



HAL
open science

Commande d'humanoïdes robotiques ou avatars à partir d'interface cerveau-ordinateur

Pierre Gergondet

► **To cite this version:**

Pierre Gergondet. Commande d'humanoïdes robotiques ou avatars à partir d'interface cerveau-ordinateur. Automatique / Robotique. Université Montpellier II - Sciences et Techniques du Languedoc, 2014. Français. NNT : 2014MON20134 . tel-01662486

HAL Id: tel-01662486

<https://theses.hal.science/tel-01662486v1>

Submitted on 13 Dec 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ MONTPELLIER 2
SCIENCES ET TECHNIQUES DU LANGUEDOC

THÈSE

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ MONTPELLIER 2

Discipline : SYAM – Systèmes Automatiques et Microélectroniques

École Doctorale : I2S – Information, Structures, Systèmes

présentée par

Pierre GERGONDET

Équipes d'accueil : LABORATOIRE D'INFORMATIQUE, DE ROBOTIQUE
ET DE MICROÉLECTRONIQUE DE MONTPELLIER
CNRS-AIST JRL, UMI3218/CRT (TSUKUBA, JAPON)

Titre :

**Commande d'humanoïdes robotiques ou avatars
à partir d'interface cerveau-ordinateur**

Présentée et soutenue publiquement le 19 décembre 2014

JURY

Etienne	BURDET	Rapporteurs
Gordon	CHENG	
Tetiana	AKSENOVA	Examineurs
Philippe	FRAISSE	
Abderrahmane	KHEDDAR	Directeur de thèse

Contents

Introduction	7
1 Brain-Computer Interface as a Control Interface: State of the Art	13
1.1 Acquisition technologies	14
1.1.1 Invasive solutions	15
1.1.2 Functional Near Infrared Spectroscopy (fNIRS)	16
1.1.3 Functional Magnetic Resonance Imaging (fMRI)	17
1.1.4 Electro-encephalography (EEG)	17
1.1.5 Summary	18
1.2 Features used in EEG-based acquisition	19
1.2.1 Motor Imagery	19
1.2.2 Event Related Potentials: P300	20
1.2.3 Steady-States Visually Evoked Potentials (SSVEP)	22
1.2.4 The Error Potential (ErrP)	26
1.3 BCI Applications	27
1.3.1 BCI-based spelling devices	27
1.3.2 Entertainment, Games and Virtual Reality Applications	29
1.3.3 Robotic Applications	33
1.4 Conclusion	46
1.4.1 Limitations of menu-based control	47
1.4.2 Limitations of NF-based control	47
1.4.3 Introducing “object-centric” BMI	48
2 Building Interface for Robot Control with SSVEP	51
2.1 Introducing the bci-interface: A Framework for Dynamic BCI Application .	52
2.2 Dynamic Background and SSVEP Stimuli Interactions	53
2.2.1 Motivation	53
2.2.2 Material and Methods	55
2.2.3 Experimental Results	57
2.2.4 Discussion	58
2.2.5 Conclusion	59

2.3	SSVEP stimuli design for object-centric BCI	60
2.3.1	Material and methods	61
2.3.2	Pilot study – Results and discussion	67
2.3.3	SSVEP display optimization – Results and discussion	70
2.3.4	SSVEP Display Optimization: Conclusion	82
2.4	Conclusion	83
3	Whole-Body Control with Steady-State Visually Evoked Potentials (SSVEP)	85
3.1	Introducing Object-Centric BMI	86
3.1.1	From Cartesian-Motion to Task-Space Control	86
3.1.2	From menu-based BMI to object-centric BMI	87
3.2	A framework for object-centric BMI	88
3.2.1	Overview	88
3.2.2	BCI Module	88
3.2.3	SSVEP Interface Module	90
3.2.4	Objects and Affordance Extraction Module	91
3.2.5	Task selection module	91
3.3	SSVEP Interface Module for Object-Centric BMI	91
3.3.1	Recursive and Enforced SSVEP Selection	92
3.3.2	State Transition in BMI	92
3.4	Objects and Affordance Extraction Module	93
3.4.1	Objects Recognition - 2D Recognition: Algorithm	94
3.4.2	Objects Recognition - 2D Recognition: Shape Extraction	95
3.4.3	Objects Recognition - 3D Recognition	95
3.4.4	Robot localization in indoor environments	95
3.5	Implementation of the Task Selection Module	96
3.5.1	Definition	97
3.5.2	Using a Bayesian network as a Task Selection Module	97
3.5.3	Improving the network	98
3.6	Object-oriented Robot Control	99
3.6.1	Walking with a Humanoid Robot in BMI	99
3.6.2	Navigation assistance for humanoid control through BMI	100
3.6.3	Object soft-drop with guarded motion	103
3.6.4	Action Cancellation	104
3.7	User Experiments	106
3.7.1	Initial User Experiment	107
3.7.2	Assessing the benefits of Navigation Assistance	109
3.7.3	A Bayesian-network assisted BCI for Humanoid Robot Control	112
3.8	Conclusion	115

4	Humanoid Control with a Brain-Computer Interface: Applications	117
4.1	Studying the Impact of Sound in Embodiment Experience	118
4.1.1	Material and Methods	119
4.1.2	Hardware and software integration	121
4.1.3	Results	122
4.1.4	Discussion	126
4.1.5	Conclusion	127
4.2	Impact of Sound in Embodiment Experience with a Strong Visual Component	129
4.2.1	Introduction	129
4.2.2	Participants	130
4.2.3	Methods	130
4.2.4	Data Analysis: Current Results	131
4.2.5	Discussion and Perspectives	134
4.3	Conclusion	136
5	ECoG-based BCI using a humanoid robot	137
5.1	An introduction to ECoG-based BCI	138
5.1.1	ECoG technology	138
5.1.2	ECoG-based BCI: the CSP method	138
5.2	One degree of freedom ECoG-based BCI	139
5.3	Two degrees of freedom ECoG-based BCI	141
5.3.1	Patient implantation	142
5.3.2	Results of Brain Mapping	142
5.3.3	A synchronized motor-imagery paradigm	144
5.3.4	Experiments	146
5.3.5	Results	146
5.4	Challenges of ECoG-based BCI	147
5.4.1	Problems induced by the surgery	147
5.4.2	Experimenting within the therapeutic process	147
5.5	Perspectives and Conclusion	148
	Conclusion and Perspectives	151
	Bibliography	155
	Appendices	167
A	A Framework for Dynamic BCI Application: bci-interface	169
A.1	Motivation and Goals	169
A.2	Command Model in bci-interface	170
A.3	Framework Elements	170
A.3.1	BCIInterface	170
A.3.2	EventHandler	171
A.3.3	Background	171

A.3.4	DisplayObject	172
A.3.5	CommandReceiver	173
A.3.6	CommandInterpreter	173
A.3.7	CommandOverrider	174
A.4	Workflow	175
A.4.1	The paradigm workflow approach	176
A.5	Examples	177
B	Example uses of the bci-interface library	179
B.1	Minimal example: an application for steering	179
B.2	Switching example: control steering and gazing	181
C	Users' questionnaires employed	185
C.1	SSVEP display optimization – user experience questionnaire	185
C.2	– user experience questionnaire	187

Introduction

In recent works of fiction such as the movies “Avatar” and “Surrogates”, a system allows a user, using only his thoughts, to experience and interact with the world through a physical copy (e.g. a robot) as if it was her/his own experience. Although state-of-the-art technologies do not allow to do this, recent technological advancements lay the initial bricks of such a system as seen in Figure 1. In order to build a “surrogate” system, one would need to combine these technologies in an efficient manner.

