UNIVERSIDADE D
COIMBRA

João David França Santos

Development of a BCI framework: hardware/software architecture and control of domotic appliances

Dissertação no âmbito do Mestrado Integrado em Engenharia Eletrotécnica e de Computadores, na especialização de Automação, orientada pelo Professor Doutor Urbano José Carreira Nunes e pelo Professor Doutor Gabriel Pereira Pires e apresentada ao Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Setembro de 2019

1 2  9 0

UNIVERSIDADE D
COIMBRA

João David França Santos

Development of a BCI framework: hardware/software architecture and control of domotic appliances

Dissertation supervised by Professor Doctor Urbano José Carreira Nunes and Professor Doctor Gabriel Pereira Pires and submitted to the Electrical and Computer Engineering Department of the Faculty of Science and Technology of the University of Coimbra, in partial fulfilment of the requirements for the Degree of Master in Electrical and Computer Engineering, branch of Automation.

Supervisors:
Prof. Dr. Urbano José Carreira Nunes
Prof. Dr. Gabriel Pereira Pires

Jury:
Prof. Dr. Paulo José Monteiro Peixoto
Prof. Dr. Mahmoud Tavakoli
Prof. Dr. Urbano José Carreira Nunes

Coimbra, September 2019

Acknowledgments

Antes demais gostaria de agradecer aos meus orientadores, Professor Doutor Urbano Nunes e Professor Doutor Gabriel Pires, pela oportunidade de ter trabalhado na área de Brain-Computer Interfaces. Um obrigado pela orientação, conselhos e motivação que me deram para o desenvolvimento deste trabalho.

Esta dissertação foi realizada no âmbito do projecto B-RELIABLE: PTDC/EEI-AUT/30935/2017, sendo financiados pelo FEDER através dos programas CENTRO2020 e a Fundação para a Ciência e a Tecnologia (FCT).

Queria realçar as amizades feitas neste percurso em que agradeço os anos cheio de boas memórias. Queria também agradecer a todos o que me ajudaram a atingir os meus objectivos.

Um especial obrigado à minha família pelo o apoio e motivação incansável em todas as fases da minha vida. Por fim, à Ana, um grande obrigado pela paciência, pelo apoio e motivação constante.

Esta dissertação foi desenvolvida em colaboração com:

1 2 9 0
UNIVERSIDADE D COIMBRA

deer.uc
DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES

INSTITUTE OF SYSTEMS AND ROBOTICS
UNIVERSITY OF COIMBRA

CENTRO 2020 PORTUGAL 2020

Abstract

A Brain-Computer Interface (BCI) allows the direct communication between the brain and a digital computer. BCI systems are still impractical for everyday use and therefore are mostly confined to lab experimentation. There is a need to convert BCI systems into wearable, reliable and standalone devices that can be used in real-world applications outside the lab, so that they can serve potential target users, in communication/control applications for people with severe motor disabilities, or as a tool for neurorehabilitation of motor or neurodevelopmental disorders. In this work, it is proposed the development of a BCI system that can control domotic appliances based on open-source equipment (e.g. OpenBCI). One of the main goals of this thesis is to evaluate whether this device provides some of the desired features for a BCI, regarding wearability, reliability stand-alone operation and low-cost. The proposed BCI is based on a neural mechanism called steady-state visual evoked potential (SSVEP) which is elicited by visual stimulation. Several signal processing methods (FFT, Welch, CCA-Standard, CCA-IT, CCA-Comb and CCA-Lite) were implemented and compared to extract SSVEP features. Initially, the signal processing methods were tested offline on benchmark datasets and then, they were tested offline and online with our BCI setup/framework. Two types of visual stimulation were compared, one elicited by a LED matrix and the other elicited by screen flashes. Offline results showed that Combinational Canonical Correlation Analysis (CCA) achieves the best accuracy in comparison to the other feature extraction methods, but in terms of signal processing time, the CCA-Lite method is faster decreasing to 58% comparing to CCA-Comb. The results obtained in online experiments showed that it was possible to control the BCI with an accuracy of 84.6% (for 5-second identification EEG segments), which shows the feasibility of the system, although several limitations of the OpenBCI system were identified.

Keywords: Brain-Computer Interface (BCI), Electroencephalography (EEG), Steady-State Visual Evoked Potential (SSVEP), Open Source BCI, Domotic Appliances.

Resumo

Uma interface cérebro-computador (ICC) permite a comunicação direta entre o cérebro e um computador digital. Os sistemas de ICC ainda não são exequíveis para o uso diário e, portanto, estão confinados principalmente à experimentação em laboratório. É necessário converter os sistemas ICC em dispositivos portáteis, confiáveis e independentes que possam ser usados em aplicativos do mundo real fora do laboratório, para que possam atender a potenciais usuários-alvo, em aplicativos de comunicação / controlo para pessoas com deficiências motoras graves, ou como uma ferramenta para a neuro-reabilitação de distúrbios motores ou distúrbios do desenvolvimento neurológico. Neste trabalho, propõe-se o desenvolvimento de um sistema ICC que possa controlar dispositivos domóticos com base em um equipamento de open source (OpenBCI). Um dos principais objetivos desta tese é avaliar se este dispositivo fornece alguns dos recursos desejados para um ICC, no que diz respeito ao desgaste, operação autónoma de confiabilidade e baixo custo. O ICC proposto é baseado num neuromecanismo chamado potencial evocado visual em estado estacionário (PEVEE), que é estimulado por estimulação visual. Vários métodos de processamento de sinal (FFT, Welch, CCA-Standard, CCA-IT, CCA-Comb e CCA-Lite) foram implementados e comparados para extrair características do PEVEE. Inicialmente, os métodos de processamento de sinal foram testados offline em bancos de dados de referência e, em seguida, testados offline e online com nossa estrutura / configuração do ICC. Foram comparados dois tipos de estimulação, um provocado por uma matriz de LEDs e o outro provocado por flashes de ecrã. Os resultados offline mostraram que a Análise Combinacional de Correlação Canónica (CCA) atinge melhor precisão em comparação com os outros métodos de extração de recursos, mas em termos de tempo de processamento do sinal, o método CCA-Lite diminui para 58 % do CCA-Comb. Os resultados obtidos online mostraram ser possível controlar o ICC com uma precisão de 84,6 % (para segmentos de EEG de 5 segundos), o que mostra a viabilidade do sistema, embora várias limitações do sistema OpenBCI tenham sido identificadas.

Palavras-chave: Interface Cérebro-Computador (ICC), Eletroencefalografia (EEG), Potencial Evocado Visual em Estado Estável (PEVEE), Open Source ICC, Aparelhos Domóticos.

Contents

List of Acronyms

BCI Brain Computer Interface

CCA Canonical Correlation Analysis

CCA-Comb Canonical Correlation Analysis Combinational

ECoG Electrocorticography

EEG Electroencephalography

EOG Electrooculography

ERP Event Related Potential

FFT Fast Transform Fourier

IT-CCA Individual Template-Canonical Correlation Analysis

ITR Information Transfer Rate

SNR Signal-to-Noise Ratio

SSVEP Steady State Visual Evoked Potential

VEP Visual Evoked Potential

List of Figures

List of Tables

1

Introduction

This chapter contextualizes the reader about this dissertation, providing the motivation, main objectives and key contributions.

1.1 Motivation and context

A Brain-Computer Interface (BCI) allows users to communicate with the external world with a direct channel with the brain without having to use common pathways (muscles, peripheral nerves, speech, etc). This technology is becoming a reality in nowadays and is no longer restricted to human imagination.

A pathway between the brain and a machine can provide a powerful tool to establish a reliable source of human-machine communication. This can be achieved by detecting patterns in the recorded brain activity and matching these patterns with specific commands. Establishing this kind of interaction is probably the only resource for independence and autonomy of subjects in a locked-in state, in which an individual has a total lack of motor control or very low dexterity, affecting head, limbs, eyes and speech, but who is still aware of the surrounding world. This state may result from advanced stages of amyotrophic lateral sclerosis, Parkinson's disease and other neurodegenerative diseases.

The advantage of BCI technology lies in the possibility of establishing a communication channel independent of peripheral nerves and muscles, helping people with severe motor impairments to increase their interaction with the external world. Multiple BCI applications can be implemented such as communication spellers, control of robotic devices, control of appliances, games, etc.

Electroencephalography (EEG) records electrical activity on the scalp and is one of the is most used techniques to control a BCI. EEG measures voltage fluctuations resulting from ionic current between neurons.

Several neural mechanisms can be used to control a BCI, the most common being motor imagination, event-related potentials (ERP) and steady-state visual evoked potentials (SSVEP). Both ERP and SSVEP require external stimulation, while motor imagination is internally induced.

SSVEP is the neural mechanism that will be focused on this dissertation to implement a BCI. SSVEP BCI approaches have several attractive points, namely, they usually do not require user or classification training and they provide high information transfer rates (ITR). In an SSVEP based BCI the external stimulus is a repetitive flicker at a constant frequency that stimulates the visual cortex. The visual stimulation can be produced by an LCD screen or LEDs.

Despite the potential offered by a BCI, this is still far from becoming a plug and play system since it still faces several limitations regarding reliability, wearability, processing capacity, etc. One step in this direction is to use low-cost and wearable devices to make technology available to target users at their homes.

1.2 Main Objectives

The main goal of this work is to build up a BCI system, based on a low-cost device able to perform signal acquisition and signal processing. OpenBCI board was selected as it provides these desired features, opening the possibility of making the entire BCI system free from an external processing source. OpenBCI platform is opensource which allows to expand it and combine it with different other tehcnologies. In particular, we propose to design an SSVEP-based BCI to control home appliances (as a proof-of-concept).

Firstly, we want to validate the quality of the EEG signals, to realize whether the quality of the EEG signal is suitable to discriminate different SSVEPs frequencies. To identification of each frequency will correspond to a BCI command that can be used tp control (or simply switch ON/OFF) a specific home appliance. Different types of visual stimulation were tested to analyse whether it is possible to have stimulation independent of a computer.

Several signal processing approaches for SSVEP feature extraction were implemented, tested and validated offline with benchmark datasets, and also with our own collected data. Then, two of the algorithms were implemented online using the OpenBCI platform, to design an SSVEP-based BCI that controlled home appliances, as a proof-of-concept.

Fig.1.1 shows a schematic representation of the intended overall BCI system.

Figure 1.1: Schematic representation of the intended SSVEP-based BCI.

1.3 Implementations and key contributions

The main developments and contributions of the presented work:

- Development of an offline BCI framework simulating the online operation;
- Offline implementation and analysis of feature extraction algorithms (FFT, Welch, Standard-CCA, CCA-IT, CCA-Comb and CCA-Lite);
- Implementation and comparison of different visual stimulations;
- Offline validation of the methods in online benchmark datasets;
- Development and setup of an oline SSVEP-BCI, based on the OpenBCI framework, to control domotic appliances.

The diagram presented in Fig.1.2 shows a summary of the implemented work.

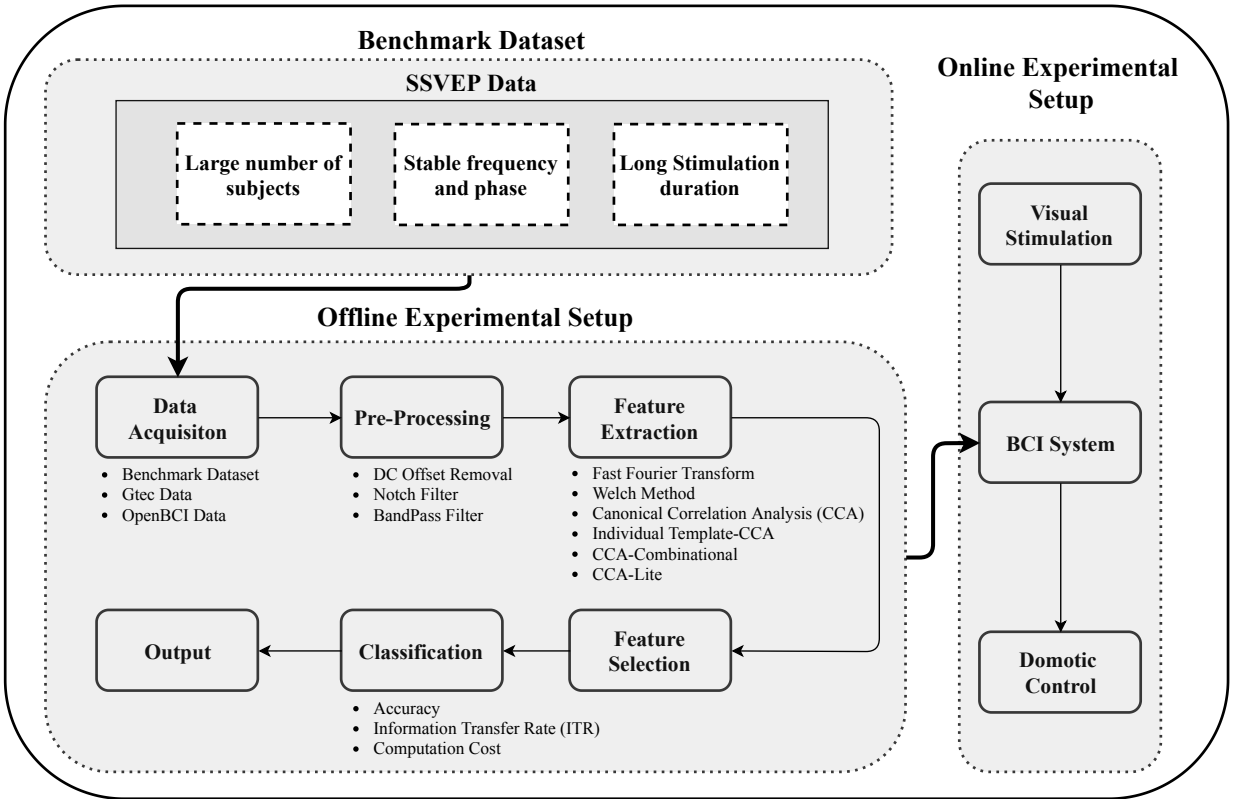


Figure 1.2: Diagram representing the main development in the dissertation.

2

State of the art

This chapter presents the general architecture of a Brain-Computer Interface (BCI), the main neurophysiological concepts and the main neuromechanisms. It also introduces the challenges of BCI and the current state-of-art.

2.1 General Architecture of a Brain-Computer Interface

A BCI based system measures and uses electric signals generated by the brain and interprets/decodes them. Through the extraction of features of the signal and it is possible to convert these brain signals into commands. A command is the execution of the user intention, which in its turn is the action that the user wants to accomplish. Therefore, these commands translate an intended action that can be performed in an output device.

A BCI system includes these main components: (1) signal acquisition; (2) feature extraction; (3) feature conversion; and (4) output commands. A BCI aims to fulfil the user's intention. This can be achieved by identifying and computing features of brain signals and to convert these features into output commands (see Fig.2.1).

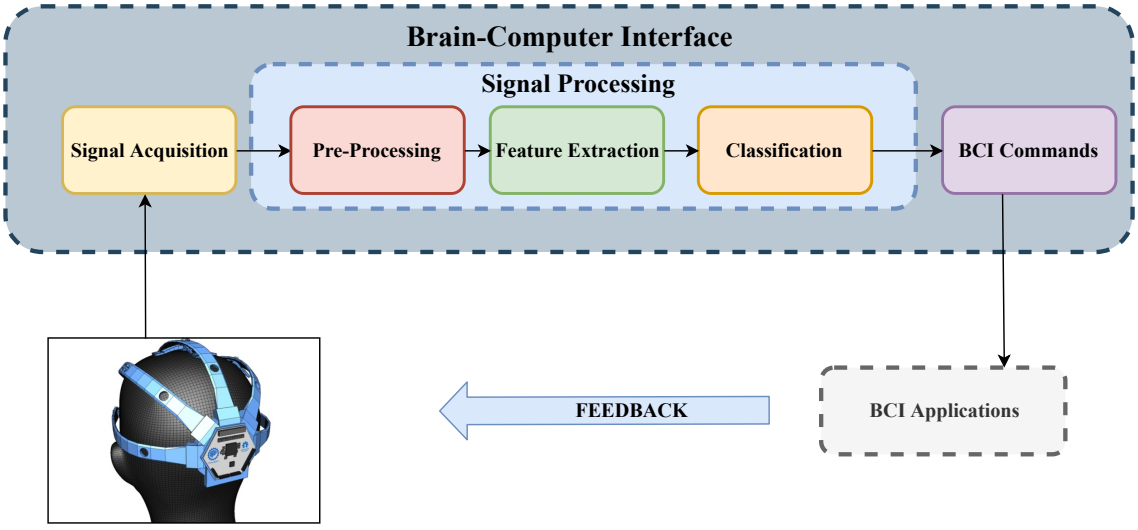


Figure 2.1: BCI design.

