes	96,29	77,60	77,66	77,63
Purse lips [u]	95,05	78,91	47,44	59,25
Smile left	98,51	92,69	89,71	91,17
Smile right	98,75	94,19	90,65	92,39
Blink left	97,58	85,54	85,29	85,41
Blink right	96,29	89,74	59,63	71,65
Avg.	**97,48**	**85,72**	**84,30**	**84,11**

Figure 5.1: Testing dataset confusion matrix (left) and statistical results (right).

From the confusion matrix and the relative metrics in Fig. 5.1, the best results expressions are *smile* (94.94% F1-score and 99.19% accuracy) and *clench teeth* (94.22% F1-score and 99.02% accuracy), confirming the trend from the previous thesis work [43]. Similar high results are shown by *smile right* and *smile left*, confirming the possibility of distinguishing unilateral expressions with the mixed electrodes configuration. Nonetheless, lateral smiles are rarely confused with an entire smile. On the contrary, *purse lips* and *blink right* expressions show the worst results. It is evident from the confusion matrix that a *purse lips* expression is often classified as an open jaw. The reason for these misclassifications is that, as reported in Fig. 3.13 from Sec. 3.4, both the expressions generate similar activation signals from the *dygastric* channel. In addition, the activity generated from the *orbicularis oris* in pursing the lips is not monitored in the present configuration, even though a part of this activity is detected under the *zygomatic* channel in a few subjects. The relatively high number of *purse lips* misclassified as *rest* is caused by a very low muscular activation level detected from some subjects. The other considerable misclassifications in the matrix affect expressions involving almost only the activity from the eyebrows or from the area surrounding the eyes (e.g., *close eyes*, *raise eyebrows*, and *blink right*), as a result of the different strategy the subjects uses to contract those muscles selectively. It is evident that in these last cases (i.e., blinks), the mixed configuration used to distinguish unilateral expressions has been less effective compared with smiles, possibly worsening the classification for the aforementioned eyebrows-related expressions. Looking at the *Rest* column of the confusion matrix, the number of active expressions misclassified as *rest* causes a drop in the precision metric. Except for the *purse lips* class already discussed, the problem arose for those subjects who struggled to maintain a prolonged muscle activation level. The classification metrics results by test subjects are presented in appendix A.

5.2 System latency

The timing latency of the system was measured using a 6 MHz timer of the Apollo3. During the execution of the ML mode, the timer was started immediately before the execution of the ANN prediction. As soon as the output prediction is returned, the timer value is read, converted into ms, and saved. The values obtained from 1 min acquisition were almost constant because the ANN embedded implementation (detailed in Sec. 4.4) always operates the same amount of calculations. The averaged data resulted in a mean prediction latency of 0.627 ms. Considering an HMI possible application of the system, the maximum application latency to control an external device would be 205.627 ms, below the 300 ms threshold requirement for real-time control of prostheses [62]. The application latency takes into account the following time contribution: the ATC window contribution (130 ms), the maximum BLE connection interval (75 ms), and the prediction latency (0.627 ms).

5.3 MCU current consumption

MCU power consumption analysis has been performed using external equipment. An INA240A1 [63] version of the current sense amplifier was used, which has a 10 Ω sense resistor at the amplifier inputs and a 20 V/V gain. The measurements were made by removing a jumper between the board power and the MCU V_{DD} and inserting the amplifier's inputs. The amplifier's output was captured with a digital oscilloscope [64] and saved. For each of the active states of the MCU board 10 s of data were acquired:

- Advertising: the board begins broadcasting its presence to prospective listening devices a short time after being turned on, with a 500 ms interval;
- Connection: once the board and the external BLE device are connected, both devices wait for the other to send data. They periodically (i.e., 75 ms max BLE connection interval) check if the other is still active. After 6 s of no data exchange, the periodic check occurs every 5 elapsed connection intervals (i.e., 375 ms;
- ATC mode: when ATC notifications are enabled, the board starts counting the TC events. Every fixed time window (i.e., 130 ms), the board wakes from *deep sleep* mode, computes the ATC related calculation, and requests the notification to be sent on the next available connection interval;
- ML mode: similar to the ATC mode, the board counts the TC events. However, only the ML notifications are sent. In addition, the board computes ANN forward propagation calculations each time an ATC values set is acquired;
- ATC mode and ML mode: the two previous modes are combined, with the board sending ATC and ML notifications.