However, traditionally, reproducing the task at the will of the operator on a humanoid robot or an avatar is done by capturing his/her movements. In this thesis, we wish to move away from gesture and movement, which control a task, toward the interpretation of the user’s thoughts. That is the translation of intention into action as seen in Figure 2. At first we plan to achieve only simple control tasks that follow closely telepresence schemes between an operator and a humanoid robot. The demonstrators we build and present in this thesis progressively move away from this simple task to allow complex interactions between the user and its environment through the embodied device.

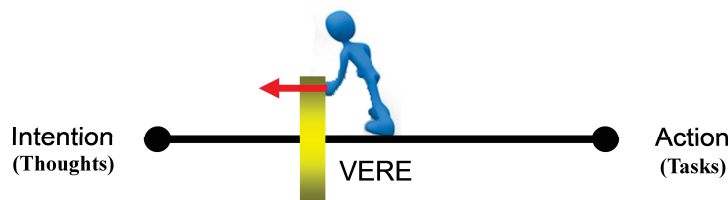
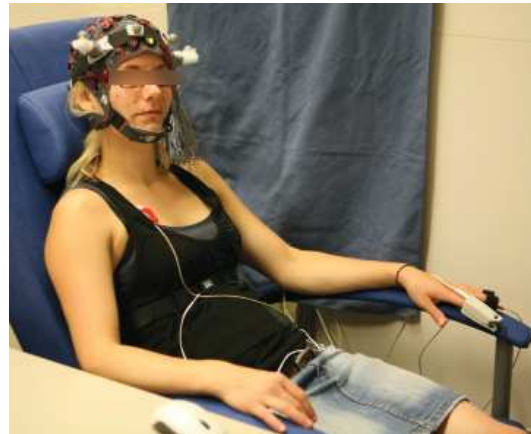


Figure 2: Concept of the VERE project

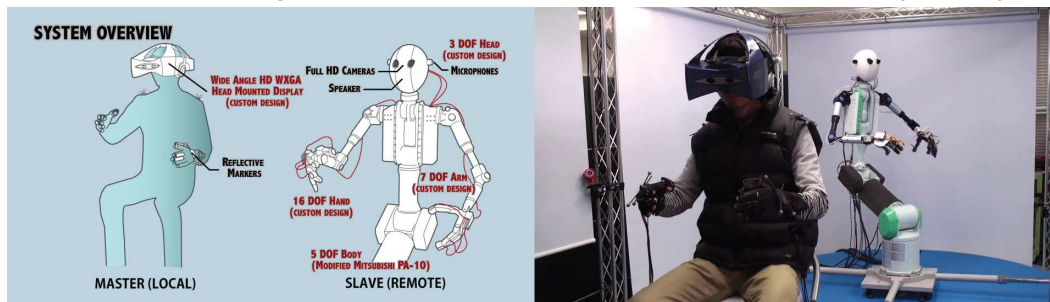
This work is conducted in the framework of the European integrated project VERE (Virtual Embodiment and Robotics re-Embodiment). Our goal in this project is to devise a software architecture that integrates a set of control and sensory feedback strategies to physically re-embody a human user into a humanoid robot surrogate or a virtual avatar using his/her own thoughts. The demonstrator we aim to achieve can be described as the following: a user is equipped with a brain-computer interface (BCI) that monitors brain waves activities from which we want to extract intentions and translate them into robotic commands. Sensory feedback should be delivered to the user so that s/he can embody the controlled device: virtual or real.



(a) The Geminoid robot aims at investigating the question of human presence by mimicing the appearance of a human being.



(b) A user equipped with several physiological sensors allowing to measure brain and body activity.



(c) Telesar V: a complex telexistence system.

Figure 1: Existing technologies towards a “surrogate” system.

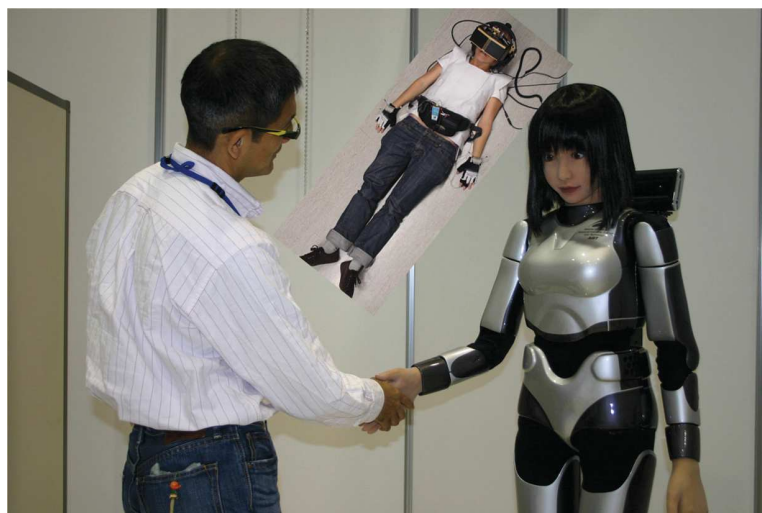


Figure 3: VERE demonstrator with HRP-4

Figure 3 shows such a complex interaction. The operator is equipped with the sensors

and feedback devices that compose the embodiment station built within the VERE project by another partner (University of Pisa). The intentions of the operator are guessed from his/her thinking and are then transformed into robotic tasks. The humanoid robot then executes these tasks and its sensors outputs are interpreted and properly reproduced for the operator in “close-loop” fashion. The research challenges involved within the realization of such a system are numerous and we now explicit them.

1. Complexity in fully controlling the surrogate’s motion

With technologies currently available, we are not able to extract a precise (motor) intention from the user’s brain activity. Hence, our approach requires an important level of shared autonomy, that is the capacity for the robot to make decision on its own while respecting the wishes of the user, and also requires the robot to be able to understand its environment. This later statement involves many things: the robot should know–or learn–how its environment is built, where it is located within this environment and which tasks it can perform. This is made necessary by two equally important aspects. First, the robot needs to understand its environment in order to interact with it in a safe and efficient manner. Second, it also has to be able to communicate its knowledge to the BCI so that it is used to better understand the user’s intentions and provide meaningful feedback. In the case of a virtual avatar, this is easier to achieve as the environment is simulated.

2. Minimal motion-based control/coupling

The technologies currently available to recover the user’s intentions from his/her brain are far from achieving the kind of understanding that we can observe in fictional works. They are actually very limited with regards to the number of recognizable intentions, the available bit rate and the accuracy of the recognition. Therefore, we have to devise a number of strategies to obtain the user’s intention through the B-BCI (Body-Brain Computer Interface). Furthermore, these intentions should be mapped into surrogate (robotic) actions.