The measurement of brain signals utilizing a sensor (e.g., scalp or intracranial electrodes for electrophysiologic activity) is called signal acquisition. The brain signals are then amplified to proper ranges for electronic processing. After this amplification, the signals are digitised and transmitted to a computer [1],[2].

Feature extraction is the proceeding of examining the digital signals. The main goal of this process is to differentiate the relevant signal features so that it can be possible to extract a signal feature related to the person's intent. Such features must have robust correlations with the user's intent [3].

Subsequently, a feature conversion algorithm is applied to the output of the feature extraction, that turns the features into suitable commands (associated with the user's intention). The feature conversion algorithm should ideally be dynamic so that it could adjust to sudden changes in the signal features. And it should also guarantee that the variety of possible features covers the full range of device control [1].

Finally, the commands originated from the feature conversion algorithm can be applied to an external device fulfilling its purpose (executing the user's intention). Then, feedback is provided to the user.

2.1.1 Methods to measure brain activity

The approaches to measuring brain activity can be divided into invasive and non-invasive methods. These methods differ in terms of spatial and temporal resolution, complexity, cost and portability.

Invasive methods need surgery to place the required sensor in the desired area. Electrocorticography (ECoG) is considered an invasive method because the signal is captured from the surface of the cortex or capture signals from within the cortical tissue. Due to the necessity of a surgical procedure, this method is costly, complex and risky. Nevertheless, this method has a clear advantage over non-invasive methods, regarding signal quality, spatial resolution or frequency range [3].

On the other hand, non-invasive methods do not need any kind of surgery, because they measure the brain signal through its electrical and magnetic activities. The most popular method is Electroencephalography (EEG), which records the electrical activity from the scalp, using electrodes. Even though non-invasive methods have lower spatial and temporal resolution than invasive methods, they are the most used, because they can be relatively inexpensive, lightweight and comparatively easy to apply [3].

2.1.1.1 Electroencephalography (EEG)

EEG is very popular due to its simple, low-cost and fast implementation. EEG can be recorded with wet or dry electrodes (Fig.2.2). To enhance the conductivity between the scalp and the elec-

trodes, it is common to use an abrasive gel for skin cleaning and a conductivity gel. With dry electrodes, there is no need to skin preparation on the user's scalp enabling the setup implementation to be faster, in contrast to wet electrodes. However, the dry electrodes have a smaller signal-to-noise ratio (SNR), which reduce the quality of the signal.

The EEG signal obtained can have amplitudes in the order of 0 to 100 mV. The EEG signal that is collected from the electrodes can be susceptible to several types of artefacts (cranial muscular activity, eye movements and blinks, external electromagnetic interferences or even power supply noise).

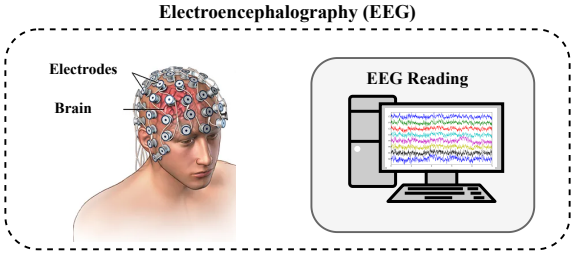


Figure 2.2: Example of a EEG signal recording session.

2.1.2 Brain Signal Frequency Bands

Brain waves can be represented by six typical bands based on the frequency range between 1 and 100 Hz [4], in Table 2.1.

Table 2.1: Principal brain waves [6].

Brain Wave	Location	Mental State	Frequency (Hz)
Delta δ	Everywhere	Reduced consciousness or during sleep	<4
Theta θ	Temporal and parietal	During emotional stress	4-7.9
Alpha α	Occipital and parietal	Associated to mental relaxation and have a reduce amplitude during mental imagery	8-12
Mu μ	Frontal	Associated to intension of movement	8-13
Beta β	Parietal and frontal	Consciously alert, thinking activities	12-30
Gamma γ	Originated in the thalamus	Associated with attention, perception and cognition	25-100

2.2 Neural mechanisms

Several neural mechanisms can be used to control a BCI system. An event-related potential (ERP) is the measured brain signal, which is the outcome of several activities types (sensory, cognitive or motor) [9], [10]. When the user is trying to focus on a mental task, he/she generates changes in mu and beta rhythms in different regions of the motor cortex that can be matched with several different commands. Another ERP is called P300, it is evoked when a significant target event occurs, which represents a peak in the amplitude of the EEG signal that occurs with latency at about 300 ms after the triggering event.

The visual evoked potential (VEP) is the most extensively studied ERP. A VEP is a response to a visual stimulus, which occurs in the first few hundred milliseconds after the user exhibited a visual stimulus. A variation of VEP is the steady-state visual evoked potential (SSVEP), which is caused by repetitive stimulation [12].

2.2.1 Steady State Visual Evoked Potentials (SSVEP)

The SSVEP is evoked by repetitive visual stimulation (flickering stimulus at a certain frequency). The repetition of a stimulus at a specific frequency evokes an EEG signal with the corresponding frequency in the visual cortex. Frequency ranges matching the flickering frequency of the stimulus could be associated with commands to control the BCI.

This neural mechanism is evoked in the visual cortex as a response to flickering stimuli at rates over 6 Hz [19]. It presents a higher ITR, compared to other neural mechanisms. Figure 2.3 shows an example of a SSVEP response.

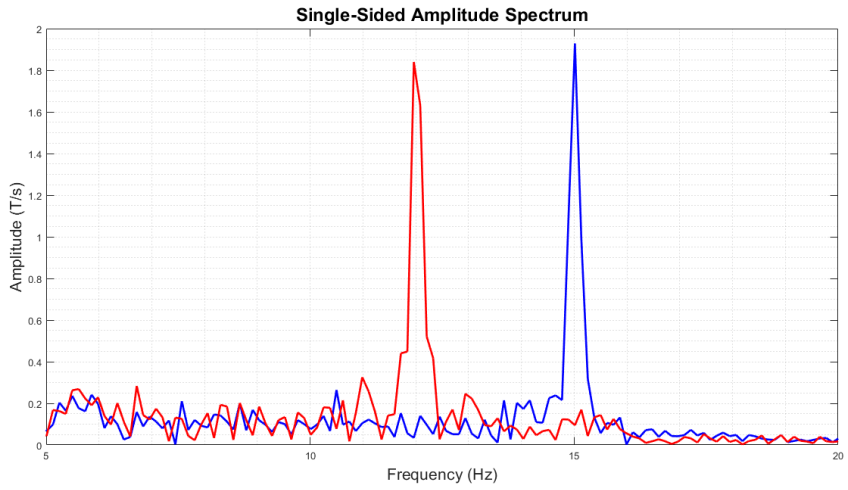


Figure 2.3: Frequency spectrum of SSVEP responses to 13Hz and 15Hz flicker stimuli, measured at channel Oz.

2.3 Existing technologies

Table 2.2 lists the most popular low-cost EEG devices, that could be used in BCI scenarios.

Table 2.2: Table containing Low-Cost BCI Hardware [4],[5].

Device	Channels	Sample Rate (Hz)	ADC Bits	SD Card Support	Motion Sensors	Battery Length	Cost (dollars)
Muse	4-6	256	12	No accessory	3 axis	5 hours	200
Insight	5+2 references	2048	15	With accessory	9 axis	4 hours using Bluetooth	300
Epoc	14+2 references	256	16	With accessory	9 axis	6 hours using Bluetooth Low-Energy	799
OpenBCI	Up to 16 channels	250-2000	24	Yes	3 axis	∼26 hours	500 for 8 channels, 949 for 16 channels
Neurosky Mindwave	1+1 reference	512	12	N/A	N/A	8 hours	99.99

Table 2.3 shows BCI studies/applications based on low-cost equipment. One of the goals of these applications was to allow the user to control external devices used in the daily routine. Another goal of these implementations was to test methods/approaches on low-cost devices that could achieve performances comparable to those of research grade devices, boosting their use as reliable and wearable devices. Some problems from these applications are the feasibility of the system and the quality of the EEG signal acquired. Nevertheless, some reported studies show that a BCI system can be implemented in low-cost equipment and achieve good accuracy rates (see Table 2.3). And considering that most BCI users can not afford a high cost of medical EEG equipment, no matter how good its performance is, researching the effectiveness of these systems is a step forward to bring these devices to more people and to new scenarios out of the lab.

Table 2.3: Literature of BCI implementation in low-cost devices.

Article	Type	Feature Extraction Method	Acquisition System	Electrodes Number	Number of commands	Accuracy
Ruhunage 2018 [14]	EEG, EOG, SSVEP and P300	CCA	OpenBCI	7	4	84,5 %
Alvarado 2017 [15]	EEG and Matlab	Deep Learning	OpenBCI	4	-	-
Perera 2016 [16]	EEG and SSVEP	FFT	OpenBCI	8	3	87,88 %
Shivappa 2018 [17]	EEG and Auditory Steady State Response	FFT	OpenBCI	4	2	92 %
Lin 2014 [18]	EEG and SSVEP	CCA	Neurosky	14	4	76,6 %

The performance of a BCI depends primarily on the subject's performance and the signal quality. A large number of BCI publications were conducted based on SSVEPs in order to explore new feature extraction methods that could improve their discriminattion and detection.

Table 2.4 shows a brief summary of relevant studies of feature extraction methods for SSVEP detection. Best results have been achieved with CCA and variants.

Table 2.4: Literature of BCI implementation with SSVEP.

Article	Stimulus Time	Phase sync	Feature Extraction Method	Acquisition System	Individual Template	Electrodes Number	Targets Number	Offline Accuracy	Online Accuracy
Bin 2009 [27]	2 s	no	CCA	BioSemi Active 2	no	9	6	-	95,3 %
Gao 2006 [40]	2,25 s	no	PSDA/CCA	BioSemi Active 2	no	8	9	PSDA 70 % CCA 76 %	-
Jia 2010 [19]	2 s	yes	FFT	BioSemi Active 2	no	32	15	87,1 %	-
Wei 2011 [39]	-	no	PSDA/CCA	BioSemi Active 2	no	64	24	-	PSDA 86,9 % CCA 97,62 %
Nakanishi 2014 [35]	1 s	no	CCA/CCA-Comb	gUSBamp	yes	16	8	-	CCA 88,4 % CCA-Comb 95,2 %
Nakanishi 2015 [33]	1 s	no	CCA/ IT-CCA/ CCA-Comb	BioSemi Active 2	yes	8	12	CCA 55 % IT-CCA 81,17 % CCA-Comb 92,78 %	CCA 57 % IT-CCA 81 % CCA-Comb 92 %
Dokyun 2017 [37]	1,5 s	no	PSDA/CCA/ CCA-Comb/ CCA-Lite	BioSemi Active 2	yes	1	12	-	PSDA 31,61 % CCA 58,67 % CCA-Comb 86,06 % CCA-Lite 85,39 %

3

Background Material

This chapter describes the hardware and software modules that will be used to implement the SSVEP-based BCI, as well as the signal processing methods that were considered to detect the SSVEPs.

3.1 Hardware

The EEG acquisition system (electrodes, cap and bio amplifier) is based on the OpenBCI platform, which is a low cost, programmable, open-source EEG platform that gives any consumer the possibility to access their brain waves. This tool can take advantage of the open-source movement to accelerate the development through collaborative hardware and contributions of the OpenBCI community. This platform is mainly composed of a high powered analogue front-end, a programmable microcontroller and a wireless communication module, which are described in the following sections.

3.1.1 OpenBCI 32bit Cyton board

OpenBCI Cyton board is built around Texas Instrument's ADS1299 (see Fig.3.1). The ADS1299 is an 8-channel, low-noise, 24-bit analogue-to-digital converter designed specifically for measuring very low EEG signals. It has the capacity of reading bioelectrical signals amplifying and eliminating noises. It also can generate internal signals for testing and calibration, as well as EEG-specific functions like a lead-off detection, to ensure that the electrodes are making good contact with the scalp.

The OpenBCI Cyton Board comes with an onboard re-programmable microcontroller, the PIC32MX250F128B, which provides local memory and processing power. The data received from the electrodes can be processed in this module, not needing to have an external computer for signal processing.

This microcontroller is responsible for receiving the digital data of ADS1299 and processing in a format suitable for transmission. It also has incorporated a LIS3DH accelerometer, which is

an ultra-low-power high-performance three-axis linear accelerometer. The device features ultra-low-power operational modes allow advanced power-saving and smart embedded functions, like generate interrupt signals using two independent inertial wake-up/free-fall events, according to the device's position.

This board has a sample rate of 250 Hz, but if the Daisy module is plugged in, which gives the board the ability to increase the number of channels to 16, the sample rate is only 125 Hz. The board communicates wirelessly to a computer via the RFDuino Bluetooth module.

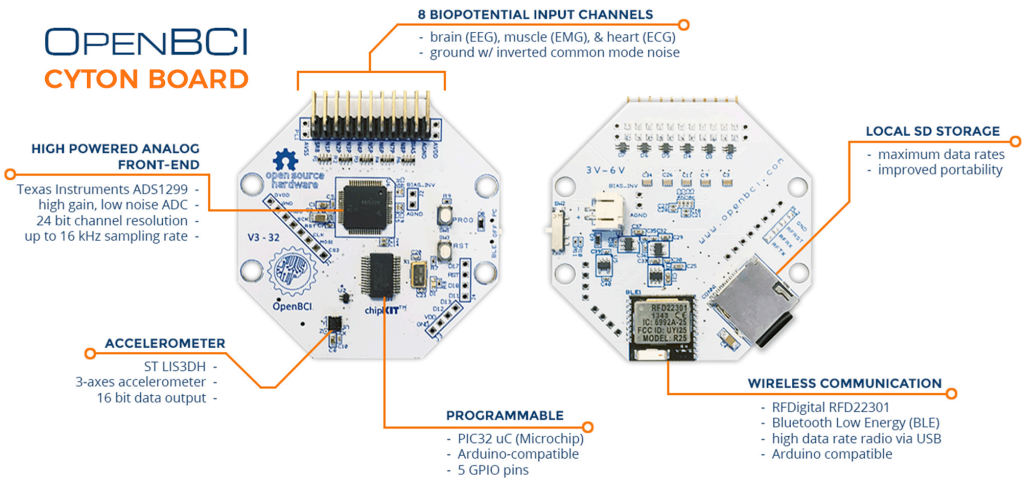


Figure 3.1: OpenBCI Cyton Board: Main modules and features (Fig. reproduced from [21])

3.2 Software

OpenBCI provides software for streaming, recording and visualizing data in a PC. The software in the PC allows the visualization per channel as well as the application of filters and a Fast Fourier Transform.

For data processing, several Software Development Kits (SDKs) are available. In the PC, EEG data can be processed online in Matlab using a Python-based intermediate application (Lab Streaming Layer - LSL) that receives data from OpenBCI 32bit Cyton board.

3.2.1 Lab Streaming Layer

LabStreamingLayer (LSL) is an interface application for the unified collection of time-series measurements in research experiments that handles both the networking, time-synchronization, (near-) realtime access, as well as (optionally) the centralized collection, viewing and disk recording of the data. The LSL distribution consists of two main components:

- The core transport library (liblsl) and its language interfaces (C, C++, Python, Java, C, MATLAB);

- A suite of tools built on top of the library, including a recording program, online viewers, importers and apps that make data from a range of acquisition hardware available on the lab network (for example audio, EEG or motion capture).