The averaged MCU current absorption for the different active states is summarized in Table 5.1. As expected, the lower power consumption is associated with Advertising and Connection states, while the highest is associated with ML and ATC combined mode.

	Advertising	**Connection**	**ATC**	**ML**	**ATC and ML**
Avg. Current Absorption (mA)	0.2914	0.2922	0.5881	0.5824	0.5934

Table 5.1: MCU Current Absorption averaged over a 10 s period in different active states.

In Fig. 5.2, a current absorption waveform of 500 ms for ML and ATC combined mode is reported to observe the system's behavior. BLE connection intervals are represented by high peaks in current absorption (i.e., 12 mA) with a 75 ms interval. Slightly lower current peaks are measured at the end of each ATC window (i.e.,130 ms) when the board exits from *deep sleep* to compute ATC and prediction-related operations. The periodic current peak request is short in time, so the mean absorption value (i.e., 0.593 mA) is still similar to the baseline.

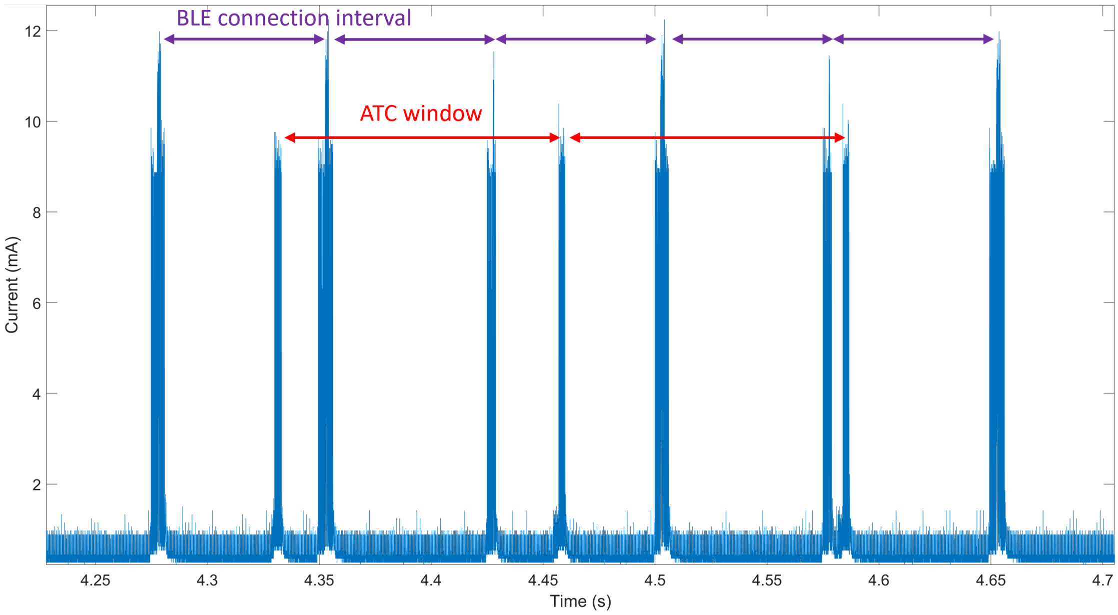


Figure 5.2: Digitized waveform of the MCU current absorption during ATC and ML mode functioning. The consumption related to ATC and ML operations at the end of each ATC window is indicated in red. The peaks related to BLE connections are in purple. The average current absorption measured is 0.593 mA.

Chapter 6

Conclusions

This thesis proposed the integration of an embedded prototype for real-time facial expression recognition based on the surface ElectroMyoGraphic (sEMG) signal from the muscles of the face and neck. The Averaged Threshold Crossing (ATC) technique, an event-driven bio-inspired approach, has been used as a feature extraction method. The final system met the requirements for real-time applications thanks to low response latency and promising classifier performance.

The device comprises an Apollo3 MicroController Unit (MCU) and 5 Analog Front Ends (AFE). The AFEs, under I^2C commands from the MCU, filter the sEMG signal from face muscles and extract the TC quasi-digital signals. The TC output channels are connected as inputs to the MCU General Purpose Input/Output (GPIO) to calculate the ATC and make predictions. The MCU streams the data through its Bluetooth Low Energy (BLE) v.4.2 transceiver to a control platform. The platform's software, developed in Python, handles the data stream and allows control of the system functionality by interacting with a Graphical User Interface (GUI).