3. Full synergy and beyond telepresence transparency

The notion of embodiment, central within the VERE project, goes beyond the classical approaches of teleoperation and telepresence. In teleoperation and telepresence, a user controls a remote device and receives feedback (e.g. visual feedback or haptic cues) about the task s/he is executing through the device. In telepresence, this allows the user to feel present at the scene where s/he is executing a task through the device. In embodiment, the user should not only feel present at the remote location but s/he should feel as if the device’s body was his/her own. This requires the introduction of new sensory feedback strategies.

These issues are all dependent upon constantly evolving scientific fields that also drive our overall progress. They cover a large number of research fields and go beyond the scope of a single thesis. We however demonstrate significant progress on each of these issues.

In the first chapter of this thesis, we introduce the general concept of brain-computer interface (BCI) or brain-machine interface (BMI) based on the approach of the most prominent frameworks in this field. That is the division between an acquisition layer, a signal processing layer and an application layer. The following sections are organized based on these layers. The most common brain signals' acquisition technologies are presented and the choice of the electroencephalogram (EEG) technology for the majority of the presented work is justified. Then, an overview of the most common paradigm used in EEG-based BCI to obtain the users' intentions is presented. Finally, several applications of BCI are then introduced with a particular emphasis on previous works for the control of robotic devices. Based on the examination of those previous works, we propose a novel approach to BMI within the context of humanoid robot and complex tasks execution through embodiment.

In the second chapter, we discuss the tools we developed and problems we encountered within the context of BCI-based robot control. First, we tackle the need for dynamic applications by introducing a novel library to build a user interface for BCI that is (a) easy to use, and (b) able to switch between different behaviors. Then, we show that steady-state visually evoked potentials (SSVEP) recognition performance are affected when the stimuli used to elicit the potentials are overlaid upon a dynamic feedback such as the video stream from a robot's embedded camera. Finally, we introduce several ways to overlay SSVEP stimuli upon objects recognized by the robot's vision system. These methods are compared based on both objectives, e.g. recognition accuracy, and subjective ones, e.g. user appreciation, criteria.

In the third chapter, we focus on the use of the SSVEP paradigm to perform whole-body control of a humanoid robot and further introduce the concept of *object-centric* BMI. In the first part, a general introduction of the object-centric BMI needs and principle is given. Then, a framework to realize object-centric BMI is proposed and its components are introduced individually. The second part of the chapter focuses on the technical modules that are required to implement this framework on a humanoid robot; including SSVEP-based paradigms allowing reliable and recursive selections, robot control bricks that allow us to perform task-oriented control and finally the vision elements that permits objects recognition and object-oriented navigation. Finally, several user experiments show that the framework allows an efficient control of the humanoid robot using a BCI.

In the fourth chapter, we use the tools introduced in the previous two chapters to perform embodiment studies using the humanoid robot controlled with BCI as a platform. The chapter is particularly focused on two studies conducted in collaboration with the Fondazione Santa Lucia (FSL). In the first study, the subjects control the robot while being subject to different modalities of sound feedback. A second experiment is also introduced. In this experiment, the subjects are equipped with a head-mounted display (HMD) which implies that the experiment is more immersive from the get-go. Therefore, the effect of the different sound modalities is different from the first experiment. Moreover, patients afflicted with spinal cord injuries (SCI patients) participated in this later experiment.

The fifth and final chapter describes preliminary results obtained from experiments that used electrocorticography (ECoG) to acquire the signals from the user's brain. The experiments were performed on two patients who had undergone surgery within an epilepsy-related

treatment. The results of these initial experiments are presented and we discuss the challenges posed by ECoG-based BCI.

We end our thesis with a summary of our contribution and future works and openings. In the next chapter, we present succinctly work from the state-of-art that serves as background of our achievements.

Brain-Computer Interface as a Control Interface: State of the Art

Brain-computer interfaces (BCI) [1] [2], or brain-machine interfaces (BMI), allow bypassing the usual means of communication channels between a human and a computer or a machine such as hand or voice input devices (e.g. keyboard, mouse, joysticks, voice command microphones, etc.). Instead, they allow the users to communicate their *intentions* to the computer by processing the source of the intentions: the brain activity. In return, the user is able to control different application (software or device) systems connected to the BCI. Combining this technology with a humanoid robot, we envision the possibility of incarnating one's body into a robotic avatar. Such a goal is pursued by the VERE project¹ in which frame the present work takes place. In many ways the problem at hands is reminiscent of telepresence technology. However, the specificities that are brought by the use of a brain-computer interface goes beyond telepresence aims and objectives while at the same time they backtrack substantially the performances. Indeed, in the current state of technology, brain-computer interfaces differ from traditional control interfaces [3], as they have a low information rate, high input lag and erroneous inputs. While the two terms—BCI and BMI—hold the same signification, in this thesis, we prefer the later when referring to the common idea of a BCI while using the second for BCI within the context of robot control.

In recent years, several frameworks such as OpenViBE [4], BCI2000 [5] or TOBI hBCI [6] have introduced a similar three-layer model to produce BCI application as shown in Figure 1.1. The signal acquisition layer monitors the physiological signals from the brain through one or several physical devices and digitizes these signals to pass them to the signal-processing unit. The signal-processing unit is in charge of extracting features —e.g. power spectrum, signal energy— from the raw signals, and pass them onto a classification algorithm to distinguish the patterns of *intentions* of the user. Finally, these decoded intentions

¹<http://www.vereproject.eu/>

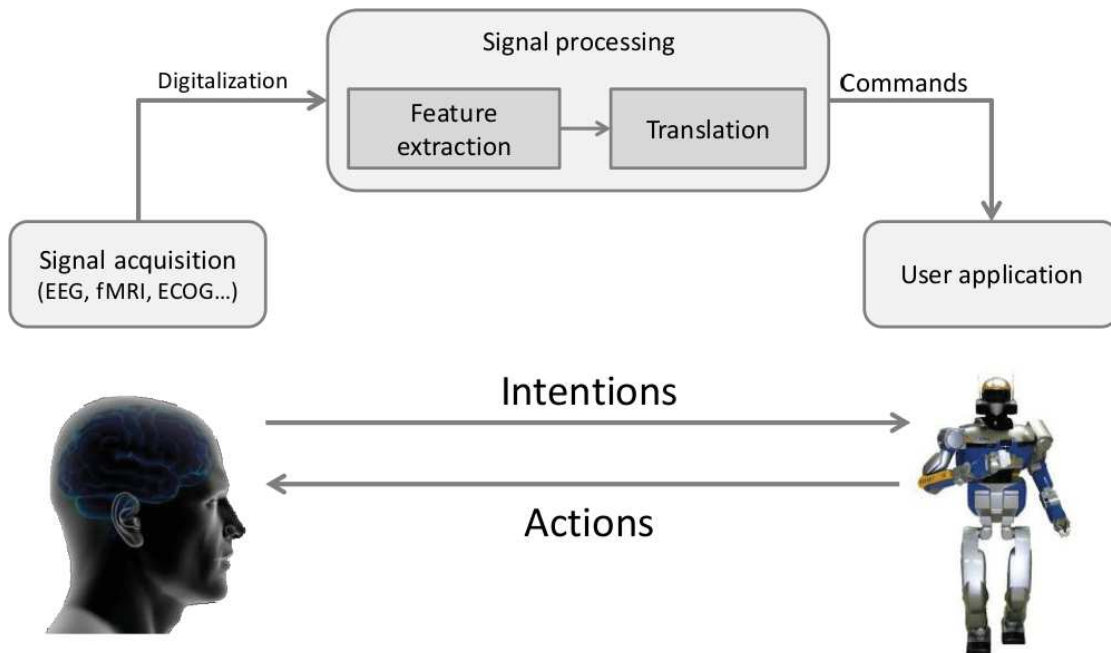


Figure 1.1: General design of a BCI system

are passed to the user application.