This library was used to connect the OpenBCI Cyton Python SDK with Matlab.

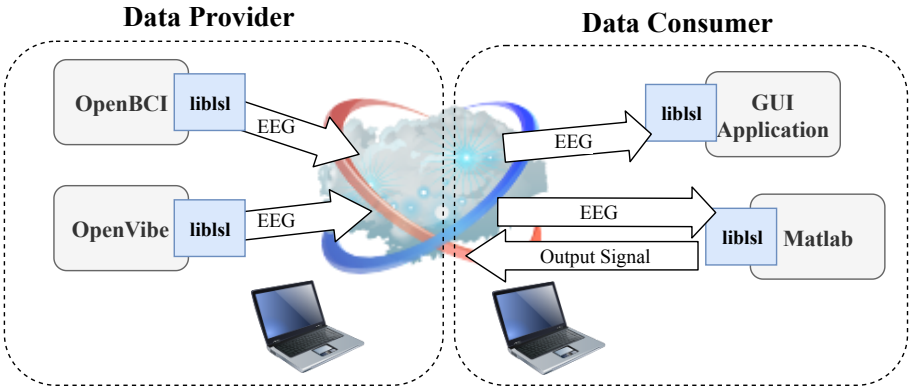


Figure 3.2: Schematic view of the Lab Streaming Layer (LSL) software framework for collecting, storing, and processing multi-modal laboratory data including data collected. LSL runs on a local area network (or, conceptually, a compute cloud network) and efficiently links data providers physiological and/or behavioral recording systems) with data consumers (data viewer, recorder, or analysis facilities),(Fig. adapted from [7]).

3.3 Visual stimulation

Visual stimuli in current SSVEP-based BCIs are displayed either on a computer screen (by subsequent changes of a selected area of the screen driven by software) or by using LEDs driven by hardware frequency generators. In terms of form and colour of visual stimuli, using a computer screen provides high flexibility, but they are constrained by their refresh rate and non-realtime nature of operating systems.

On the other hand, LEDs powered by hardware generators are not limited in frequency but have limitations in terms of format, colour and patterns that can be played back - basically, a static menu with fixed symbols should be linked to the stimulator's design.

3.4 EEG signal processing

Before classification, there is a pre-processing of the EEG signal, feature extraction and a feature selection. The preprocessing phase aims to eliminate signal frequencies outside the range of interest and interferences that the signal may have suffered from the surroundings. EEG signals are often affected by noise from contact between the electrode and the skin, muscle artefacts and the source of electric energy (50 Hz in Europe) [40].

3.4.1 Pre-processing

In pre-processing, an EEG signal is filtered in order to eliminate irrelevant information. The elimination of known noise and limiting the signal frequency to the band of interest allows the increase of the signal-to-noise ratio (SNR). A low SNR means that detectable patterns will be difficult to find. Even with a high SNR, EEG patterns of interest are difficult to identify, which makes necessary the application of effective signal processing methods to extract discriminative features.

To eliminate these known interferences, EEG signals are usually filtered to remove the DC component, the 50 Hz powerline interference, and to remove irrelevant band frequencies. Frequencies above 50 Hz should are usually discarded to avoid muscle artefacts contamination [24].

3.4.2 Feature Extraction Methods

Feature extraction is one of the most important steps to identify the signals of interest and the most discriminative features. This section describes the signal processing methods that were considered in this dissertation to detect SSVEPs, which includes methods in the time and frequency domains and methods that take advantage of EEG spatial correlation [25]. It is important to refer that it was not applied any classifier because the output vector from the feature extraction methods has a dimension of 1×1 and thefore the decision was based on thresholds. There was no user's training conducted and the data collected before online experiments were used only to previously adjust thesholds to participants. Particularly in IT-CCA (section 3.4.2.4) a SSVEP template of the participant was needed.

3.4.2.1 Power Spectral Density Analysis

Traditional, Power Spectral Density Analysis (PSDA) method is widely used in SSVEP-based BCIs, which is a frequency domain processing method. It finds the frequency at which a maximum magnitude in the power spectrum of the SSVEP signal occurs. PSDA is usually implemented using a Fast Fourier Transform (FFT) for a single-channel time-series signal.

The Fourier transform converts a signal in the time domain to the frequency domain (spectrum) and it is implemented through the Discrete Fourier Transform (DFT) according to:

$$X_k = \sum_{n=0}^{N-1} x_n . e^{-i2\pi kn/N} \tag{3.1}$$

The Fast Fourier Transform refers to a very efficient algorithm. The central insight which leads to this algorithm is the realization that a discrete Fourier transform of a sequence of N points can be written in terms of two discrete Fourier transforms of length $N/2$. Thus, if N is a power of

two, it is possible to recursively apply this decomposition until we are left with discrete Fourier transforms of single points.

3.4.2.2 Welch's Method

Welch's method (also called the periodogram) is used to improve the estimation of the power spectrum. The method consists of dividing the signal into successive blocks, forming the periodogram for each block, and averaging [26].

Let us consider the m th windowed, zero-padded frame from the signal x by

$$x_m(n) \triangleq w(n)x(n+mR), \quad n = 0, 1, \ldots, M-1, m = 0, 1, \ldots, K-1, \tag{3.2}$$

where R is defined as the window hop size, and K denote the number of available frames. Then, the periodogram of the m th block is given by

$$P_{x_m,M}(w_k) = \frac{1}{M}|FFT_{N,k}(x_m)|^2 \triangleq \frac{1}{M}\left|\sum_{n=0}^{N-1} x_m(n).e^{-i2\pi kn/N}\right|^2 \tag{3.3}$$

and the Welch estimate of the power spectral density is given by the average of periodograms across time

$$\hat{S}_x^W(w_k) \triangleq \frac{1}{K}\sum_{m=0}^{K-1} P_{x_m,M}(w_k). \tag{3.4}$$

Different types of windows, $w(n)$ can be applied (rectangular, Hamming, etc), as well as different overlap percentages allowing to adjust power spectral resolution, leakage effects, standard deviation, etc.

3.4.2.3 Canonical Correlation Analysis

The Canonical Correlation Analysis (CCA) method is one of the most effective methods used for SSVEP detection. It is applied to multi-channel EEG and usually shows high detection accuracy [27] (see Table 2.4). It optimizes the recognition procedure because it combines information from multiple channels to improve the SNR [28]. CCA is a method of exploring the relationships between two multivariate sets of variables or vectors, to infer their similarity (see Fig.3.3).

In the SSVEP detection context, this method is used to detect the similarity between stimuli frequency and the reference signal frequencies. For two multivariate variables X and Y, CCA transforms them into 1D variable x and y, through a pair of vectors w_x and w_y, to maximize the correlation between x and y [27].

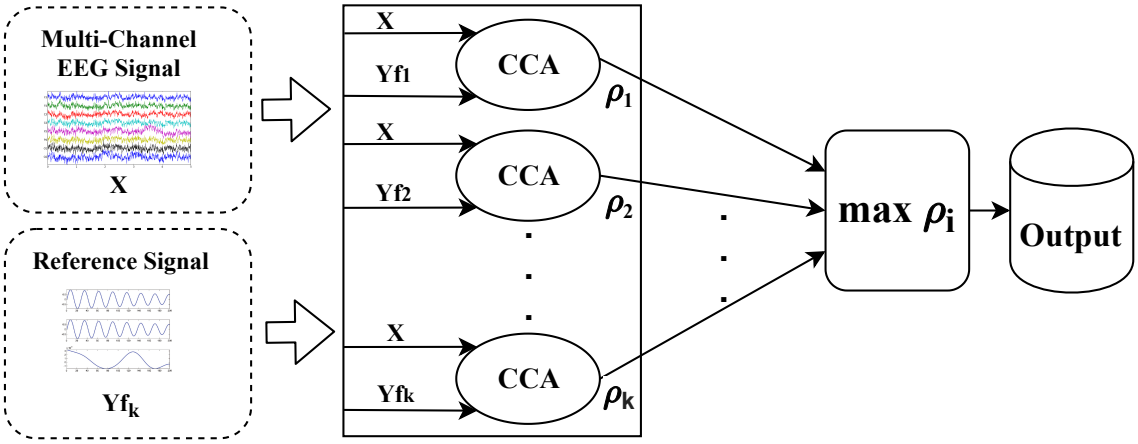


Figure 3.3: Usage of CCA in EEG signals analysis (Fig. adapted from [27]).

Reference signal Y is designed as a group of sine and cosine waveform with frequencies containing the frequency of stimuli f and its harmonics, as described as $Y(f)$ [27],[28].

$$Y(f) = \begin{bmatrix} \sin(2\pi \times f \times n) \\ \cos(2\pi \times f \times n) \\ \dots \\ \sin(2\pi \times m \times f \times n) \\ \cos(2\pi \times m \times f \times n) \end{bmatrix}, n = \frac{1}{S}, \frac{2}{S}, \dots, \frac{N}{S} \tag{3.5}$$

where N is the number of samples in an EEG epoch, S is the sampling rate and f is the base frequency to be detected.

CCA finds the weight vectors w_x and w_y, through eigenvalue decomposition of the covariance matrix (see Fig.3.4).

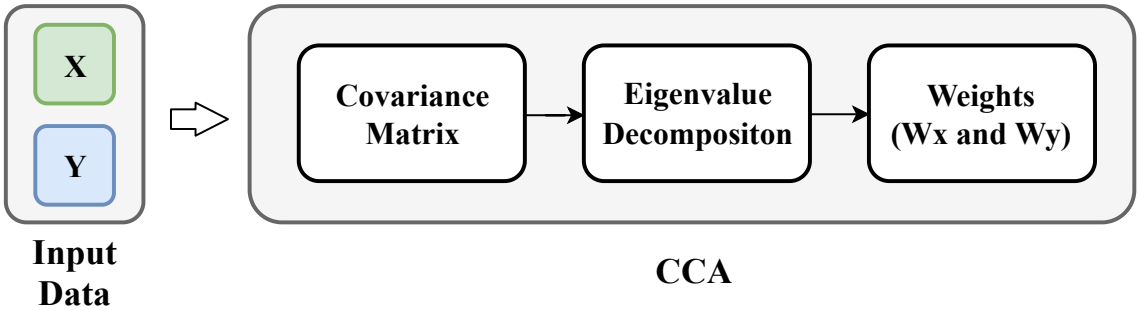


Figure 3.4: Calculation of the weights in the CCA.

The covariance matrix can be calculated by:

$$C = E[(X - E[X])(Y - E[Y])] \tag{3.6}$$

where the operator $E[x]$ represents the average of x data.

Then, the correlation between x and y is maximized by:

$$\max_{w_x,w_y} \rho(x,y) = \frac{E[w_x^T X Y^T w_y]}{\sqrt{E[w_x^T X X^T w_y] E[w_x^T Y Y^T w_y]}} \tag{3.7}$$

where X is the two dimensional EEG data matrix with $N \times T$ dimensions (N: number of channels and T: time samples). Y corresponds to the reference signal with $M \times T$ dimensions (M: number of harmonics x 2). The operator $E[x]$ represents the average of x data, the weight vectors w_x and w_y have $N \times N$ and $M \times N$ dimensions, respectively.

The maximum of ρ concerning to w_x and w_y is the maximum canonical correlation. Projections onto w_x and w_y, i.e. x and y, are called canonical variants.

The multi-channel EEG signals and each of the reference signals is used as an input of the CCA method, which is applied to each frequency of the reference signal. The canonical correlation output ρ can then be used for frequency recognition [27],[29].

$$L = \max_i \rho_i, i = 1, 2, ..., K, \tag{3.8}$$

where ρ_i are the CCA coefficients obtained with the frequency of reference signals being f_1, f_2, ... , f_K.

3.4.2.4 Individual Template Based CCA

Recently, various extensions of CCA have been proposed to incorporate individual EEG calibration data in CCA to improve the detection performance. The Individual Template Based CCA (IT-CCA) approach was proposed to detect temporal features of EEG signals using a canonical correlation between test data and individual template signals (subject-specific) in an SSVEP based BCI. This approach has been also applied for SSVEP detection [30],[36].

For each frequency target, an individual template is obtained by averaging multiple training trials as $\bar{X}_{nj} = \frac{1}{N_t} \sum_{h=1}^{N_t} X_{nj}$, N_t is total number of samples, n is the number of the sample and j is the number of the trial. In this case, reference signals Y_n of the Standard-CCA are replaced by the individual template $\bar{X}_n$ and then, the IT-CCA is computed the same way as in Standard-CCA:

$$\rho_n = \max_{w_x,w_{\bar{x}}} \frac{E[w_x^T X \bar{X}_n^T w_{\bar{X}}]}{\sqrt{E[w_x^T X X^T w_x] E[w_{\bar{x}}^T \bar{X}_n \bar{X}_n^T w_{\bar{x}}]}} \tag{3.9}$$

3.4.2.5 CCA-Combinational

The CCA-Combinatorial (CCA-Comb) combines the Standard-CCA and the IT-CCA approaches [35],[36]. Figure 3.5 shows the CCA-Comb architeture. Correlation coefficients between projections of a test set $\hat{X}$ and an individual template $\bar{X}_n$ using CCA-based spatial filters are used as

features for target identification [35]. Specifically, the following three weight vectors are used as spatial filters to enhance the SNR of SSVEPs: (1) $W_x(\hat{X}\bar{X}_n)$ between the test set $\hat{X}$ and the individual template $\bar{X}_n$, (2) $W_x(\hat{X}Y_n)$ between the test set $\hat{X}$ and sine-cosine reference signals Y_n, and (3) $W_x(\bar{X}_nY_n)$ between the individual template $\bar{X}_n$ and sine-cosine reference signals Y_n. A correlation vector r_n is defined by:

$$r_n = \begin{bmatrix} \rho \\ r_{n,1} \\ r_{n,2} \\ r_{n,3} \end{bmatrix} = \begin{bmatrix} \rho \\ r(\hat{X}^T W_x(\hat{X}\bar{X}_n), \bar{X}_n^T W_x(\hat{X}\bar{X}_n)) \\ r(\hat{X}^T W_x(\hat{X}Y_n), \bar{X}_n^T W_x(\hat{X}Y_n)) \\ r(\hat{X}^T W_x(\bar{X}_nY_n), \bar{X}_n^{TT} W_x(\bar{X}_n^T Y_n)) \end{bmatrix} \tag{3.10}$$

where $r(a, b)$ indicates the Pearson's correlation coefficient between two one-dimensional signals a and b. An ensemble classifier can be used to combine the four features. In practice, the following weighted correlation coefficient ρ_n is used as the final feature in target identification:

$$\rho_n = \sum_{l=1}^{4} sign(r_{n,l}).r_{n,l}^2 \tag{3.11}$$

where $sign()$ is used to retain discriminative information from negative correlation coefficients between test set $\hat{X}$ and individual template $\bar{X}_n$, The individual template that maximizes the weight correlation value is selected as the reference signal corresponding to the target.