In the first phase, exploiting the knowledge gained from a feasibility study carried out in previous thesis work, the expressions to perform and the related convenient electrode positioning were studied. A new acquisition campaign involving 24 healthy subjects was launched to obtain a sufficiently large dataset to train an Artificial Neural Network (ANN) to classify 11 facial expressions. The Machine Learning (ML) model selection led to an embedded implementation of an ANN with 5 input features, 1 hidden layer with 42 nodes, and 12 output nodes (11 facial expressions and the rest state).

An *in vivo* test phase involved 6 more subjects in testing the system's performance, obtaining an average classification accuracy of 97.4%. During active ATC mode and

ML combined mode functioning, power consumption measurements on the MCU showed a 0.593 mA mean current absorption. The average latency of the classifier measured using an MCU timer was 0.627 ms. The sum of the ATC window, the prediction latency, and the maximum BLE connection interval give an application latency of 205.627 ms for a possible HMI application.

6.1 Future directions

Future developments of this thesis work could concern different aspects of the system.

The following works could focus on the implementation of other Machine Learning embedded algorithms (e.g., Support Vector Machine (SVM), k-Nearest Neighbors (k-NN), and Decision trees) to evaluate how they impact classification, power consumption, and latency. Unsupervised Machine Learning techniques could be used for clustering the data before training a supervised learning algorithm. Clustering could give additional insight into the dataset to upgrade the output labels. Another critical improvement could aim at the adaptability of the system to individuals through an automatic calibration routine to ease the manual adaptations of electrode placement.

New acquisition protocols and different electrode positioning could be planned for specific applications not comprehensively investigated in this work. For example, focusing on the perioral and neck muscles could provide helpful information about the movements and positions of the mouth and larynx for speech recognition. Moreover, it has been suggested that emotional facial expressions can be detected with sufficient accuracy from sEMG eyebrow activity alone, as basic emotions are associated with distinct eyebrow movements. As facial electrodes can potentially hinder spontaneous facial expressions, focusing on eyebrows activity could be a strategy to explore this application.

The hardware can be upgraded or modified based on the application's specific requirements. As the presented system is an evaluation prototype, integrating the MCU and acquisition channels on the same support would minimize the size and weight of the system for ease of use and portability. In addition, to create a wearable device for long-term monitoring, support for carrying the device and holding the electrodes in position may be added to make the device more comfortable and convenient for the user. An headcap or a lightweight, flexible, and adjustable headband would be viable options.

Appendix A

ANN classification results by test subject

Subject 1

Evaluation metrics (%)				
Expression	Accuracy	Precision	Recall	F1-score
Rest	99,22%	93,28%	99,04%	96,07%
Smile	99,78%	99,53%	97,70%	98,61%
Clench teeth	97,79%	79,30%	98,39%	87,82%
Open jaw [a]	96,79%	72,25%	97,94%	83,15%
Raise eyebrows	98,61%	88,66%	95,56%	91,98%
Frown	99,39%	96,57%	95,91%	96,24%
Close eyes	98,46%	89,10%	91,04%	90,06%
Purse lips [u]	96,14%	98,61%	50,96%	67,19%
Smile left	99,72%	98,47%	98,26%	98,36%
Smile right	99,61%	97,05%	98,50%	97,77%
Blink left	99,39%	95,70%	97,16%	96,42%
Blink right	95,77%	85,58%	60,00%	70,54%
Avg.	**98,39%**	**91,17%**	**90,04%**	**89,52%**

Subject 2

Evaluation metrics (%)				
Expression	Accuracy	Precision	Recall	F1-score
Rest	97,48%	82,16%	96,35%	88,69%
Smile	99,27%	93,46%	97,80%	95,58%
Clench teeth	99,72%	98,47%	97,97%	98,22%
Open jaw [a]	95,30%	69,61%	80,87%	74,82%
Raise eyebrows	94,79%	61,59%	95,65%	74,93%
Frown	94,60%	83,55%	44,88%	58,40%
Close eyes	94,24%	64,15%	72,92%	68,26%
Purse lips [u]	95,64%	85,25%	52,79%	65,20%
Smile left	98,35%	90,53%	90,11%	90,32%
Smile right	99,35%	94,42%	97,83%	96,09%
Blink left	98,13%	83,19%	96,07%	89,17%
Blink right	95,81%	95,12%	48,99%	64,68%
Avg.	**96,89%**	**83,46%**	**81,02%**	**80,36%**