In this chapter we present each layer independently. We first present the different acquisition technologies that have been used in the BCI context, we discuss their benefits and limitations and we justify the choice of electroencephalography (EEG) in most of our works. The second section focuses on the different paradigms that can be used in EEG-based BCI. Finally, we review previous works that have used BCI as a control input, focusing on previous works in the robotics domain.

1.1 Acquisition technologies

The “reading” of one’s brain activity signals can be done by means of different technologies. However, within the context of brain-computer interface only those that allow measuring the brain activity in real-time or close to real-time are considered. In this section, we present some invasive solutions that have been used and BCI and then we focus on the three most used non-invasive technologies in BCI: functional near infrared spectroscopy (fNIRS), functional magnetic resonance imaging (fMRI) and electroencephalography (EEG). The practical pros of these technologies, e.g. the quality of the recorded signals or their spatial localization accuracy — how precisely one can tell from where in the brain a signal originates, are discussed together with other practical issues such as cost and more importantly the intrusiveness level. Finally, we explain our choice to settle on EEG for most of the applications developed within this work.



Figure 1.2: A typical ECoG array with 20 electrodes, courtesy of Dr Kamada

1.1.1 Invasive solutions

In this section, we introduce a few invasive solutions that have been employed to read the signals of the brain activity. We first introduce the electrocorticography (ECoG) technique, a technology that is considered to be “semi-invasive” and we then present invasive solutions as well as some of their achievements. Obviously those technologies have outstanding qualities from the signal acquisition point of view. However, their very nature also limit their usage.

1.1.1.1 Electrocoorticography (ECoG): a “semi-invasive” solution

Electrocorticography (ECoG), also known as intra-cranial EEG (iEEG), is an invasive technology used to measure the brain activity. In this case, the electrodes are placed directly in contact with the surface of the brain, typically using electrodes encompassed in a silastic membrane such as the one shown in Figure 1.2. This solution can be classified as “semi-invasive” because while the ECoG array is put inside the cranium of the patient, the electrodes do not penetrate the brain. Therefore it is considered relatively safe compared to the invasive solutions we present later in this section.

Compared to non-invasive technologies the quality of the signal obtained from the ECoG is much higher, offering a spatial resolution of 1cm [7] or even less, with an excellent signal to noise ratio. Furthermore the acquisition rate is also very good: typically reaching 1 kHz.

Recently, ECoG has been featured in impressive works related to robotics [8]. However, the main application of ECoG remains therapeutic as it is considered the “gold standard” for defining epileptogenic zones, a key step in the resolution of severe epilepsy cases. It is most often in this context that ECoG-based BCI experiments are conducted. Therefore, the use of this technology in BCI remains exceptional and implantations last for a few weeks at most.

1.1.1.2 Neuroprosthetics

Neuroprosthetics is the science that deal with the development of neural prostheses. Those prostheses are devices that can substitute a motor or sensory modality that was damaged

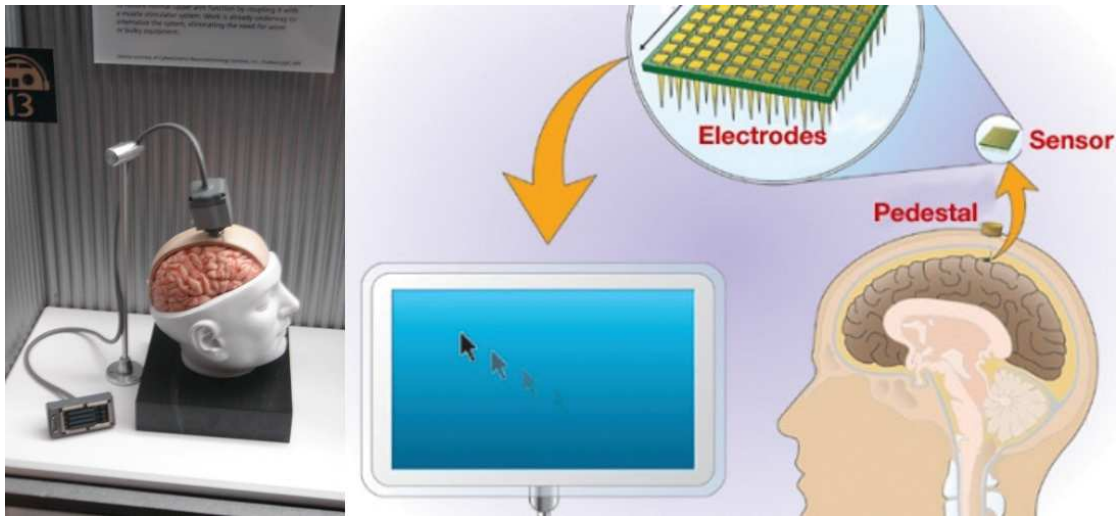


Figure 1.3: The BrainGate system, a promising neural prosthesis in development, adapted from [9]

following a disease or an injury. A relatively common example of this kind of device are cochlear implants which aim is to restore the auditory sense for deaf or severely hard of hearing persons. Another notable device is the “Dobelle-eye” which aims at restoring the vision of blind people by stimulating the visual cortex based on the images received by a digital camera worn by the patient.

Motor neuroprosthetics aim to either restore movement capabilities for paralyzed patients or to allow them to use computer and robotic devices to assist them. The first successful example of such devices in humans goes back to 1998. However, in 2005, Matt Nagle became the first tetraplegic patient able to control an artificial hand through a neural prosthesis: the BrainGate system.

BrainGate consists of a sensor implanted in the brain and an external decoder device, the signal acquisition and signal processing layers of the BCI model. The sensor uses 100 hair-thin electrodes that sense the electromagnetic signature of neurons firing in specific areas of the brain, for example, the area that controls arm movement.

Recently, in a remarkable work using the BrainGate system, a tetraplegic patient was able to control a robotic device to grab a bottle of water and drink from her own volition [10]. We discuss the details of this experiment later in this chapter.

1.1.2 Functional Near Infrared Spectroscopy (fNIRS)

Near-infrared spectroscopy is a general spectroscopic method that uses the near-infrared region of the electromagnetic spectrum. It is used in a wide variety of applications. In particular, it can be applied to perform functional neuroimaging, in which case it is called functional near infrared spectroscopy (fNIRS). It measures the hemodynamic responses associated with neuron behavior thus providing a reading of the brain activity.

fNIR is a non-invasive imaging method involving the quantification of chromophore concentration resolved from the measurement of near infrared (NIR) light attenuation, temporal

or phasic changes. NIR spectrum light takes advantage of the optical window in which skin, tissue, and bone are mostly transparent to NIR light in the spectrum of 700-900 nm, while hemoglobin (Hb) and deoxygenated-hemoglobin (deoxy-Hb) are stronger absorbers of light. Differences in the absorption spectra of deoxy-Hb and oxy-Hb allow the measurement of relative changes in hemoglobin concentration through the use of light attenuation at multiple wavelengths. Two or more wavelengths are selected, with one wavelength above and one below the isobestic point of 810 nm at which deoxy-Hb and oxy-Hb have identical absorption coefficients. Using the modified Beer Lambert law (mBLL), relative concentration can be calculated as a function of total photon path length. Typically the light emitter and detector are placed ipsilaterally on the subjects skull so recorded measurements are due to back-scattered (reflected) light following elliptical pathways. NIRS is functionally very similar to MRI but it is less costly and more portable. However, it cannot be used to measure cortical activity at more than 4cm deep and its spatial resolution is lower than MRI.