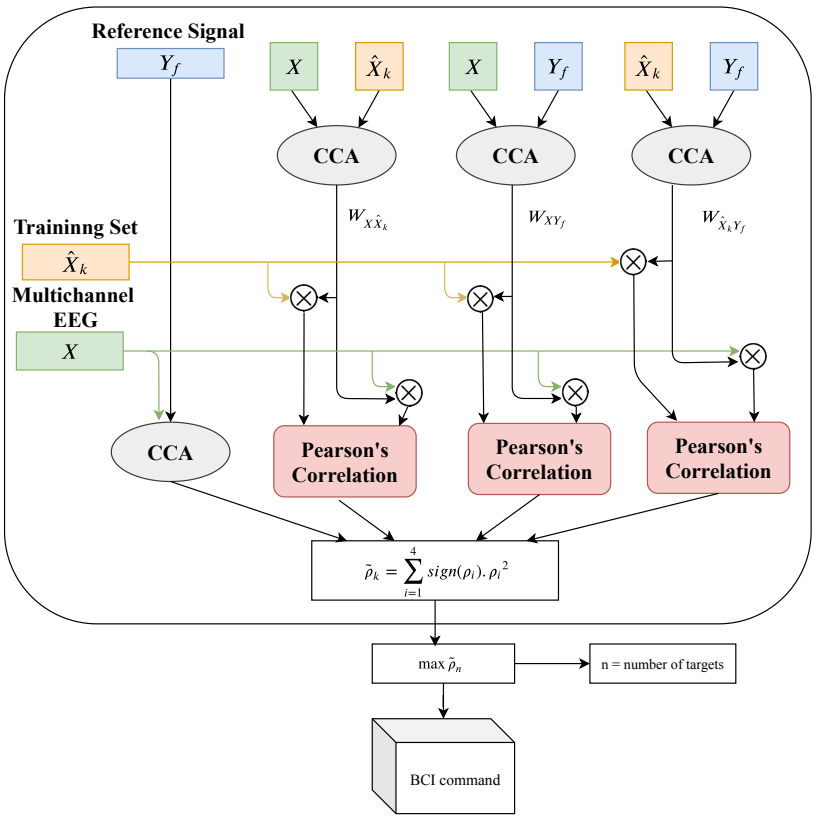


Figure 3.5: Usage of CCA-Comb in EEG signals analysis.

4

Developed Work

This chapter describes the online and offline implementation of the selected signal processing methods, as well as the implementation of the overall BCI framework (from EEG acquisition to appliances control).

4.1 Offline Experimental Setup

This section details each aspect of the offline implementation, which simulates the online operation. The classification pipeline is composed of four modules, as shown in Fig.4.1: the Pre-Processing Module, the Feature Extraction Module, the Feature Selection Module and the Output Detection Module.

Signal processing methods and classification pipeline were initially validated with an online benchmark dataset. This allowed to validate the methods, as well as to compare their performance with state-of-the-art results. On the other hand, it served as a reference for comparison with our own collected datasets.

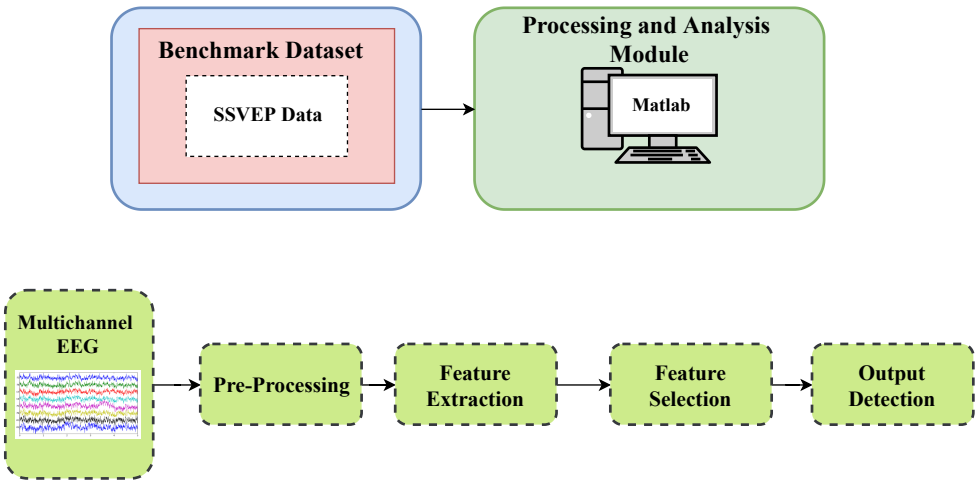


Figure 4.1: Offline experimental setup.

4.1.1 Benchmark Dataset

It is difficult to compare the performance of algorithms when the data collected is very noisy, as it was the case of the SSVEPs collected. Therefore, we needed to validate separately: 1) signal processing methods, 2) quality of EEG recorded from OpenBCI acquisition system, and 3) effectiveness of visual stimulation. So, an online benchmark dataset was initially used to analyse SSVEP neurophysiologically and to test and validate signal processing methods.

This dataset was used as a benchmark dataset to compare the methods for stimulus coding and target identification in SSVEP based BCIs. It also provides high-quality data for computational modelling of SSVEPs. The offline simulation gives the possibility of designing new modules, develop computational models and evaluate their BCI performance without collecting any new data [38].

The dataset has the following features: 1) A large number of subjects (8 experienced and 27 naive, 35 in total) were recorded; 2) A large number of stimulation frequencies (40, range: 8–15.8 Hz with an interval of 0.2 Hz) were included, and a 5-s stimulation for each frequency was repeated six times in recording; 3) Stimulus events (onsets and offsets) were accurately synchronized to EEG data; and 4) 64-channel whole-head EEG data were recorded [38].

The EEG was recorded with a Synamps2 EEG system (Neuroscan, Inc.) at a sampling rate of 1000 Hz. Then all epochs were down-sampled to 250 Hz to reduce storage and computation costs. The data matrix consists of 240 trials (40 targets × 6 blocks) and each trial consists of 64 channels of 1500-point data [38] (see Fig.4.2).

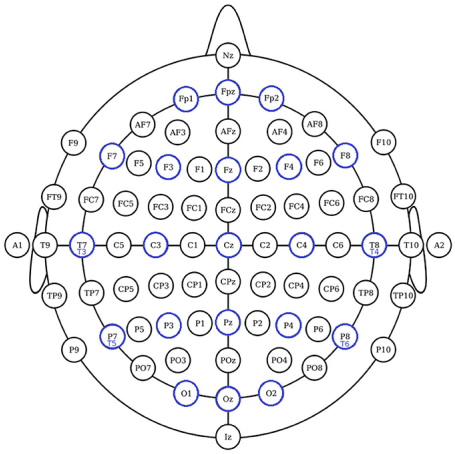


Figure 4.2: 64 electrode positions for 10-20 extended system.

For each subject, the experiment included six blocks, each containing 40 trials corresponding to all 40 characters in random order. Each trial started with a 0.5-s target cue. Subjects were asked to shift their gaze to the target as soon as possible. After the cue, all stimuli started to flicker on the screen concurrently for 5 s.

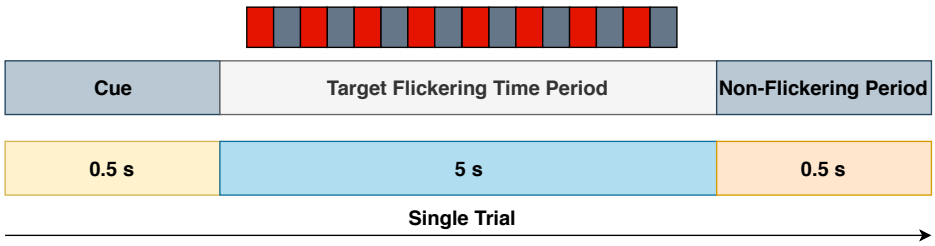


Figure 4.3: Single trial from benchmark dataset.

Then, the screen was blank for 0.5 s before the next trial began. Each trial lasted 6 s in total (Fig.4.3). Subjects were instructed to avoid eye blinks during the 5-s stimulation duration. There was a rest period for several minutes between two consecutive blocks [38]. It is important to notice that the dataset is provided as raw data, i.e., without any pre-processing.

4.1.2 Pre-processing

As mentioned before in section 3.4.1, EEG signal pre-processing goes through 3 steps. The EEG signals were filtered using a DC Offset Removal, a 50 Hz Notch Filter and a Butterworth Filter (Bandpass) third order with a minimum cut-off frequency of 5 Hz and maximum cut-off frequency 50 Hz, as illustrated in Fig.4.4.

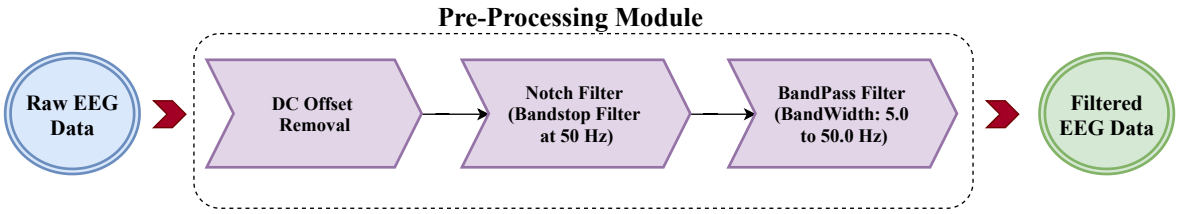


Figure 4.4: Pre-Processing Module.

4.1.3 Feature Extraction Methods

4.1.3.1 Fast Fourier Transform

For the FFT approach, it was necessary to come up with an algorithm that made the feature detection more efficient than the FFT applied to a single EEG channel. The FFT was implemented for single-channel time-series signal [39], but as EEG data are multi-channel, so the approach accumulates the FFTs of each channel, given by:

$$\bar{X}_n = \frac{1}{N_t} \sum_{h=1}^{N_t} X_N \tag{4.1}$$

where N_t is the total number of channels, N is the number of a single channel. For each channel, a spectrum amplitude normalization was applied [40], according to:

$$P = \frac{|FFT(x)|}{\sum |FFT(x)|} \tag{4.2}$$

where x is the filtered single-channel EEG data and $FFT(x)$ is the fast Fourier transform of x. Hence, the amplitude spectrum was normalized to one by summing all the frequency points of the spectrum.

Only with the detection of the fundamental frequency of each target, the target detection was not effective, i.e., the accuracy of the feature detection was very low. As preliminary results were not satisfactory, the spectrum amplitudes of the second and third harmonics of the stimulation frequency were averaged.

The multiplicity of the harmonics does not interfere with the other target's frequencies, because the final target frequency was 15.8Hz and the second harmonic of the first target (8Hz) was 16Hz, so there was no frequency overlap. If the frequency of the maximum amplitude in the frequency spectrum belonged to a possible target frequency, that frequency was considered the selected target. If the FFT outcome was not one of the target frequencies, the larger amplitude-frequency of the two adjacent target frequencies become the selected target.

Target frequencies were 0.2 Hz apart. As the sampling rate was 250 Hz and the SSVEP segment had 1250 samples (5 seconds of duration), the number of FFT points was selected as the nearest power of 2, higher than 1250, i.e., 2048 points, yielding a frequency resolution of 0.12 Hz ($\frac{Fs}{window} = \frac{250Hz}{2048points}$). See the frequency spectrum of Fig.4.5 and an EEG segment elicited by an 8 Hz stimulation.

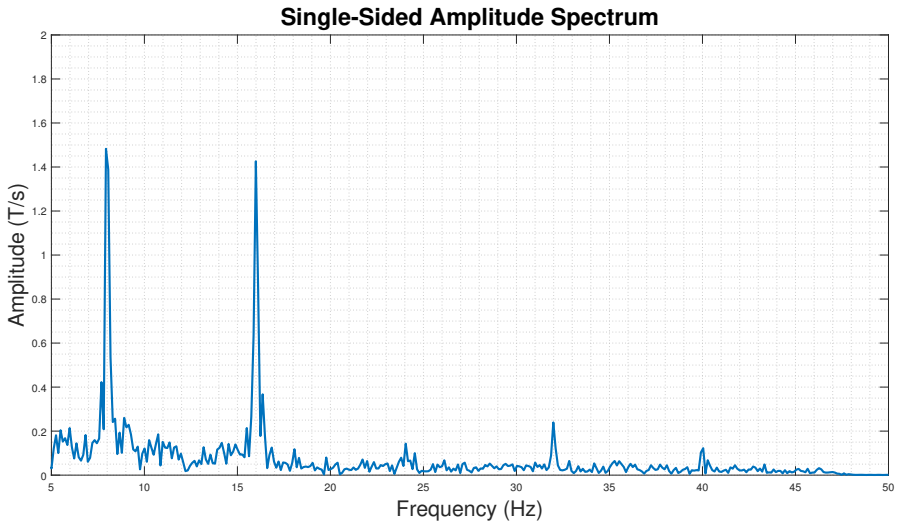
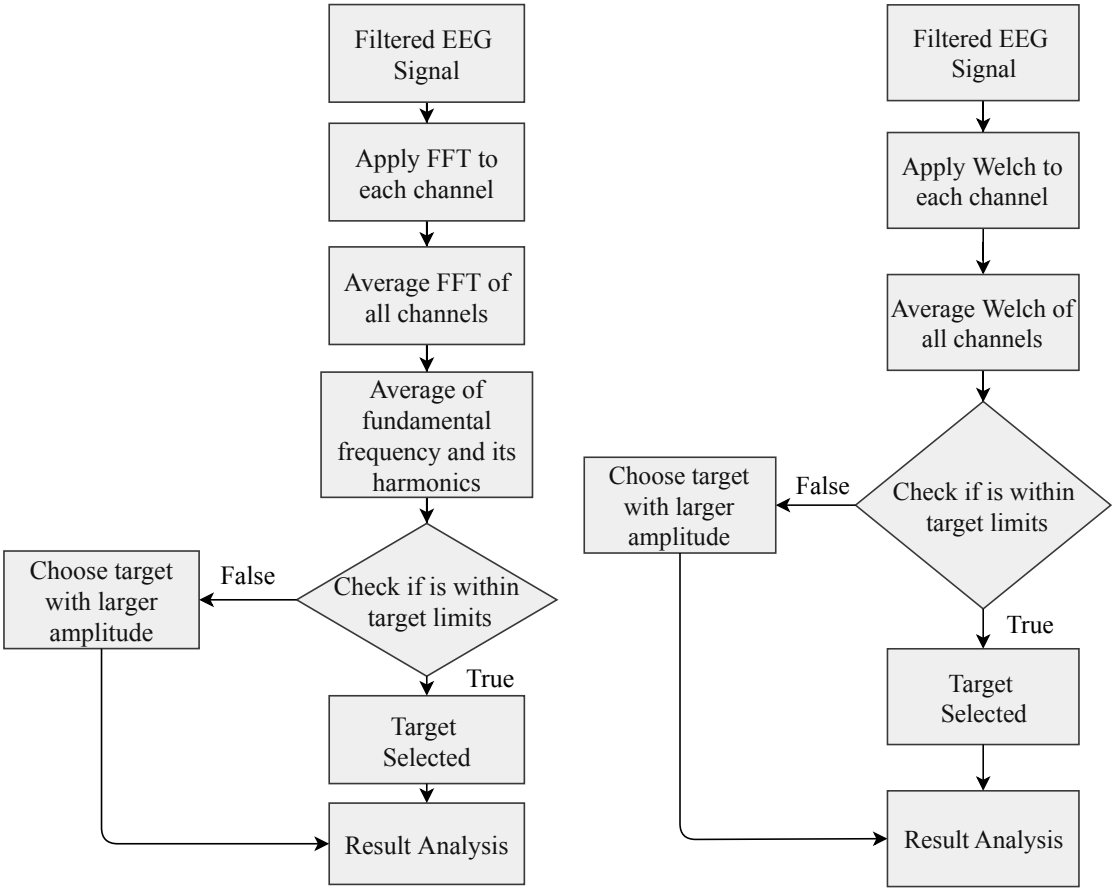


Figure 4.5: Frequency spectrum of SSVEPs responses to 8.0 Hz stimulation, measured at channel Oz. Fundamental frequency (8 Hz), second and third harmonics (16 and 24 Hz) are also clearly visible.



(a) FFT algorithm. **(b)** Welch algorithm.

Figure 4.6: Algorithms implemented in the offline simulation.

4.1.3.2 Welch's Method

Welch's method required the adjustment of the input data to maximize the spectral stability and resolution of the method, which was adjusted through window size and the number of blocks. Like the FFT, Welch's method had only 1250 points available, which greatly limits the effective use of this method.

For a window of 2048 points, the frequency resolution was 0.12 Hz ($\frac{Fs}{window} = \frac{250Hz}{2048points}$). Hence, no information would be lost, because the online benchmark dataset's frequency resolution is 0.2 Hz.

This solution could not be implemented, so the possible solution due to the online benchmark dataset size was to implement a window with 1024 points (equal to 4 seconds), which gave a frequency resolution of 0.244 Hz ($\frac{Fs}{window} = \frac{250Hz}{1024points}$). It was used an overlapping of 50%, which corresponded to 512 points.

