Universidad del Rosario

MACC
Matemáticas Aplicadas y Ciencias de la Computación

A Brain-Computer Interface for labour market inclusion of people suffering severe upper-limb impairments

by

Juan Lucas García Osorio

Bachelor Thesis

Bachelor's Degree Programme in Applied Mathematics
and Computer Sciences
Universidad del Rosario, Bogota, Colombia

June, 2023

Universidad del Rosario

MACC Matemáticas Aplicadas y Ciencias de la Computación

A Brain-Computer Interface for labour market inclusion of people suffering severe upper-limb impairments

by

Juan Lucas García Osorio

Bachelor Thesis

Bachelor's Degree Programme in Applied Mathematics
and Computer Sciences
Universidad del Rosario, Bogota, Colombia

Advisor

PhD. Mario Fernando Jiménez Hernández

Co-Advisor

PhD. Denis Delisle Rodriguez, Edmond and Lily Safra International Institute of Neuroscience, Santos Dumont institute

June, 2023

Abstract

Robotic assistive devices, such as exoskeletons are used in labour environments to promote social inclusion of diverse types of impairments as for example upper-limb. Robotic exoskeletons can be controlled by surface electromyography signals. However, people with severe neural impairments and absence of residual muscular activity are unable of using these sEMG-based systems due to the absence of residual muscular activity. Alternatively, robotic hand prostheses and exoskeletons commanded by Brain-Computer Interfaces (BCIs) have been successfully applied in these people. This study aims to develop a low-cost steady-state visual evoked potential (SSVEP)-based BCI for social inclusion, using unsupervised calibration. A low-cost flicker visual stimulator with geometric shapes is proposed to elicit brain commands. Both Canonical Correlation Analysis (CCA) and Power Spectral Density (PSD) are used to classify SSVEP stimuli. As a first step, the proposed BCI was tested in a serious game, which was developed to simulate the workspace, and provide feedback to the subject. CCA presented the best classification results with an accuracy of 71.6 $\pm$ 9.7% and an Information Transfer Rate (ITR) of 37.6 $\pm$ 15.4 bits/min and averaged latency of 0.77 $\pm$ 0.39 s to provide an output associated to the stimulus observed by the subject.

Keywords: Brain-Computer Interface, Steady State Visual Evoked Potential, social inclusion, serious game, upper-limb disability.

Dedicatoria

To my father, Gabriel Garcia, without you this would not have been possible.

Acknowledgements

The moment I was most looking forward to writing about this thesis was the acknowledgements, I feel so much pride and gratitude for having made it this far that I wanted to express it to all the people who made it possible, without them, my university career and the development of this work would not have been possible. First of all, I would like to thank my supervisors, Mario and Denis, for all their support, mentoring, teaching and motivation to pursue my scientific ideas and interests, and for allowing me to expand my knowledge in other environments and welcoming me as one of their family.

Also, I would like to thank my friends who accompanied me along this beautiful path that was my undergraduate, I learned wonderful things by their side and shared moments that I will always carry in my heart. Especially to Dana, Daniel and Elizabeth, thank you for always being by my side, in the worst and in the best moments, without your unconditional support my passage through the university would not have been so pleasant and the days of hard work would have been impossible.

Last but not least I want to thank my family, the engine of my life, the people who motivate me every day to achieve my goals. Whatever I needed, no matter how impossible it was, they were there to solve it. To Gabriel, my father, thank you for believing in me and always betting everything you had in my favor, definitely, this would not have been possible without you. To Juan Ga, my brother, thank you for sowing in me that scientific curiosity since I was so young, the Arduino and the printer that you gave me were gifts that changed my life, without you I would not have found my path. To Maria (tata), my sister, thank you for your advice, values and life teachings, without you I would be a completely different person, you motivate me to be a better person every day, whenever I think of you I always smile and feel that I can conquer the world. To Luisa, my mother, thank you for always being there for me and supporting me throughout my life. This is the first of many achievements, thank you all for everything.

Contents

Abstract i

Dedication ii

Acknowledgement iii

List of Figures vi

1 Introduction 1

1.1 Motivation . . . 1
1.2 Objectives . . . 4
1.3 Document organization . . . 4

2 Context and State of the art 6

2.1 Brain overview . . . 6
2.1.1 Neural communication . . . 6
2.1.2 Cerebral regions . . . 7
2.2 Brain Computer Interface (BCI) . . . 8
2.3 From neural to digital measurements . . . 9
2.3.1 Invasive techniques . . . 9
2.3.2 Non-invasive techniques . . . 10
Electroencephalography (EEG) . . . 10
Functional Magnetic Resonance Imaging (fMRI) . . . 12
Magnetoencephalography (MEG) . . . 12
2.4 BCI control approaches . . . 13
2.4.1 Stimulus Evoked Potentials . . . 14
2.4.2 P300 potential . . . 14
2.4.3 Steady State Visually Evoked Potential (SSVEP) . . . 14
2.4.4 Auditory Evoked Potential . . . 15

2.5 Spatial filter . . . 16
2.5.1 Laplacian filter . . . 16
2.5.2 Common Average Referencing (CAR) . . . 17
2.6 Feature Extraction . . . 18
2.6.1 Frequency Domain Analysis . . . 18
Power Spectral Density (PSD) . . . 18
2.6.2 Statistical features . . . 18
Canonical Correlation Analysis (CCA) . . . 19
2.7 BCI performance . . . 19
2.7.1 Accuracy (ACC) . . . 21
2.7.2 Kappa score . . . 21
2.7.3 True Positive Rate (TPR) and False Positive Rate (FPR) . . . 22
2.7.4 Information Transfer Rate (ITR) . . . 22
2.8 Related works . . . 23

3 Methods and Materials **26**
3.1 Signal acquisition . . . 27
3.2 Stimuli presentation . . . 27
3.2.1 Stimuli presenter . . . 29
3.2.2 Device features of the visual stimulator . . . 31
3.3 Signal preprocessing . . . 32
3.4 Feature extraction . . . 36
3.4.1 PSD . . . 36
3.4.2 CCA . . . 37
3.5 Serious game . . . 38
3.6 Experimental assessment . . . 39
3.6.1 Participants . . . 39
3.6.2 Experimental setup . . . 40
Offline phase . . . 41
Online phase . . . 42

4 Results **44**
4.1 Preliminary data analysis . . . 44
4.2 Performance during offline phase . . . 47
4.3 Performance during online phase . . . 52
4.4 Discussion . . . 54

5 Conclusion **56**
5.1 Future works 57

Bibliography **58**

A Appendix **65**
A.1 Python Code 65
A.2 OpenVibe scenario 68

List of Figures

2.1 Cerebral lobes(from [6]) 8

2.2 Main invasive techniques. In (a), one electrode measures the potential difference between the inside and outside of the neuron. In (b) the top part show the process of ECoG array implantation and in the bottom it can be seen an x-ray of the implanted array (from [6]) . . . 10

2.3 In (a) EEG cap montage with 64 wet electrodes (from [30]), In (b) the 10 - 20 international system (from [31]) 12

2.4 Main brain waves found in EEG readings (from [35]) 13

2.5 In (a) P300 waveform comparing epochs with 'odd' and 'normal' stimuli. In (b) P300 speller design (image b from [39]) 15

2.6 The plots show amplitude spectra computed using FFT averaged over 30 trials. (a) and (b) show when the subject is looking at stimulation flickering at 12.2 Hz and 8.2 Hz respectively. There is a predominant amplitude peak at those frequencies 16

2.7 Illustration of spatial filter. (a) an example of Laplacian filter applied to Cz channel, (b) an example of CAR filter applied to Cz channel taking its seven neighbours 17

2.8 An illustration for usage of CCA in EEG signal analysis. $x_1, ..., x_8$ are signals from 8 EEG channels and $y_1, ..., y_8$ are Fourier series of a given frequency period signal. The CCA finds the linear combination coefficients $w_{x1}, ..., w_{x8}$ and $w_{y1}, ..., w_{y8}$, which gives the largest correlation between X and Y (from [47]) 20

2.9 Illustration of confusion matrix. The triangle denoted in green refers to the positive class. The square denoted in red refers to the negative class 20

2.10 Literature review flow chart of articles on BCI system using SSVEP as control paradigm to control external devices for upper limbs assistance 24

2.11 Methodology used in [51]. In bottom left, the EEG montage with the augmented reality glasses. In the middle the user perspective wearing the glasses. In bottom right, the robotic arm used (Figure from [51]) 25

3.1 Methodology used for the development of a BCI capable of classifying SSVEP 26
3.2 Position of the electrodes according to the 10-20 system. In red reference electrode. In blue ground. In green the lecture electrodes . . . 28
3.3 Stimulator display. (a) Selected layout for the 16 8x8 led matrix modules. In (b) the two stimuli used to generate SSVEP 30
3.4 Illustration of frequency separation to be able to make dual stimuli with Arduino 31
3.5 Final result of the visual stimulator. In (a) the orthographic views, where it can be seen all available connectors for programming, digital outputs and power supply. In (b) the display of the visual stimuli. . . 32
3.6 An example of the configuration file that allows the use of different parameters tuning 33
3.7 Flowchart with the procedure performed on the signal for noise reduction 34
3.8 An example of the blink detection algorithm, threshold of 30 uV was applied to the signal. In (a) is the signal of Fp1 channel, then, in orange and green there are blinks detected by the algorithm. In (b) the view of the signal from the occipital channel O1, the time where the blink occurs is highlighted in orange and green respectively. In (c) the result of linear interpolation between the start and end point from the blink epoch. 36
3.9 PSD of signal extracted from channel O1 (2 seconds chunk), with an $\epsilon = 1$. In (a) is when the subject is looking at the triangle, in (b) is when the subject is looking at the square. The green dotted line indicates the region where the energy peak must be for the signal to be classified as triangle or square 37
3.10 Serious game view 39
3.11 Experimental setup 41
3.12 Initial 60 second sequence for experiment setup 42
3.13 (a, b) Indication of the figure on which the participant should concentrate his attention. (c) thank you screen displayed at the end of the experiment 42

3.14 Serious game control mode. At the top you can see the movement of the hands when the participant is looking at the triangle. At the bottom is the movement of the hands when the participant is looking at the square. Finally, note that at the bottom a point is added to the counter due to the completion of the sequence 43

4.1 Raw signal of the offline phase of O1 channel obtained from participant 1 45
4.2 Filtered signal of the offline phase of O1 channel obtained from participant 1. (a) shows the temporal filtered signal and the baseline correction. In (b) the spatial corrected signal by CAR filter 46
4.3 Signal average of the epochs grouped by triangle and square of participant 1. In (a) the average of triangle epochs. In (b) the average of square epochs 46
4.4 PSD of the averaged signals. In (a) PSD of triangle epochs. In (b) PSD of square epochs 47
4.5 Correlation coefficients using CCA algorithm 47
4.6 Average system accuracy using different time windows. In (a) the preprocessing without the eye blink subtraction algorithm. In (b) preprocessing with eye blink subtraction stage 48
4.7 Accuracy achieved by the system with both preprocessing stages . . . 51
4.8 System response confusion matrix 52
4.9 System response latency 53
4.10 Time taken by Participants to complete the task through serious game play 53

A.1 Online phase scenario 69
A.2 Offline phase scenario 70

Chapter 1

Introduction

This chapter is intended to be a brief preface to this research, presenting to the reader the motivation behind the development of this project, followed by the main ways in which externals aids such as orthotics and prosthetics are activated. Then it is showed the lack of systems implementation in people with nervous system disabilities, simultaneously a short explanation is presented of what a Brain Computer Interface is, and the common ways to interact between the brain and external devices. Finally it is explained the purpose to be achieved in this project, and the products delivered throughout the implementation of the thesis.

1.1 Motivation

When physical work is required, people with motor disabilities are incapable of participating because physical work needs to exert force, sudden movements, or even just moving around. Therefore there are no vacancies for these people in these kinds of jobs. According to the European Commission, labour market inclusion refers to when vulnerable and disadvantaged people can participate in quality paid work [1]. In the scenario where physical work is involved the above definition is not always accomplished. One solution for this is the use of assistive robotic devices that try to replicate human motion in order to do physical work [2]. The exoskeletons are the robotic devices used to aid physical work, an exoskeleton is a mechanic device worn on the outside of the human body that augments: motion, strength, and endurance [3].

In the presence of motor impairment a special method needs to be used to control the motion of the robotic device. The motion control of this robotic device is often accomplished via sensors that acquire superficial signals [4], [5], such as sur-

face electromyography (sEMG). These sensors acquire electrical activity of muscles. When a muscle contracts it generates an electric signal that can be measured in the surface of the skin. When the disability is due to an amputation, either by trauma or birth, the brain continues to give signals to an amputee because the nerves that used to connect the missing limb are still intact and are continuously sending electric signals [6].

Although the method of sEMG to control the motion of an exoskeleton covers a wide spectrum of motor disabilities, there are a large number of disabilities that cannot be controlled by sEMG because of the poor signal activity in the affected limb, caused by damage in the nerve pathway or even in the brain zone in charge of the movement of the specific part of the body [7]. Diseases or traumas such as stroke, spinal cord injury, multiple sclerosis, and traumatic brain injuries could affect the communication between the brain and the limb. That is why the population before mentioned cannot participate in a workplace doing physical work, even with the help of robotic devices [6].

On the other hand, 20% fewer women are performing hard physical work than men [8] and according to the definition given by the European Commission about labour market inclusion, there is an inequality in the workplace when we talk about gender. Thus, these aids provided by the robotic exoskeleton can be implemented within a wider spectrum and help to include a greater number of women and contribute to closing the social gap by balancing the number of women working in areas related to physical labour. In this fashion, alternatives have been proposed so that people belonging to the aforementioned groups can control exoskeletons oriented to physical work.

One alternative way to control external devices such as computers, robotic exoskeletons, or prosthetic limbs is through the use of brain signals. This is known as a Brain Computer Interface (BCI). The idea is to translate neural activity into commands that can operate some kind of systems [9], [10]. The most common approach for brain signal acquisition is electroencephalography (EEG), as it is a non-invasive method that only needs a special cap and a few minutes to prepare the subject for the readings [11].

There are many ways to stimulate the human brain in order to generate the desired EEG pattern. One of them is known as P300, it is produced when the subject is stimulated by two different shapes, each with different probability of occurrence. When the lower probability figure appears on the screen, the subject will produce in the EEG readings a positive voltage deflection 300 ms after the appearance of

the stimulus [12]. One example, it is the uses of P300 in spellers, where letters are randomly presented on the screen and when the subject looks at the one that he/she wants the P300 potential is activated [13].

Other methods utilized for BCI are shown bellow: Motor Imagery (MI) that consists in imagining oneself performing a physical movement without actually executing the movement, as a result of the activation of the same neural networks involved in the execution of the actual movement [14]. MI is widely used in rehabilitation processes since it has been presented as a quick and short way to recover lost neuronal connections [15]. However, the higher the sophistication of the paradigm, the more training it needs to have good classification results. That is why methods such as P300 and MI need training sessions in order to get good accuracy levels [16].

Lately, BCI technologies have implemented another paradigm known as Steady State Visual Evoked Potentials (SSVEP). This method does not require previous training, and has high levels of accuracy and Information Transfer Rate (ITR) [17]. SSVEP generate a neural response when a subject is looking at a visual stimulus that flickers at a specific frequency. When the person focuses their attention on the flicker stimulus, it will cause oscillatory neural responses in the occipital regions at the same frequency as the stimulus [18]. There are many techniques of feature extraction in order to detect if the subject is looking at the desired frequencies, such as Power Spectral Density (PSD), Independent Component Analysis (ICA), Linear Discriminant Analysis (LDA), and Canonical Correlation Analysis (CCA) [19], [20].

In this way, this thesis aims to explore the use of BCI as an alternative in elbow exoskeleton control used for assistance in repetitive tasks. So that the idea is to develop a BCI that allows by means of evoked potentials to control an elbow robotic exoskeleton that will be emulated by means of a serious game, that contains two tasks in a packing activity. (1) taking a box and move the box to the next section, (2) release the box in the destination place. SSVEP will be used to identify the activation signal to control the elbow exoskeleton that will be simulated by the movement of the package in the serious game. SSVEP was chosen because of its versatility, and its no need of training, these specifications fit perfectly for a workplace application.

To achieve the work proposed in this document subsequent tasks need to be done. Initially, design a visual stimulator to generate the desired evoked potentials in the occipital region. Then, design a protocol for data acquisition that allows us to have reproducibility in the results. Followed by the evaluation of performance among different feature extraction algorithms and choose the more appropriate. Finally,

generate communication between all of the systems: stimulator, data acquisition system, python, and unity.

1.2 Objectives

General

To develop a steady-state visual evoked potential-based brain-computer interface for assistance of people with severe upper-limb impairments by controlling a virtually emulated robotic elbow exoskeleton in a serious gaming environment.

Specific

1. To measure steady-state visual evoked potential brain signals required for the activation of the brain-computer interface by implementing a visual stimulator of two-frequency oscillating geometric shapes.

2. To determine the efficiency of power spectral density and canonical correlation analysis algorithms for feature extraction and classification of steady-state visual evoked potential brain signals.

3. To evaluate the performance of the brain computer interface for the control of an emulated robotic exoskeleton by developing a serious game with two specific work activities.

1.3 Document organization

This document contains six chapters divided into Introduction, State of the art, Methodology, Results, Discussion, and Conclusions.

The state of the art in the second chapter begins by presenting the theoretical framework, showing definitions and concepts used during the development of the project. Concepts such as: Electroencephalography and Brain Computer Interface (processing methods and techniques). Finally, a literature review is made, in current and relevant works in the field of neuroengineering and development of BCIs for upper limb device control.

The third chapter discusses the methodology used in the project. It shows the phases implemented to meet the proposed objectives. The construction of the low-

cost stimulator for the production of visual evoked potentials is explained. Finally, the experimental protocol carried out for data acquisition is discussed.

The results obtained are shown in chapter 4, where CCA and PSD are compared as methods of feature extraction and classification. In addition to this, different time windows and overlays are evaluated to find a balance between speed and accuracy. Subsequently, an analysis is made about the response times between the system and the subject's intention. In this way, Chapter 6 discusses the results obtained with similar investigations presented in Chapter 2, thus validating the results obtained in this work.

Finally, chapter 6 describes the conclusions based on the previously established objectives, as well as considerations and future work on the research carried out.

Chapter 2

Context and State of the art

This chapter presents the concepts, definitions and techniques used throughout the development of this work. Physiological definitions are revised, about the brain and how it produces electrical signals that can be captured by specialized devices. Subsequently, it is discussed how to acquire brain information and how to extract relevant characteristics for the control of external devices based on the subject's intention. Finally, a literature review is conducted about information of BCI for the control of limbs assistive devices.

2.1 Brain overview

This section does not pretend to have very detailed and deep information about the brain because the brain is one of the most complex and enigmatic organs of the body. Rather, it is intended to show the required concepts to understand how it is possible to acquire neural signals and the main challenges involved. The signal acquisition process is the fundamental pillar for the construction of devices controlled by the brain.

2.1.1 Neural communication

The brain communicates through the electrical and chemical activity of nerve cells, known as neurons. Neurons are specialized in the communication and transmission of information in the nervous system [6]. Each neuron has a complex structure consisting of a cell body, dendrites, and an axon. Dendrites receive signals from other neurons and transmit them to the cell body, where the signals are integrated and processed [21]. The axon transmits the signal via a chemical called a neurotrans-

mitter to another neuron, or to a muscle or gland [6].

When the neuron receives sufficiently strong inputs from other neurons, a cascade of events is triggered, changing the resting potential found in the cell body (approximately -65 mV) to a positive potential (approximately 20 mV), which is known as action potential [6]. Finally, the action potentials produced by a neuron communicate to the dendrites of adjacent neurons via the axon [22].

When a neuronal process is performed (no matter how small) millions of neurons are activated, so that the action potentials produced individually by each neuron add up resulting in a significant potential differential. Which can be captured on the scalp by amplifiers and electrodes. The EEG uses this principle to acquire brain signals, which will be explained in later sections.

2.1.2 Cerebral regions

The brain is divided into four cerebral lobes, named according to the bone on which they are located in the skull. The lobes found in the human brain are the frontal, parietal, temporal, and occipital lobes (Fig. 2.1), each of which has specific functions [21].

The frontal lobe is the part of the brain located at the front, just behind the forehead. It is responsible for a variety of higher cognitive functions [6]. The parietal lobe is located at the top and back of the brain and is responsible for sensory perception and the integration of sensory information from different parts of the body [6]. The temporal lobe is located in the lower part of the brain and is responsible for hearing, facial recognition and memory [6]. The occipital lobe is located at the back of the brain and is responsible for visual perception. It is the region of the brain that is activated when we see objects and colors. The occipital lobe is responsible for conscious visual perception and the ability to recognize objects, shapes and colors [6].

It is important to keep in mind that the brain lobes do not function in isolation, but work together in complex networks to control our cognitive, sensory and motor functions. For example, the parietal lobe and occipital lobe work together to perceive space and depth, while the frontal lobe and parietal lobe work together to plan and execute complex movements [6].