Finally, while fNIRS has been used in BCI, and particularly in robotics [11], it is not as popular as EEG as it is more expensive and slower: as with fMRI, due to the nature of the measured phenomenon, the system is able to detect the user intention a few seconds — 5 to 10 seconds — after this intention is formed in the user's mind.

1.1.3 Functional Magnetic Resonance Imaging (fMRI)

Magnetic resonance imaging (MRI) is a non-invasive brain signal acquisition technology. The primary form of MRI uses the blood-oxygen-level-dependent (BOLD) contrast. This is a type of specialized brain and body scan used to map neural activity in the brain or spinal cord of humans or other animals by imaging the change in blood flow (hemodynamic response) related to energy use by brain cells.

fMRI is primarily used in brain mapping research as, contrary to methods that can provide similar information, it does not require people to undergo shots, surgery, to ingest substances (radioactive markers), or to be exposed to radiation [12]. However, it has also been used to perform BCI activities. For example, in 2006, the Honda Research Institute Japan (HRI-J) and the Advanced Telecommunications Research Institute International (ATR) presented a robotic hand controlled by fMRI [13]. However, they later switched to an EEG/NIRS hybrid system to control the Asimo robot [14] mainly for practical reasons. Furthermore, as mentioned earlier, the temporal resolution of fMRI makes it difficult to use it effectively in applications requiring the control of multiple behaviors. Current research aims at resolving this issue.

1.1.4 Electro-encephalography (EEG)

Electroencephalography (EEG) is the recording of electrical activity along the scalp. EEG measures voltage fluctuations resulting from ionic current flows within the neurons of the brain. The biggest drawback of this technology is its very poor signal-to-noise ratio, which is mostly due to the fact that the electrical activity of the brain is measured through the skull

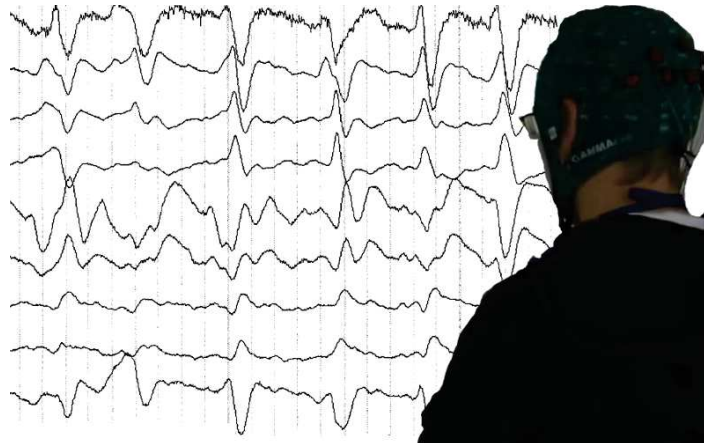


Figure 1.4: An EEG user wearing a cap with mounted electrodes and obtained signals

of the subject. Therefore, the spatial resolution of the system is very low and sophisticated signal processing techniques are required to extract useful information.

However, EEG also has significant advantages. First, it has a very high temporal resolution. Typically, EEG is recorded between 250 and 2000Hz but the latest technologies are able to achieve a 20000Hz sampling. Secondly, it is relatively tolerant to subject movement. Finally, it is significantly cheaper than other technologies: for example, an fMRI machine typically requires a large magnet and a magnetically shielded room, a very expensive setup.

This is why EEG is by far the most popular acquisition technology used in BCI research.

1.1.5 Summary

The following table shows, for each technology, the different qualities we are looking at when considering our choice for acquisition technology. “Localization” designates the spatial-localization accuracy. “SnR” designates the signal-to-noise ratio. “Cost” designates the cost of the system. Finally “Real-Time” designates the real-time capability of the technology.

Technology	Localization	SnR	Cost	Real-Time	Invasive
Brain implants	+++	+++	-	++	++
ECoG	++	++	-	++	Yes
NIRS	-	-	+	+	No
fMRI	++	++	-	-	No
EEG	-	-	++	++	No

For BCI applications, EEG has proven to be the most used technology, although it suffers from poor spatial localization accuracy and signal to noise ratio, it has remarkable practical qualities such as low cost, real-time acquisition and most importantly non-invasiveness. Moreover, a wide range of paradigms has been researched using EEG and provide as many ways to understand the BCI user’s intentions.

For those practical reasons we decided to adopt EEG in most of our work. However, the framework we will introduce in Chapter 3 is independent of the acquisition technologies or of the intentions' extraction method employed on the BCI side.

1.2 Features used in EEG-based acquisition

In the field of neuroscience a tremendous work is made to identify brain signal patterns relating to one's intentions or emotions. Nevertheless, the state-of-the-art BCI stands very far from being able to read, as in a book, one's mind thoughts or intentions by any technology means existing today. The brain is still a mystery in many aspects despite the notable amount of research conducted in the field. Nevertheless, brain function understanding progress substantially from year to year. Therefore, today's BCI technology is still confined in a *stimuli-response monitoring scheme*.

Three major methodologies are used in the BCI field because of their consistency among different untrained users. In this section, we first focus on the motor imagery paradigm where the system detects imagined movements. This paradigm is the only one where intention is guessed from the strength of one's thought. Then, an event related potential, the P300 is presented. Then, we introduce the steady-state visually evoked potential (SSVEP) that we are using extensively in this thesis. Finally, the error-related potential that could be potentially very useful in control applications, is presented.

1.2.1 Motor Imagery

Motor imagery [15] consists in detecting so-called event related de-synchronization–*ERD*– and event related synchronization–*ERS*–which are the decrease or increase of power in the alpha (8–13Hz) and beta (14–30Hz) frequency ranges of the brain activity and which occurs when the user executes or, more interestingly, imagines a movement.

In [16], Pfurtscheller *et al.* performed a study with 9 subjects to determine the feasibility of 4-classes classification of motor imagery, namely: left-hand, right-hand, feet and tongue movement imagination. In this study, the subjects were seated in an electrically shielded cabin and watched a 15" monitor. A trial lasted 7 seconds. At the beginning, the screen was blank then a cross was displayed in the middle of the screen at second 2. At second 3, an arrow indicated the motor imagery task that should be performed by the subject. They were asked to imagine the kinesthetic experience of movement rather than a visual representation of it and to avoid any motion during the trial. An ERD/ERS is expressed as a percentage in power decrease (ERD) or power increase (ERS) relatively to the 1 second interval prior to the appearance of the cross in a trial at 2 second.