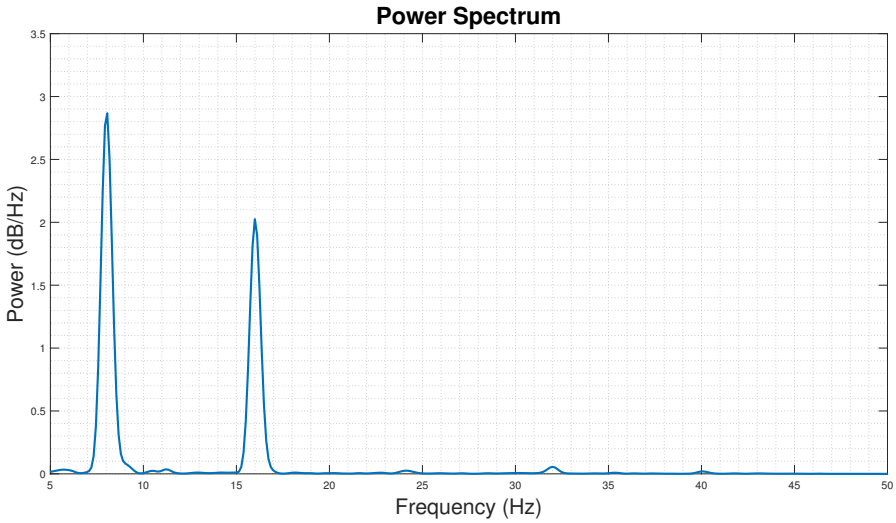


Figure 4.7: Power spectrum of SSVEPs responses at 8.0 Hz through Welch's estimation, measured at channel Oz.

4.1.3.3 CCA and IT-CCA

The CCA was applied to the filtered data and then, a target was extracted based on the CCA output. The first step was to create the reference signal for CCA. In the IT-CCA, it was used an individual template (64 channels $\times$ 1250 samples).

The reference signal was built using eq.3.5 from section 3.4.2.3. Due to the characteristics of the benchmark dataset, there were 40 reference signals, each one corresponding to the frequency of a single target (40 targets in total). So, these reference signals used 4 harmonics of each frequency and its size was 8 $\times$ 1250 samples (the first dimension comes from 4*2 harmonics). The size of the filtered EEG signal was 64 channels $\times$ 1250 samples.

The next step was to apply the CCA as shown in Fig.4.8. The output was a vector with 64 $\times$ 1 dimensions. The first value of this vector was selected and then, the procedure represented in Fig.4.8 was replicated for all the existing reference signals (40 signals in this case).

At last, the canonical correlation output ρ had a dimension of 1×40, where the maximum value corresponds to the target with a higher correlation with the reference signal.

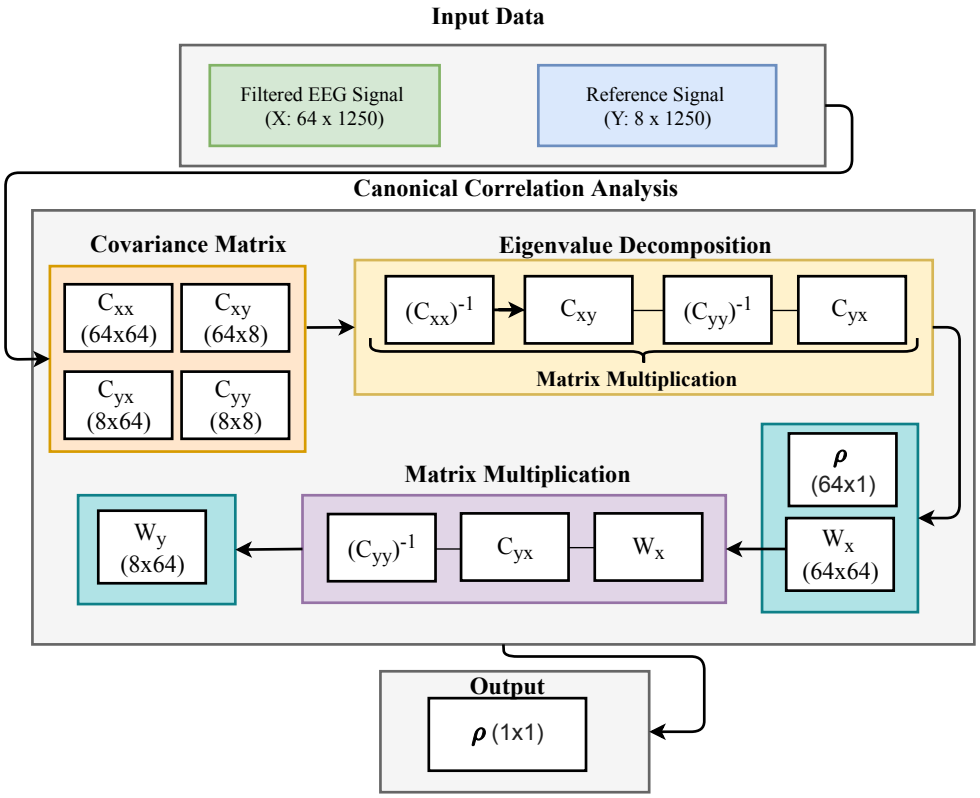


Figure 4.8: CCA algorithm implemented in the offline simulation.

4.1.3.4 CCA-Combinational

In this method, it was used the individual template of the IT-CCA and the reference signal of the Standard-CCA. The first step was to calculate the weights of each CCA implemented (same implementation in Fig.4.8), but here the output was the weight W_x (64×64).

Then, after finding the weights, they were multiplied by the EEG signal (X) and by the individual template ($\hat{X}$). Next step was to perform a Pearson's correlation to find the correlation vector r_n (eq.3.10). The output of each Pearson's correlation was a vector of 1×1 dimension.

Thus, all outputs from Pearson's correlations and from the CCA method were used in the eq. 3.11, that generated a vector with 1×1. This procedure was executed for all 40 frequencies of the individual template ($\hat{X}_k$) and the reference signal (Y_f). After all frequencies have been covered, the maximum value was extracted of the output vector (1×40). This maximum value corresponded to the selected target.

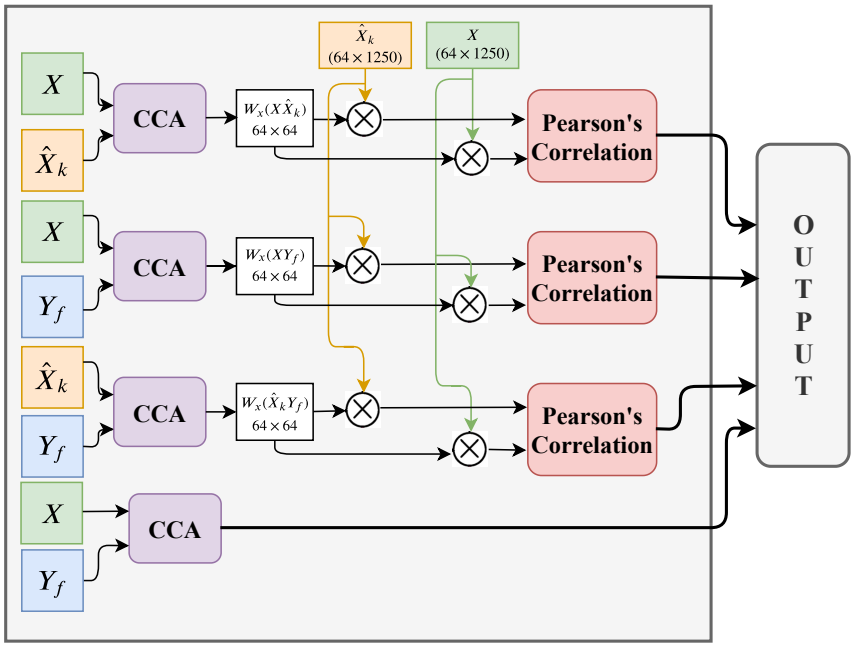


Figure 4.9: CCA algorithm implemented in the offline simulation.

4.1.3.5 CCA-Lite

CCA and its variants are computational demanding. Their implementation in low-computation microcontrollers require a more optimized algorithm for real-time to improve the target identification algorithm (CCA-Combinational) and turning it into an optimized algorithm for wearable devices, and CCA-Lite is a method that was proposed in [37] with that purpose.

The method was implemented in Matlab to analyse its effective computational benefit, in view of possible implementation in the OpenBCI microcontroller.

The high computational complexity of the CCA-Combinational can be solved with the optimizations methods proposed in [37]. These optimizations aim to reduce the amount of data processed, the memory requirement and computational complexity.

Two optimizations were incorporated in CCA-Lite. The first was a signal binarization, which consists of mapping the amplitude of SSVEP to +1 or -1 (see Fig.4.10).

Combinational CCA deals with a multi-bit EEG signal, multi-bit reference signal and multi-bit training set, which causes the high computational complexity and high memory requirement. This problem could be solved with the signal binarization.

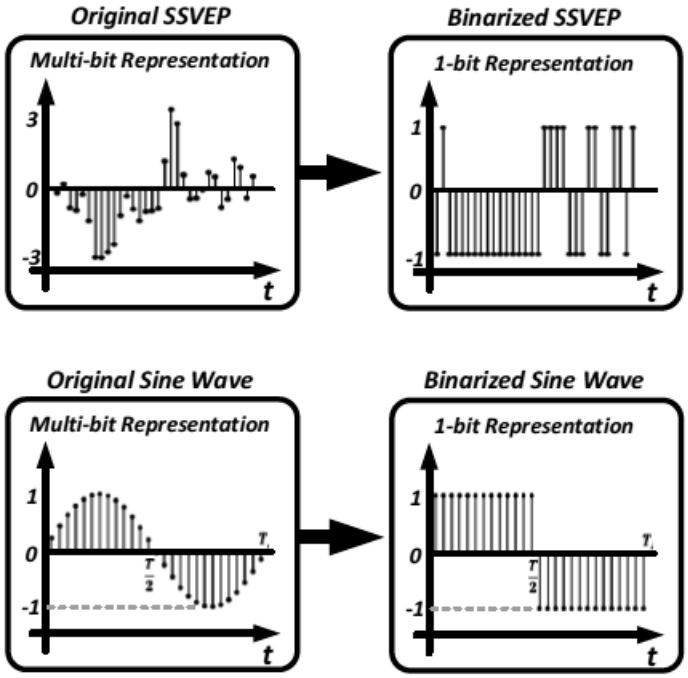


Figure 4.10: Example of signal binarization on SSVEP and reference sine signal (Fig. reproduced from [37])

The signal binarization could be applied to 3 datasets (the EEG signal, the reference sinusoidal signal and the training set signal). In order to maintain the features of each signal intact, the signal binarization was done only in the individual template dataset and in the reference signal dataset. This implementation had a higher accuracy with small memory requirement considering the other combinations possibles (signal binarization in all three datasets or only in the reference signal dataset, etc).

This happened because if this optimization was implemented in the SSVEP, much of its main features disappeared. Since the training data and the reference data were pre-stored in the memory and the signal binarization implemented in these two data sets reduced the memory requirement.

The second optimization was on-the-fly covariance calculation. The Standard-CCA covariance matrix was responsible for the correlation analysation between two datasets (eq.3.6). Two datasets, X and Y, composed an input matrix M. To apply the covariance in this matrix, it was needed to find the mean value of each signal (eq.3.6).

If the mean value of the matrix was close to zero when compared to the range of the amplitude, it was possible to receive time samples, without waiting for the last sample to calculate the mean value.

$$\begin{aligned} C_{ij} &= cov(M_i, M_j) \\ &= E[(M_i - E[M_i])(M_j - E[M_j])] \\ &= E[M_i M_j] - E[M_i]E[M_j] \end{aligned} \tag{4.3}$$

where C is the covariance matrix, C_{ij} is an element of covariance matrix C and M_i is the ith row vector of M.

If $E[M_i M_j] \gg E[M_i]E[M_j]$, then $E[M_i]E[M_j]$ can be omitted. Looking at (eq.4.3), it is possible to see that the covariance matrix consists of averaging the vector samples. In a real-time signal processing point of view, the system should wait for all the samples of X and Y signal to apply the covariance matrix, that is, the averaging of the vector samples cannot be captured until the end of the EEG recording.

So, the signal processing can only be done after the EEG recording has finished, but with this optimization, it was not required the subtraction operation and there was no need to wait to cease of the EEG recording, as represented in Fig.4.11. The BCI system performed the covariance matrix calculation while the SSVEP was still recording, improving the communication speed significantly [37]. After these optimizations, the algorithm implementation was similar to CCA-Comb.

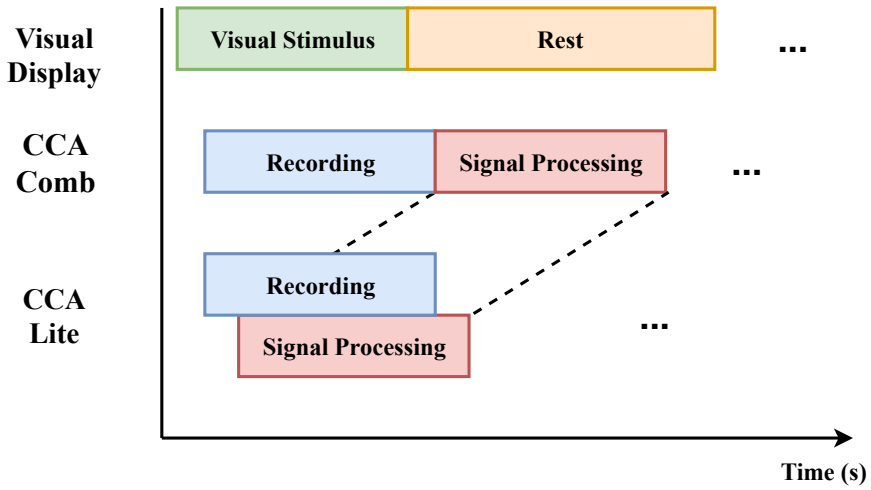


Figure 4.11: Advantage of On-the-fly Covariance calculation in CCA-Lite.

4.2 Real-Time Experimental Setup

This section details each aspect of the real-time experimental setup. It is composed of four modules, as we can see in Fig.4.12: the Stimulation Module, the Data Acquisition Module, the Signal Processing Module and the Domotic Appliance Module.

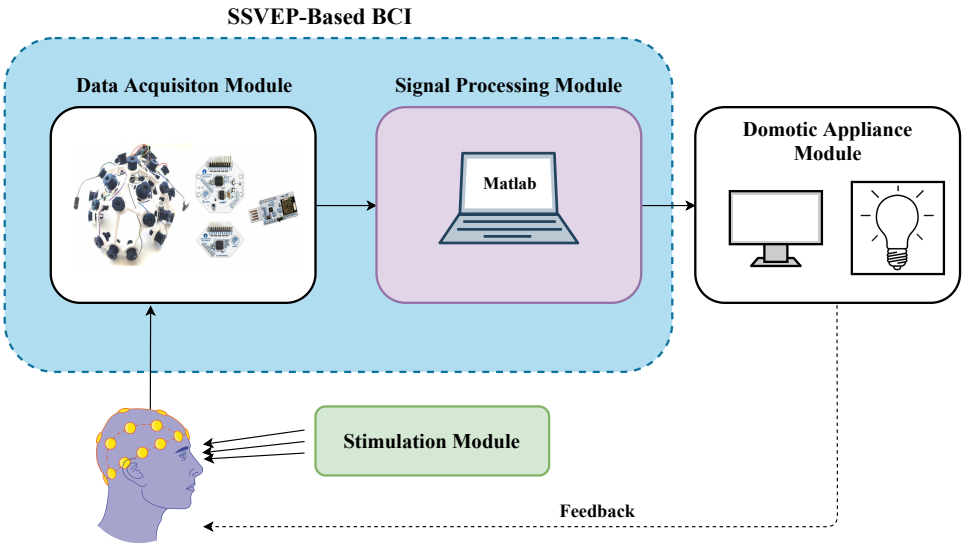


Figure 4.12: Real-time experimental setup.