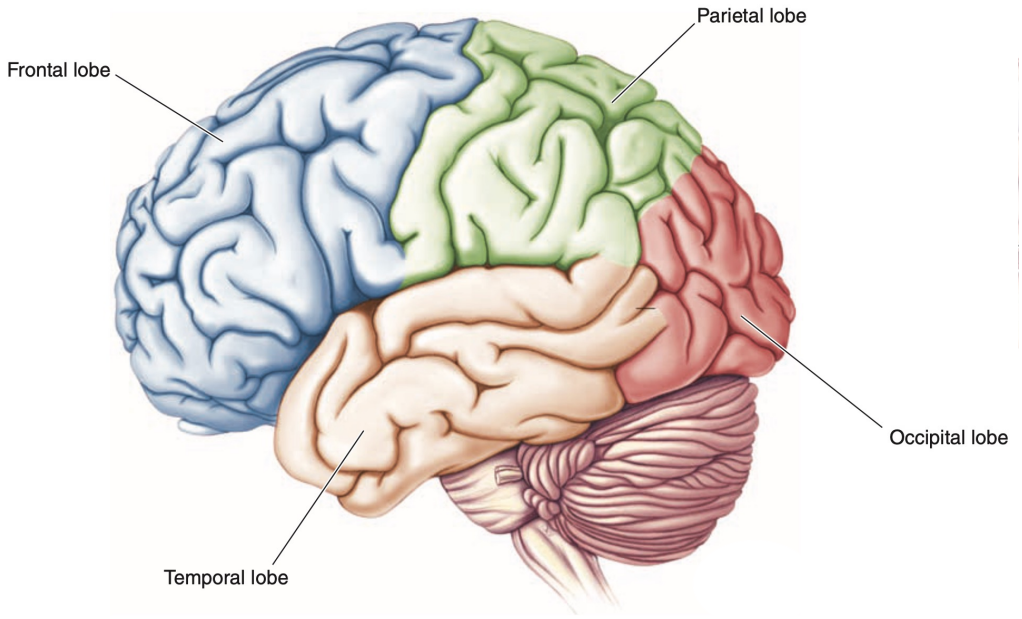


Figure 2.1: Cerebral lobes(from [6])

2.2 Brain Computer Interface (BCI)

A BCI is a communication system between the brain's electrical activity and external devices, most commonly a computer or robotic limb [23]. Also, BCIs are used in rehabilitation of patients who have suffered some kind of injury in their central nervous system [24]. BCI is a highly researched technology that aims to improve the quality of life of individuals with disabilities or motor limitations [25].

There are several types of BCI, which are classified according to the way brain signals are recorded and the method used to interpret them. BCIs are classified into two types: invasive and noninvasive. This classification is based on the way in which the neural signals are obtained, the definition of these methods can be found in the following section.

Invasive BCI techniques involve implanting electrodes or sensors inside the skull after surgery to sense the electrical activities of the brain cells, this technique provides high-quality output. However, surgery is a complex process that can only be carried out by a professional in addition to the recovery that is a slow and delicate process [25].

Non-invasive BCIs are of great interest to the developers because the measurement of brain signals is performed superficially and no surgical procedure is required. Some acquisition methods for noninvasive BCI signals include EEG, magnetoen-

cephalography (MEG), functional magnetic resonance imaging (fMRI), and BCI based on electrocorticography (ECoG) [25].

EEG is the most popular method, and consists of recording brain signals using electrodes placed on the surface of the scalp. These signals, known as EEG signals, are processed by a signal processing software that identifies specific patterns in brain activity related to the subject's intention, such as moving a cursor to the right or left.

BCI can be used for a variety of applications such as restoring mobility in people with motor disabilities, to improve performance in cognitive tasks and to monitor brain activity in real time. For example, a BCI can enable a person with paralysis to send emails or control a wheelchair by thought alone [26].

2.3 From neural to digital measurements

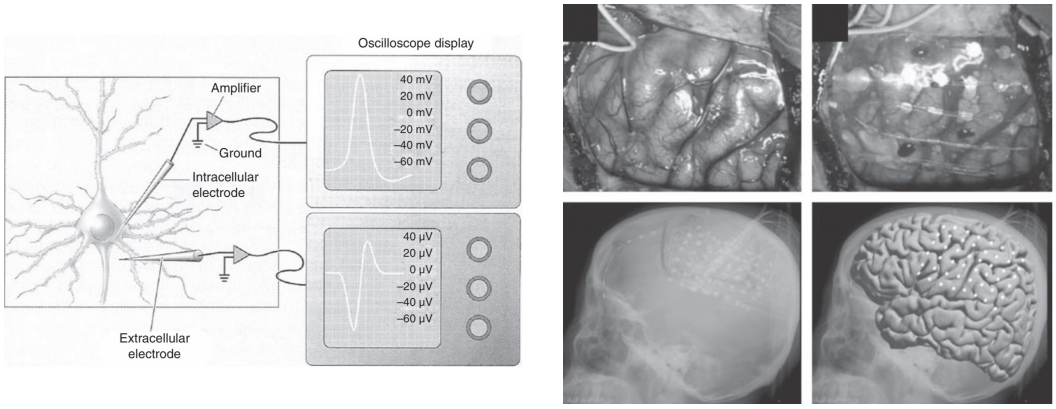
Since sets of neurons command cognitive, motor and sensory processes, a chemical, and electrical exchange are necessary. Existing methods for measuring brain activity which can be divided into two main categories: invasive and non-invasive.

2.3.1 Invasive techniques

These techniques commonly allow the measurement of individual neurons. Being invasive means that it involves a surgery, where a part of the skull is removed and a set of electrodes is implanted there. Although it is a very precise measurement method and allows accurate information about the neurons studied, it is risky because the surgery can generate an infection [27]. Finally, over time, the measurements lose resolution and become contaminated with noise as glial cells surround the implanted device, forming scarring and thus isolating the electrical measurement [25]. Therefore, this type of implantation is commonly performed in monkeys and rats [6]. An illustration of intracellular and extracellular measurement using an electrode is shown in Figure 2.2.a.

On the other hand, there is a less invasive method that consists in the implantation of electrodes on the surface of the brain. However, it is still necessary a surgical intervention that goes through the meninges and places the electrode array in the cerebral cortex. This is known as Electrocorticography (ECoG). Typically, an m x n sized electrode array is implanted, where m and n vary between 1 and 8 electrodes. ECoG has attracted the attention of BCI developers because it is an intermediate

between invasive and noninvasive methods, since it is possible to have a higher spatial resolution. In addition, being closer to neuronal activity, it is less vulnerable to artifacts from muscle activity and environmental noise. However, there are still risks involved since a special environment is required to perform the surgery [25]. Figure 2.2.b shows the ECoG implantation process.



(a) One electrode measurement (b) ECoG array

Figure 2.2: Main invasive techniques. In (a), one electrode measures the potential difference between the inside and outside of the neuron. In (b) the top part show the process of ECoG array implantation and in the bottom it can be seen an x-ray of the implanted array (from [6])

2.3.2 Non-invasive techniques

Non-invasive methods for the acquisition of brain signals are techniques that make it possible to obtain information about brain activity without the need for surgical intervention. Nevertheless, it has great disadvantages since the signal is very weak and has to go through several amplification and filtering processes for its interpretation. In addition to this, it has a spatial resolution that is not as good compared to the methods seen previously. However, due to their ease of implementation, these methods are used throughout this work. Non-invasive measurement techniques will be explained below.

Electroencephalography (EEG)

This is the most popular noninvasive method that captures brain signals from the brain using electrodes placed on the scalp (Fig. 2.3.a). When an action or cognitive process is performed millions of neurons cause synaptic potentials, EEG captures the sum of these potentials which are radially projected to the scalp [25]. Therefore,

only potentials generated around the cortex are detected due to the voltage field produced falls off; as this signal is equivalent to the square of the distance from the source. Thus very deep activity produced in the brain is not detected by EEG [25]. This causes the spatial resolution to be limited to a range of a few centimeters. However, its temporal resolution is very good, being in the millisecond range.

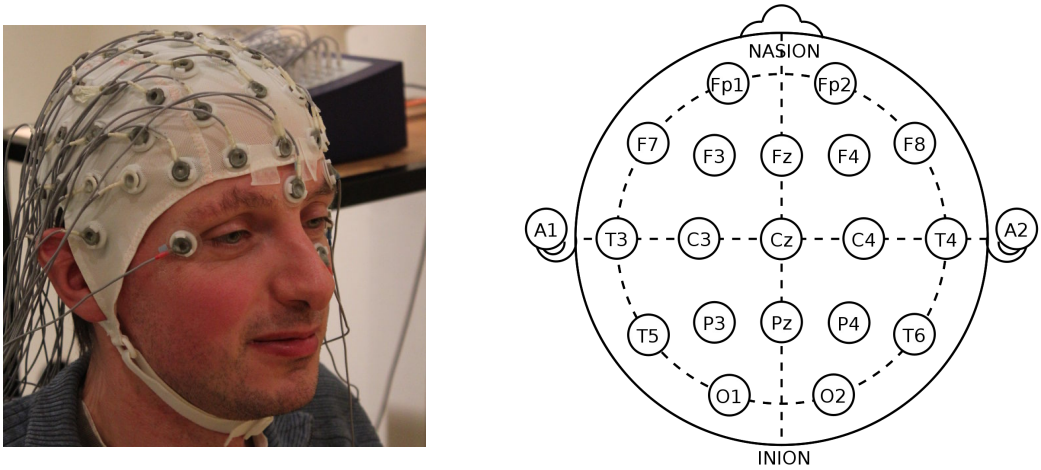
Due to the fact that the layers of the meninges and the skull are interposed between the brain and the electrodes, the captured signal is attenuated, so it is necessary to amplify the signal. However, this process also increases the noise present in the environment. This is why it is necessary to apply temporary filters to reduce the contamination coming from the electrical network and devices (50 or 60 Hz) [25].

EEG electrodes can be found mainly in wet and dry presentations. Wet electrodes use an electrolytic gel as a conductor between the scalp and electrode. The disadvantage of wet electrodes is that the application of this gel requires a previous preparation that may take a considerable amount of time, and this process may be uncomfortable for subjects. However, they are cheaper, and there is greater availability of this technology in research environments. On the other hand, dry electrodes consist of a single "pointed" metal that acts as a conductor between the skin and the electrodes; a disadvantage of this type of electrode is that it generates discomfort in long sessions [28].

In order to generalize the process and allow reproducibility between experiments, the electrodes are placed in specific positions. This is known as the international 10 - 20 system which describes the location of the electrodes on the head [25]. This system is based on the relationship between the location of an electrode and the corresponding area of the brain (Fig. 2.3.b). In addition, the zones refer to the brain lobes and their derivations: Fp for prefrontal, F for frontal, C for central, P for parietal, T for temporal, and O for occipital. Besides, the number refers to how close to the central line they are located, being Z when they are right in the center. Odd numbers indicate that the electrode is located in the left hemisphere, and even numbers indicate that it is located in the right hemisphere (see Fig. 2.3.b).

Brain waves have characteristics and can be classified according to their frequency and amplitude [25]. In addition, there is a correlation between the captured brain waves and the area in which they are being produced. The classification groups are shown in Figure 2.4. Alpha waves can be captured in the occipital region of people who are awake, with their eyes closed, and in a relaxed state. Beta waves are detected over the parietal and frontal regions, usually when a person is alert and

concentrated. In the lower-frequency group, we can find Delta waves that are found in babies or adults during sleep. Theta waves are reported in states of drowsiness or moments of leisure for children and adults. In the waves with a higher frequency range, we find Gamma waves, which have been reported in tasks involving short-term memory and multisensory integration [29]. Moreover, there is a correlation between the amplitude and frequency, where, at lower frequencies, we find signals with a higher amplitude.



(a) 64 electrodes EEG cap (b) 10 – 20 system

Figure 2.3: In (a) EEG cap montage with 64 wet electrodes (from [30]), In (b) the 10 - 20 international system (from [31])

Functional Magnetic Resonance Imaging (fMRI)

fMRI measures neuronal activity indirectly, detecting changes in blood flow due to activation of neurons during specific tasks. This is due to neuronal activity triggers an increase in highly oxygenated blood that replaces oxygen-depleted blood. This response appears after a few seconds following neuronal activity, with a delay between 3 to 6 seconds. This method has a high spatial resolution. However, it has a high response delay because the acquisition and processing is time consuming. Therefore, fMRI is not suitable for the implementation of real-time applications [32].

Magnetoencephalography (MEG)

MEG measure the magnetic fields produced by electrical activity in the brain. MEG is sensitive to activity from the cortical tissue and can detect currents moving perpendicular to the scalp. Compared to EEG, MEG provides a higher temporal reso-

lution and superior spatial resolution [33]. Neural activity generates magnetic fields that are unaffected by the meninges and biological matter such as the skull. A disadvantage of this system is the price, EEG is considerably less expensive than MEG. Moreover, MEG systems are no portable and require a magnetically shielded room to prevent interference from external magnetic signals [34].

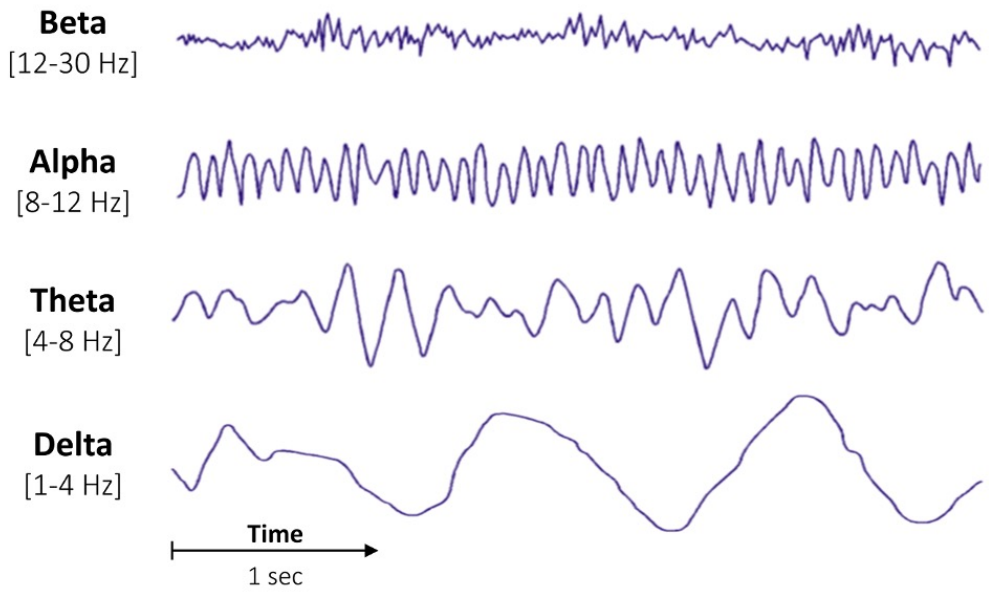


Figure 2.4: Main brain waves found in EEG readings (from [35])

2.4 BCI control approaches

As EEG reflects potential changes in large neuronal populations, these changes can be elicited by training the subject for a period of time or by special external stimulation that can activate large brain regions. Synchronous BCIs have a fairly high popularity. Such BCIs are based on the detection of stereotypical brain responses that are generated after has been presented a stimulus to the subject [36]. This stimulus is associated with a command or a decision of the BCI. Thus, control of the system is not properly initiated by the subject but is tied to the presentation of a stimulus by the BCI. As an advantage, synchronous BCIs usually do not require training by the subject and high levels of accuracy can be obtained by almost anyone [25].

2.4.1 Stimulus Evoked Potentials

Evoked potentials (EP) are responses generated by the brain when the subject is focused on a stimulus. There is a great variety of these types of responses studied in the literature. A brief explanation of some of these methods used in the development of BCIs is presented below.

2.4.2 P300 potential

The P300 evoked potential is an electrical signal generated by the brain in response to a "rare" or unexpected stimulus, its name is given because this potential usually appears approximately 300 milliseconds after stimulus presentation. P300 is shown as a positive voltage deflection of about 5 to 10 uV above the EEG baseline [37] (Fig. 8). In order to be able to observe this voltage change, the stimulus onset must be unpredictable and relevant to the subject. The amplitude of the P300 correlates positively with how relevant the stimulus is and correlates negatively with the probability of the stimulus. This phenomenon is usually seen at parietal regions, although, some of the components are found within temporal and frontal regions [38].

One example of P300 is the speller development, where a matrix of letters or words is presented on a screen [39]. Simultaneously highlighting a row and column randomly, the subject must fix his attention on the desired letter or word. So, when the letter or word the subject is thinking of is highlighted the potential P300 is activated. Usually the subject is asked to count the number of times his or her decision has been highlighted, in order to maintain the subject's concentration and to have the same effectiveness in producing the P300 [40]. An example of a speller can be seen in Figure 2.5.

2.4.3 Steady State Visually Evoked Potential (SSVEP)

SSVEP is a technique used to detect patterns of brain activity in response to flickering visual stimuli [41]. This way, commands can be encoded by lights or buttons. For example, the subject fixes his attention on a flashing light with a specific frequency of 10 Hz, subsequently, his brain can generate in the occipital regions an electrical activity that also oscillates at that same frequency and its respective harmonics (in this case 20 Hz, 30 Hz, 40 Hz, etc) [25]. This type of signal is known as: Steady State Visually Evoked Potential. Using frequency decomposition techniques on the

EEG signal, the BCI can detect the frequency of the stimuli to which the subject is paying attention (see Fig. 2.6).

The stimulation must have an oscillation frequency greater than 5 Hz, where, the best performing frequencies for SSVEP can be found approximately between 6 to 24 Hz [18]. Besides, the highest information transfer rates in noninvasive BCIs have been obtained using this type of signals. Finally, the SSVEP technique is one of the most popular techniques in BCI due to its high accuracy and its ability to control multiple options simultaneously [42].

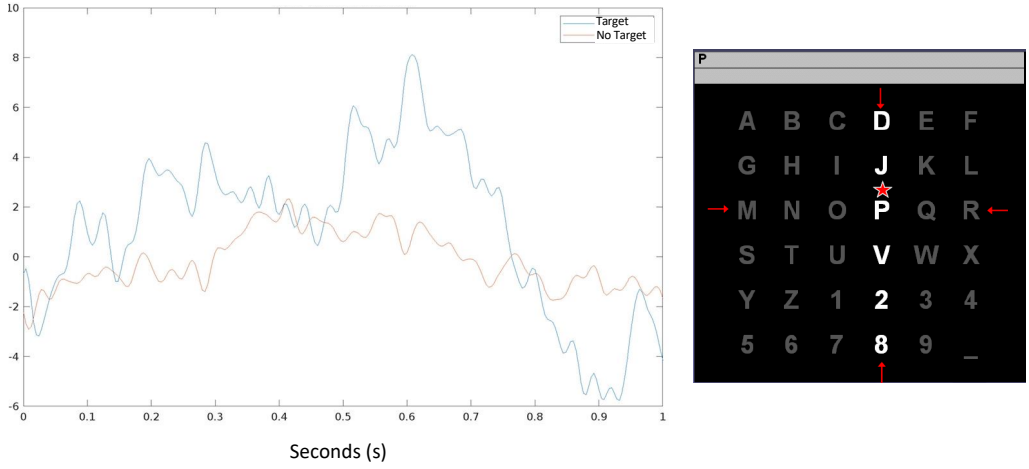


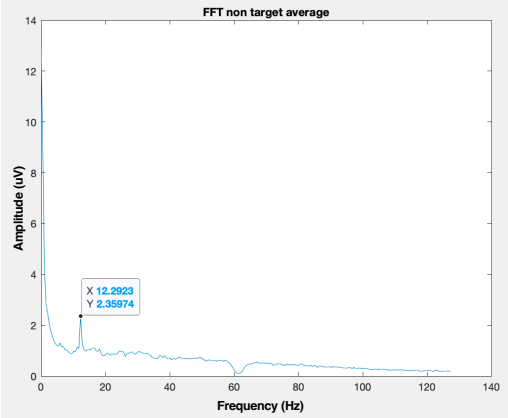
Figure 2.5: In (a) P300 waveform comparing epochs with 'odd' and 'normal' stimuli. In (b) P300 speller design (image b from [39])

2.4.4 Auditory Evoked Potential

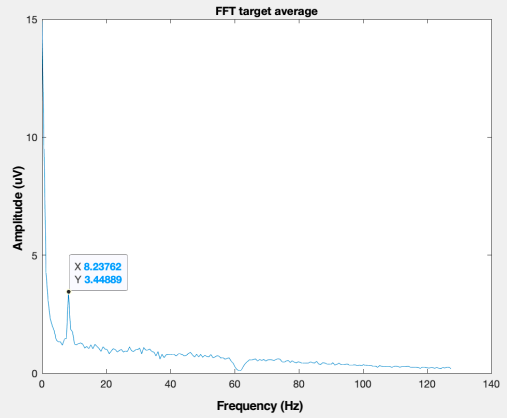
A variation of the P300 evoked potential can be found by changing the stimulation method from visual to auditory [43]. A series of beeps are presented at a certain frequency, where one of them has a different frequency. Therefore, it is detected by the subject who unconsciously generates the evoked potential. This type of auditory evoked potential is used in people who have lost all motor function and who also have a reduced concentration field, since, unlike visual stimulation, auditory stimulation is done for periods of milliseconds [44].