Subject 3

Evaluation metrics (%)				
Expression	Accuracy	Precision	Recall	F1-score
Rest	98,98%	90,94%	99,22%	94,90%
Smile	99,83%	99,53%	98,39%	98,96%
Clench teeth	98,69%	86,35%	99,31%	92,37%
Open jaw [a]	97,76%	79,92%	96,82%	87,56%
Raise eyebrows	97,52%	78,06%	97,31%	86,63%
Frown	99,69%	97,41%	98,91%	98,15%
Close eyes	98,04%	81,90%	97,51%	89,03%
Purse lips [u]	97,04%	98,30%	59,69%	74,28%
Smile left	99,50%	98,03%	96,15%	97,08%
Smile right	99,44%	98,86%	94,54%	96,65%
Blink left	99,22%	95,01%	95,84%	95,42%
Blink right	94,85%	88,89%	46,25%	60,85%
Avg.	**98,38%**	**91,10%**	**89,99%**	**89,32%**

Subject 4

Evaluation metrics (%)				
Expression	Accuracy	Precision	Recall	F1-score
Rest	97,17%	75,36%	99,61%	85,81%
Smile	99,69%	97,61%	98,59%	98,10%
Clench teeth	99,70%	98,43%	98,04%	98,23%
Open jaw [a]	97,93%	80,87%	97,98%	88,61%
Raise eyebrows	99,87%	98,62%	99,80%	99,21%
Frown	99,70%	97,83%	98,61%	98,21%
Close eyes	94,97%	70,07%	74,05%	72,01%
Purse lips [u]	95,07%	86,96%	45,36%	59,62%
Smile left	99,22%	99,13%	91,38%	95,10%
Smile right	99,37%	94,82%	97,82%	96,30%
Blink left	95,06%	72,25%	66,99%	69,52%
Blink right	99,39%	98,52%	93,94%	96,17%
Avg.	**98,10%**	**89,20%**	**88,52%**	**88,07%**

Subject 5

Evaluation metrics (%)				
Expression	Accuracy	Precision	Recall	F1-score
Rest	93,58%	60,68%	99,23%	75,31%
Smile	99,39%	100,00%	92,92%	96,33%
Clench teeth	98,89%	90,26%	97,80%	93,88%
Open jaw [a]	97,10%	78,94%	91,03%	84,55%
Raise eyebrows	97,41%	81,67%	89,71%	85,50%
Frown	99,62%	97,84%	97,84%	97,84%
Close eyes	97,87%	94,01%	80,22%	86,57%
Purse lips [u]	94,68%	70,98%	38,06%	49,55%
Smile left	97,27%	87,30%	81,57%	84,34%
Smile right	98,04%	95,90%	79,95%	87,20%
Blink left	96,65%	81,91%	74,94%	78,27%
Blink right	96,26%	74,47%	56,27%	64,10%
Avg.	**97,23%**	**84,50%**	**81,63%**	**81,95%**

Subject 6

Evaluation metrics (%)				
Expression	Accuracy	Precision	Recall	F1-score
Rest	92,86%	57,16%	98,65%	72,38%
Smile	97,18%	92,99%	69,95%	79,84%
Clench teeth	99,29%	96,03%	94,92%	95,47%
Open jaw [a]	94,26%	61,09%	87,85%	72,06%
Raise eyebrows	96,49%	75,38%	86,80%	80,68%
Frown	97,04%	87,47%	75,70%	81,16%
Close eyes	94,26%	69,70%	51,80%	59,43%
Purse lips [u]	91,79%	47,70%	38,34%	42,51%
Smile left	96,89%	82,31%	80,90%	81,60%
Smile right	96,63%	83,25%	74,45%	78,60%
Blink left	97,35%	84,86%	83,19%	84,01%
Blink right	95,26%	89,18%	46,82%	61,40%
Avg.	**95,77%**	**77,26%**	**74,11%**	**74,25%**

Bibliography

[1] Caifeng Shan, Shaogang Gong, and Peter W McOwan. «Facial expression recognition based on local binary patterns: A comprehensive study». In: *Image and vision Computing* 27.6 (2009), pp. 803–816 (cit. on p. 1).

[2] Erik Murphy-Chutorian and Mohan Manubhai Trivedi. «Head pose estimation in computer vision: A survey». In: *IEEE transactions on pattern analysis and machine intelligence* 31.4 (2008), pp. 607–626 (cit. on p. 1).

[3] Md Rezwanul Ahsan, Muhammad I Ibrahimy, Othman O Khalifa, et al. «EMG signal classification for human computer interaction: a review». In: *European Journal of Scientific Research* 33.3 (2009), pp. 480–501 (cit. on p. 1).