4.2.1 Stimulation Module

The stimulation module was assembled with 4 LED matrices 8x8 (3 yellow and 1 blue) and it is controlled by an Arduino Nano. This module generates visual stimuli independent of the use of a computer screen. It is worth to mention that initially, a computer screen with checkerboard stimuli (and variants) was used to validate the existence of SSVEPs.

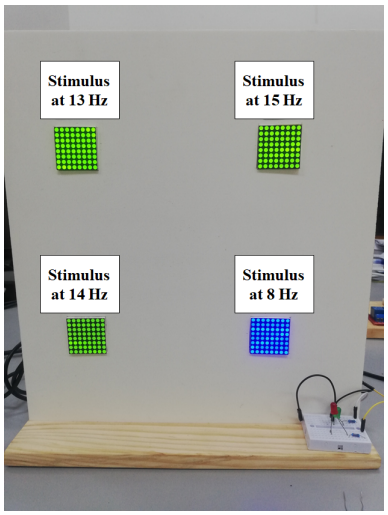


Figure 4.13: Stimulation based on LEDs matrix.

The LEDs flicker at 8 Hz, 13 Hz, 14 Hz and 15 Hz frequencies. The system has also 2 indication LEDs to know if the system is ON or OFF, and a button to start/stop the stimulation. The approach used in Arduino to control the LEDs was based on the following algorithm:

Algorithm 1: Algorithm of the control of the Stimulation.

```
unsigned long currentMillis = millis(); // current time
matrix[array].clear(); // clear matrix
interval = 1 / desired_frequency;
if ((currentMillis - previousMillis[array]) >= interval)
    previousMillis[array] = millis(); //stores the millis value in the selected array
    if (ledState[array] == LOW) // if the LED is off turn it on and vice-versa
        ledState[array] = HIGH;
         Turn ON
     else
        ledState[array] = LOW;
        Turn OFF
matrix[array].writeDisplay();
```

4.2.2 Online implementation of the SSVEP-BCI

Figure 4.15 shows the overall architecture and the data pipeline of the online implementation, including data acquisition, real-time processing and appliance control.

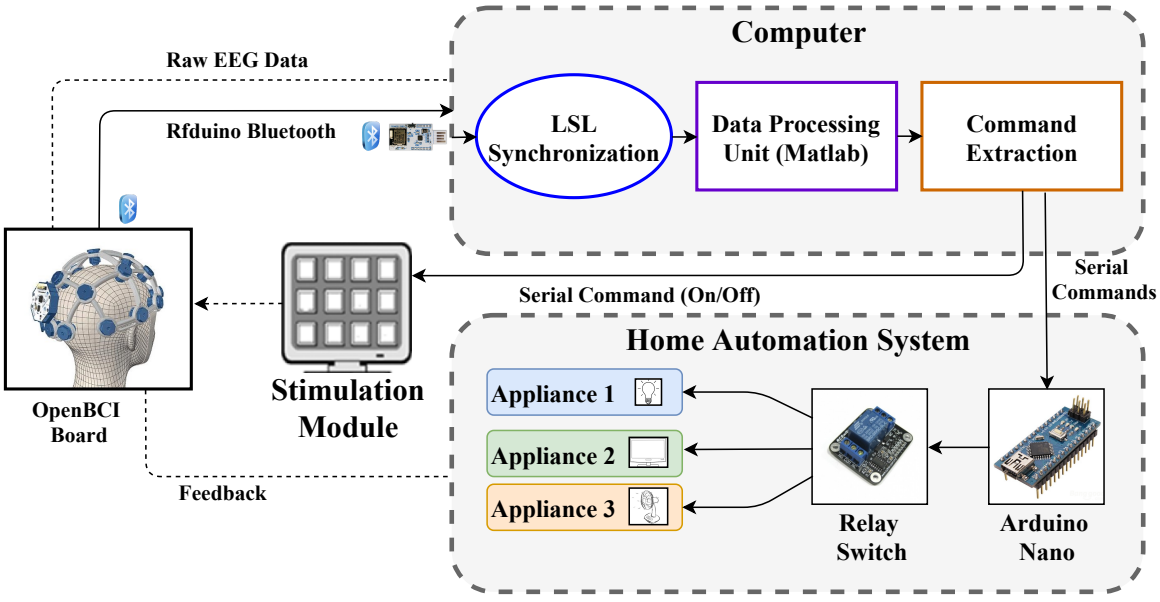


Figure 4.14: Experimental setup/architecture of real-time SSVEP based BCI.

The EEG acquisition was made by the OpenBCI 32 bit Cyton board. The electrodes were placed at O1, O2, PO3, PO4, PO7, PO8, P3 and P4 locations. The reference pin and the ground pin were placed at the 2 ear lobes. Although the OpenBCI electrodes were dry electrodes, we had to use a conductive gel(g.GAMMA gel) to improve the signal acquisition.

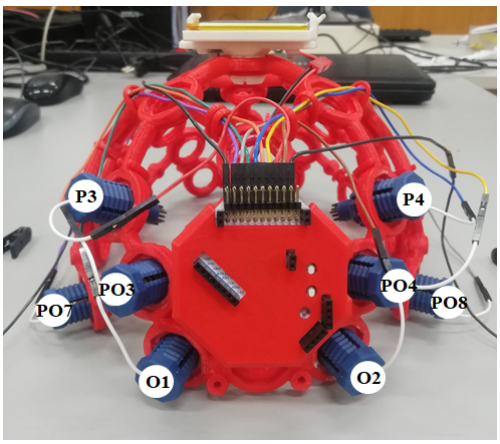


Figure 4.15: OpenBCI Electrodes locations.

The software framework was based on the lab streaming layer (LSL), handling networking, time-synchronization and near real-time access over different machines in the same network. The OpenBCI provided an interface of the LSL for Python, making possible to stream EEG data of the Ultracortex into Python interface. LSL transferred raw EEG data to Matlab, where signal processing algorithms occurred.

The EEG data were collected at a sampling rate of 250 Hz and transmitted by the Bluetooth module of the OpenBCI to the Rfduino Bluetooth plugged in the computer. A buffer was created, in order to receive the EEG data and maintaining the system in real-time. The buffer was filled with 2500 samples (equal to 10 seconds) for subsequent processing.

The buffer worked as a first in first out (FIFO) update, with a window of 50 samples (equal to 0.2 seconds). The buffer was constantly renovated, as shown in Fig.4.16. The signal processing algorithm is computed during one sample interval (4 ms).

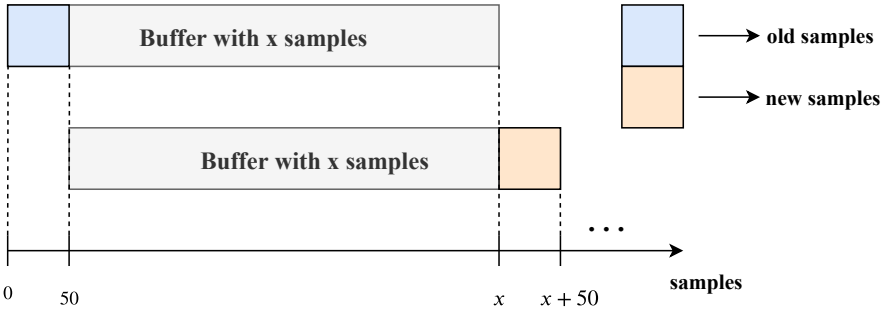


Figure 4.16: Implementation of the buffer.

In the real-time operation, two methods were implemented, the FFT approach described in section 3.4.2.1 and the CCA approach described in section 3.4.2.3, as they did not require any previous models.

After performing the pre-processing in the raw EEG data acquired, the FFT (with 4096 points) was computed for each channel taking the data stored in the buffer. Then, the FFT of each channel was subsequently averaged across channels.

In order to select the SSVEP frequency, the method verified if the higher absolute value calculated by FFT was in the boundary of a possible stimulus in an interval, as exemplified in Fig.4.17.

The target frequency was selected only if the amplitude was higher than a given threshold previously defined for each participant (based on a 10-second stimulation). In these boundaries, it was also verified which band had the largest sum of amplitude components. A successful detection only happened when the sum of the amplitude of the FFT components in the stimulus band was higher than its neighbour's bands (see Fig.4.17).

Thereby, the target detection had two verifications, yet a successful detection occurred if only one was verified.

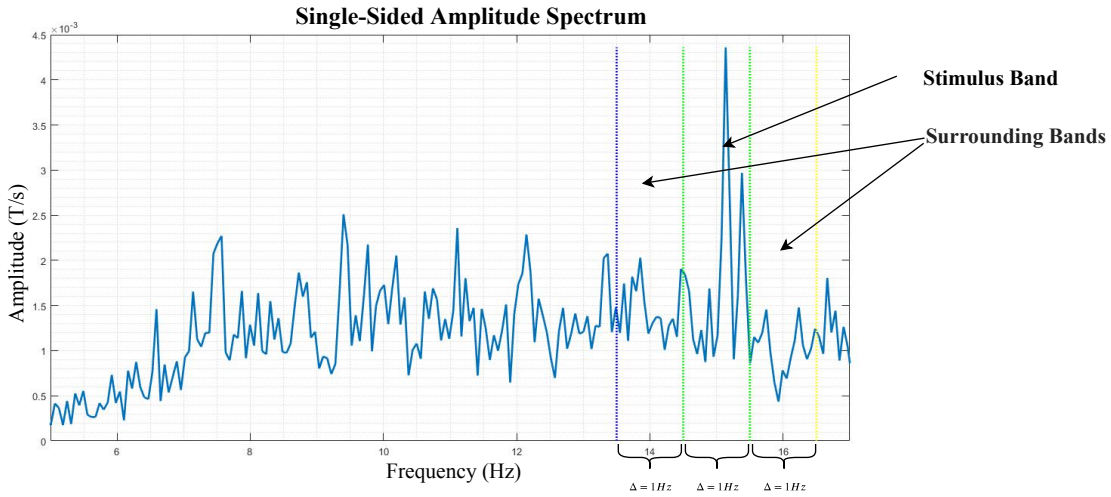


Figure 4.17: Target detection based on FFT approach. Example for a 15 Hz stimulus

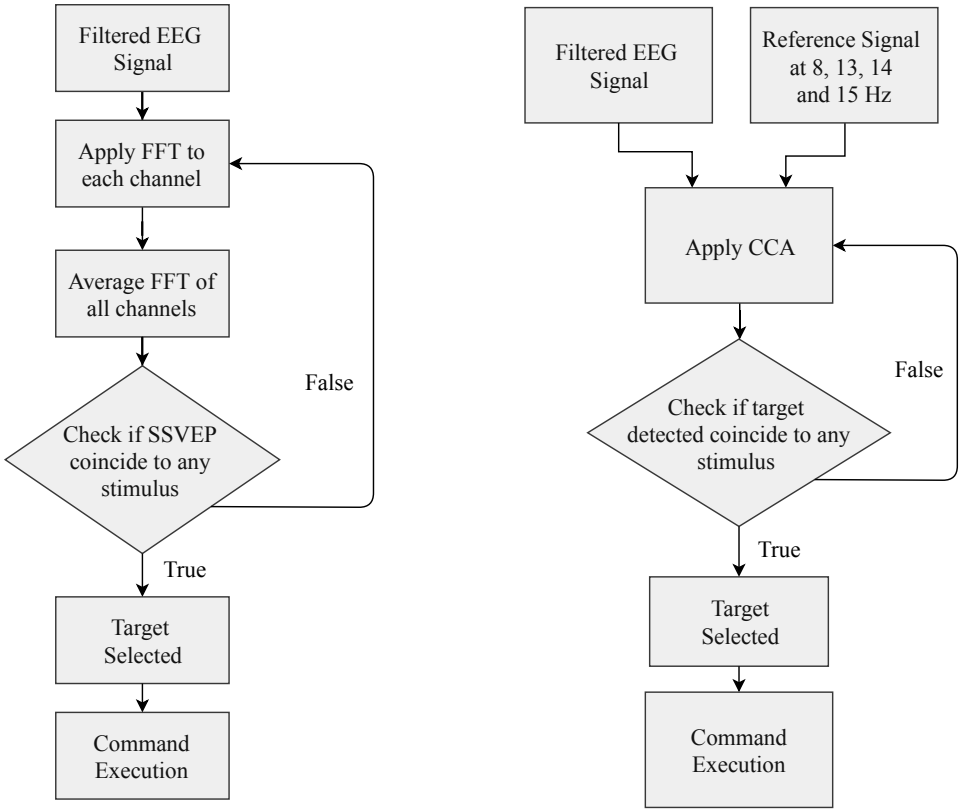
To have a successful target detection, 20 consecutive successful detections were required. When these 20 consecutive successful detections were achieved, a command was generated and the buffer was cleared, to avoid false positives in the next iterations. The algorithm is presented in Fig.4.18a.

The implementation of the CCA was heavier computationally than the FFT implementation. After performing the pre-processing in the raw EEG data, the Standard-CCA method was applied as explained in section 4.1.3.3. The reference signal had as target frequencies 8 Hz, 13 Hz, 14 Hz and 15 Hz, and it was calculated considering 3 harmonics, so the reference signal had a size of 6 $\times$ 2500 time samples.

In the real-time implementation, the filtered EEG had a size of 8×2500 time samples, because the number of channels was 8. The difference between the real-time implementation and the offline implementation was the number of channels being used and the number of targets. Based on Fig.4.8, the implementation was the same, only the input data was different. So, the ρ that was produced by the eigenvalue decomposition had a size of 8×1, as well as the weights of the spatial filters that changed size to 8×8 for W_x and 6×8 for W_y.

Subsequently, this process was repeated for all 4 reference signals and the canonical correlation output ρ had a dimension of 1×4, where the maximum value corresponded to the target with higher correlation with the reference signal.

After 20 consecutive successful detections, a command was sent to control the appliance and the buffer was cleared. The algorithm is presented in Fig.4.18b.



(a) FFT algorithm. **(b)** CCA algorithm.

Figure 4.18: Algorithms implemented in the real-time experience.

The command associated with the detected target was sent only if, in the Stimulation Module, the button on/off was activated. To turn the system on, the user has to focus on the 8 Hz stimulus and turn on the button. The purpose of an ON/OFF BCI switch was to ensure that there were no false positives occurred when the user did not want to send any command (this mode of operation was used here as a preliminary step to a full automatic switch ON/OFF). The state diagram of the operating mode is shown in Fig.4.19.

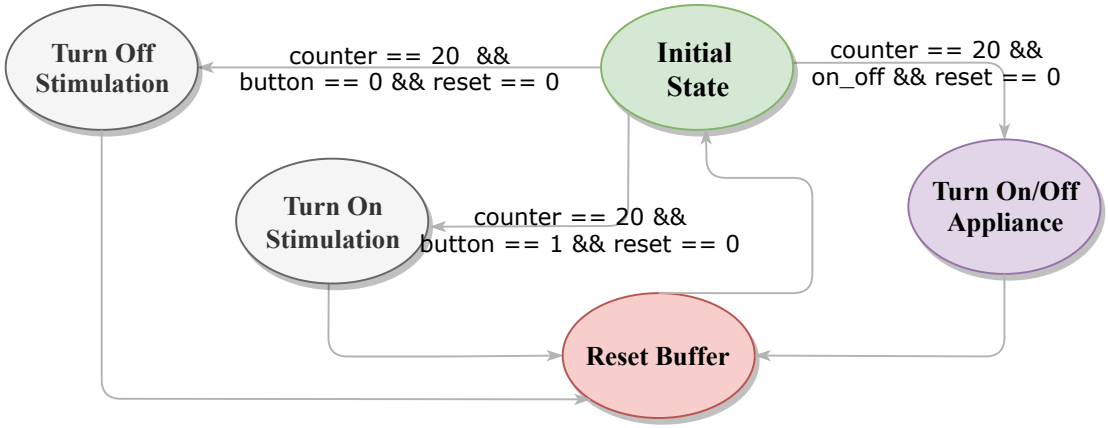


Figure 4.19: State Machine for the control of the domotic appliances.