2.5 Spatial filter

The signal coming from EEG has a low temporal resolution, which means that recognizing the exact location where the brain activity comes from is very complicated, so it cannot provide detailed information about brain activity in specific areas of the brain, since neuronal activity in the cortex must pass through several layers (meninges, brain spinal fluid, skull, and scalp) until it reaches the sensor located in the scalp. Spatial filters perform various linear combinations among multiple electrodes, which makes it easier to locate the origin of the signal and reduce the noise that is common in all channels. Two different spatial filtering techniques were discussed for the development of this project: Laplacian and Common Average Referencing (CAR) filters. These filters consist of re-referencing the readings obtained from the raw signal.



(a) 12.2 Hz visual stimulation



(a) 8.2 Hz visual stimulation

Figure 2.6: The plots show amplitude spectra computed using FFT averaged over 30 trials. (a) and (b) show when the subject is looking at stimulation flickering at 12.2 Hz and 8.2 Hz respectively. There is a predominant amplitude peak at those frequencies

2.5.1 Laplacian filter

This filter extracts local activity from the selected electrode, subtracting the average activity of the four nearest orthogonal electrodes [25]. This causes signals from muscle movements, blinks or saccades to be removed from the electrode of interest. The equation representing this filter is found in the equation 2.1.

$$\tilde{\mathbf{s}}_i = \mathbf{s}_i - \frac{1}{4}\sum_{i\in\Theta}\mathbf{s}_i, \tag{2.1}$$

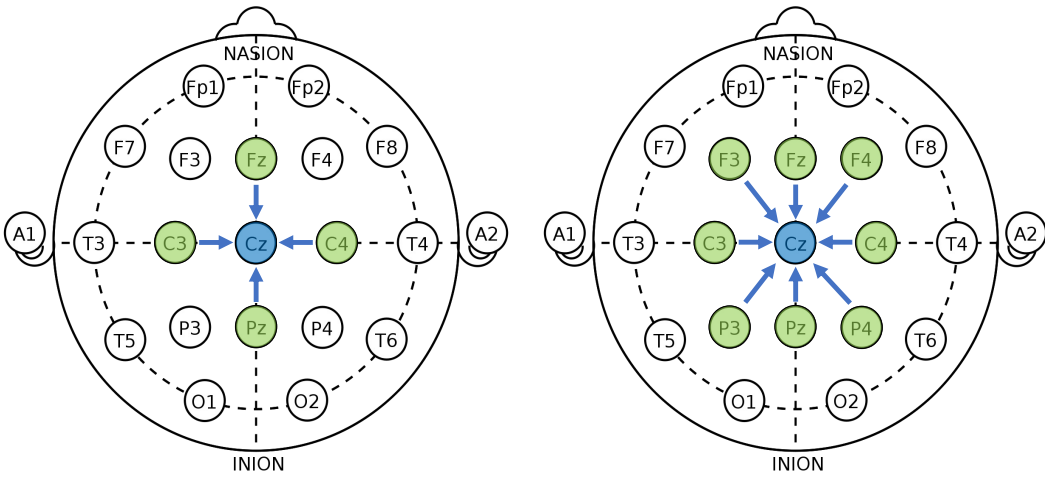
where Θ is the set of the four nearest orthogonal eletrodes, and s_i is the signal from channel i which is the channel of interest. The laplacian filter is denoted by $\bar{s}$. An illustration of the filter applied to the Cz channel can be seen in Figure 2.7a.

2.5.2 Common Average Referencing (CAR)

Common Average Referencing (CAR) takes the average activity of all electrodes and subtracts it from the electrode of interest. A variant of this filter is that instead of taking the activity of all electrodes, a subset of electrodes is selected that are located in one or more areas of interest (motor cortex, visual cortex, among others). This makes that the common activity in all electrodes is reduced and thus highlight the local activity of the electrode of interest. The equation representing this filter is found in the equation 2.2.

$$\tilde{\mathbf{s}}_i = \mathbf{s}_i - \frac{1}{N}\sum_{i=1}^{N}\mathbf{s}_i, \tag{2.2}$$

where N is the number of taken electrodes, and s_i is the signal from channel i, and CAR filter is denoted by $\bar{s}$. An illustration of the filter applied to the Cz channel with $N = 7$ electrodes subtraction can be seen in Figure 2.7b.



(a) Laplacian

(b) Common Average Referencing

Figure 2.7: Illustration of spatial filter. (a) an example of Laplacian filter applied to Cz channel, (b) an example of CAR filter applied to Cz channel taking its seven neighbours

2.6 Feature Extraction

Feature extraction is the process of identifying and extracting relevant features or patterns from the raw EEG signal [25]. These features can be used to represent different aspects of brain activity. These features can be frequency, amplitude or temporal features. For model training or classification algorithms it is more efficient and accurate to use the extracted features than the raw signal. Some extraction techniques used in EEG analysis include: Time domain features, Frequency domain features, Time frequency domain features and Statistical features. Some extraction methods used in this work are presented below.

2.6.1 Frequency Domain Analysis

For the analysis of SSVEP signals, it is necessary to know at what frequency the occipital channel readings are oscillating. For this reason, it is necessary to study the frequency properties of the acquired signal.

Power Spectral Density (PSD)

PSD is calculated by taking the Fourier transform of a signal and squaring the magnitude of the resulting complex values [45]. The resulting values represent the power of each frequency component in the signal. The PSD is often normalized by dividing the power of each frequency component by the total power of the signal, which provides the relative contribution of each frequency component to the total power of the signal [25].

The PSD of a signal is typically plotted as a graph, with frequency on the x-axis and power spectral density on the y-axis. The shape of the PSD plot can provide important information about the frequency distribution of the signal. In a EEG signal, a PSD plot during deep sleep would typically show high power in the low frequency range (delta and theta bands), whereas a PSD plot of an EEG signal during an active cognitive task would typically show high power in the high frequency range (alpha and beta bands) [46].

2.6.2 Statistical features

Statistical features can be extracted from EEG signals in order to reduce dimensionality, separate independent components or study the relationship among signals. Statistical features are very important in the analysis of SSVEP signals, since it al-

lows comparing the SSVEP signal with a set of fundamental signals oscillating at specific frequencies and thus looking at the relationship between them.

Canonical Correlation Analysis (CCA)

Canonical correlation analysis (CCA) is a multivariate statistical technique that analyzes the relationship between two sets of variables [47]. It attempts to identify the linear combinations of variables in each set that are maximally correlated with each other. CCA is usually used to explore the relationships between multiple variables and identify underlying patterns. In CCA, the two sets of variables are called X and Y variables. The objective is to find the linear combinations of X and Y that maximize their correlation. The resulting correlations are called canonical correlations. CCA can be used to identify how the variables in the two sets are related and which variables contribute most to the relationship.

For the analysis of SSVEP with CCA, as variable X we have the signal corresponding to the occipital channels, such that $X \in \mathbb{R}^{ch \times n}$, where ch is the number of channels and n is the length of the signal. As variable Y, it is has the fundamental frequencies (sine and cosine) and their harmonics with a frequency equal to the one we want to study, such that $Y \in \mathbb{R}^{2(f+1) \times n}$, where f is the number of harmonic to be studied [47]. For example, if we want to study the relationship that our signal has against a frequency of 8 Hz and two of its harmonics. Then, X would have the information of the occipital channels (O1 and O2) and Y would be composed by:

$$y(t) = \begin{pmatrix} \sin(8\pi \times t) \\ \cos(8\pi \times t) \\ \sin(16\pi \times t) \\ \cos(16\pi \times t) \\ \sin(24\pi \times t) \\ \cos(24\pi \times t) \end{pmatrix}$$

Finally, different linear combinations are calculated to see which one has the highest correlation between X and Y data, an illustration of the CCA process can be seen in Figure 2.8.

2.7 BCI performance

To evaluate the performance of classification models there are several metrics that can be used. Many of these metrics are derived from the confusion matrix. The

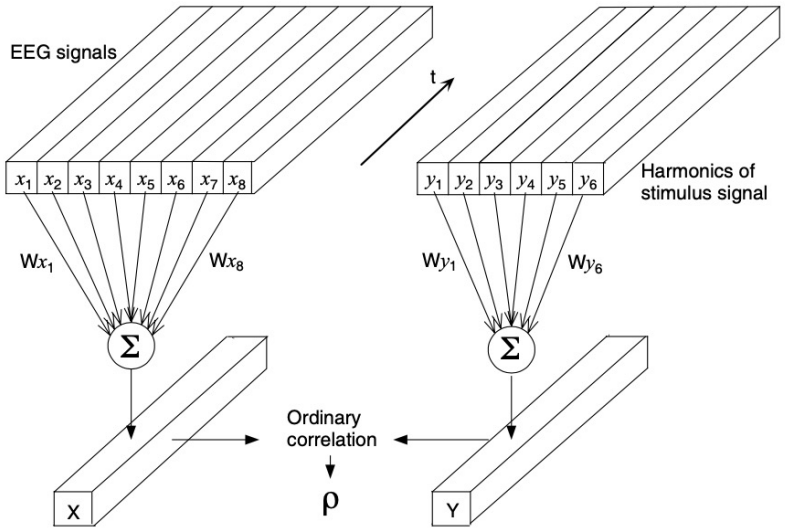


Figure 2.8: An illustration for usage of CCA in EEG signal analysis. $x_1, ..., x_8$ are signals from 8 EEG channels and $y_1, ..., y_8$ are Fourier series of a given frequency period signal. The CCA finds the linear combination coefficients $w_{x1}, ..., w_{x8}$ and $w_{y1}, ..., w_{y8}$, which gives the largest correlation between X and Y (from [47])

confusion matrix is responsible for counting the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). For example, observing a triangle oscillating at 8 Hz triggers (command 1) and a square oscillating at 12 Hz triggers (command 2). This example would have a confusion matrix for SSVEP-BCI as follows:

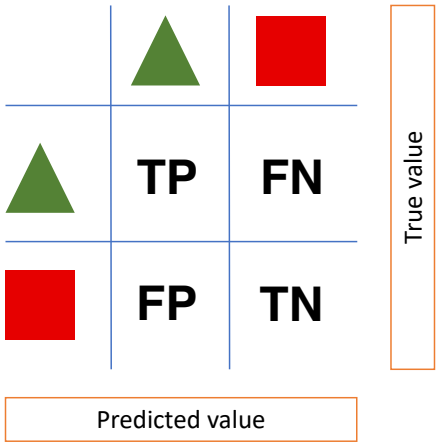


Figure 2.9: Illustration of confusion matrix. The triangle denoted in green refers to the positive class. The square denoted in red refers to the negative class

In addition, there is a special metric in the development of BCIs that indicates

the efficiency and effectiveness in the transmission of information between a subject and an external device, which is known as Information Transfer Rate (ITR). A description of the evaluation metrics commonly used in BCIs is presented below.

2.7.1 Accuracy (ACC)

It is a measure of the precision of a model to correctly predict the right class or label of a sample. This is the proportion of correct predictions made by the model compared to the total predictions (see formula 2.3). The prediction value varies from 0 to 1, where 0 means that the model has not made any correct predictions and 1 means that it has made all predictions correctly. Therefore, the higher the accuracy, the better the performance of the model.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.3}$$

2.7.2 Kappa score

The kappa score evaluates the agreement between two observers, where each one classifies n items within c mutually exclusive classes. This method is similar to accuracy, but also, takes into account the possibility that the agreement between the observers is due to chance. The kappa score can take values between -1 and 1, where 1 means that there is perfect agreement among observers, and therefore, no misclassification errors. On the other hand, if the kappa score is equal to 0, it means that the agreement is equal to what would be expected by chance. This indicates that there is no significant relationship between the classifications of the observers or systems. Finally, if the kappa score is negative it means that the observers disagree beyond what would be expected by chance. The way to calculate the kappa score metric can be found in equation 2.4.

$$\kappa = \frac{P_o - P_e}{1 - P_e} \tag{2.4}$$

where,

$$P_o = \frac{TP + TN}{TP + TN + FP + FN}$$

$$P_e = \frac{(TP + FN) \times (TP + FP) + (FN + TN) \times (FP + TN)}{(TP + TN + FP + FN)^2}$$

P_o is the proportion of agreements between observers (it has the same equation as the accuracy). P_e is the proportion of random agreements among observers.

2.7.3 True Positive Rate (TPR) and False Positive Rate (FPR)

The true positive rate measures the proportion of true positive cases that are correctly identified as positive by the model (see equation 2.5). Otherwise, the false positive rate measures the proportion of negative cases that are incorrectly classified as positive by the model (see equation 2.6). Both measures provide a notion of sensitivity and specificity of the model, for TPR by saying how well the model can classify positive cases, and for FPR by measuring the frequency in which the model makes false alarms by identifying negative cases as positive.

$$TPR = \frac{TP}{TP + FN} \quad (2.5) \qquad \text{and} \qquad FPR = \frac{FP}{FP + TN} \quad (2.6)$$

2.7.4 Information Transfer Rate (ITR)

ITR refers to the amount of information that a subject can transmit through the BCI in a given period of time. It can be calculated using different equations based on the correct classification rate, the time available to send the command, the mutual information among the EEG signal and the commands that the subject wishes to send to the BCI. For the development of this work, we will use the equation stipulated in [48] (see equation 2.8), where the different parameters to take into account for a correct ITR calculation are analyzed.

$$B = \log_2 N + P \log_2 P + (1 - P) \log_2[(1 - P)/(N - 1)], \quad (2.7)$$

where, B, is the ITR in bit rate (bits/symbol), N, is the number of possible choices, P, classifier accuracy and B_t in bits/min is used to indicate the BCI ITR:

$$B_t = B(60/T), \quad (2.8)$$

where, T, is the time needed to convey each symbol.

2.8 Related works

Twelve scientific articles on BCI development for the control of assistive devices were found using the process proposed in Figure 2.10. As an initial step, a search equation containing the following keywords was proposed: BCI, SSVEP, CCA, upper limb impairments, and serious game (see equation 2.9). A query was made in the main scientific databases: IEEE Xplore, Scorpus, PudMed, and Google Scholar. After this, we filtered by language (English), year of publication (from 2015 to 2023), and eliminated duplicates that were found with records stored in Google Scholar. Then, the results were sorted by the number of citations and those that the number of citations was greater than 10 were chosen. Finally, they were manually analyzed by title, keywords, abstract and introduction. Those papers related to the development of BCI-SSVEP for control of lower limb assist devices were included.

BCI & SSVEP & CCA & (“upper limb” | “lower limb”) | “serious game” (2.9)

Several stimulation systems have been proposed for generating SSVEP signals. One method that stands out for its simplicity and versatility is the use of augmented reality [49], [50], [51]. Specialized glasses, such as Google HoloLens, are used to project visual stimuli. Using this idea of operation in [51] they managed to develop an SSVEP-BCI that was tested in 10 healthy subjects with an average accuracy of 94.97%. Those glasses projected a matrix of 12 commands where each command indicated the rows and columns of a 11 x 11 board. On the board a robotic hand accommodated objects in the desired position (Fig. 2.11). The robotic arm used was a six-degree- of-freedom mechanical arm (Kinova jaco2, Canada). While using the glasses, the subject could control the device while being able to see the background, as it was developing movement control by the robotic hand. Although the presentation of stimuli through the use of augmented reality glasses presents greater comfort for the subject, it has very little versatility because it cannot handle a high range of frequencies oscillating at the same time, which is why the use of technologies for SSVEP stimulation is carried out through LEDs or LCD screens.

Regardless of the method of visual stimulation used, SSVEP has a great disadvantage in presenting visual discomfort and fatigue when used for long periods of time, which has led to the exploration of hybrid BCIs, that combine two or more control methods to provide a more accurate response, such as the use of SSVEP and MI [52]. This mix has been used to control a NAO robot [53], which by SSVEP controlled three motion actions (forward, left, and right), and by MI, controlled the

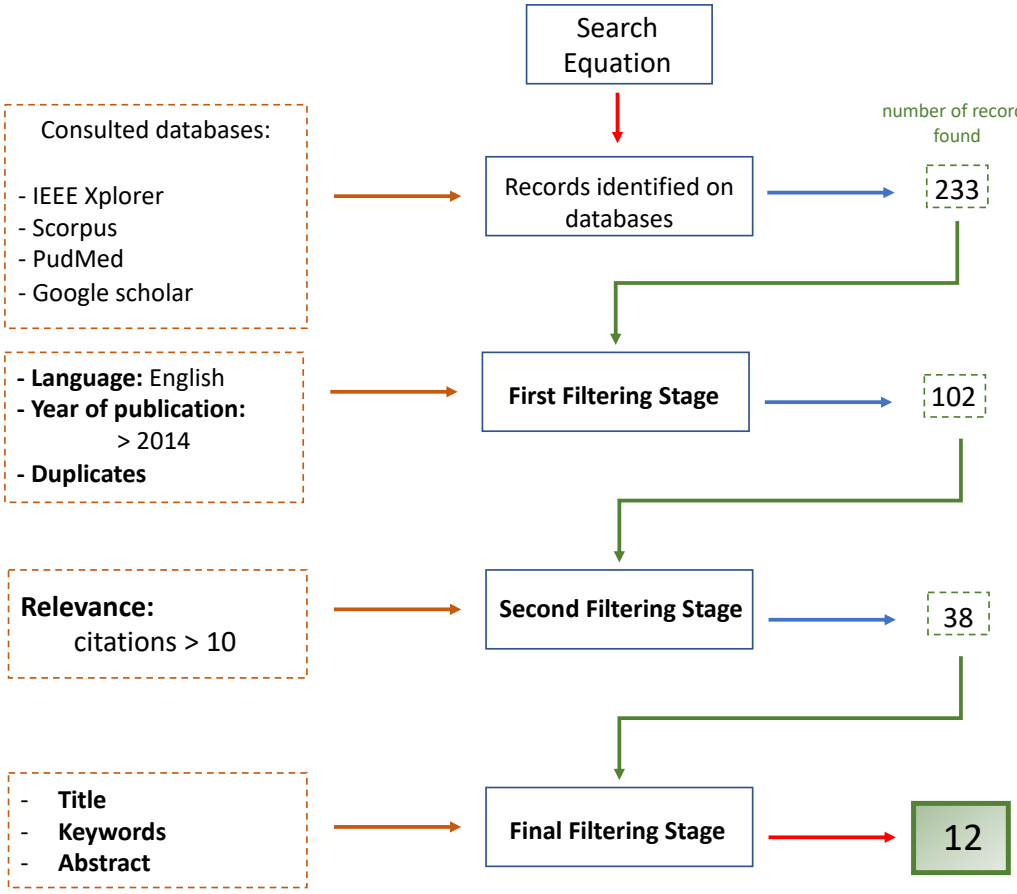


Figure 2.10: Literature review flow chart of articles on BCI system using SSVEP as control paradigm to control external devices for upper limbs assistance

grip of the NAO robot hand. This change in control occurred automatically when the amplitude of the μ band exceeded a threshold (0.03 for μ band). This study was conducted on three healthy subjects, with an average accuracy of 89%. Similarly, this hybrid BCI was used to control the horizontal and vertical movement of a drone [54], where the subject could change the control mode by blinking his eyes; in the SSVEP mode, two LED lights were located above a monitor, where each one meant a direction of rotation. Simultaneously, in the MI mode, the subject had to imagine the drone turning left or right to perform the drone's movement. Another study in [55], MI and SSVEP were used to improve the accuracy of the system by using electrodes located at C3 and C4 to detect frequencies related to visual stimulation in order to use signals coming from imagination to detect SSVEP signals.

Just as there is a combination of control paradigms for BCI, SSVEP-BCI has also been combined with computer vision algorithms [56], [50]. This was performed to make the movement carried out by the robot error-free. In [50] the subject

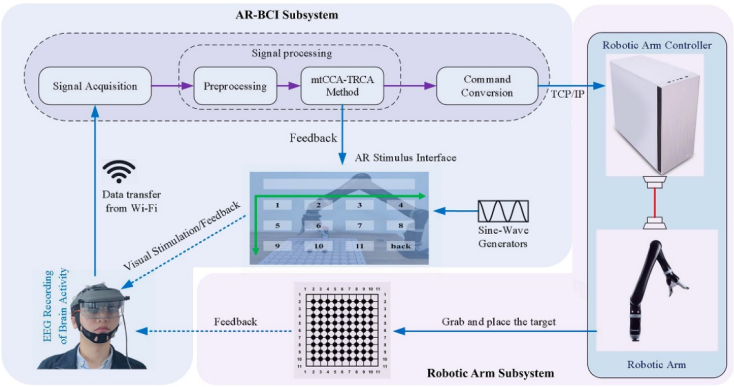


Figure 2.11: Methodology used in [51]. In bottom left, the EEG montage with the augmented reality glasses. In the middle the user perspective wearing the glasses. In bottom right, the robotic arm used (Figure from [51])

could control a robotic hand to move on a board with three different objects (blue, yellow, and red objects). The subject has four commands shown via an LCD display, where three commands represents the selection of one of the three objects and one command represent a undo button which undo the previous operation. Once the object is selected by SSVEP, the computer vision system recognizes the location of the object and returns it to its final location. This system was tested on 12 healthy subjects with an average accuracy of 93.96%.