[4] Marco Crepaldi, Marco Paleari, Alberto Bonanno, Alessandro Sanginario, Paolo Ariano, Duc Hoa Tran, and Danilo Demarchi. «A quasi-digital radio system for muscle force transmission based on event-driven IR-UWB». In: *2012 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. 2012, pp. 116–119. DOI: `10.1109/BioCAS.2012.6418406` (cit. on pp. 1, 19).

[5] D.A. Neumann. *Kinesiology of the Musculoskeletal System: Foundations for Rehabilitation*. Volve learning system. Mosby/Elsevier, 2010. ISBN: 9780323039895 (cit. on p. 3).

[6] G.J. Tortora and B.H. Derrickson. *Principles of Anatomy and Physiology, 14th Edition: 14th Edition*. Wiley, 2013. ISBN: 9781118808856 (cit. on pp. 3, 5).

[7] M. Schuenke, E. Schulte, and U. Schumacher. *General Anatomy and Musculoskeletal System - Latin Nomencl. THIEME Atlas of Anatomy*. THIEME Atlas of Anatomy Series. Thieme, 2011. ISBN: 9781604062939 (cit. on p. 4).

[8] L. Mesin. *Introduction to Biomedical Signal Processing*. La community di ilmiolibro.it. ilmiolibro self publishing, 2017. ISBN: 9788892332485 (cit. on p. 6).

[9] E.N. Marieb, K. Hoehn, and Pearson. *Human Anatomy and Physiology, Global Edition*. Always learning. Pearson Education, Limited, 2015. ISBN: 9781292096971 (cit. on pp. 6–8, 10).

[10] B. Zebian, W. Lam, and R. Aggarwal. *One Stop Doc Musculoskeletal System*. One Stop Doc. CRC Press, 2005. ISBN: 9781444114386 (cit. on p. 7).

[11] Andrew J Fuglevand, DAVID A Winter, and AFTAB E Patla. «Models of recruitment and rate coding organization in motor-unit pools». In: *Journal of neurophysiology* 70.6 (1993), pp. 2470–2488 (cit. on p. 7).

[12] Nikolaus P Schumann, Kevin Bongers, Hans C Scholle, and Orlando Guntinas-Lichius. «Atlas of voluntary facial muscle activation: Visualization of surface electromyographic activities of facial muscles during mimic exercises». In: *Plos one* 16.7 (2021), e0254932 (cit. on pp. 7, 21, 37–41).

[13] L. Mesin. *Neuromuscular System Engineering*. La community di ilmiolibro.it. ilmiolibro self publishing, 2019. ISBN: 9788892361881 (cit. on pp. 12–14).

[14] Elwood Henneman, George Somjen, and David O Carpenter. «Functional significance of cell size in spinal motoneurons». In: *Journal of neurophysiology* 28.3 (1965), pp. 560–580 (cit. on p. 12).

[15] Susanna Rampichini, Taian Martins Vieira, Paolo Castiglioni, and Giampiero Merati. «Complexity Analysis of Surface Electromyography for Assessing the Myoelectric Manifestation of Muscle Fatigue: A Review». In: *Entropy* 22.5 (2020). ISSN: 1099-4300. DOI: 10.3390/e22050529. URL: https://www.mdpi.com/1099-4300/22/5/529 (cit. on pp. 12, 13).

[16] David Chambers, Christopher Huang, and Gareth Matthews. *Basic Physiology for Anaesthetists*. 2nd ed. Cambridge University Press, 2019. DOI: 10.1017/9781108565011 (cit. on p. 14).

[17] Roberto Merletti, Alberto Botter, and U Barone. «Detection and conditioning of surface EMG signals». In: *Surface electromyography: Physiology, engineering, and applications* (2016), pp. 1–37 (cit. on p. 13).

[18] *Delsyseurope*. https://delsyseurope.com/. Accessed: 2023-02-25 (cit. on p. 15).

[19] Christopher Spiewak, M Islam, A Zaman, Mohammad Habibur Rahman, et al. «A comprehensive study on EMG feature extraction and classifiers». In: *Open Access Journal of Biomedical Engineering and Biosciences* 1.1 (2018), pp. 1–10 (cit. on p. 15).