Figure 1.5 show the results of one subject for the 4 motor imagery tasks and 3 of the 60 electrodes as presented in [16]–where a more thorough discussion of the results is provided.

These results show that hand motor imagery induces significant ERD in the mu band while foot and tongue motor imagery triggers significant ERS. Pfurtscheller *et al.* then showed that it was possible to build a classifier for single trial detection with an accuracy

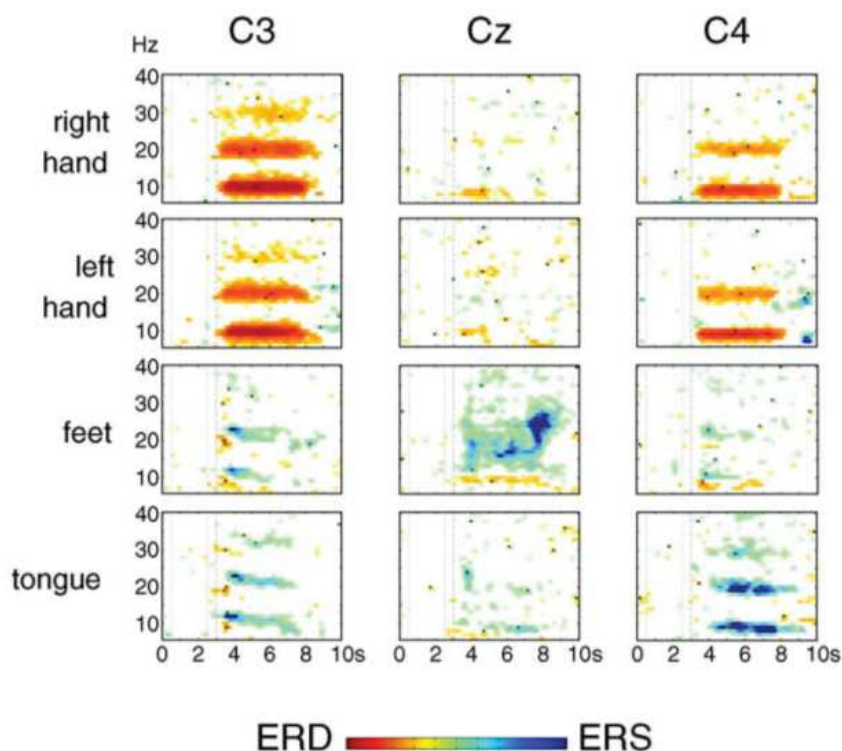


Figure 1.5: Time-frequency maps that shows significant ERD-ERS for one subject, 4 motor imagery tasks and 3 electrodes. Adapted from [16]

between 30% and 50%. However, it has been shown that the subjects could learn to control ERD/ERS thus making such classifiers more accurate. This process is quite long though.

While progress has been made since this early work, this method still requires a long training with the user to achieve useful performance when the system distinguishes multiple classes of movement.

1.2.2 Event Related Potentials: P300

The P300 [17] is an event related potential (ERP). It describes a positive wave in the cerebral activity which occurs 300ms after the occurrence of an expected event among other similar events. It is thought to be related to the process of decision-making.

The P300 is well adapted for systems with a large number of choices - e.g. selecting keys on a keyboard [18] —and has proven to provide very good performance with little training among the population. In [19], a study conducted with 100 persons showed that 90% were able to achieve an accuracy level between 80 and 100% after a 5 minutes training. However, to obtain this kind of performance it is better to have the rare event occur multiple times to properly detect the P300, therefore, while a high bit rate can be achieved with this paradigm, the decision rate is rather low, typically reaching 6 decisions a minute [18].

In Figure 1.6, we can see how the paradigm works for a typical keyboard application.

The user is shown a 6×6 grid with grayed letters. When the paradigm starts, each row and column is highlighted successively. In this example, the user wishes to select the letter 'N'. If the letter 'N' is not active then nothing particular can be observed from the brain activity. However, when the 'N' letter is active, then we can see a positive wave around 300ms after the occurrence of the stimulus, this is the P300. Typically, each row and column is highlighted multiple times to elicit as many potentials as possible and ensure the selection of the user's choice.

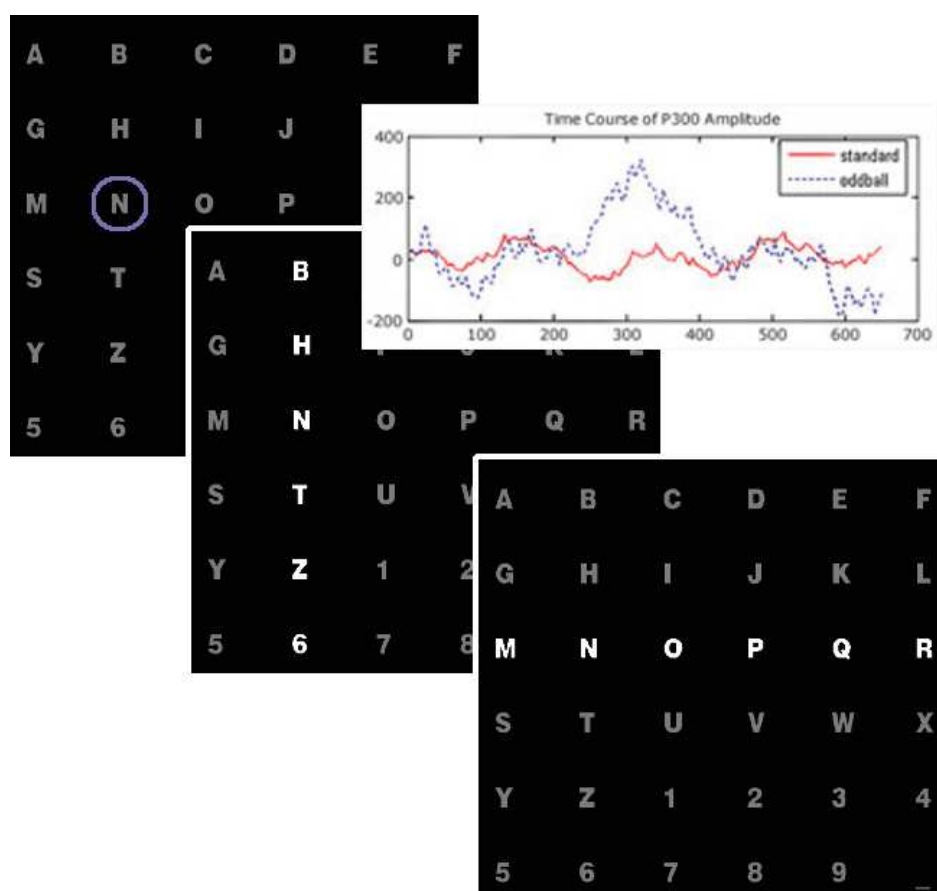


Figure 1.6: The P300 paradigm and associated brain answer: red indicates standard activity, blue shows the P300 triggered, adapted from [20]

While we only presented the paradigm for visual stimuli, it can be noted that the paradigm can also be applied with audio and tactile stimuli [21]. The principle is the same: for audio-based P300, different audio-cues are used instead of the visual stimuli while in tactile-based P300, different tactile stimuli are given to the user. However the performances of audio-based and tactile-based P300 systems are much lower [22] since it is more difficult for the user to distinguish the different stimuli.