In addition complete assistance systems for people with physical disabilities have been developed [57], [58]. A lower-limb exoskeleton with an integrated stimulation plate at chest height with four green LEDs were developed. The LEDs were used for motion control using SSVEP, where each LED represents turn right, turn left, stand up, and sit down control. This system was tested on 11 healthy subjects with an accuracy of 91.3% and an ITR of 32.9 bits/min in the online mode. In order to be able to have more commands for the control of assistive devices in [59] a system for the control of up to 48 commands using SSVEP with 6 different frequencies and eye tracking was proposed. For this, a matrix of six columns by eight rows is placed, where in each column, six different frequencies are located. This process is repeated in all the rows but with different order in the frequencies of stimulus that appear in each column, eye tracking is performed, and when the subject changes row, the system detects this change and so, it is already recognized in which row the subject is focusing his attention. Then, he/she fixes his/her attention in the column he/she wants to choose, and through CCA, feature extraction is performed. Owing to the high number of commands, the system had an average ITR of 184 bits/min and an accuracy of 90% among 20 healthy subjects.

Chapter 3

Methods and Materials

The aim of this chapter is to present the procedures that were executed to develop the BCI and subsequently integrate it with Unity to provide the subject with visual feedback. The process was conducted in five distinct phases. The initial phase involved data acquisition where the subject was prepared with EEG instrumentation and the data collection process was initiated. The second phase used a visual stimulation for the subject with a device designed specifically for this project. The third phase involved preprocessing the acquired data, followed by classification. Finally, feedback regarding the classification of the system was provided to the subject. Figure 3.1 illustrates the general methodology used in this study.

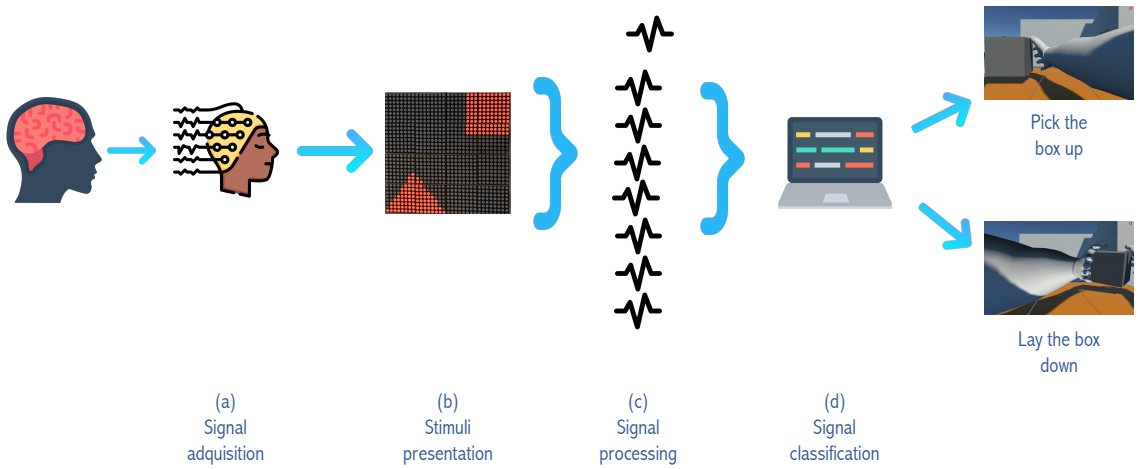


Figure 3.1: Methodology used for the development of a BCI capable of classifying SSVEP

3.1 Signal acquisition

To measure the electrical signals produced by the brain, a noninvasive EEG method was used. The device used to acquire the brain signals was G.NAUTILUS RESEARCH, which is a wireless EEG cap designed for research applications. Eight wet electrodes were used. The locations of such electrodes were used to extract the information to meet our goal of classifying the brain signals. This location was selected in compliance with the parameters given by the 10-20 system (see Fig. 2.3.b).

Two electrodes were placed in the prefrontal areas (FP1 and FP2) to detect blinking and eye movements [60]. An algorithm was designed in Python that eliminated the pieces of the signal containing blink information. Subsequently, four electrodes were placed in the primary and supplementary motor zones (C3, C4, P3, P4), to perform a spatial filter that would minimize the appearance of artifacts and unwanted signals. This location was selected to study these channels for future projects involving motor intention and evoked potentials ([61], [62]). Finally, two electrodes were located in the visual cortex (O1, O2) to extract signals resulting from the subject looking at the stimuli [63]. GND and REF were placed in the FZ and left earlobe, respectively. A graph of this setup is shown in Figure 3.2.

Signals were obtained at a sampling rate of 250 Hz. As G.NAUTILUS is a wireless cap, the connection between the computer and the device was made using a program designed by the same manufacturer called g.needaccess, which allows the sensors to be read by third-party programs. Thus, OpenViBE software was used for more comfortable and efficient processing. This software is open source and uses a system of "boxes" that are placed in scenarios. Each scenario is capable of performing a sequence of processing steps. Also, such scenarios can be used in both online and offline studies. Among its functions are spatial filters, temporal filters, artifact detectors, and pre-trained machine learning models. Additionally, OpenViBE has the ability to be programmed in Python and run scripts with the acquired data in real time.

3.2 Stimuli presentation

For the generation of SSVEP-related potentials, visual stimulation of the subject is necessary. As explained in the previous section (2.4.3), the brain produces electrical signals in the occipital region that oscillate at the same frequency as the visual stimulus, which is constantly flickering at a specific frequency. Generally the range

of frequencies used for these stimuli can be classified into three groups: low (4 - 12 Hz), medium (12 - 30 Hz) and high (> 30 Hz) [64]. These stimuli can be generated using specialized software.

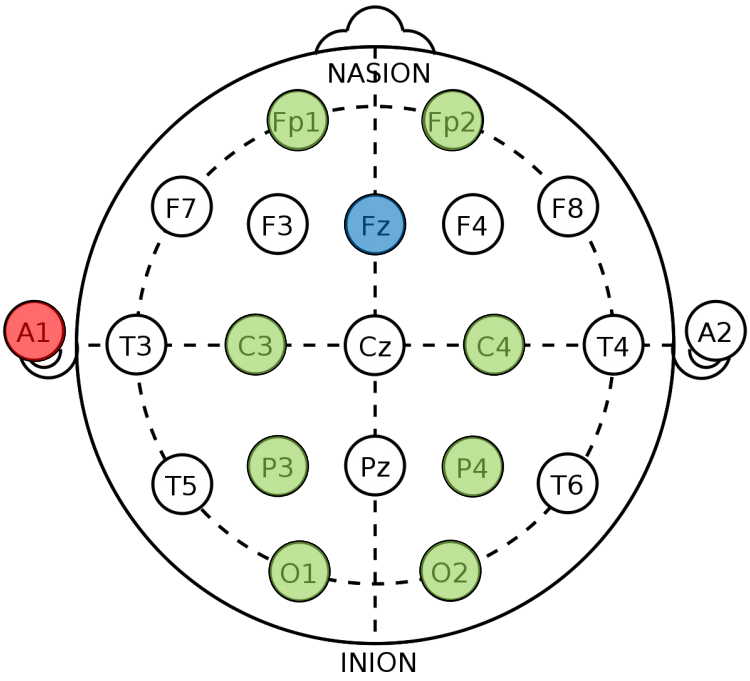


Figure 3.2: Position of the electrodes according to the 10-20 system. In red reference electrode. In blue ground. In green the lecture electrodes

PsychoPy is an open-source platform for the creation of psychological and neuroscientific experiments that features a graphical interface to facilitate the creation and editing of visual and auditory stimuli. It has a Python implementation which makes the modification of experimental characteristics agile and simple [65]. In addition, we have a MATLAB library produced by g.tec, called g.HIsys. This is similar to the tool explained above; it allows the creation of visual and auditory stimuli but also has preprocessing blocks [66]. However, this program has several limitations, as it is expensive and works only for products of the same brand.

In addition to these software programs, there are many alternatives for the generation of visual stimuli. However, the solutions provided by the software are not very suitable for SSVEP because a computer performs a large number of tasks simultaneously. This means that the frequency at which the stimulus is presented is not the desired one and has values that vary from epoch to epoch. Besides, the presentation of several stimuli at the same time is very complicated and the error increases significantly, owing to the limitations of the peripherals of a computer such as its screen and processing. In this sense, it does not provide precise time control

for an adequate presentation of SSVEP stimuli.

In addition to this, for the objective of the project where an application focused on use in the industry environment is proposed, it is necessary that the device used for stimuli will be portable and compact. This way can be located in any area and has no cable complications. Furthermore, resistance to withstand any fall or possible liquid falling on it. Finally, we want it to be economical so that it can be used by several employees. This is why a low-cost visual stimulator was designed using Arduino and 3D printing.

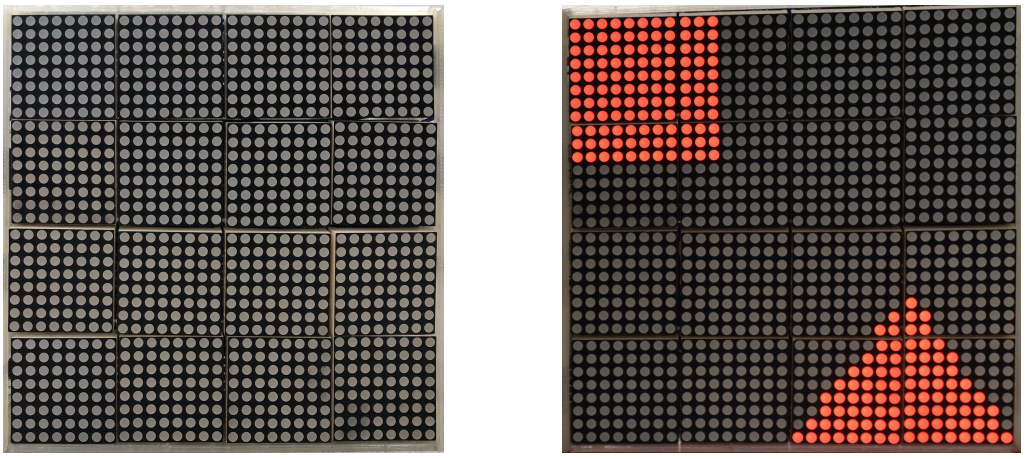
3.2.1 Stimuli presenter

The Arduino UNO was chosen for the construction of the stimulator because of its versatility, low cost, and friendly programming. For the visualization of the stimulus were used 16 modules of 8 x 8 led matrix controlled by the Max7219 controller, which allows the incorporation of several matrices to have a synchronous and simple operation. The communication between these devices is performed by 4 pins on an SPI interface. The color of the matrix was chosen as red due to studies show that colors with large wavelengths, such as red and orange, can capture more attention from the subject and generate SSVEP waves of greater amplitude than other colors with smaller wavelengths [67]. A square placement of the LED arrays was chosen, with four rows and four columns, as shown in figure 3.3.a.

The proposed BCI has the ability to control two commands. Therefore, it is necessary to have two visual stimuli oscillating at different frequencies, each one associated with the desired command. In this manner, two figures were placed in the LED array: a triangle in the upper left and a square in the lower right as shown in Figure 3.3.b. A certain action is linked to the moment when the subject looks at the triangle and another action is carried out when the subject looks at the square. The stimuli are presented simultaneously so that the subject can choose the desired command at the desired time and does not have to wait for it to appear on the screen, as would be the case with a P300 speller.

To generate the desired brain response, we need to flash the figures at different frequencies. To do this with Arduino is not a simple task, due to having two figures oscillating at different frequencies, it is necessary to perform parallel processes. There are several alternatives for the solution of this problem, one of them is the use of embedded devices with higher computational power, such as a raspberry pi or a Jetson nano. However, the use of these devices would increase the cost of the stimulator, which would not meet one of the development objectives. On the other hand,

two Arduino boards could be used and each one would be in charge of managing a stimulus. The problem with this is that it increases the complexity and simplicity of the device, which is not practical.



(a) LED matrix layout (b) Figures location

Figure 3.3: Stimulator display. (a) Selected layout for the 16 8x8 led matrix modules. In (b) the two stimuli used to generate SSVEP

Finally, an algorithm that allows the presentation of two figures oscillating at different frequencies in one Arduino board was developed. For this, two low oscillation frequencies were chosen: for the triangle, 8 Hz, and for the square, 12 Hz. To perform this process sequentially, a common point was found where the oscillation cycles meet and repeat, as illustrated in figure 3.4.

As can be seen in Figure 3.4, the procedure consists of finding a division of the periods such that the sequence is repeated every certain time. In this case, by dividing the 125 ms period corresponding to the triangle into 6, we have fractions of 20.83 ms. Therefore, two of these fractions are equivalent to half a period where the square is ON and three of these fractions are equivalent to half a period where the triangle is ON. Repeating this process for each period of 125 ms we obtain a simulation of the parallelization process for the operation of both visual stimuli. Finally, a check of the correct frequency operation is performed. For this, digital outputs are added for each figure, and then, these values are measured with an oscilloscope, having a result of 8.02 ± 0.04 Hz for the triangle and 12.06 ± 0.04 Hz for the square.

Once the algorithm design was finished and the correct operation of the system was checked, a 3D case was designed to make the system compact and easy to transport. In addition to this, six (6) Arduino digital outputs were left available to

be used at the disposal of the experiment. In this case, they were used to make the marking of stimuli in the OpenViBE data acquisition software. The final result of the stimulator can be seen in Figure 3.5.

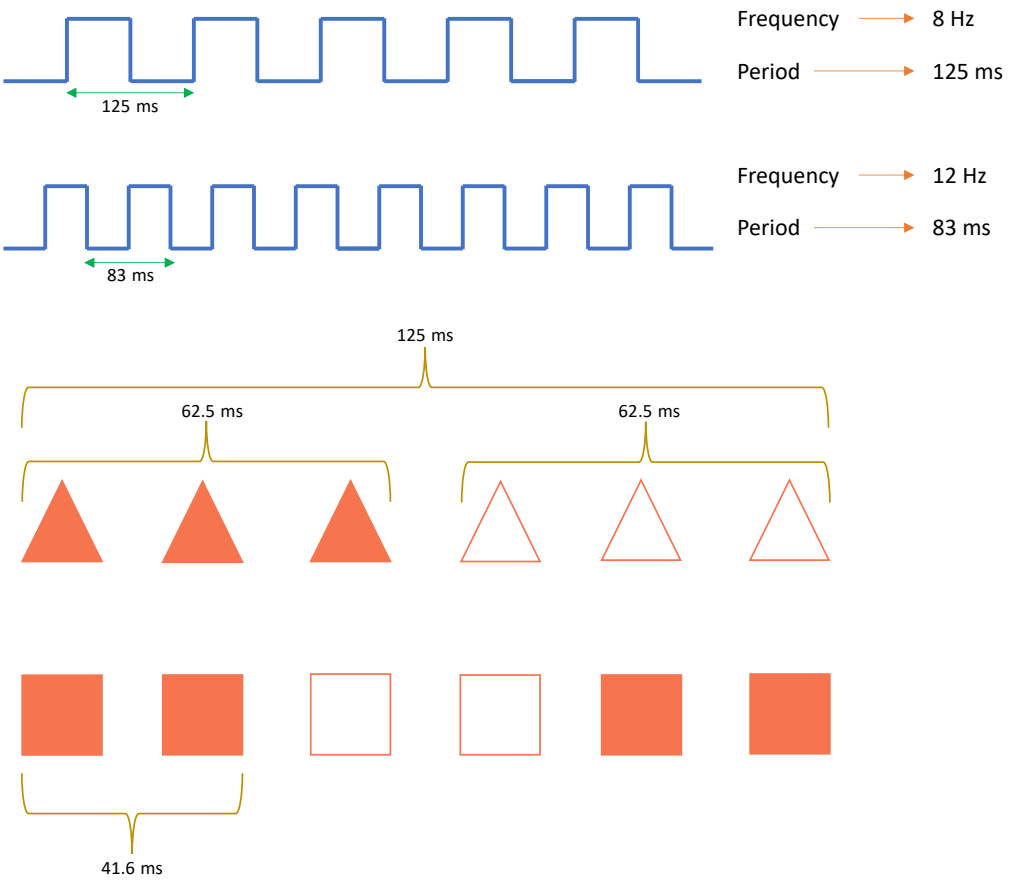
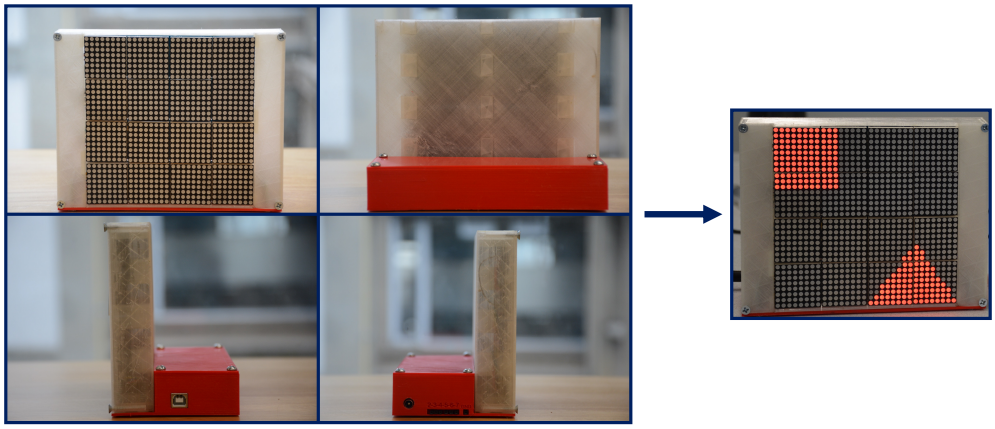


Figure 3.4: Illustration of frequency separation to be able to make dual stimuli with Arduino

3.2.2 Device features of the visual stimulator

In addition to the already explained functions of the stimulator, extra functions were programmed to enable the tweak of different characteristic for SSVEP paradigm. The stimulation for SSVEP is carry out where two geometric figures (triangle and square) oscillate at 8 Hz and 12 Hz, respectively. Within this mode, we can find two types of operations: (1.1) Offline mode, where every certain amount of time an indication appears randomly under some figure to indicate the subject where to focus their attention. This, in order to have marks in the periods of the signal where the target frequency is known and thus, have evaluation metrics on the classification

models implemented. (1.2) Online mode, where the subject focuses their attention on the stimulus of their choice and it is given feedback on the classification made through a serious game.



(a) Stimulator orthographic views (b) Stimuli

Figure 3.5: Final result of the visual stimulator. In (a) the orthographic views, where it can be seen all available connectors for programming, digital outputs and power supply. In (b) the display of the visual stimuli.

Similar to other commercial programs for stimulus generation, a modification of the master file called "master.txt" is sufficient to change the type of paradigm to be used and tune the parameters of the experiment. An example is shown in Figure 3.6. Some of the parameters are shared between the two types of paradigm: `''number_of_trials''` is the number of times any of the figures is shown, `''stimuli_time''` (seconds) is the time for which the figure should be shown, `''inter_stimuli_time''` (seconds) is the time that must elapse between the appearance of each figure, `''target_probability''` (between 0 and 1. 0) is the probability that the triangle will appear, i.e., if this value is 0.3 it means that out of 10 trials the triangle is expected to appear in 3. Finally, the parameters with "pin" in the name, assigns that number (between 2 and 7) to the corresponding pin. In the case of SSVEP, `''operation_mode''` refers to whether it is offline mode (1) or online mode (2).

3.3 Signal preprocessing

For the preprocessing stage, the emphasis was placed on noise reduction and trying to eliminate high and low frequencies that could contaminate the measurement. A complete flowchart of the applied process can be seen in figure 3.7. For this purpose, a band-pass filter (Butterworth of order four) was applied. The Butterworth

bandpass filter is a signal-processing filter designed to have a flat frequency response in the passband. This is also referred to as a maximally flat-magnitude filter. The order of the filter determines the steepness of the transition between the passband and the stopband. In this case, an order 4 filter was used to achieve the desired filtering level.

```
paradigm_type: SSVEP

operation_mode: 1
number_of_trials: 60
stimuli_time: 5
inter_stimuli_time: 0.3
target_probability: 0.5

target_pin: 4
non_target_pin: 5
```

(a) SSVEP configuration file

Figure 3.6: An example of the configuration file that allows the use of different parameters tuning

As input signal we receive the eight channels mentioned in section 3.1, then, we select all the channels to which we apply a temporal filtering. A Butterworth type band pass filter is applied, preserving the frequencies within the range of 0.1 Hz to 30 Hz, effectively eliminating unwanted signals from the electrical network (48 Hz - 52 Hz) [68] and artifacts from the muscles that occur at high frequencies [69]. Additionally, within this range, not only the frequencies of interest related to our SSVEP were intended to be conserved, but also, the frequencies of interest (8 Hz and 12 Hz) and their first harmonic (16 Hz and 24 Hz). In addition, the frequency bands related to MI were preserved to use the collected information for future analysis. A more in-depth description of these values is provided in the section 2.3.2. By preserving the frequencies within a specified range, it is possible to obtain a clean signal that can be effectively analyzed using feature extraction and classification algorithms. This allows for a more accurate analysis and interpretation of the data.