[20] Cheng-Ning Huang Chun-Han Chen and Hung-Yuan Chung. «The review of applications and measurements in facial electromyography». In: *Journal of Medical and Biological Engineering* 25.1 (2004), pp. 15–20 (cit. on pp. 17, 20).

[21] Fabio Rossi, Andrea Mongardi, Paolo Motto Ros, Massimo Ruo Roch, Maurizio Martina, and Danilo Demarchi. «Tutorial: A versatile bio-inspired system for processing and transmission of muscular information». In: *IEEE Sensors Journal* 21.20 (2021), pp. 22285–22303 (cit. on pp. 18, 19, 29, 31, 32).

[22] Fabio Rossi, Paolo Motto Ros, Ricardo Maximiliano Rosales, and Danilo Demarchi. «Embedded Bio-Mimetic System for Functional Electrical Stimulation Controlled by Event-Driven sEMG». In: *Sensors* 20.5 (2020). ISSN: 1424-8220. DOI: `10.3390/s20051535`. URL: `https://www.mdpi.com/1424-8220/20/5/1535` (cit. on p. 18).

[23] Amirhossein Shahshahani, Masoud Shahshahani, Paolo Motto Ros, Alberto Bonanno, Marco Crepaldi, Maurizio Martina, Danilo Demarchi, and Guido Masera. «An all-digital spike-based ultra-low-power IR-UWB dynamic average threshold crossing scheme for muscle force wireless transmission». In: *2015 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE. 2015, pp. 1479–1484 (cit. on p. 18).

[24] Paolo Motto Ros, Marco Paleari, Nicoló Celadon, Alessandro Sanginario, Alberto Bonanno, Marco Crepaldi, Paolo Ariano, and Danilo Demarchi. «A wireless address-event representation system for ATC-based multi-channel force wireless transmission». In: *5th IEEE International Workshop on Advances in Sensors and Interfaces IWASI*. IEEE. 2013, pp. 51–56 (cit. on p. 18).

[25] Andrea Mongardi, Paolo Motto Ros, Fabio Rossi, Massimo Ruo Roch, Maurizio Martina, and Danilo Demarchi. «A Low-Power Embedded System for Real-Time sEMG based Event-Driven Gesture Recognition». In: *2019 26th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. 2019, pp. 65–68. DOI: `10.1109/ICECS46596.2019.8964944` (cit. on p. 19).

[26] Xugang Xi, Yan Zhang, Xian Hua, Seyed M Miran, Yun-Bo Zhao, and Zhizeng Luo. «Facial expression distribution prediction based on surface electromyography». In: *Expert Systems with Applications* 161 (2020), p. 113683. ISSN: 0957-4174. DOI: `https://doi.org/10.1016/j.eswa.2020.113683` (cit. on p. 19).

[27] Mahyar Hamedi, Sh-Hussain Salleh, Mehdi Astaraki, and Alias Mohd Noor. «EMG-based facial gesture recognition through versatile elliptic basis function neural network». In: *Biomedical engineering online* 12.1 (2013), pp. 1–22 (cit. on p. 20).

[28] M Hamedi, Sh-Hussain Salleh, TS Tan, K Ismail, J Ali, C Dee-Uam, C Pavaganun, and PP Yupapin. «Human facial neural activities and gesture recognition for machine-interfacing applications». In: *International journal of nanomedicine* (2011), pp. 3461–3472 (cit. on p. 20).

[29] Mahyar Hamedi, Sh-Hussain Salleh, Chee-Ming Ting, Mehdi Astaraki, and Alias Mohd Noor. «Robust facial expression recognition for MuCI: a comprehensive neuromuscular signal analysis». In: *IEEE Transactions on Affective Computing* 9.1 (2016), pp. 102–115 (cit. on p. 20).

[30] Michael Wand and Tanja Schultz. «Pattern learning with deep neural networks in EMG-based speech recognition». In: *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE. 2014, pp. 4200–4203 (cit. on p. 20).

[31] Eduardo Lopez-Larraz, Oscar M Mozos, Javier M Antelis, and Javier Minguez. «Syllable-based speech recognition using EMG». In: *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*. IEEE. 2010, pp. 4699–4702 (cit. on p. 20).

[32] Sebastian Roldan-Vasco, Sebastian Restrepo-Agudelo, Yorhagy Valencia-Martinez, and Andres Orozco-Duque. «Automatic detection of oral and pharyngeal phases in swallowing using classification algorithms and multichannel EMG». In: *Journal of Electromyography and Kinesiology* 43 (2018), pp. 193–200 (cit. on p. 20).