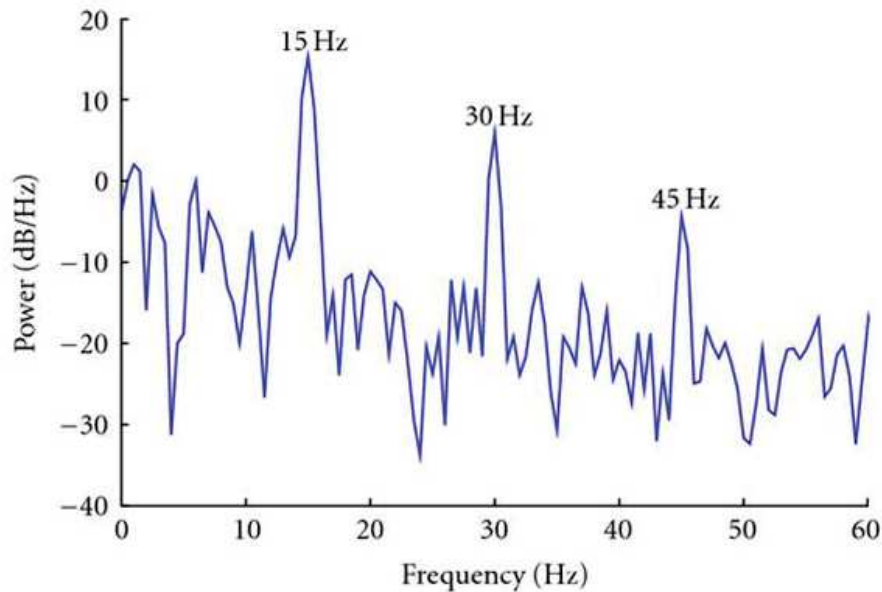


Figure 1.7: EEG spectrum when showing a 15Hz stimulus to the user, courtesy of [24].

1.2.3 Steady-States Visually Evoked Potentials (SSVEP)

The Steady-States Visually Evoked Potential (SSVEP) is elicited by presenting the subject with a flickering stimulus. As the user focuses her/his visual attention on the stimulus, a peak can be observed in the power spectrum of the EEG signal recorded from the primary visual cortex at the frequency of the flickering stimulus and other local peaks at this frequency odd harmonics as seen in Figure 1.7. By analyzing the signal we can predict whether the user is attending to a particular stimulus or not. The stimuli can be of various types, including physical LEDs to on-screen flickering objects. State-of-the-art detection systems [23] can operate at a good success rate, i.e. above 80%. The method relies uniquely on the user's attention to the stimulus, it also allows to detect that the user is maintaining her/his attention on a given stimulus and to detect a shift of attention in a few seconds. This makes SSVEP our preferred feature for reactive control through BCI such as steering control.

Various parameters can affect the strength of the SSVEP answer: the stimulus color, the stimulus shape and size [24] [25] [26] [27]. Moreover, the frequencies have to be carefully selected to have neither common first or second harmonics and to be below 20Hz to minimize the risk of eliciting an epileptic crisis in healthy subjects as advised in [28].

This paradigm can also be a bit tiring because of the annoying nature of the stimulus at low frequencies—typically, because of technical and medical considerations, SSVEP frequencies are chosen below 20Hz. It is possible to retrieve SSVEP at very high frequencies [29] but the intensity of the response is much lower and therefore much more difficult to detect.

Given that we are extensively using SSVEP in our work we present the extraction method we use to detect SSVEP from the user's brain signals. Furthermore, we also introduce the

specific method we used to display SSVEP stimuli on a computer screen instead of using a separate flashing box.

1.2.3.1 Minimum Energy Method

To extract the SSVEP from the brain signals, we rely on the minimum energy classification method presented in [30]. It has also been extended to implement a zero-class; that is a classification output that indicates that the user is not attending any stimuli. The signal acquired at a given electrode is modeled as:

$$y_i(t) = \sum_{k=1}^{N_h} a_{i,k} \sin(2\pi kft + \phi_{i,k}) + \sum_j b_{i,j} z_j(t) + e_i(t)$$

where $y_i(t)$ is the signal acquired at an i -th electrode. The first part of the equation is the SSVEP response, consisting of sinusoids with frequencies of the stimulus frequency and harmonics of this frequency. The second part of the equation represents the nuisance signals, e.g. non SSVEP activity or movement artifacts. Those are common to all electrodes but have different weight factors $b_{i,j}$ depending on the electrode i . Finally, $e_i(t)$ represents the specific measurement noise of electrode i .

Consecutive N_t samples are regrouped into the vector form:

$$\mathbf{y}_i = \mathbf{X}\mathbf{a}_i + \mathbf{Z}\mathbf{b}_i + \mathbf{e}_i$$

where $\mathbf{y}_i = [y_i(1), \dots, y_i(N_t)]^\top$ and \mathbf{e}_i is a similar vector. \mathbf{X} is the SSVEP model matrix of size $N_t \times 2N_h$ containing $\sin(2\pi kft)$ and $\cos(2\pi kft)$ pair in its columns. \mathbf{a}_i contains the corresponding amplitudes. \mathbf{Z} and \mathbf{b}_i are similar constructs for the nuisance signal. Finally, given that the time segment is short enough, the noise can be assumed constant. Hence, N_y electrodes can be written as a single matrix $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_{N_y}]$ and the model can be generalized to:

$$\mathbf{Y} = \mathbf{X}\mathbf{A} + \mathbf{Z}\mathbf{B} + \mathbf{E}$$

In order to maximize the SSVEP response while minimizing the noise, a channel is constructed by a weighted sum of N_y electrodes using a $N_y \times 1$ vector of weights w , that is:

$$\mathbf{s} = \sum_{i=1}^{N_y} w_i \mathbf{y}_i = \mathbf{Y}\mathbf{w}.$$

furthermore, N_s channels can be combined by constructing different electrodes combination $\mathbf{S} = \mathbf{Y}\mathbf{W}$ where \mathbf{W} is a $N_y \times N_s$ matrix containing the different weights combination. The construction of \mathbf{W} is the matter of the minimum energy combination technique.