With the signal temporally filtered, a separation is made between the prefrontal channels and the rest of the electrodes. These signals given by the 6 channels

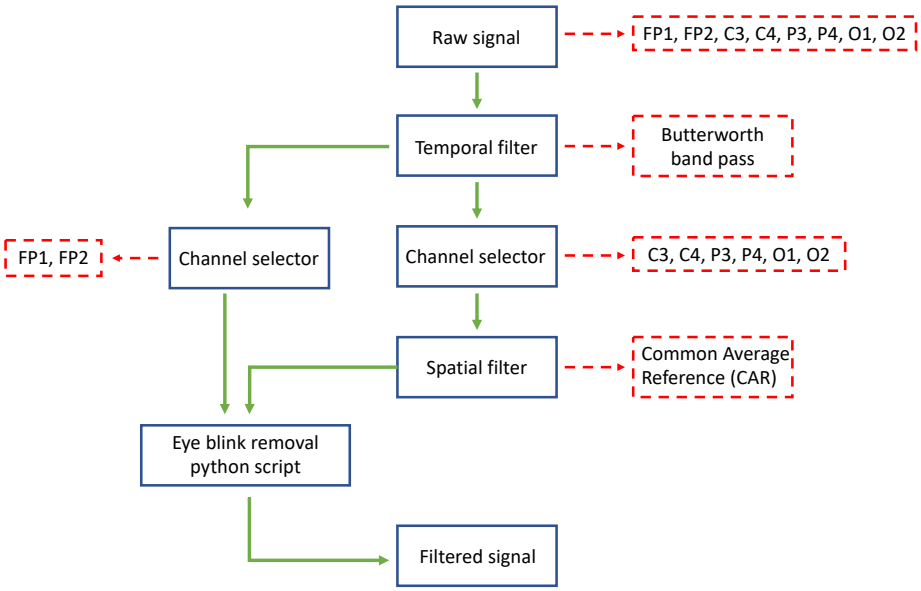


Figure 3.7: Flowchart with the procedure performed on the signal for noise reduction

resulting from the division (C3, C4, P3, P4, O1, O2) are filtered by means of a spatial filter. The importance of these filters is given due to the low temporal resolution that EEG measurements have. This is because millions of neurons perform a specific process and there is not enough precision to know specifically where the cortical activation was given. For spatial filtering, signals from multiple electrodes are linearly combined, which makes it easy to locate the point of origin. To work with SSVEP-related signals, we intend to eliminate artifact information from eye movements, blinks, and involuntary muscle responses. Therefore, we will use the CAR filter, which was explained in the previous section (2.5.2). Mainly, this method takes the average of the channels and subtracts them from the channel of interest, thus reducing the impact of signals that are present in common in all channels and highlighting the signal that is specific to the channel to be studied.

Simultaneously, the remaining channels of the division (FP1 and FP2) were analyzed to determine the parts of the signal containing flicker information. A Python algorithm that received two input parameters was developed: (1) prefrontal channel information and (2) signals in which flicker sections were to be removed. As the output of the program, the signal given as input (2) is returned with a linear interpolation located in the areas where the eye blink or blinks were detected. The pseudocode used for this algorithm is given in Algorithm 2. From algorithm 1 *find*

maxima is intend to return the index of the absolute maximum in an subset, then *findDerivativeChange* receives as input the index of the absolute maximum and returns the index where there is a change of sign in the derivative. That change means that the "spike" referent to the blink is over. Finally, *linearInterpolation* computes a simple linear interpolation between the signal and the points at which the blink occurs.

To implement this algorithm with the signals obtained by the G.NAUTILUS, a connection was made between OpenViBE and Python using the Python script "box". In addition to this, a call is made to the script every time blocks of 32 observations are obtained. An example of how the algorithm works can be seen in Figure 3.8. It can be seen how the algorithm detects several blinks in a signal with a length of 2 seconds.

Algorithm 1 `FindBlinkIndex`

Require: $chunk$, $threshold$
  $index \leftarrow \text{find}(chunk > threshold)$
  if $\text{length}(index) = 0$ **then return** $\emptyset$
  else
    for $i = 0$ to $\text{length}(index)$ **do**
      $maxima \leftarrow \text{findMaxima}(chunk)$
      $leftChange \leftarrow \text{findDerivativeChange}(chunk, maxima, 1)$
      $rightChange \leftarrow \text{findDerivativeChange}(chunk, maxima, -1)$
      $result \leftarrow [leftChange, rightChange]$
    end for
     return $result$
  end if

Algorithm 2 RemoveBlinkIndex

Require: $signal$, $threshold$
  $Fp1Signal \leftarrow signal[0]$
  $result \leftarrow FindBlinkIndex(Fp1Signal, threshold)$
  for $i = 0$ to $\text{length}(result)$ **do**
    $blinkNum \leftarrow result[i]$
    $begin \leftarrow blinkNum[0]$
    $end \leftarrow blinkNum[1]$
    $interpolationResults \leftarrow \text{linearInterpolation}(signal, begin, end)$
    $signal \leftarrow interpolationResults$
  end for
     return $signal$

Finally, the signal is reassembled, joining all the channels into a single array

as it originally came from the device. This way, we have the signal completely preprocessed and filtered to be used in feature extraction algorithms for SSVEP. This process will be explained in detail in the next section.

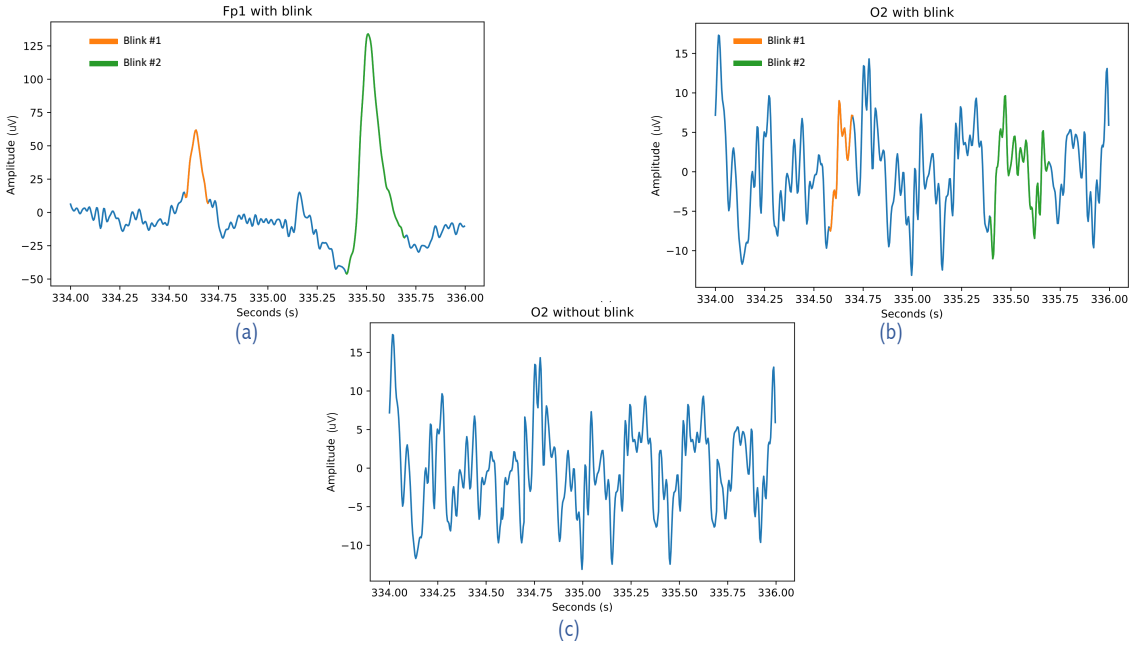


Figure 3.8: An example of the blink detection algorithm, threshold of 30 uV was applied to the signal. In (a) is the signal of Fp1 channel, then, in orange and green there are blinks detected by the algorithm. In (b) the view of the signal from the occipital channel O1, the time where the blink occurs is highlighted in orange and green respectively. In (c) the result of linear interpolation between the start and end point from the blink epoch.

3.4 Feature extraction

For the analysis of SSVEP signals, the objective is to study the frequency properties. Properties related to efficiency and performance such as PSD, ICA , LDA, CCA are usually employed. During this project the use of PSD and CCA for feature extraction was explored.

3.4.1 PSD

As explained in section 2.6.1 PSD represents the energy distribution of the signal in the frequency domain. The PSD can be plotted as a function of frequency to visualize the energy distribution at different frequencies of the signal. Finally, a tolerance ϵ is set such that if the highest energy is at $8 \pm \epsilon$ Hz, it is established that

the subject is looking at the triangle (fig. 3.9.a). However, if the highest energy is $12 \pm \epsilon$ Hz, the subject is looking at the square (fig. 3.9.b). If neither of these are true, we can say that the subject is not looking at any of the stimuli of interest.

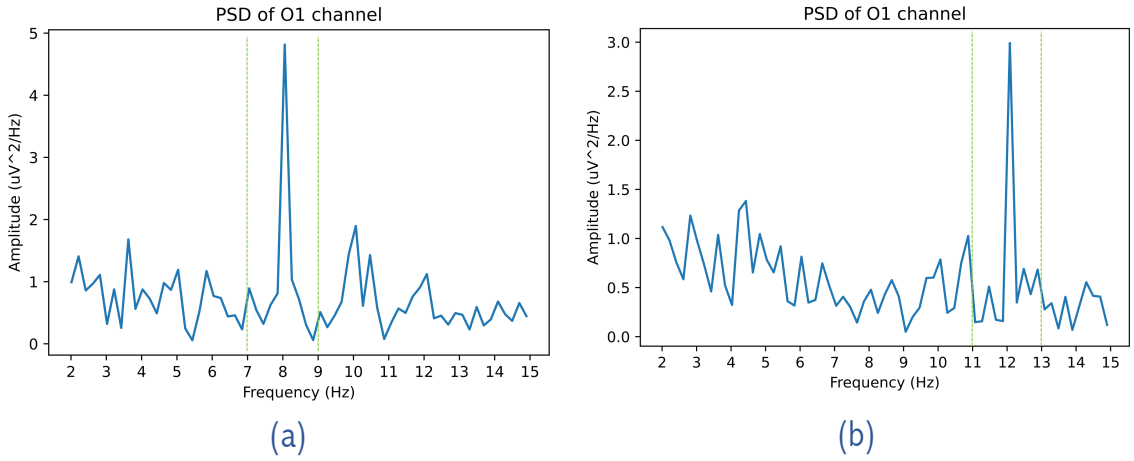


Figure 3.9: PSD of signal extracted from channel O1 (2 seconds chunk), with an $\epsilon = 1$. In (a) is when the subject is looking at the triangle, in (b) is when the subject is looking at the square. The green dotted line indicates the region where the energy peak must be for the signal to be classified as triangle or square

3.4.2 CCA

As explained in the previous chapter (2.6.2) CCA is a technique used to find the strongest linear relationships between two sets of multivariate variables, which allows to identify patterns of neural activity with greater accuracy and stability in noisy environments. In this study, information from the signal and its first harmonic was obtained. For the experiment, the subject was stimulated with frequencies of 8 Hz and 12 Hz (3.2.1). The algorithm used for CCA calculation is shown in the following table:

The algorithm for CCA is run twice: (1) to observe the correlation between signal X (where X is the signal referring to occipital channels (O1 and O2 Fig. 3.2)) and 8 Hz (p_8). (2) to observe the correlation between the signals X and 12 Hz (p_{12}). As an input the algorithm receives the X signal. Finally, if $p_8 > p_12$, we can say that the input signal X corresponds to the subject looking at the triangle (Fig. 3.13.a), whereas if $p_{12} > p_8$, we can say that the subject is looking at the square (Fig. 3.13.b).

Table 3.1: CCA algorithm for feature extraction

Input: (1) the signal epoch $X_t \in \mathbb{R}^{ch \times n}$, where t is the epoch number, where ch is the number of channels used and n is the length of the signal. (2) a set of fundamental frequencies $Y \in \mathbb{R}^{vf \times n}$ where vf is the number of fundamental signals

Output: A value $p \in [0, 1]$, where p explains the similarity between X and Y

i. Compute the mean of X and Y, named $\bar{X}$ and $\bar{Y}$

ii. Compute the centered data matrices,

$$Z_X = X - \bar{X}, Z_Y = Y - \bar{Y}$$

iii. Compute the covariance matrices,

$$S_{xx} = \frac{1}{n} Z'_X Z_X,$$
$$S_{xy} = \frac{1}{n} Z'_X Z_Y,$$
$$S_{yy} = \frac{1}{n} Z'_Y Z_Y$$

iv. Compute the matrix square roots of the inverse covariance matrices,

$$S_{sx} = (\sqrt{S_{xx}})^{-1},$$
$$S_{sy} = (\sqrt{S_{yy}})^{-1}$$

v. Compute the transformed data matrices,

$$A = S_{sx} S_x y S_{sy},$$
$$B = S_{sy} S_x y S_{sx}$$

vi. Compute the 2 largest eigenvalues and eigenvectors of A'A and B'B,

$$\lambda_A, A_\lambda = eig(A' * A, 2),$$
$$\lambda_B, B_\lambda = eig(B' * B, 2)$$

vii. Compute the canonical variates and correlation coefficients,

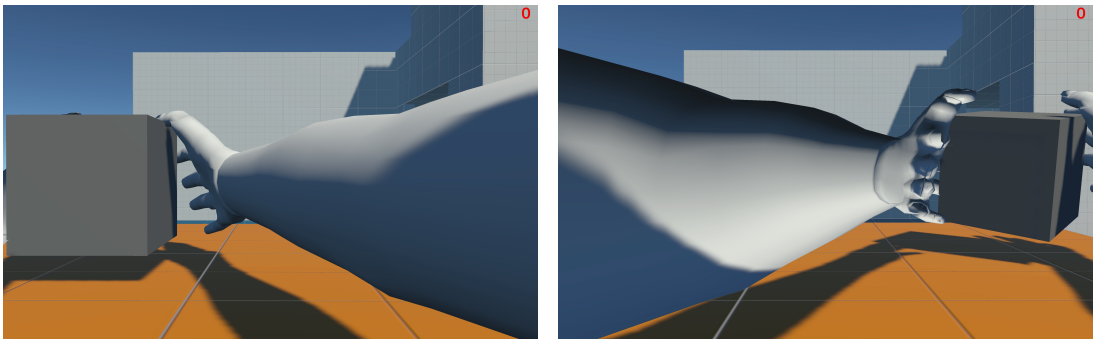
$$p = diag(\sqrt{diag(\lambda_A)^{-1}}) \lambda_A * \sqrt{diag(\lambda_A + \lambda_B)^{-1}} \lambda_B \sqrt{diag(\lambda_B)^{-1}}$$

viii. Return the outputs, p

3.5 Serious game

To simulate the work environment and give feedback of the system control to the subject, a serious game was developed (Fig. 3.10). The serious game is set in an environment which has a table and a box. Its view is in first person and the camera is located at shoulder height, so the subject can see all the joints and movements of the arms. In the middle of the two hands, there is a box which is resting on the table. The objective of the game is that the subject can move the box from their initial position to the final position. To accomplish the task with the box, there are

two movement states: (1) The serious game avatar picks up a box and carries it to its final position (Fig. 3.10.a); (2) The avatar drops the box at the destination and returns to the initial position (Fig. 3.10.b). Upon completion of states (1) and (2) the subject adds a point, which is reflected in a counter located at the top right of the screen. The execution of the serious game and the data acquisition were carried out on different computers via TCP communication between Python and unity.



(a) Initial motion (b) Final motion

Figure 3.10: Serious game view

The development of the serious game was part of a collaboration with the Edmond and Lily Safra International Institute of Neuroscience, Santos Dumont institute, by the student of the graduate program in neuroengineering André F. Oliveira under the supervision of PhD. professor Denis Rodriguez. This game was designed in unity software.

3.6 Experimental assessment

An experimental protocol was designed to test all the features of the system developed during this project. For this, two main parts are proposed: (1) offline phase, where the subject is told which stimulus to look at and, (2) online phase, where a serious game is used to give feedback. In this case, the subject is free to look at the stimulus he/she wants to complete a determined score.

3.6.1 Participants

The study conducted involved a group of eight individuals (five males and three females) with an average age of 22 $\pm$ 4 years. The selection criteria for these participants were chosen based on the parameters presented in a previous study [70]

to optimize the performance of the system being studied. Additionally, all participants were required to have no prior neurological history of epilepsy. None of the participants had prior experience with BCI or even the use of EEG.

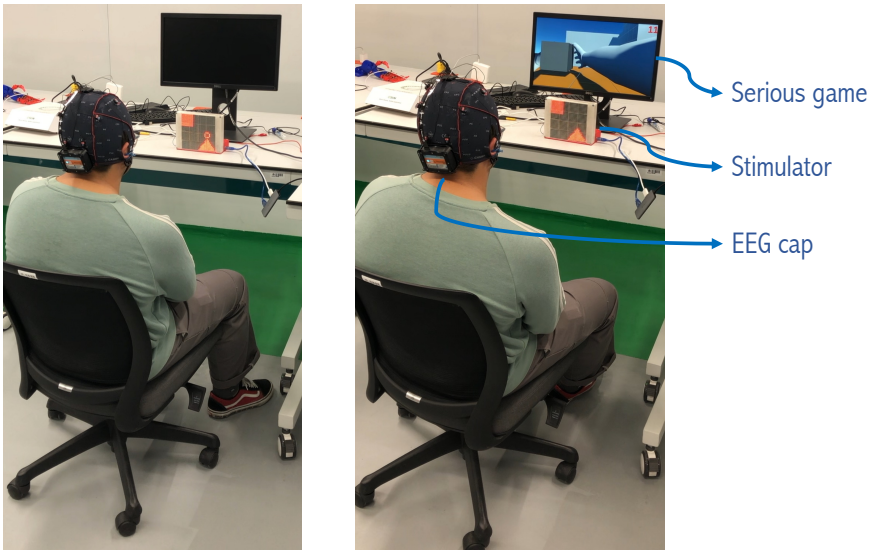
To exclude potential confounding factors, individuals with epilepsy, history of frequent migraine episodes, visual disorders, use of contact lenses, neurological history, intoxication, or medication were excluded from the study. Furthermore, all participants were screened to ensure that they possessed full visual, cognitive, and auditory abilities to follow instructions and were capable of providing informed consent before data acquisition. Informed consent was obtained from all participants prior to data acquisition to ensure that they were fully aware of the study's purpose, procedures, and potential risks.

3.6.2 Experimental setup

The experiment was conducted in a spacious, comfortable, and quiet laboratory with the absence of electrical devices that could compromise the EEG signals acquired. Three main phases were conducted during data collection: (1) explanation of the experiment, (2) preparation of the electrodes, and (3) execution of the experimental protocol. The subjects were seated in a chair, in front of them was the stimulator at the bottom and a monitor at the top. The monitor was used to give feedback by the serious game; therefore, this was off for the offline part of the experiment and on for the online part. An example of this setup is shown in Figure 3.11.

After verification of the inclusion criteria by the participants, the explanation of the experiment for the offline and online phase was carried out. In this point, a demo was executed in the stimulator which showed the stimuli and indications that were going to appear throughout the session. Likewise, some considerations for the participant to take into account at the moment of executing the experiment are commented: the participant is asked to be relaxed at all times, to try to blink as little as possible and to minimize the amount of muscular movements, i.e., try not to move arms, fingers or feet. Finally, as each stimulus consists of approximately 20 LEDs representing a figure (triangle and square), the subject is asked to try to concentrate on the figure rather than on one or a few LEDs, in order to find higher amplitude values related to SSVEP.

Simultaneously, the participant was prepared for the placement of the electrodes. The forehead is cleaned with cotton soaked in alcohol to remove grease that can induce noise in the measurements of prefrontal channels. Also, conductive gel is applied to each of the electrodes to reduce the impedance. Thus, it is attempted to



(a) Offline phase (b) Online phase

Figure 3.11: Experimental setup

maintain the impedance value less than 30 $k\Omega$. Once all the electrodes meet these requirements, we proceed with the execution of the offline phase and the online phase.

Offline phase

For this phase, the presentation of stimuli with indication is made. In this sense, a supervised method where the subject is told which stimulus to looking at. This way, the signals obtained with the indication label are grouped, in order to obtain metrics for the evaluation of the methods applied for signal classification. To carry out the data acquisition in the offline part, the stimulator is programmed in SSVEP offline mode, where in the first 60 seconds a sequence of dots appears (Fig. 3.12), indicating the remaining time to start the experiment. During this time, a verification of the signal is made, the subject is asked to blink, tense the muscles of the mouth and close his eyes to looking at whether there is an increase in amplitude in the Delta and Theta waves.