4.2.3 Home appliances

The home appliances were switched ON/OFF by relays controlled by an Arduino Nano. This board, which is connected to 3 four-module relays, received the commands from the BCI system. These relays activated the device that was attached to it. The relays and domotic appliances are shown in Fig.4.20.

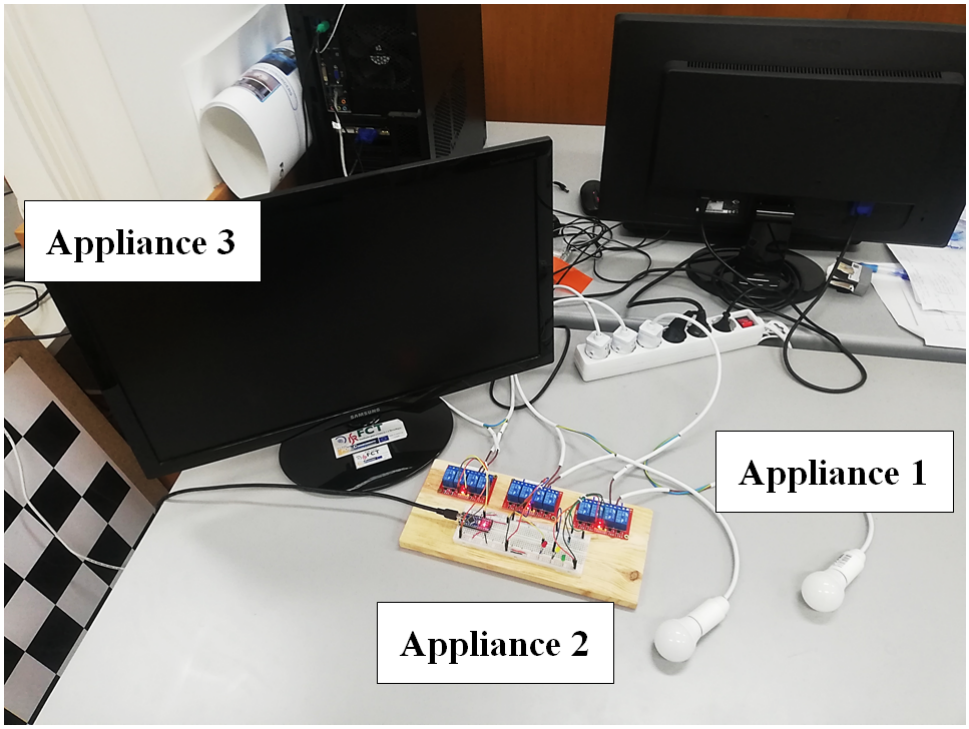


Figure 4.20: Domotic setup.

5

Results and Discussion

This chapter presents the offline results obtained with the benchmark dataset and with our own data. The online results were achieved with the SSVEP-BCI implementation to switch on/off home appliances. The accuracy of two signal processing methods proposed in the previous chapter are compared. First, all methods are tested and classified on the benchmark dataset. Then, EEG signals recorded from Gtec and OpenBCI acquisition systems are compared as well as two different visual stimulation approaches (LEDs and LCD). Finally, the results of the real-time experiment are shown.

5.1 Offline Results

5.1.1 Benchmark Dataset

In the first stage of the experiments, the online benchmark dataset [38] was tested with various feature extraction methods. The objective was to evaluate different feature extraction methods for SSVEP detection and decide on an optimal configuration.

5.1.1.1 Information transfer rate

Information transfer rate (ITR) is commonly applied to assess the performance of BCI systems in bit/min, which considers various factors such as classification accuracy P, average target selection time T (seconds/selection) and the number of target frequencies N_f (number of possible selected commands) [37]. It is calculated from:

$$ITR = log_2 \left(N_f.P^P.\left[\frac{1-P}{N_f-1}\right]^{1-P}\right) * \left(\frac{60}{T}\right) \tag{5.1}$$

For the online benchmark dataset, the target selection time T is 2.5 seconds (2.0 seconds of target gazing time plus 0.5 seconds of gaze shifting time for target identification).

5.1.1.2 Cross Validation

For the detection methods in which a statistical model was required (for example, the individual template in IT-CCA, CCA-Comb and CCA-Lite), the individual template was estimated using leave-one-out cross-validation, thereby ensuring that the results were never obtained from seen data, and providing a reliable metric.

In cross-validation, the data is split into train, validation and test set. The dataset was divided into 6 blocks. In each of the 6 rounds from each target (40 targets for the online benchmark dataset), cross-validation was performed using 5 blocks for training and 1 block for testing, as exemplified in Fig.5.1.

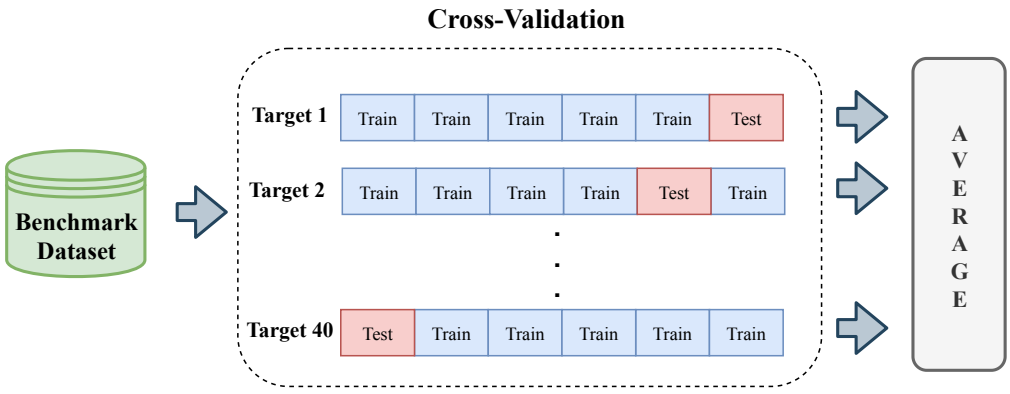


Figure 5.1: Implemention of cross-validation in CCA template-based scenarios. The SSVEP frequency detection was always based on unseen data.

5.1.1.3 Comparison of Feature Extraction algorithms

Six feature extraction methods were applied in the online benchmark dataset. Table 5.1 shows the classification accuracy and ITR of all six methods. For a direct comparison with our own collected data (using only 8 channels), the methods were applied considering the 64 channels of the dataset and only 8 channels (CB1, CB2, O1, O2, Oz, PO4, PO6 and PO8), location near the visual cortex.

For 64 channels, the method with better accuracy was the CCA-Comb with 99.14 %. It is possible to see that the methods with a training component (IT-CCA, CCA-Comb and CCA-Lite) had better classification results than FFT and Welch. It was also observed that the results were always better for 8 channels than for the 64 channels, except for CCA-Comb where the accuracy is almost the same (99.14 vs. 99.07).

Table 5.1: Classification results for the offline mode in the benchmark dataset (based on 5 second stimulation).

	64 channels		8 channels	
	Accuracy [%]	ITR [bpm]	Accuracy [%]	ITR [bpm]
FFT	76,26	86,28	90,00	107,11
Welch	26,06	20,18	41,01	35,28
Standard-CCA	86,45	102,14	93,95	115,10
IT-CCA	90,36	106,76	95,13	116,55
CCA-Comb	99,14	125,49	99,07	125,47
CCA-Lite	96,43	119,68	98,14	123,31

These results can be explained by several reasons, the selected channels are from the visual cortex, where mainly SSVEPs are evoked, additional electrodes provide irrelevant features with higher contamination (eye and muscular movement), and computational methods are more efficient for lower matrices dimensions. The particular low performance of Welch's method was due to the low number of data for the estimation.

One other thing that we wanted to compare was the amount of computation processing for each method. In Fig.5.2, it was compared the processing time of each features extraction method. The fastest method was the Welch Method. For the CCA approaches, CCA-Lite was the one that had a lower processing time, as expected, which was due to the optimizations applied in the CCA-Comb, as explained in section 4.1.3.5. The time processing reduced to 58% comparing to CCA-Comb. Although the implementation was on Matlab, it is expected similar reductions with C/C++ implementations. The processing time dropped drastically for the 8 channels experiments compared to the 64 channels experiments.

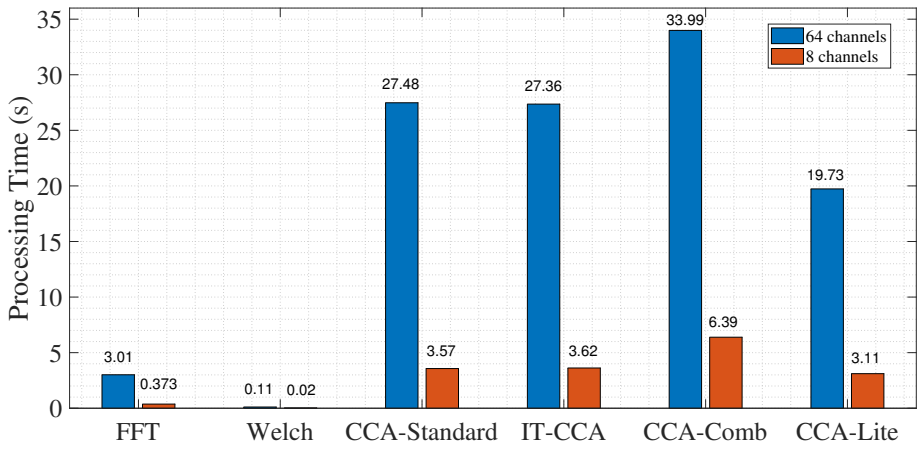


Figure 5.2: Processing time for the methods implemented offline on the benchmark dataset.

5.1.2 Own datasets

Several SSVEP experiments were performed to collect data with two acquisition systems: low-cost OpenBCI and gUSBamp, a research-grade equipment certified for clinical experiments. The data obtained with gUSBamp was recorded from 8 wet active electrodes placed at positions O1, O2, O8, O9, PO7, PO8, PO9 and PO10. The reference and the ground electrodes were placed at the 2 ear lobes. The OpenBCI setup was already explained in section 4.2. These two setups were used to compare the quality of the EEG signals recorded and to validate their effectiveness for SSVEP detection. Two tests were made with OpenBCI, using the dry electrodes with and without conductive gel (g.GAMMA gel) to analyse the influence of the gel on the quality of the signal.

Two different types of visual stimulation were used. The first stimulation was based on 4-target matrix presented on a 23.6-in LCD monitor with a resolution of 1920 × 1080 pixels at 60 Hz. The stimulus program was developed under MATLAB (MathWorks, Inc.) using the Psychophysics Toolbox (see Fig.5.3). The second type of visual stimulation was based on a LED matrix as presented in section 4.2.1.

Figure 5.3: Visual stimulation in a LCD screen.

The initial tests were made for stimulation at 15 Hz, and SSVEP data were analysed using the FFT. Figures 5.4a, 5.4b, and 5.4c show respectively the FFTs of data recorded at channel Oz with gUSBamp, OpenBCI with dry electrodes without a conductive gel, and OpenBCI with dry electrodes plus conductive gel (g.GAMMA gel).

The 15 Hz peak was traceable with gUSBamp and with OpenBCI with gel, but it was not visible when no gel was used. This suggests that the OpenBCI dry electrodes did not provide enough quality for SSVEP detection. The conductive gel seemed to decrease the electrode-skin impedance improving the signal quality.

The same tests were performed with a matrix LED stimulation, using the three acquisition approaches. Considering the figures 5.4d, 5.4e and 5.4f, the SSVEP only fails in the experiment using OpenBCI equipment with dry electrodes. The Gtec equipment produced a cleaner signal than OpenBCI equipment, but the OpenBCI with dry electrodes plus conductive gel (g.GAMMA gel) also filled the purpose of SSVEP detection.

The SNR was defined as the ratio of $y(f)$ to the mean value of 10 neighbouring frequencies (five frequencies on each side) [40]:

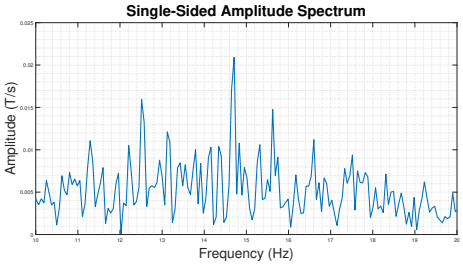
$$SNR = 20\log_{10}\frac{10 \times y(f)}{\sum_{k=1}^{5}[y(f - m \times k) + y(f + m \times k)]} \tag{5.2}$$

where the amplitude spectrum $y(f)$ is calculated by FFT and m is the frequency resolution.

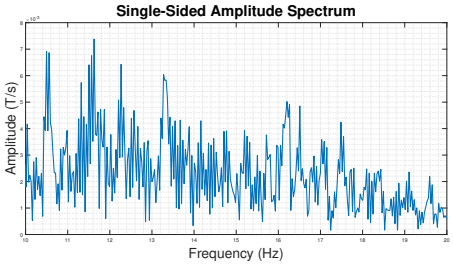
The signal quality of these data acquisition systems can be inferred from the SNR measured from eq.5.2, as shown in Table 5.2. From this table, we can conclude that stimulation based on the LCD screen and the setup with gUSBamp were the most effective, concerning the quality of the EEG signal and the detection of SSVEP.

Table 5.2: SNR of each EEG recorded test.

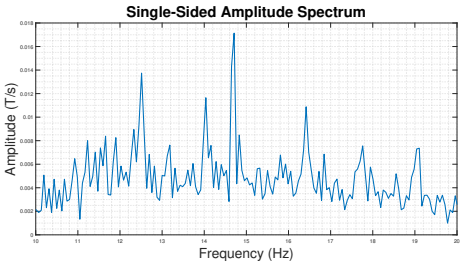
	LCD			LED		
	Gtec	OpenBCI		Gtec	OpenBCI	
		dry	wet		dry	wet
SNR (dB)	21,56	14,39	16,26	20,03	14,89	18,02



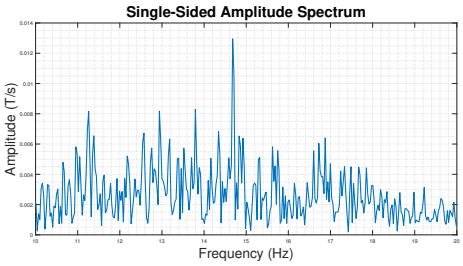
(a) EEG recorded with gUSBamp using a LCD screen.



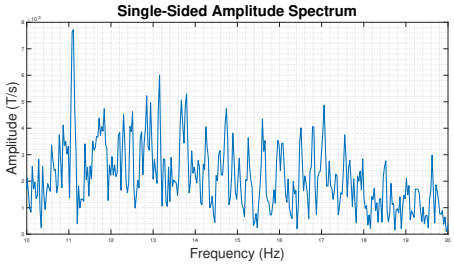
(b) EEG recorded with OpenBCI board and dry electrodes using a LCD screen.



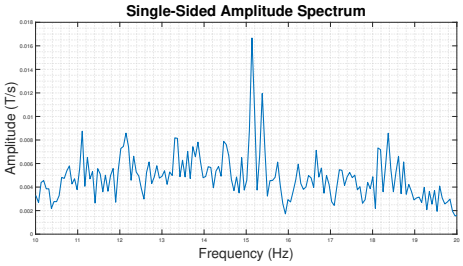
(c) EEG recorded with OpenBCI board and wet electrodes using a LCD screen.



(d) EEG recorded with gUSBamp using a LED matrix.



(e) EEG recorded with OpenBCI board and dry electrodes using a LED matrix.



(f) EEG recorded with OpenBCI board and wet electrodes using a LED matrix.

Figure 5.4: FFT of EEG recorded at channel Oz. Visual stimulus at 15 Hz.

Comparing the stimulation in the LCD screen and LED matrix, no differences were detected in the offline implementation.