[33] Zhiyuan Lu and Ping Zhou. «Hands-free human-computer interface based on facial myoelectric pattern recognition». In: *Frontiers in neurology* 10 (2019), p. 444 (cit. on p. 20).

[34] Hiroki Tamura, Takayuki Murata, Yuki Yamashita, Koichi Tanno, and Yasufumi Fuse. «Development of the electric wheelchair hands-free semi-automatic control system using the surface-electromyogram of facial muscles». In: *Artificial Life and Robotics* 17 (2012), pp. 300–305 (cit. on p. 20).

[35] Yinfeng Fang, Dalin Zhou, Kairu Li, and Honghai Liu. «Interface prostheses with classifier-feedback-based user training». In: *IEEE transactions on biomedical engineering* 64.11 (2016), pp. 2575–2583 (cit. on p. 20).

[36] Karsten Wolf, Thomas Raedler, Kai Henke, Falk Kiefer, Reinhard Mass, Markus Quante, and Klaus Wiedemann. «The face of pain-A pilot study to validate the measurement of facial pain expression with an improved EMG method». In: *Pain Research and Management* 10.1 (2005), pp. 15–19 (cit. on p. 20).

[37] Riitta Mieronkoski, Elise Syrjälä, Mingzhe Jiang, Amir Rahmani, Tapio Pahikkala, Pasi Liljeberg, and Sanna Salanterä. «Developing a pain intensity prediction model using facial expression: A feasibility study with electromyography». In: *PloS one* 15.7 (2020), e0235545 (cit. on p. 20).

[38] Thomas Richard Dawes, Ben Eden-Green, Claire Rosten, Julian Giles, Ricardo Governo, Francesca Marcelline, and Charles Nduka. «Objectively measuring pain using facial expression: is the technology finally ready?» In: *Pain management* 8.2 (2018), pp. 105–113 (cit. on p. 20).

[39] Nikolaus Peter Schumann, Kevin Bongers, Orlando Guntinas-Lichius, and Hans Christoph Scholle. «Facial muscle activation patterns in healthy male humans: A multi-channel surface EMG study». In: *Journal of neuroscience methods* 187.1 (2010), pp. 120–128 (cit. on p. 21).

[40] Anna Gruebler and Kenji Suzuki. «Design of a Wearable Device for Reading Positive Expressions from Facial EMG Signals». In: *IEEE Transactions on Affective Computing* 5.3 (2014), pp. 227–237. DOI: `10.1109/TAFFC.2014.2313557` (cit. on p. 21).

[41] Yumiao Chen, Zhongliang Yang, and Jiangping Wang. «Eyebrow emotional expression recognition using surface EMG signals». In: *Neurocomputing* 168 (2015), pp. 871–879. ISSN: 0925-2312. DOI: `https://doi.org/10.1016/j.neucom.2015.05.037` (cit. on p. 21).

[42] Vikram Kehri, Rahul Ingle, Sangram Patil, and RN Awale. «Analysis of facial EMG signal for emotion recognition using wavelet packet transform and SVM». In: (2019), pp. 247–257 (cit. on p. 21).

[43] Angela Carnevale. «Machine learning algorithms for facial gesture recognition: a first analysis based on event-driven sEMG acquisition». PhD thesis. Politecnico di Torino, 2021 (cit. on pp. 22, 37, 42, 66).

[44] Andrea Mongardi, Paolo Motto Ros, Fabio Rossi, Massimo Ruo Roch, Maurizio Martina, and Danilo Demarchi. «A low-power embedded system for real-time sEMG based event-driven gesture recognition». In: *2019 26th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE. 2019, pp. 65–68 (cit. on p. 23).

[45] Andrea Mongardi, Fabio Rossi, Andrea Prestia, Paolo Motto Ros, Massimo Ruo Roch, Maurizio Martina, and Danilo Demarchi. «Hand Gestures Recognition for Human-Machine Interfaces: A Low-Power Bio-Inspired Armband». In: *IEEE Transactions on Biomedical Circuits and Systems* 16.6 (2022), pp. 1348–1365. DOI: `10.1109/TBCAS.2022.3211424` (cit. on pp. 23, 29, 31, 32).

[46] *ambiq Apollo3 Blue*. `https://ambiq.com/apollo3-blue/`. Accessed: 2023-02-20 (cit. on pp. 23, 27).