Once these initial 60 seconds have elapsed, the appearance of stimuli begins, randomly indicating which figure the subject should look at (Fig. 3.13). The probabilities for the appearance of each stimulus are 50% each, in order to have a balanced sample at the end of the experiment. The stimulus indication has a duration of 0.5 seconds and a new indication is shown every 5 seconds (these 5 seconds presentation are known as an epoch). This process is repeated for 60 epochs. Once the stimulus

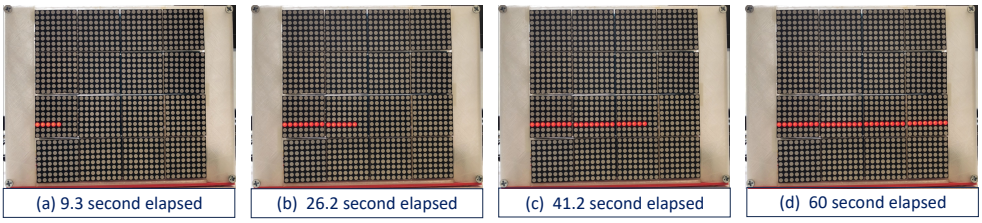


Figure 3.12: Initial 60 second sequence for experiment setup

presentation is finished, a "thank you" screen is displayed (see Fig. 3.13).c) to let the participant know that the first part of the experiment is over. Lastly, there is an approximate duration of 6:30 minutes for the offline phase of the data collection.

Finally, preparation for session 2 was performed, where the online mode was changed on the stimulator, and the screen was enabled for the playback of the serious game. Simultaneously, the participant is asked to relax his vision and be ready for the next data acquisition.

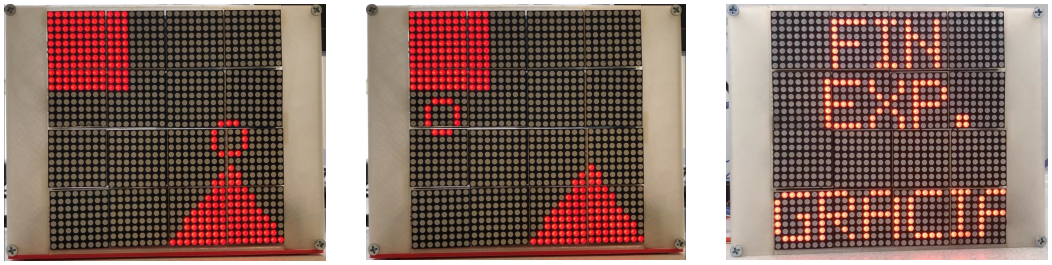


Figure 3.13: (a, b) Indication of the figure on which the participant should concentrate his attention. (c) thank you screen displayed at the end of the experiment

Online phase

In the work environment, the employee is assigned a task and must complete it, no matter what methodology is used, no matter how long it takes to complete the objective. Now, the stimulator did not show any indication, and the participant was free to control the serious game at will. The control mode of the serious game is explained to the participant. He/She has to move the box from its initial position to its final position. The participant must look at the triangle (Fig. 3.14 top). Similarly, to leave the box in its final position and return to the initial position for another box, the participant must look at the square (Fig. 3.14). Once this sequence is completed, one point is added to the counter.

The goal of this session is for the participant to complete 40 points. After that,

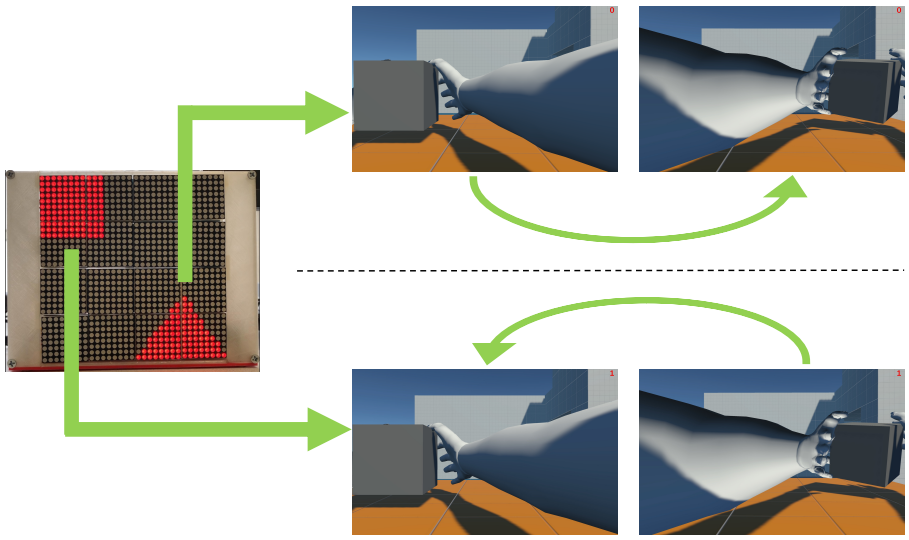


Figure 3.14: Serious game control mode. At the top you can see the movement of the hands when the participant is looking at the triangle. At the bottom is the movement of the hands when the participant is looking at the square. Finally, note that at the bottom a point is added to the counter due to the completion of the sequence

the participant is asked to repeat this task of getting 40 points three times, trying in each attempt to do it faster and with more control. A 2-minute rest period was given each time the participant completed the 40-point task.

Chapter 4

Results

This chapter presents the results obtained in this work. Also the chapter 4 is divided into two main parts. In the first part, the data collected in the offline phase were used to evaluate the effectiveness of the system using two feature extraction methods: PSD and CCA. ACC, TPR, FPR and kappa score were the metrics used to determine which of the feature extraction methods provides a more balanced classification of the data produced by our system. Simultaneously, the ITR produced by the BCI is calculated. In addition, a comparison between signals preprocessed with an eye blink reduction algorithm and signals without eye blink reduction was made. In the second part, the online phase of the experiment was studied. Here the usability of the system was analyzed by means of a serious game.

4.1 Preliminary data analysis

As explained in the methodology section, the participant was visually stimulated to produce SSVEP signals. A raw signal, which contains the necessary characteristics to classify the signal into square or triangle, was received by EEG. Thus, it is known on which stimulus the participant was focusing his/her attention. To achieve this classification objective, the raw signal (see Fig. 4.1) was subjected to filtering and characterization stages.

This way, a Butterworth temporal filter of order three was applied. This filter allowed frequencies between 0.1 and 30 Hz to pass through. In addition, a correction was implemented on the horizontal axis to eliminate the phase shift to allow the signal to return to the zero level (see Fig. 4.2.a). Besides, equation 2.2 was applied to the signal in order to perform a CAR type spatial filtering (Fig. 4.2.b). In this sense, it was completed the filtering stage.

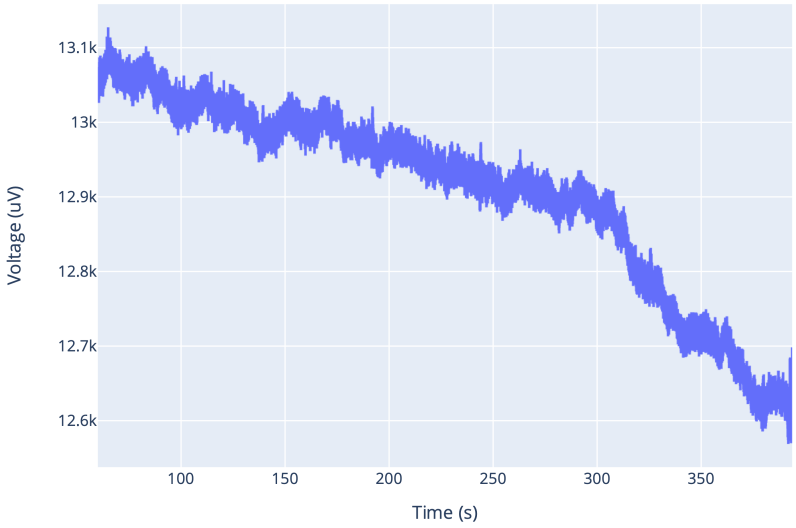
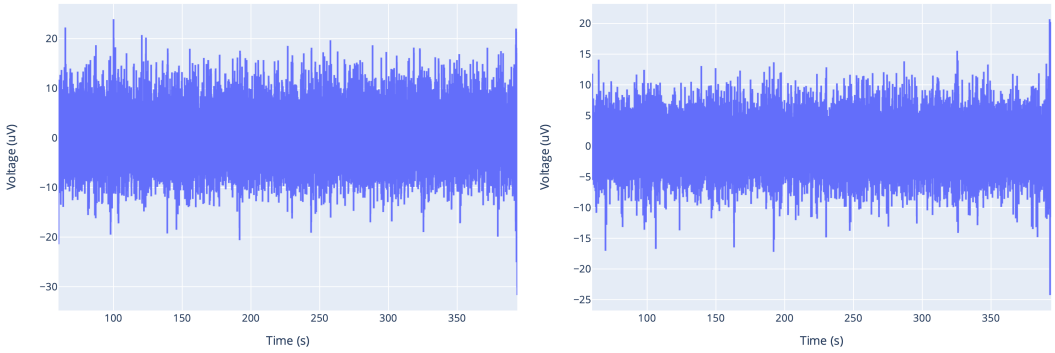


Figure 4.1: Raw signal of the offline phase of O1 channel obtained from participant 1

Once the signal was filtered, it was verified if the participant was able to visually segment the stimulus on which he/she was concentrating, since the triangle and the square were only a few centimeters away. When the participant was concentrating on a figure, he/she was able to see the other stimulus with his peripheral vision. Therefore, before starting with the classification of the signal, an average of the signals was made. This manner, all the epochs where the participant was looking at the triangle were averaged. The same process was performed with the epochs where the participant was looking at the square. The average of the epochs for participant 1 can be seen in figure 4.3. Note that there is a visual difference between the frequencies of the signals, as shown in Fig. 4.3.a and 4.3.b, where the frequency of the signal average of the triangle appears slightly slower than the frequency of the signal average of the square. This visual observation can be confirmed by the PSD in both averaged signals. The results are shown in figure 4.4.

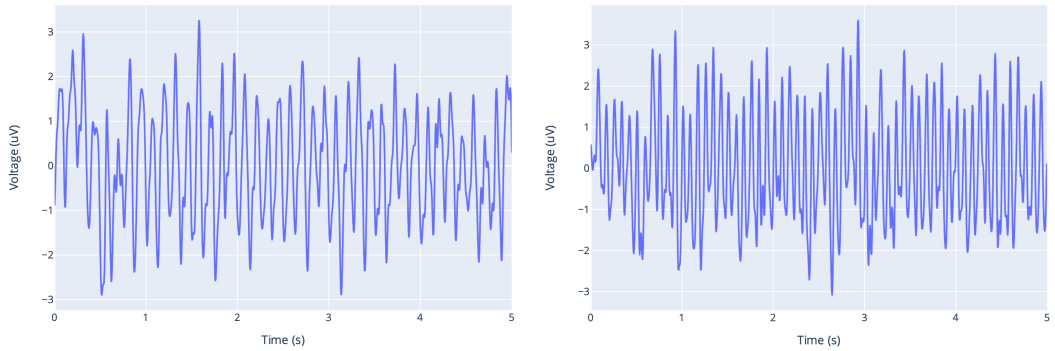
For the average of the epochs referring to the triangle the PSD showed a maximum amplitude at 8 Hz (see Fig. 4.4.a), followed by a lower-amplitude peak at 12 Hz. A significant concentration of energy was found at 16 Hz, which corresponds to the first harmonic of the frequency at which the triangle oscillates. Therefore, in the case of the triangle, it can be concluded that the participant is able to partially segment this geometric figure with respect to the square because the first harmonic has a lower amplitude than the amplitude at which the square is oscillating. On

the other hand, for the average PSD at the epochs of the square, it can be deduce that the participant is able to completely segment the square figure because the two peaks with the highest energy concentration are found at 12 and 24 Hz (see Fig. 4.4.b).



(a) Pass band filtered signal (b) CAR filtered signal

Figure 4.2: Filtered signal of the offline phase of O1 channel obtained from participant 1. (a) shows the temporal filtered signal and the baseline correction. In (b) the spatial corrected signal by CAR filter

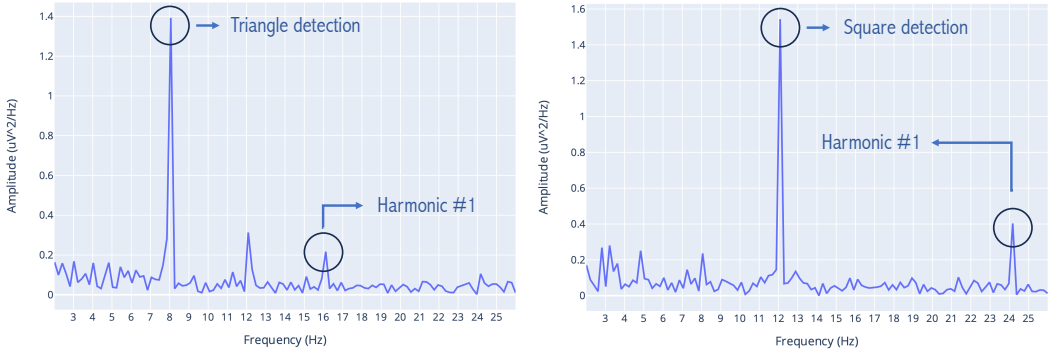


(a) Signal average of triangle epochs (a) Signal average of square epochs

Figure 4.3: Signal average of the epochs grouped by triangle and square of participant 1. In (a) the average of triangle epochs. In (b) the average of square epochs

A classification thresholds were identified for the use of CCA algorithm. that means, the values for which the correlations indicated that the signal belonged to the triangle class or to the square class (Fig. 4.5). Using the initial 60 seconds of data acquisition, it can be seen how the correlations with both signals (p_8 and p_12) are less than 10% (green dots in Figure 4.5). Therefore, in the case that p_8 and p_12 are less than 0.10, the piece of the signal will not be taken into account, since it possibly contains information coming from noise or inactivity by the participant. Finally, the delimitation between the class to which the signal belongs (triangle or

square) are clear, so that, a classification of the SSVEP signal obtained using a method such as CCA is feasible.



(a) PSD of the signal average on triangle epochs

(b) PSD of the signal average on square epochs

Figure 4.4: PSD of the averaged signals. In (a) PSD of triangle epochs. In (b) PSD of square epochs

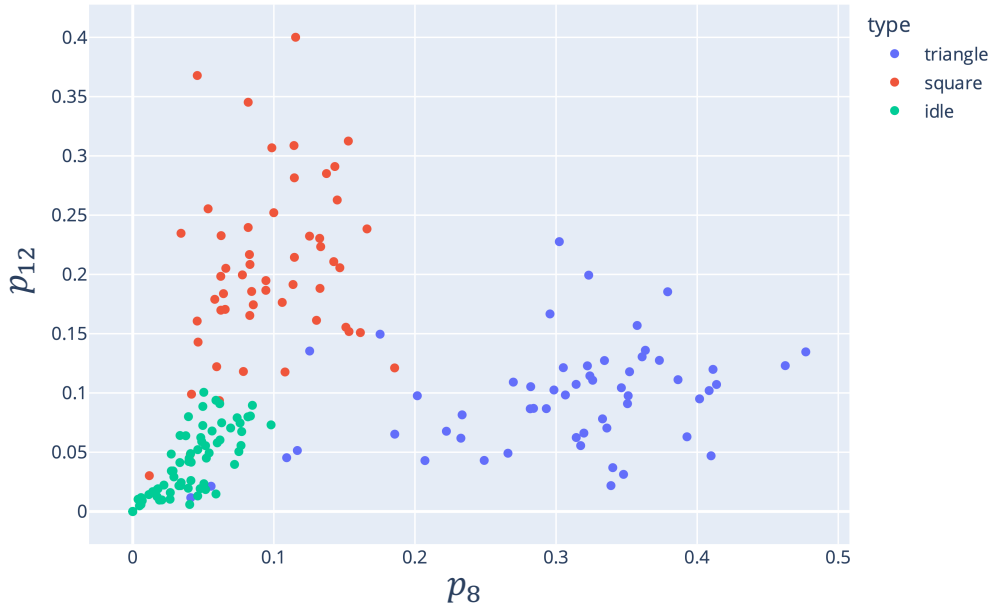


Figure 4.5: Correlation coefficients using CCA algorithm

4.2 Performance during offline phase

Different time windows were used to be applied to the feature extraction and classification algorithms. This is necessary because a short time window does not store a significant amount of information that can generalize the participant's intention

so the classification has low accuracy rates and makes the participant's experience of using the system poor. Otherwise, a long time window gives good classification results, but as it requires more data, it takes longer to give a classification result. Therefore, a balance between accuracy and response time is required. To find the value of the time window, the data obtained in the offline phase of the experiment is used.

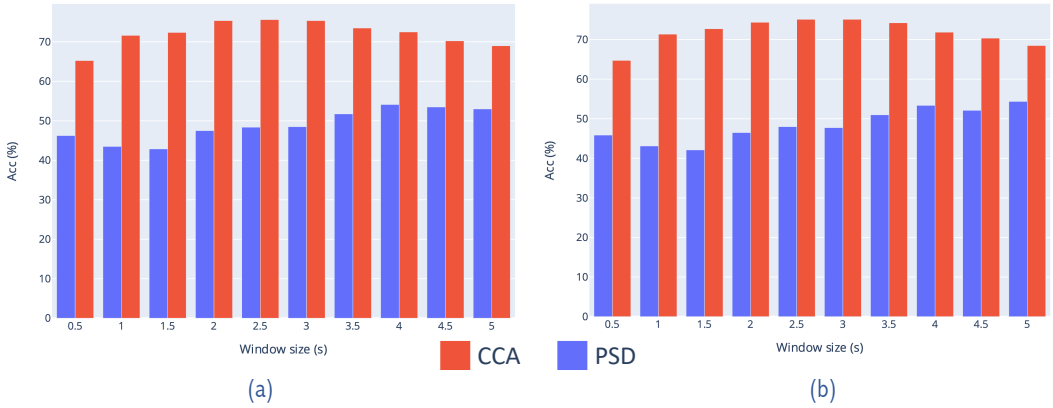


Figure 4.6: Average system accuracy using different time windows. In (a) the preprocessing without the eye blink subtraction algorithm. In (b) preprocessing with eye blink subtraction stage

Different time windows were evaluated, starting from 0.5 seconds to 5 seconds, making 0.5 second jumps. In all windows, an overlap of 50% was used. The results for preprocessing without eye blink removal are shown in figure 4.6.a. For the CCA case (Table 4.1), the average system accuracy increases as the time window grows, with a maximum average accuracy for a window size of 2.5 seconds. After this, the system accuracy starts to decrease. This is due to the design of the experiment, since the instruction for the offline phase was to look at the indicated figure for periods of 5 s. Therefore, as the time window is so long, it will constantly contain information about signals resulting from looking at the triangle and the square. However, for the case of PSD, the accuracy had its highest score for the 5-second time window (Table 4.2). Finally, in all cases, CCA performed better when classifying SSVEP signals. In fact, better values are obtained using a time window of 0.5 seconds in CCA than using a time window of 5 seconds in PSD.

Similarly, the evaluation process was performed with different time windows using the eye blink reduction algorithm shown in section 3.3. The results obtained are shown in figure 4.6.b. For CCA and PSD, the same distribution of results was obtained without the application of the eye blink reduction algorithm. In the case of CCA (table 4.3), a decrease of $0.23 \pm 0.5\%$ was obtained, and in the case of PSD

(table 4.4), there was a decrease in precision of 0.50 ± 0.7%.

Window size (s)	Acc (%)
0.5	65.250
1	71.625
1.5	72.375
2	75.375
2.5	75.625
3	75.375
3.5	73.500
4	72.500
4.5	70.250
5	69.000

Table 4.1: Accuracy of CCA using different time window sizes

Window size (s)	Acc (%)
0.5	46.250
1	43.500
1.5	42.875
2	47.500
2.5	48.375
3	48.500
3.5	51.750
4	54.125
4.5	53.500
5	53.000

Table 4.2: Accuracy of PSD using different time window sizes

Note that regardless of the preprocessing stage (with or without flicker removal) for CCA, the largest percentage increase in accuracy is between the 0.5 second window and the 1 s window, with an increase of 6.37%. However, the percentage difference in accuracy between 1 second and 2.5 seconds (where the maximum is found) is 4%. Thus, the 1-second window was chosen to be used throughout the offline and online stages of the experiment, due to the project is oriented to be use in a work environment, so, the response of the system is sought to be fast in order to be able to handle a large number of iterations per minute. Besides, the one second window offers that point where performance and speed meet. Also, this time span offers the possibility of activation of motor intention patterns, doing that the data taken could be used in motor intention BCI study.