5.2 Online Results

Five subjects participated in the SSVEP-BCI online experiments. Each participant performed 5 trials, consisting of 10 commands each (Table 5.3). There were 4 targets, each one with a different frequency. The stimulus at 8 Hz corresponded to the turned on the system. Then, there were three more stimuli at 13 Hz, 14 Hz and 15 Hz, each one corresponding to a given command that controlled a specific appliance. Therewas a rest period of 20 seconds between commands.

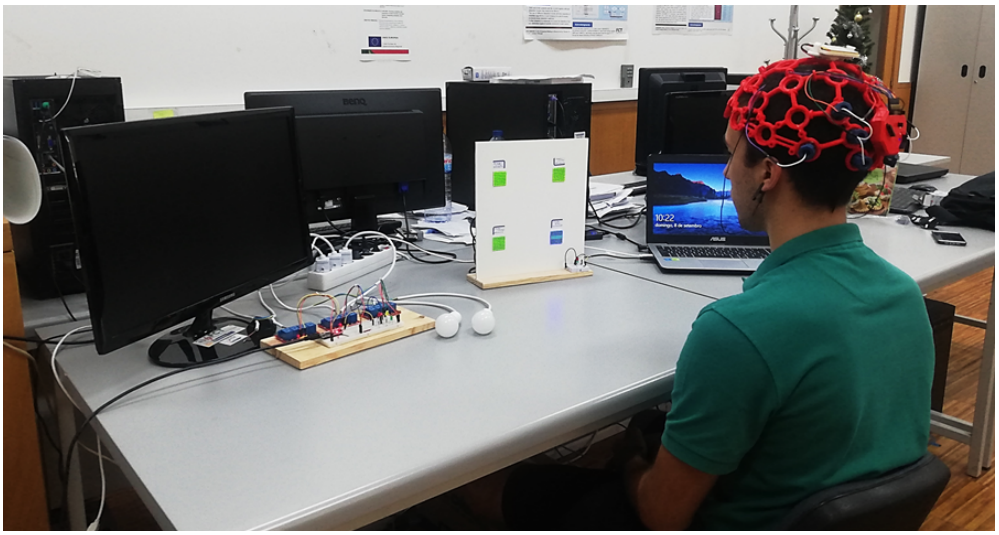


Figure 5.5: Controlling the SSVEP-based BCI system.

Table 5.3: Characteristics of of participants who attended online experiments.

Subject	1	2	3	4	5
Gender	Male	Female	Male	Male	Female
Age	50	24	22	23	52
BCI experience	experienced	naïve	experienced	naïve	experienced

Table 5.4 shows the accuracy using the FFT approach, referring to the size of the buffer used. It was clear that the buffer with 10 seconds offered a better SSVEP detection for the FFT algorithm.

Table 5.4: Online resuts of the SSVEP-BCI using the FFT detection approach (section 4.2.2).

Subject	Buffer 5s Accuracy [%]	Buffer 6s Accuracy [%]	Buffer 7s Accuracy [%]	Buffer 8s Accuracy [%]	Buffer 9s Accuracy [%]	Buffer 10s Accuracy [%]
#1	50,0	58,0	62,0	64,0	70,0	78,0
#2	50,0	57,0	61,0	65,0	72,0	69,0
#3	64,0	64,0	68,0	73,0	73,0	76,0
#4	56,0	58,0	60,0	68,0	73,0	77,0
#5	61,0	58,0	62,0	64,0	71,0	78,0
Mean	56,2	59,0	62,6	66,8	71,8	**75,6**
SD	6,34	2,83	3,13	3,83	1,30	3,78

Using the Standard-CCA implementation, the 5 s buffer was the one yielding the best accuracies, as shown in Table 5.5. A buffer with 4 s was not used, because for a successful command was needed 20 consecutive successful detection, which it was equivalent to 4 s, so a buffer with this size could not be implemented in Standard-CCA implementation.

Table 5.5: Online resuts of the SSVEP-BCI using the Standard-CCA detection approach (section 4.2.2).

Subject	Buffer 5s Accuracy [%]	Buffer 6s Accuracy [%]	Buffer 7s Accuracy [%]	Buffer 8s Accuracy [%]	Buffer 9s Accuracy [%]	Buffer 10s Accuracy [%]
#1	87,0	81,0	80,0	76,0	74,0	71,0
#2	82,0	83,0	83,0	77,0	75,0	70,0
#3	86,0	84,0	79,0	75,0	74,0	72,0
#4	84,0	81,0	81,0	79,0	73,0	71,0
#5	85,0	82,0	82,0	78,0	74,0	72,0
Mean	**84,8**	82,2	81,0	77,0	74,0	71,2
SD	1,92	1,30	1,58	1,58	0,71	0,84

Comparing table 5.4 and 5.5, the Standard-CCA had a higher accuracy in the online implementation. It was also noticeable that in the Standard-CCA, there was a better accuracy for the buffers with smaller sizes. On the contrary, in the FFT the buffers with larger sizes had better accuracies.

6

Conclusion and Future Work

This dissertation presented several feature extraction methods for SSVEP detection, assessed offline in benchmark datasets and online on an OpenBCI framework. Analysing the results obtained in the offline experiment, it could be verified that the more complex feature extraction method (CCA-Comb) was the one which had better results. It was also compared to the processing time of each method and the relation between the data and each method used. This analysis gave a thorough insight into the capability of each method. It was possible to see that the feature extraction methods that were more computationally complex were more demanding and took more time to process the EEG signal. From the application of the CCA-Lite method, it was possible to conclude that it is a much lighter method in terms of memory requirement in comparison to CCA-Comb.

Regarding the visual stimulation, it was concluded that it is possible to detect SSVEP through LED matrix stimulation. For the signal acquisition, it was a side back from the OpenBCI equipment. The signal acquired by OpenBCI was too noisy, much because of its dry electrodes. When a conductive gel was used in OpenBCI electrodes, the signal increased the quality and it was possible to perform an effective SSVEP detection.

For the implementation in the online experiment, the feature extraction methods used were the FFT and Standard-CCA and the tests done gave empirical evidence that the system reasonable performance, which achieved better accuracy in the Standard-CCA.

It is presented as proposals for future work:

- Implement the signal processing algorithm in the OpenBCI microcontroller, to have a fully stand-alone system;
- Implement more feature extraction methods in the real-time experiment;
- Implement a machine learning algorithm to improve the performance of each feature extraction method;
- Use a larger group of participants to validate more effectively the implemented approaches;
- Improve the functionalities of the current version of the BCI system.

Bibliography

[1] Shih, Jerry J and Krusienski, Dean J and Wolpaw, Jonathan R, "Brain-computer interfaces in medicine," *Mayo Clinic Proceedings*, Elsevier, 2012, pp. 268-279.

[2] Wolpaw, Jonathan R and Birbaumer, Niels and McFarland, Dennis J and Pfurtscheller, Gert and Vaughan, Theresa M,"Brain–computer interfaces for communication and control," *Clinical neurophysiology*, Vol. 113, No. 6, 767–791,(2002). Elsevier.

[3] Graimann, Bernhard and Allison, Brendan and Pfurtscheller, Gert, "Brain-computer interfaces: A gentle introduction," *Brain-computer interfaces*, Springer, 2009, pp. 1-27.

[4] Chu, Narisa NY. *Brain-Computer Interface Technology and Development: The emergence of imprecise brainwave headsets in the commercial world* IEEE Consumer Electronics Magazine 4.3 (2015): 34-41.

[5] Maskeliunas, Rytis, et al. *Consumer-grade EEG devices: are they usable for control tasks?*. PeerJ 4 (2016): e1746.

[6] Larsen, Erik Andreas. *Classification of EEG signals in a brain-computer interface system*. Institutt for datateknikk og informasjonsvitenskap, 2011.

[7] Samson, V. R. R., et al. *Electroencephalogram-Based OpenBCI Devices for Disabled People*. Proceedings of 2nd International Conference on Micro-Electronics, Electromagnetics and Telecommunications. Springer, Singapore, 2018.

[8] Hinterberger, Thilo and Weiskopf, Nikolaus and Veit, Ralf and Wilhelm, Barbara and Betta, Elena and Birbaumer, Niels,"An EEG-driven brain-computer interface combined with functional magnetic resonance imaging (fMRI)," *IEEE Transactions on Biomedical Engineering*, Vol. 51, No. 6, 971–974,(2004). IEEE.

[9] Duncan, Connie C and Barry, Robert J and Connolly, John F and Fischer, Catherine and Michie, Patricia T and N*etäl*, Vol. Event-related potentials in clinical research: guidelines for eliciting, recording, and quantifying mismatch negativity, P300, and N400, No. Clinical Neurophysiology, 120,(11). 1883–1908.

[10] Muller-Putz, Gernot R and Pfurtscheller, Gert,“Control of an electrical prosthesis with an SSVEP-based BCI,” *IEEE Transactions on Biomedical Engineering*, Vol. 55, No. 1, 361–364,(2007). IEEE.

[11] Jia, Chuan and Gao, Xiaorong and Hong, Bo and Gao, Shangkai,“Frequency and phase mixed coding in SSVEP-based brain–computer interface,” *IEEE Transactions on Biomedical Engineering*, Vol. 58, No. 1, 200–206,(2010). IEEE.

[12] Regan, David,“Steady-state evoked potentials,” *JOSA*, Vol. 67, No. 11, 1475–1489,(1977). Optical Society of America.

[13] Wang, Yijun and Gao, Xiaorong and Hong, Bo and Gao, Shangkai, ”Practical designs of brain–computer interfaces based on the modulation of EEG rhythms,” *Brain-Computer Interfaces*, Springer, 2009, pp. 137-154.

[14] Ruhunage, Isuru and Perera, Chamika Janith and Munasinghe, Induwara and Lalitharatne, Thilina Dulantha, ”EEG-SSVEP based Brain Machine Interface for Controlling of a Wheelchair and Home Appliances with Bluetooth Localization System,” *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, IEEE, 2018, pp. 2520–2525.

[15] Alvarado-Díaz, Witman and Meneses-Claudio, Brian and Roman-Gonzalez, Avid, ”Implementation of a brain-machine interface for controlling a wheelchair,” *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, IEEE, 2017, pp. 1–6.

[16] Perera, Chamika Janith and Naotunna, Isira and Sadaruwan, Chameera and Gopura, Ranathunga Arachchilage Ruwan Chandra and Lalitharatne, Thilina Dulantha, ”SSVEP based BMI for a meal assistance robot,” *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, 2016, pp. 2295–2300.

[17] Shivappa, Vinay Kumar Karigar and Luu, Brian and Solis, Marco and George, Kiran, ”Home automation system using brain computer interface paradigm based on auditory selection attention,” *2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, IEEE, 2018, pp. 1–6.

[18] Lin, Yuan-Pin and Wang, Yijun and Jung, Tzyy-Ping, ”Assessing the feasibility of online SSVEP decoding in human walking using a consumer EEG headset,” *Journal of neuroengineering and rehabilitation*, BioMed Central, 2014, pp. 119–127.

[19] Jia, Chuan and Gao, Xiaorong and Hong, Bo and Gao, Shangkai, ”Frequency and phase mixed coding in SSVEP-based brain–computer interface,” *IEEE Transactions on Biomedical Engineering*, IEEE, 2010, pp. 200–206.

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

[21] OpenBCI site,
`https://openbci.com/`

[22] Gramann, Klaus and Ferris, Daniel and Gwin, Joseph and Makeig, Scott, "Imaging Natural Cognition in Action.," *International journal of psychophysiology : official journal of the International Organization of Psychophysiology*, Vol. 91, ,(09 2013).

[23] Wang, Yijun and Wang, Ruiping and Gao, Xiaorong and Hong, Bo and Gao, Shangkai,"A practical VEP-based brain-computer interface," *IEEE Transactions on neural systems and rehabilitation engineering*, Vol. 14, No. 2, 234–240,(2006). IEEE.

[24] Bigdely-Shamlo, Nima and Mullen, Tim and Kothe, Christian and Su, Kyung-Min and Robbins, Kay A,"The PREP pipeline: standardized preprocessing for large-scale EEG analysis," *Frontiers in neuroinformatics*, Vol. 9, 16,(2015). Frontiers.

[25] Liu, Quan and Chen, Kun and Ai, Qingsong and Xie, Sheng Quan,"recent development of signal processing algorithms for SSVEP-based brain computer interfaces," *Journal of Medical and Biological Engineering*, Vol. 34, No. 4, 299–309,(2014)

[26] Welch, Peter,"The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms," *IEEE Transactions on audio and electroacoustics*, Vol. 15, No. 2, 70–73,(1967). IEEE.

[27] Bin, Guangyu and Gao, Xiaorong and Yan, Zheng and Hong, Bo and Gao, Shangkai,"An online multi-channel SSVEP-based brain–computer interface using a canonical correlation analysis method," *Journal of neural engineering*, Vol. 6, No. 4, 046002,(2009). IOP Publishing.

[28] Zhang, Zhuo, et al. *Spectrum and Phase Adaptive CCA for SSVEP-based Brain Computer Interface*. 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, 2018.

[29] Zhang, Nannan, Yadong Liu, and Zongtan Zhou. *A SSVEP-BCI with random moving stimuli in simulation environment.* Brain-Computer Interface (BCI), 2017 5th International Winter Conference on. IEEE, 2017.

[30] Bin, Guangyu and Gao, Xiaorong and Wang, Yijun and Li, Yun and Hong, Bo and Gao, Shangkai,"A high-speed BCI based on code modulation VEP," *Journal of neural engineering*, Vol. 8, No. 2, 025015,(2011). IOP Publishing.

[31] Wang, Yijun and Nakanishi, Masaki and Wang, Yu-Te and Jung, Tzyy-Ping,"Enhancing detection of steady-state visual evoked potentials using individual training data," *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, IEEE, 2014, pp. 3037–3040.

[32] Nakanishi, Masaki and Wang, Yijun and Wang, Yu-Te and Mitsukura, Yasue and Jung, Tzyy-Ping,"A high-speed brain speller using steady-state visual evoked potentials," *International journal of neural systems*, Vol. 24, No. 06, 1450019,(2014). World Scientific.

[33] Nakanishi, Masaki and Wang, Yijun and Wang, Yu-Te and Jung, Tzyy-Ping,“A comparison study of canonical correlation analysis based methods for detecting steady-state visual evoked potentials,” *PloS one*, Vol. 10, No. 10, e0140703,(2015). Public Library of Science.

[34] Chen, Xiaogang and Wang, Yijun and Nakanishi, Masaki and Gao, Xiaorong and Jung, Tzyy-Ping and Gao, Shangkai,“High-speed spelling with a noninvasive brain–computer interface,” *Proceedings of the national academy of sciences*, Vol. 112, No. 44, E6058–E6067,(2015). National Acad Sciences.

[35] Nakanishi, Masaki and Wang, Yijun and Wang, Yu-Te and Mitsukura, Yasue and Jung, Tzyy-Ping,“A high-speed brain speller using steady-state visual evoked potentials,” *International journal of neural systems*, Vol. 24, No. 06, 1450019,(2014). World Scientific.

[36] Wang, Yijun and Nakanishi, Masaki and Wang, Yu-Te and Jung, Tzyy-Ping,“Enhancing detection of steady-state visual evoked potentials using individual training data,” *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp.3037-3040 (2014)IEEE.

[37] Kim, Dokyun and Byun, Wooseok and Ku, Yunseo and Kim, Ji-Hoon,“High-Speed Visual Target Identification for Low-Cost Wearable Brain-Computer Interfaces,” *IEEE Access*, Vol. 7, 55169–55179,(2019). IEEE.

[38] Wang, Yijun and Chen, Xiaogang and Gao, Xiaorong and Gao, Shangkai,“A benchmark dataset for SSVEP-based brain–computer interfaces,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 25, No. 10, 1746–1752,(2016). IEEE.

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

[40] Wang, Yijun and Wang, Ruiping and Gao, Xiaorong and Hong, Bo and Gao, Shangkai,“A practical VEP-based brain-computer interface,” *IEEE Transactions on neural systems and rehabilitation engineering*, Vol. 14, No. 2, 234–240,(2006). IEEE.