[47] *Nordic Semiconductor. nRF52840 USB Dongle*. `https://www.nordicsemi.com/Products/Development-hardware/nrf52840-dongle`. Accessed: 2023-02-20 (cit. on pp. 23, 31).

[48] Andrea Prestia, Fabio Rossi, Andrea Mongardi, Paolo Motto Ros, Massimo Ruo Roch, Maurizio Martina, and Danilo Demarchi. «Motion analysis for experimental evaluation of an event-driven FES system». In: *IEEE Transactions on Biomedical Circuits and Systems* 16.1 (2021), pp. 3–14 (cit. on pp. 24, 25, 32).

[49] *Face Muscles scheme*. `https://www.istockphoto.com/it`. Accessed: 2023-03-01 (cit. on p. 24).

[50] *Kendall H124SG Electrodes*. `https://www.cardinalhealth.co.uk/en_gb/medical-products/patient-care/electrocardiography/adult-monitoring-electrodes/general-monitoring-ecg-electrodes/long-term-electrodes/kendall-electrodes-h124sg.html`. Accessed: 2023-03-02 (cit. on pp. 24, 42).

[51] David Alejandro Fernandez Guzman, Stefano Sapienza, Bianca Sereni, and Paolo Motto Ros. «Very low power event-based surface EMG acquisition system with off-the-shelf components». In: *2017 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE. 2017, pp. 1–4 (cit. on p. 24).

[52] *Cortex-M4*. `https://developer.arm.com/Processors/Cortex-M4`. Accessed: 2023-03-02 (cit. on p. 27).

[53] *CMSIS DSP Software Library*. `https://www.keil.com/pack/doc/CMSIS/DSP/html/index.html`. Accessed: 2023-01-27 (cit. on p. 29).

[54] *The freeRTOS Kernel*. `https://www.freertos.org/RTOS.html`. Accessed: 2023-02-16 (cit. on p. 29).

[55] *Kivy:The Open Source Python App development Framework*. `https://kivy.org/`. Accessed: 2023-03-02 (cit. on p. 32).

[56] N Nojd, Markus Hannula, and J Hyttinen. «Electrode position optimization for facial EMG measurements for human-computer interface». In: *2008 Second International Conference on Pervasive Computing Technologies for Healthcare*. IEEE. 2008, pp. 319–322 (cit. on p. 37).

[57] Jennifer M Vojtech, Gabriel J Cler, and Cara E Stepp. «Prediction of optimal facial electromyographic sensor configurations for human–machine interface control». In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 26.8 (2018), pp. 1566–1576 (cit. on p. 42).

[58] Louis G. Tassinary, John T. Cacioppo, and Thomas R. Geen. «A Psychometric Study of Surface Electrode Placements for Facial Electromyographic Recording: I. The Brow and Cheek Muscle Regions». In: *Psychophysiology* 26.1 (1989), pp. 1–16. DOI: `https://doi.org/10.1111/j.1469-8986.1989.tb03125.x` (cit. on p. 42).

[59] Ana Sabaneeff, Luciana Duarte Caldas, Marco Antonio Cavalcanti Garcia, and Matilde da Cunha Gonçalves Nojima. «Proposal of surface electromyography signal acquisition protocols for masseter and temporalis muscles». In: *Research on Biomedical Engineering* 33 (2017), pp. 324–330 (cit. on p. 42).

[60] C.C. Aggarwal. *Neural Networks and Deep Learning: A Textbook*. Springer International Publishing, 2018. ISBN: 9783319944647. URL: `https://books.google.it/books?id=AsTswQEACAAJ` (cit. on p. 56).

[61] *Machine Learning by Stanford University*. `https://www.coursera.org/learn/machine-learning-course`. Accessed: 2022-06-20 (cit. on pp. 56, 58).

[62] Baofeng Gao, Chao Wei, Hongdao Ma, Shu Yang, Xu Ma, Songyuan Zhang, et al. «Real-time evaluation of the signal processing of sEMG used in limb exoskeleton rehabilitation system». In: *Applied bionics and biomechanics* 2018 (2018) (cit. on p. 67).

[63] *INA240 by Texas Instrument Inc.* `https://www.ti.com/product/INA240?keyMatch=INA240&tisearch=search-everything&usecase=GPN`. Accessed: 2023-03-01 (cit. on p. 67).

[64] *Rigol Technologies, Digital oscilloscope*. `https://rigolshop.eu/product-oscilloscope-mso5000-mso5104.html`. Accessed: 2023-03-01 (cit. on p. 67).