This way, we proceeded to analyze the metrics obtained on average and for each participant who performed the experiment. In the case of 1 second window, the results of accuracy per participant without the use of the eye blink removal algorithm can be seen in Figure 4.7.a. For the case of classification using PSD, the following results were obtained: ACC of 43.5 ± 19.9% , Kappa of 0.07 ± 0.2%, TPR of 63.2 ± 0.6%, and FPR of 35.2 ± 1.3%. In addition, CCA performed better in extracting and classifying SSVEP signals, obtaining the following results: ACC of 71.6 ± 9.7%, Kappa of 0.43 ± 0.19%, TPR of 59.5 ± 0.05% and FPR of 39.3 ± 1.35%. The best performance was achieved by the participants S1 and S8 with

accuracy of 83%.

Window size (s)	Acc (%)
0.5	64.750
1	71.375
1.5	72.750
2	74.375
2.5	75.125
3	75.125
3.5	74.250
4	71.875
4.5	70.375
5	68.500

Table 4.3: Accuracy of CCA using different time window sizes with eye blink removal algorithm

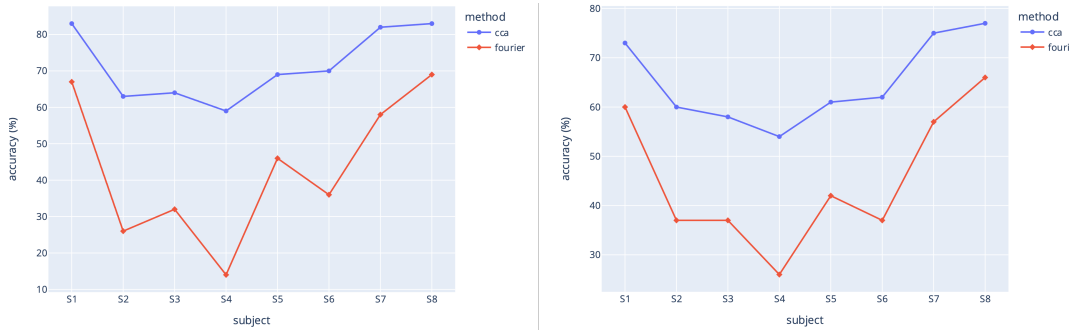
Window size (s)	Acc (%)
0.5	45.875
1	43.125
1.5	42.125
2	46.500
2.5	48.000
3	47.750
3.5	51.000
4	53.375
4.5	52.125
5	54.375

Table 4.4: Accuracy of PSD using different time window sizes with eye blink removal algorithm

Using the eye blink removal algorithm the results obtained were slightly lower (see Fig. 4.7.b), where we can say that for CCA it reduced the precision between 2 and 4 %. However, for PSD, in the participants with worse performance it increased the precision between 1 and 3 %, although in others it lost precision. On average, the results for the PSD were ACC of 45.7 $\pm$ 13.9% , Kappa of 0.018 $\pm$ 0.19%, TPR of 63.3 $\pm$ 0.58%, and FPR of 47.2 $\pm$ 1.1%. Furthermore, an average of evaluating for CCA were: ACC of 65.0 $\pm$ 8.8%, Kappa of 0.29 $\pm$ 0.16%, TPR of 61.8 $\pm$ 0.5%, and FPR of 49.0 $\pm$ 1.1%. A summary of the average results aforementioned can be found in table 4.5 where PSD_1 and CCA_1 represent the respective feature extraction techniques without eye blink removal algorithms. Lastly, PSD_2 and CCA_2 are the results with the eye blink removal algorithm.

Therefore, it can be seen that the best results were obtained using CCA as a method of feature extraction and classification (see figure 4.8 of the confusion matrix obtained). In terms of accuracy, it can be said that the results obtained for all participants are better than those obtained using a base model. The kappa score indicates that there is a moderate level of agreement between the predictions and the true labels, that means, that the model is classifying thanks to the features extracted from the signal and not by chance. With respect to the result obtained in the TPR, it can be said that the algorithm is able to correctly identify the positive

instances (detect triangle) at a consistent rate. Finally, the FPR value indicates that the algorithm incorrectly classifies negative instances (detect square) as positive (approximately 1 in 3).



(a) System accuracy without eye blink removal

(b) System accuracy with eye blink removal

Figure 4.7: Accuracy achieved by the system with both preprocessing stages

Table 4.5: Average system performance

Method	Accuracy	Kappa	TPR	FPR
PSD_1	43.5%	0.07	63.2%	35.2%
CCA_1	71.6%	0.43	59.5%	39.3%
PSD_2	45.2%	-0.01	63.3%	47.2%
CCA_2	65%	0.29	61.8%	49%

Finally, the delay of the system in giving an answer was calculated, and the difference between the moment when the stimulus was presented and the moment when the BCI gave the correct classification answer was analyzed. On average, the system took 0.77 $\pm$ 0.39 s to return the participant's intention. In figure 4.9, it is observed how the distribution of the system latency was given, having a higher number of occurrences for responses in 0.5 s. In second place, having a little less than half the system took 1.0 s to respond. Times longer than 1.5 s and 2.0 s had a

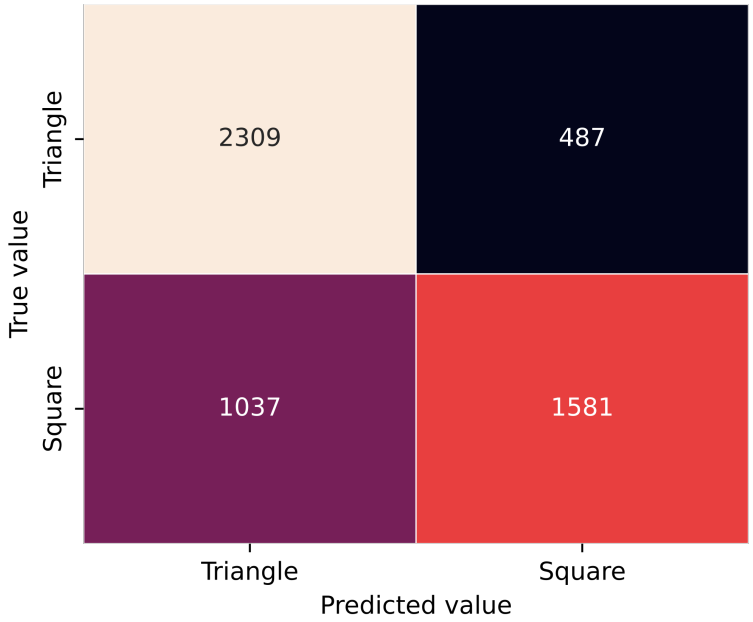


Figure 4.8: System response confusion matrix

minimal occurrence, where it can be said that the system took too long to respond, and therefore, this was considered a failure. Using the system's precision and delay information, the BCI for the offline phase has an ITR of 37.6 ± 15.4 bits/min. This information serves as an estimation value for the online phase of the system, as it is possible to find an expected value (in seconds) for how long it will take the participant to achieve the goal of completing 40 points using the serious game and thus, know if the task is being carried out correctly.

4.3 Performance during online phase

As explained in section 3.6, three trials were carried out for data collection in the online phase: for the eight participants, the task of completing the 40 points took an average time of 110.60 ± 21.5 s for attempt 1, 110.43 ± 28.5 s for attempt 2, and 119.50 ± 24.98 s for attempt 3 (Fig. A.2). For the three trials, there was a similar performance among participants. No significant change was found between trials. Some of them managed to decrease the time as the trials passed, but others did not. Participant 4 was the best performer, completing the task in less than 90.0 s in all his attempts. However, he stated that he did not have much control of the system. If we take into account that he was the participant with the lowest accuracy in the offline phase, we could say that the false positives and negatives (see Fig. 2.9) generated by the noise made the system not have much control and therefore made

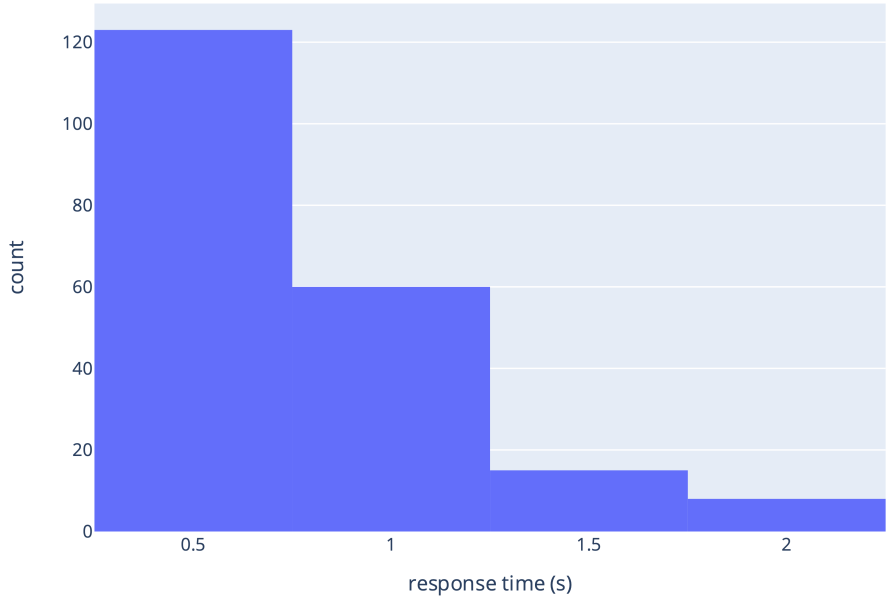


Figure 4.9: System response latency

a large number of points in a short time.

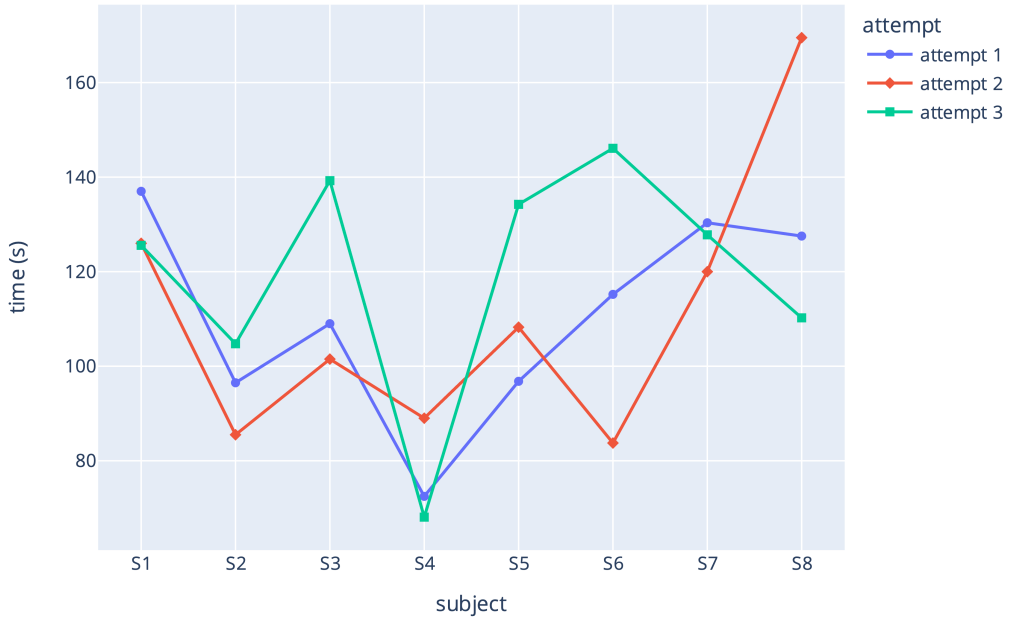


Figure 4.10: Time taken by Participants to complete the task through serious game play

Participants 1 and 8 reported having good control of the system and a response in accordance with their intentions, completing the task in an average time of 129.52 ± 6.4 and 135.52 ± 30.4, respectively. Therefore, for participants who presented high accuracy in the offline phase, the average time per operation was approximately

1.61 s. Based on the latency result presented in figure 4.9, it can be said that the participant maintained the instruction for a while even if it was already carried out, that is, if the instruction given by the participant was to move to the initial position, once this point was reached, the participant continued to follow this instruction. This in order to check if he had complete control of the system or it was just a "coincidence". Finally, for the online phase the system has an ITR of 15.2 ± 8.9 bits/min.

4.4 Discussion

This section will discuss the most relevant results of the study. To do this, we will first address the issues of comparison between CCA and PSD. Subsequently, the scores obtained about the performance of the characterization and classification algorithms will be analyzed. Finally, the performance of the developed SSVEP-BCI system will be discussed, by looking at the ITR values obtained.

A comparison between CCA and PSD as feature extraction and classification algorithms was performed, and it was found that CCA performed better in all participants analyzed, with up to 20 percentage points above PSD. These results corroborate those obtained by Wey et al. [71], where the accuracy of CCA was 10% higher than that obtained using PSD, which agrees with the results of this study and is consistent with the conclusions found in the literature. Therefore, for the use of SSVEP-BCI where an environment with a considerable amount of noise is involved, using CCA over PSD can substantially improve the performance obtained by the system.

Likewise, the average levels of accuracy using short time windows obtained in the offline part of the experiment had slightly better results than those obtained using CCA for external device control [72]. However, in the case where the time window is increased, the system developed in this study does not have such a significant increase as shown in other studies [26], where for a step from 0.5 seconds to 1 second there is an increase in accuracy of more than 30%.

However, the results obtained by [71] and [26] show a completely positive correlation between window size and accuracy results. With the best accuracy results for the highest value time window. This correlation fact is at disagreement with the findings obtained throughout this work, it could have happened since the participants were instructed to look for a period of time at the stimulus. Unlike other studies found in the literature, where the participant had no time limit for looking at

the stimulus, the system responded once it was certain of the correct classification.

Finally, the index describing the efficiency of the system (ITR) has values below those obtained in previous studies (see related works 2.8). However, this has to do with factors related to the number of commands. For instance, [56] obtained an average ITR of 48.2 bits/min with an upper limb exoskeleton that had 15 different functions, however, the response time of the system was delayed despite to the number of commands it had, it was possible to obtain an ITR with a high value. Similarly, [57] used 5 commands for an average ITR of 32.9 bits/min, but a window of 4 seconds. Therefore, in this study, some accuracy points are sacrificed for a more efficient system response. This was reflected in the developed ITR scores obtained in the SSVEP-BCI.

Chapter 5

Conclusion

A BCI to generate the inputs to move an exoskeleton in a work environment simulated by a serious game was developed. Likewise, to ensure independence in the communication between the serious game and the BCI, a TCP protocol was implemented to send information between two computers (one with data acquisition and signal classification and the other with the execution of the serious game). Besides, motion control was generated through a visual stimulator developed with Arduino to manipulated a led light arrays that triggered SSVEP signals that could be classified by feature extraction using methods such as CCA and PSD.

It was possible to deliver a stimulator that has a fairly high versatility when presenting visual stimuli and is portable and compact to be used without problems in the work environment. Additionally, it is easy to program for the execution of experiments involving SSVEP paradigm. Finally, it allows the use of several stimulators in the workspace as it is low cost.

In addition, to test the developed systems, an experimental protocol was proposed taking into account the considerations found in the literature. The data collection was divided into two phases: (1) offline and (2) online. (1) Allowed to know the system evaluation metrics, in order to know precisely how good the classification was delivered by the different methods evaluated (71.6% for CCA and 43.5% for PSD). (2) Provided information about how time delayed the response of the developed BCI and provided feedback from the subject about the usability when controlling the serious game (ITR of 15.2 $\pm$ 8.9 bits/min).

Subsequently, the system was tested with eight healthy subjects, who were specifically selected to meet the characteristics found in the literature with respect to the generation of SSVEP signals. This selection was performed to maximize the prediction results made by the system. In addition, all participants were able to

complete the experimental protocol, fulfilling each of the phases proposed in the methodology.

Finally, two feature extraction techniques, PSD and CCA, were compared. These methods were tested using SSVEP signals with and without eye blink information. In both cases, CCA was the best performing method, having a significantly higher accuracy than PSD. The high level of accuracy when using the system in some subjects and the short time window used allowed the BCI to achieve high levels of ITR, managing to translate a high number of commands per minute.

5.1 Future works

As future works we intend to develop two systems. First, we seek to use the system developed with SSVEP signal classification and add Motor Intention detection. Thus converting the system into a hybrid BCI. In this sense, in the long term of use of the system, the system will be able to unconsciously produce Motor Intention patterns, which could be detected by Machine Learning algorithms. This way, the system would give a response based on motor intention patterns, and it would not be necessary for the subject to have to fix his/her attention to a stimulus (triangle or square), so that the control of the serious game does not generate fatigue and visual fatigue from the stimulator.

Second, it is hypothesized that the subject makes some kind of association with the geometric figure when he/she wants to do some movement control, so it is believed that there may be activation of the evoked potential P300. Thus, it is intended to add this paradigm to the hybrid implementation, so it will be countered with a BCI that classifies based on three outcomes: P300, SSVEP, Motor Intention.

Finally, this is intended to be performed in a semi-supervised way, that is, the system developed at this time with SSVEP would determine whether the input signal is of a triangle or a square category, and thus label that piece of signal. This means that no explicit training is needed from the subject, but rather, training is done unconsciously to the subject and retraining is performed periodically as significant new data arrives.

Bibliography

[1] European Commission. *Employment, Social Affairs & Inclusion*. Website. 2023. URL: https://ec.europa.eu/social/main.jsp?catId=1134&langId=en.

[2] AK Dasgupta et al. "The performance of the ICEROSS prostheses amongst transtibial amputees with a special reference to the workplaceâa preliminary study". In: *Occupational medicine* 47.4 (1997), pp. 228–236.

[3] Robert Bogue. "Exoskeletons and robotic prosthetics: a review of recent developments". In: *Industrial Robot* 36.5 (2009), pp. 421–427. DOI: 10.1108/01439910910980141.

[4] Kazuo Kiguchi, Takakazu Tanaka, and Toshio Fukuda. "Neuro-fuzzy control of a robotic exoskeleton with EMG signals". In: *IEEE Transactions on fuzzy systems* 12.4 (2004), pp. 481–490.

[5] Kazuo Kiguchi et al. "Development of a 3DOF mobile exoskeleton robot for human upper-limb motion assist". In: *Robotics and Autonomous systems* 56.8 (2008), pp. 678–691.

[6] Mark F Bear, Barry W Connors, and Michael A Paradiso. *Neuroscience: Exploring the Brain*. Lippincott Williams & Wilkins, 2007.

[7] Silvija Angelova et al. "Power frequency spectrum analysis of surface EMG signals of upper limb muscles during elbow flexion–A comparison between healthy subjects and stroke survivors". In: *Journal of Electromyography and Kinesiology* 38 (2018), pp. 7–16.

[8] Institute of Entrepreneurship Development. *Gender Equality in Physically Demanding Occupations*. Project update on IED. May 2021. URL: https://ied.eu/project-updates/gender-equality-in-physically-demanding-occupations/.

[9] Jonathan R Wolpaw et al. "Brain-computer interface technology: a review of the first international meeting". In: *IEEE transactions on rehabilitation engineering* 8.2 (2000), pp. 164–173.

[10] Luis Fernando Nicolas-Alonso and Jaime Gomez-Gil. "Brain computer interfaces, a review". In: *sensors* 12.2 (2012), pp. 1211–1279.

[11] Swati Vaid, Preeti Singh, and Chamandeep Kaur. "EEG signal analysis for BCI interface: A review". In: *2015 fifth international conference on advanced computing & communication technologies*. IEEE. 2015, pp. 143–147.

[12] Walter S Pritchard. "Psychophysiology of P300." In: *Psychological bulletin* 89.3 (1981), p. 506.

[13] Erwei Yin et al. "A novel hybrid BCI speller based on the incorporation of SSVEP into the P300 paradigm". In: *Journal of neural engineering* 10.2 (2013), p. 026012.

[14] Martin Lotze and Ulrike Halsband. "Motor imagery". In: *Journal of Physiology-paris* 99.4-6 (2006), pp. 386–395.

[15] Jennifer A Stevens and Mary Ellen Phillips Stoykov. "Using motor imagery in the rehabilitation of hemiparesis". In: *Archives of physical medicine and rehabilitation* 84.7 (2003), pp. 1090–1092.

[16] Aymeric Guillot and Christian Collet. "Construction of the motor imagery integrative model in sport: a review and theoretical investigation of motor imagery use". In: *International Review of Sport and Exercise Psychology* 1.1 (2008), pp. 31–44.

[17] Christoph Guger et al. "How many people could use an SSVEP BCI?" In: *Frontiers in neuroscience* 6 (2012), p. 169.

[18] Danhua Zhu et al. "A survey of stimulation methods used in SSVEP-based BCIs". In: *Computational intelligence and neuroscience* 2010 (2010), pp. 1–12.

[19] Marcin Kołodziej et al. "Comparison of EEG signal preprocessing methods for SSVEP recognition". In: *2016 39th International Conference on Telecommunications and Signal Processing (TSP)*. IEEE. 2016, pp. 340–345.

[20] Richard MG Tello et al. "A comparison of techniques and technologies for SSVEP classification". In: *5th ISSNIP-IEEE Biosignals and Biorobotics Conference (2014): Biosignals and Robotics for Better and Safer Living (BRC)*. IEEE. 2014, pp. 1–6.

[21] Eric Kandel et al. *Principles of Neural Science*. McGraw-Hill Education, 2013.

[22] Steven M Chrysafides, Stephen J Bordes, and Sandeep Sharma. "Physiology, Resting Potential". In: *StatPearls* (2022).

[23] Max O Krucoff et al. “Enhancing nervous system recovery through neurobiologics, neural interface training, and neurorehabilitation”. In: *Frontiers in neuroscience* 10 (2016), p. 584.

[24] Priyanka A. Abhang, Bharti W. Gawali, and Suresh C. Mehrotra. *Introduction to EEG- and Speech-Based Emotion Recognition*. Academic Press, 2016.

[25] Rajesh P. N. Rao. *Brain-Computer Interfacing: An Introduction*. Cambridge University Press, 2013.

[26] Hamilton Rivera-Flor et al. “CCA-Based Compressive Sensing for SSVEP-Based Brain-Computer Interfaces to Command a Robotic Wheelchair”. In: *IEEE Transactions on Instrumentation and Measurement* 71 (2022), pp. 1–10.

[27] *Brain Surgery: Treatment & Recovery*. `https://my.clevelandclinic.org/health/treatments/16802-brain-surgery`.

[28] Hermann Hinrichs et al. “Comparison between a wireless dry electrode EEG system with a conventional wired wet electrode EEG system for clinical applications”. In: *Scientific reports* 10.1 (2020), pp. 1–14.

[29] Michal Teplan et al. “Fundamentals of EEG measurement”. In: *Measurement science review* 2.2 (2002), pp. 1–11.

[30] DaniÃ«l Lacko. “The application of 3D anthropometry for the development of headgear - A case study on ergonomic brain-computer interfaces”. PhD thesis. Mar. 2017.

[31] Patrick J. Lynch. *21 electrodes of International 10-20 system for EEG*. `https://commons.wikimedia.org/wiki/File:21_electrodes_of_International_10-20_system_for_EEG.svg`. [Accessed: April 19, 2023]. 2005.

[32] David D Cox and Robert L Savoy. “Functional magnetic resonance imaging (fMRI)âbrain readingâ: detecting and classifying distributed patterns of fMRI activity in human visual cortex”. In: *Neuroimage* 19.2 (2003), pp. 261–270.

[33] *How to decide whether to use EEG, MEG or fMRI*. `https://www.researchgate.net/post/How-to-decide-whether-to-use-EEG-MEG-or-fMRI`.

[34] CJ Stam. “Use of magnetoencephalography (MEG) to study functional brain networks in neurodegenerative disorders”. In: *Journal of the neurological sciences* 289.1-2 (2010), pp. 128–134.

[35] Raphael Vallat. *Computing EEG power spectra*. `https://raphaelvallat.com/bandpower.html`. Accessed on 2023-04-25. 2021.

[36] Eda Akman Aydin, Ãmer Faruk Bay, and Ä°nan GÃ¼ler. "Comparision of P300 based synchronous and asynchronous Brain Computer Interface". In: *2017 Medical Technologies National Congress (TIPTEKNO)*. 2017, pp. 1–4. DOI: 10.1109/TIPTEKNO.2017.8238100.

[37] Dean F Salisbury et al. "First-episode schizophrenic psychosis differs from first-episode affective psychosis and controls in P300 amplitude over left temporal lobe". In: *Archives of general psychiatry* 55.2 (1998), pp. 173–180.

[38] Axel Mecklinger and Peter Ullsperger. "The P300 to novel and target events: a spatio–temporal dipole model analysis". In: *Neuroreport* 7.1 (1995), pp. 241–245.

[39] Abeer Selim, Manal Wahed, and Yasser Kadah. "Machine learning methodologies in brain-computer interface systems". In: *CIBEC2008 Biomed. Eng.* (Jan. 2008), pp. 1–5.

[40] Reza Fazel-Rezai et al. "P300 brain computer interface: current challenges and emerging trends". In: *Frontiers in neuroengineering* (2012), p. 14.

[41] François-Benoît Vialatte et al. "Steady-state visually evoked potentials: focus on essential paradigms and future perspectives". In: *Progress in neurobiology* 90.4 (2010), pp. 418–438.

[42] Xiaogang Chen et al. "A high-itr ssvep-based bci speller". In: *Brain-Computer Interfaces* 1.3-4 (2014), pp. 181–191.

[43] John Polich. "Normal variation of P300 from auditory stimuli". In: *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section* 65.3 (1986), pp. 236–240.

[44] Terence W Picton. *Human auditory evoked potentials*. Plural Publishing, 2010.

[45] *How to Calculate the Power Spectral Density (PSD) for Vibration Analysis*. https://blog.endaq.com/calculate-power-spectral-density-using-the-endaq-open-source-python-library.

[46] Sukhada A.Unde and Revati Shriram. "PSD based Coherence Analysis of EEG Signals for Stroop Task". In: *International Journal of Computer Applications* 95 (June 2014), pp. 1–5. DOI: 10.5120/16675-6778.

[47] Zhonglin Lin et al. "Frequency recognition based on canonical correlation analysis for SSVEP-based BCIs". In: *IEEE transactions on biomedical engineering* 53.12 (2006), pp. 2610–2614.

[48] Dennis J McFarland and Jonathan R Wolpaw. “Brain-computer interfaces for communication and control”. In: *Communications of the ACM* 54.5 (2011), pp. 60–66.

[49] Pasquale Arpaia et al. “Wearable brain–computer interface instrumentation for robot-based rehabilitation by augmented reality”. In: *IEEE Transactions on instrumentation and measurement* 69.9 (2020), pp. 6362–6371.

[50] Xiaogang Chen et al. “Combination of augmented reality based brain-computer interface and computer vision for high-level control of a robotic arm”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 28.12 (2020), pp. 3140–3147.

[51] Lingling Chen et al. “Adaptive asynchronous control system of robotic arm based on augmented reality-assisted brain–computer interface”. In: *Journal of Neural Engineering* 18.6 (2021), p. 066005.

[52] Can Wang et al. “Implementation of a brain-computer interface on a lower-limb exoskeleton”. In: *IEEE access* 6 (2018), pp. 38524–38534.

[53] Feng Duan et al. “Design of a multimodal EEG-based hybrid BCI system with visual servo module”. In: *IEEE Transactions on Autonomous Mental Development* 7.4 (2015), pp. 332–341.

[54] Xu Duan et al. “Quadcopter flight control using a non-invasive multi-modal brain computer interface”. In: *Frontiers in neurorobotics* 13 (2019), p. 23.

[55] Li-Wei Ko, Oleksii Komarov, and Shih-Chuan Lin. “Enhancing the hybrid BCI performance with the common frequency pattern in dual-channel EEG”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 27.7 (2019), pp. 1360–1369.

[56] Xiaogang Chen et al. “Combination of high-frequency SSVEP-based BCI and computer vision for controlling a robotic arm”. In: *Journal of neural engineering* 16.2 (2019), p. 026012.

[57] No-Sang Kwak, Klaus-Robert Müller, and Seong-Whan Lee. “A lower limb exoskeleton control system based on steady state visual evoked potentials”. In: *Journal of neural engineering* 12.5 (2015), p. 056009.

[58] Xiaogang Chen et al. “Control of a 7-DOF robotic arm system with an SSVEP-based BCI”. In: *International journal of neural systems* 28.08 (2018), p. 1850018.

[59] Malik M Naeem Mannan et al. “A hybrid speller design using eye tracking and SSVEP brain–computer interface”. In: *Sensors* 20.3 (2020), p. 891.

[60] Dennis M O'Toole and William G Iacono. "An evaluation of different techniques for removing eye-blink artifact from visual evoked response recordings". In: *Psychophysiology* 24.4 (1987), pp. 487–497.

[61] Alexander Remsik et al. "A review of the progression and future implications of brain-computer interface therapies for restoration of distal upper extremity motor function after stroke". In: *Expert review of medical devices* 13.5 (2016), pp. 445–454.

[62] Andrea Valenti et al. "A deep classifier for upper-limbs motor anticipation tasks in an online BCI setting". In: *Bioengineering* 8.2 (2021), p. 21.

[63] Gernot R Muller-Putz and Gert Pfurtscheller. "Control of an electrical prosthesis with an SSVEP-based BCI". In: *IEEE Transactions on biomedical engineering* 55.1 (2007), pp. 361–364.

[64] Pablo F Diez et al. "Asynchronous BCI control using high-frequency SSVEP". In: *Journal of neuroengineering and rehabilitation* 8.1 (2011), pp. 1–9.

[65] *Home â PsychoPyÂ®*. `https://www.psychopy.org/`. Accessed: 2023-04-19.

[66] *Highspeed Online Processing under Simulink - MathWorks*. `https://www.mathworks.com/products/connections/product_detail/g-hisys.html`. Accessed: 2023-04-19.

[67] Xavier Duart et al. "Evaluating the effect of stimuli color and frequency on SSVEP". In: *Sensors* 21.1 (2020), p. 117.

[68] Xinbo Qian, Yong Ping Xu, and Xiaoping Li. "A CMOS continuous-time low-pass notch filter for EEG systems". In: *Analog Integrated Circuits and Signal Processing* 44 (2005), pp. 231–238.

[69] Suresh D Muthukumaraswamy. "High-frequency brain activity and muscle artifacts in MEG/EEG: a review and recommendations". In: *Frontiers in human neuroscience* 7 (2013), p. 138.

[70] Brendan Allison et al. "BCI demographics: How many (and what kinds of) people can use an SSVEP BCI?" In: *IEEE transactions on neural systems and rehabilitation engineering* 18.2 (2010), pp. 107–116.

[71] Qingguo Wei, Meixia Xiao, and Zongwu Lu. "A comparative study of canonical correlation analysis and power spectral density analysis for SSVEP detection". In: *2011 Third International Conference on Intelligent Human-Machine Systems and Cybernetics*. Vol. 2. IEEE. 2011, pp. 7–10.

[72] Guangyu Bin et al. “An online multi-channel SSVEP-based brain–computer interface using a canonical correlation analysis method”. In: *Journal of neural engineering* 6.4 (2009), p. 046002.

Appendix A

Appendix

A.1 Python Code

```
import timeit
import importlib.util
import importlib.machinery
import pandas as pd
import numpy as np
np.seterr(all="ignore")

from datetime import datetime

now = datetime.now()
fecha_y_hora = now.strftime("%Y.%m.%d-%H.%M.%S")

# Import mymodule
loader = importlib.machinery.SourceFileLoader(
    'cca', '//Mac/Home/Desktop/Tesis/BCI 2023-1/Codigo python/
                                    Modulos/SSVEP_cca_classifier.
                                    py')
spec = importlib.util.spec_from_loader('cca', loader)
cca_class = importlib.util.module_from_spec(spec)
loader.exec_module(cca_class)

# Import mymodule
loader_2 = importlib.machinery.SourceFileLoader(
    'fourier', '//Mac/Home/Desktop/Tesis/BCI 2023-1/Codigo python/
                                    Modulos/
                                    SSVEP_fourier_classifier.py')
spec_2 = importlib.util.spec_from_loader('fourier', loader_2)
fourier_class = importlib.util.module_from_spec(spec_2)
```

```
loader_2.exec_module(fourier_class)


# Import mymodule
loader_3 = importlib.machinery.SourceFileLoader(
    'UdpComms', '//Mac/Home/Desktop/Tesis/BCI 2023-1/Codigo python/
                                        Modulos/UdpComms.py')
spec_3 = importlib.util.spec_from_loader('UdpComms', loader_3)
U = importlib.util.module_from_spec(spec_3)
loader_3.exec_module(U)



# Create UDP socket to use for sending (and receiving)
sock = U.UdpComms(udpIP="192.168.50.33", portTX=8000, portRX=8001,
                  enableRX=False, suppressWarnings=True)

lista_tiempos = []
lista_resultados = []

inicio = timeit.default_timer()

class MyOVBox(OVBox):
    def __init__(self):
        OVBox.__init__(self)

        self.signalHeader = None

        self.target_frequency = 8.1 #8.25
        self.non_target_frequency = 12.1 #12.37

        self.sampling_rate = 250
        self.time_based_epoc = 2

        self.f1_fund = cca_class.generate_fundamental_frequencies(
            self.sampling_rate*self.time_based_epoc, self.
                                                target_frequency,
                                                self.sampling_rate)

        self.f2_fund = cca_class.generate_fundamental_frequencies(
            self.sampling_rate*self.time_based_epoc, self.
                                                non_target_frequency,
                                                 self.sampling_rate)
```

```
        self.tiempoInicio = inicio
        self.tiempoActual = 0

        self.lista_tiempos = []
        self.lista_resultados = []

    def process(self):
        self.tiempoActual = round(
            timeit.default_timer() - self.tiempoInicio, 2)

        for chunkIndex in range(len(self.input[0])):
            if(type(self.input[0][chunkIndex]) == OVSignalHeader):
                self.signalHeader = self.input[0].pop()

                self.tiempoInicio = timeit.default_timer()

            elif(type(self.input[0][chunkIndex]) == OVSignalBuffer)
                                                :
                chunk = self.input[0].pop()

                numpyBuffer = np.array(chunk)
                numpyBuffer = numpyBuffer.reshape(-1, 1)

                classificacion_cca = cca_class.cca_sveep_class(
                    numpyBuffer, self.sampling_rate,
                    self.target_frequency,
                    self.non_target_frequency,
                    self.f1_fund, self.f2_fund)

                #print("El resultado de clasificacion CCA es: ",
                                                classificacion_cca
                                                )

                if classificacion_cca[0] == 'Nada':
                    print("El resultado de CCA es: ------> NADA")
                    self.lista_resultados.append(10000)
                    self.lista_tiempos.append(self.tiempoActual)
                elif classificacion_cca[0] == 'Triangulo':
                    sock.SendData(str(1))
                    print("El resultado de CCA es: ------>
                                                    TRIANGULO")
                    self.lista_resultados.append(96)
                    self.lista_tiempos.append(self.tiempoActual)
                else:
```

```
                    sock.SendData(str(2))
                    print("El resultado de CCA es: ------> CUADRADO
                                                        ")
                    self.lista_resultados.append(160)
                    self.lista_tiempos.append(self.tiempoActual)


    def uninitialize(self):
        df = pd.DataFrame({'time': self.lista_tiempos, 'value':
                                            self.lista_resultados})
        df.to_csv("//Mac/Home/Desktop/Tesis/BCI 2023-1/datos/
                                            Sujeto_8/Indicacion/
                                            prediccion/" + "
                                            prediccion-[" +
                                            fecha_y_hora + "].csv")

box = MyOVBox()
```

A.2 OpenVibe scenario

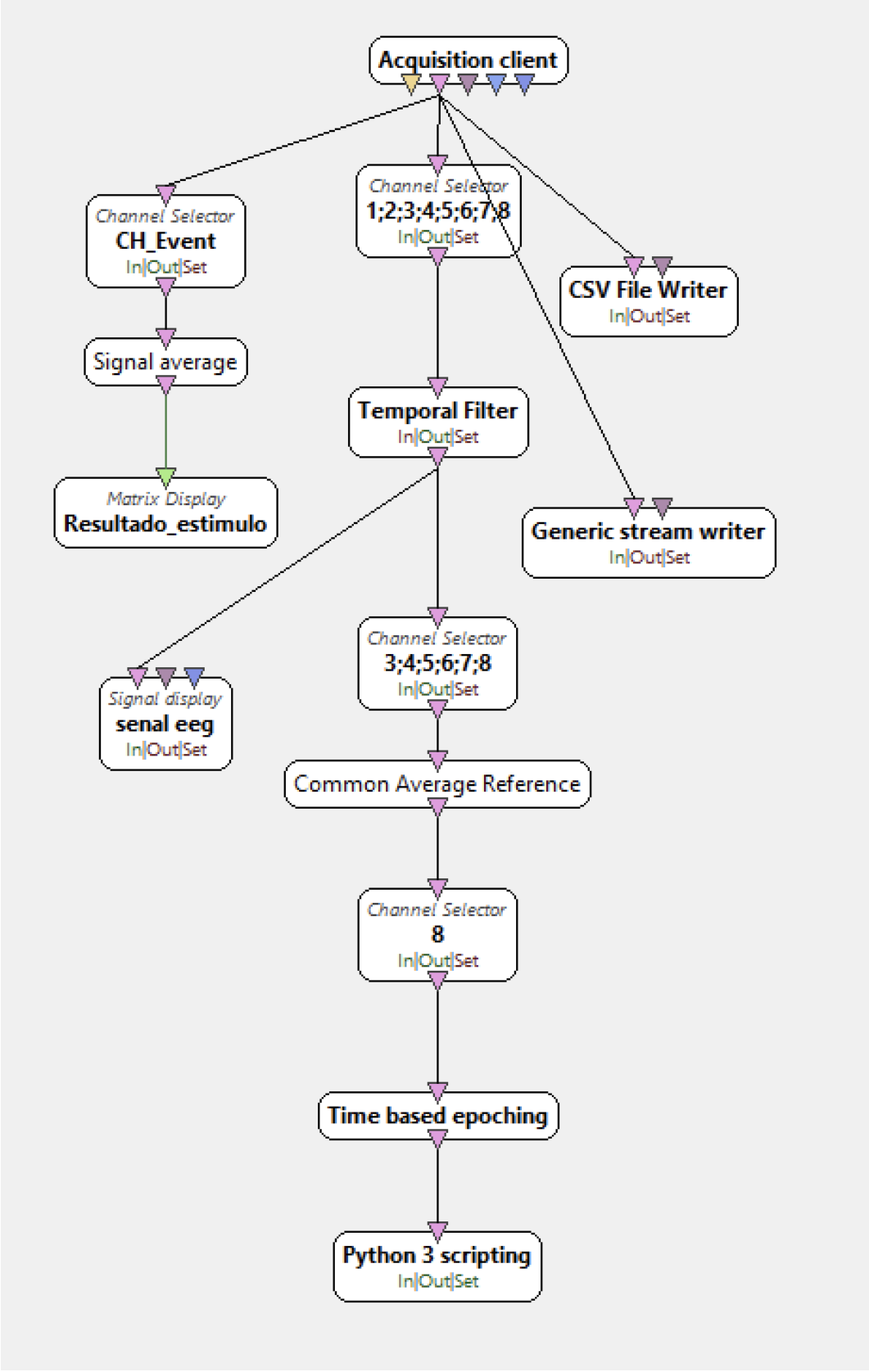


Figure A.1: Online phase scenario

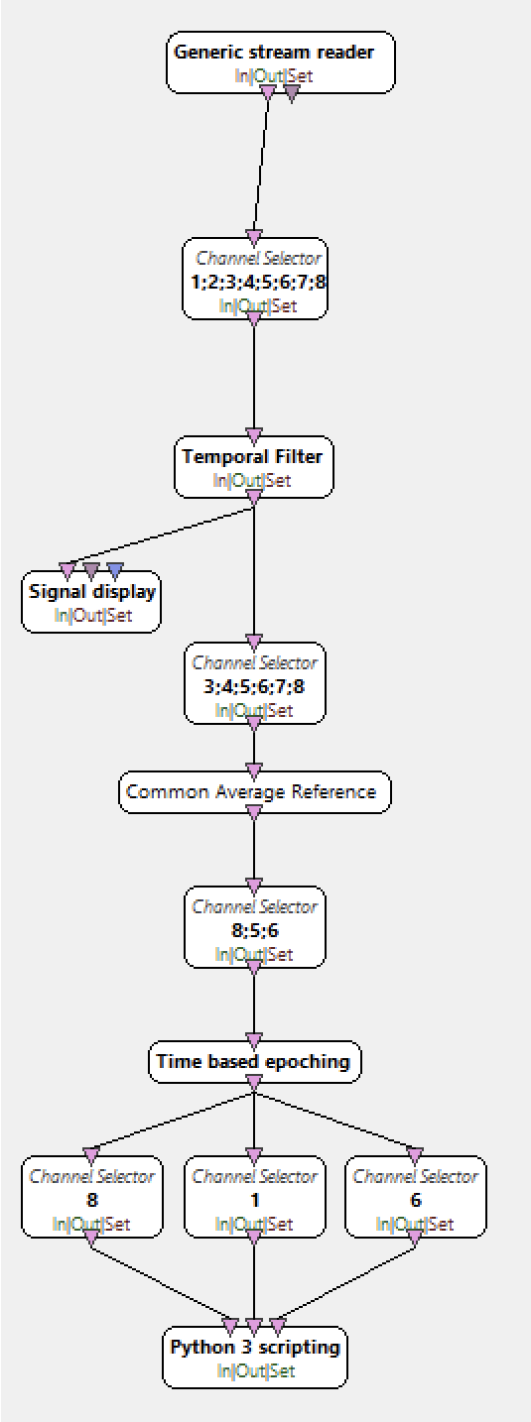


Figure A.2: Offline phase scenario