The idea of this method is to build a combination that cancels the noise signals. To do so, we first remove any potential SSVEP components from the electrode signals, hence:

$$\tilde{\mathbf{Y}} = \mathbf{Y} - \mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{Y} \approx \mathbf{Z}\mathbf{B} + \mathbf{E}$$

then we have to find the weight vector $\hat{\mathbf{w}}$ that minimize the resulting energy of the electrodes combination. Therefore, we are left with the optimization problem:

$$\min_{\hat{\mathbf{w}}} \|\tilde{\mathbf{Y}} \hat{\mathbf{w}}\|^2 = \min_{\hat{\mathbf{w}}} \hat{\mathbf{w}}^\top \tilde{\mathbf{Y}}^\top \tilde{\mathbf{Y}} \hat{\mathbf{w}}$$

the solution to this optimization problem is the smallest eigenvector \mathbf{v}_1 of the symmetric matrix $\tilde{\mathbf{Y}}^\top \tilde{\mathbf{Y}}$ and the energy of this combination is equal to the smallest eigenvalue λ_1 of this matrix. Moreover, as $\tilde{\mathbf{Y}}^\top \tilde{\mathbf{Y}}$ is symmetric, its eigenvectors are orthogonal, thus providing us with N_y uncorrelated channels with increasing noise energy. The number of channels N_s retained from the minimum energy method is the smallest number that satisfies:

$$\frac{\sum_{i=1}^{N_s} \lambda_i}{\sum_{i=1}^{N_y} \lambda_i} > 0.1$$

where the denominator is the total energy in the nuisance signals and noise, and the numerator is the total energy retained when combinations are used, i.e. N_s discards close to 90% of the nuisance signal energy. Finally, the channels being selected, the following test statistic is used to test the presence of an SSVEP response:

$$T = \frac{1}{N_s N_h} \sum_{l=1}^{N_s} \sum_{k=1}^{N_h} \frac{\hat{P}_{k,l}}{\hat{\sigma}_{k,l}^2}$$

where $\hat{P}_{k,l}$ is an estimate of the power in SSVEP harmonic frequency k in the channel s_l and $\hat{\sigma}_{k,l}^2$ is an estimate of the noise in the same frequency and channel. Details regarding those estimations are provided in [30].

The class that receives the highest score is normally selected. However, this does not account for moments where the user does not look at a specific stimulus. To do so, the score value for each class is converted to the corresponding probability value using a softmax transformation. If the probability value of the class with the highest score and thus the highest probability is less than a selected limit, the sample is assigned to the zero class instead. Otherwise, the initial class assignment is kept. The limit is selected by defining the residual error probability, i.e. the residual chance that the class assignment may be wrong, which is empirically set to 3%.

This process requires a short training -typically under 10 minutes- and provides a 80% accuracy rate [31]. However, 60% of errors are miss-classification as “no decision” when a stimulus was actually attended, so the system is actually mistaken about the intent of the user about 10% of the time. This low error rate makes this system somehow reliable in a control context. The method implementation can provide a new command every 200ms, which is a satisfactory performance for a BCI system [32]. However, as the method relies on a 3 seconds window data input, a user shift of attention cannot be detected in less than 3 seconds.

1.2.3.2 Displaying SSVEP Stimuli on Screen

In SSVEP-based BCI, different elements can be used to trigger the potential in the human brain (namely at the visual cortex) [24]. Such methods include the use of a LED that flickers at a chosen frequency or a graphical stimulus displayed on a computer screen.

However, the display of SSVEP on a computer screen is limited by the refresh frequency of the screen as the display of the stimulus is locked to the frame display. Therefore, most SSVEP-based BCI that use on-screen stimuli are limited to the choice of stimuli frequencies that “fit” with the screen frequency. For example, on a monitor having a 60Hz refresh rate, one can easily display 8.57Hz, 10Hz or 12Hz stimuli as they would respectively correspond to 7, 6 and 5 frames per period. However, an 11Hz stimulus would require a 5.5 frames per period which is not suitable. This is illustrated in Figure 1.8.

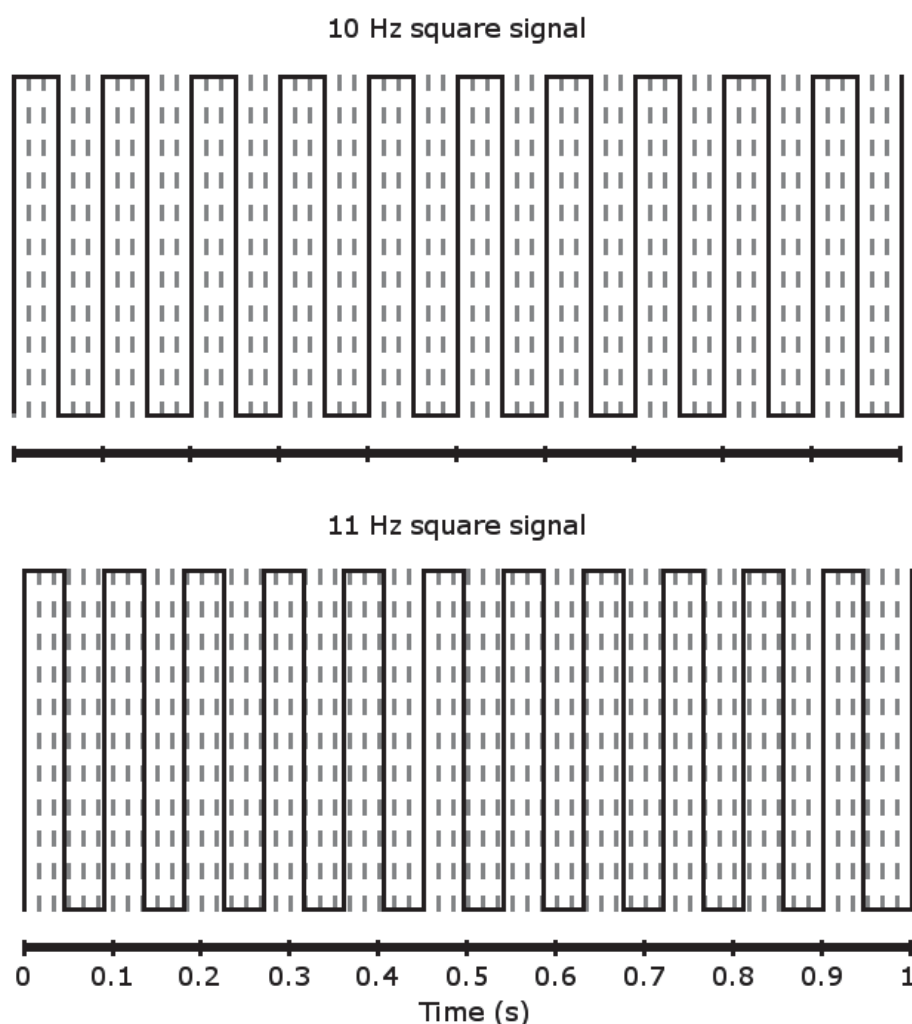


Figure 1.8: Square signals for 10 and 11Hz stimuli. The gray dotted-lines represent frames, as you can see the 10Hz signal is synchronized with (reverse every 3 frames) while the 11Hz signal is not (reverse every 2.72 frames)

In [33], a new frame-based method is proposed to display SSVEP stimulus of any frequency on a screen despite the refresh rate limitation. The idea is to approximate the presentation rate (e.g. reverse the stimulus every 2.72 frames) by using a varying number of frames in each cycle. So the stimulus signal for frequency f is given by:

$$stim(f, i) = square[2\pi f(i/RefreshRate)]$$

where $square(2\pi ft)$ generates a square wave with frequency f , and i is the frame index. Figure 1.9 shows the results of this function for an 11Hz stimulus. A one second frame sequence would be [3 3 3 2 3 3 3 2 3 3 2 3 3 3 2 3 3 3 2 3 3 2] which includes 11 cycles of a varying length of five or six frames. This approach allows any frequency, up to half the refreshing frequency, to be displayed on-screen. In [33], the authors proposed this method and demonstrated it by implementing a SSVEP-based BCI using 16 different frequencies. Frequencies ranged from 9 to 12.75Hz with an interval of 0.25Hz. The system reached 97.2% accuracy.

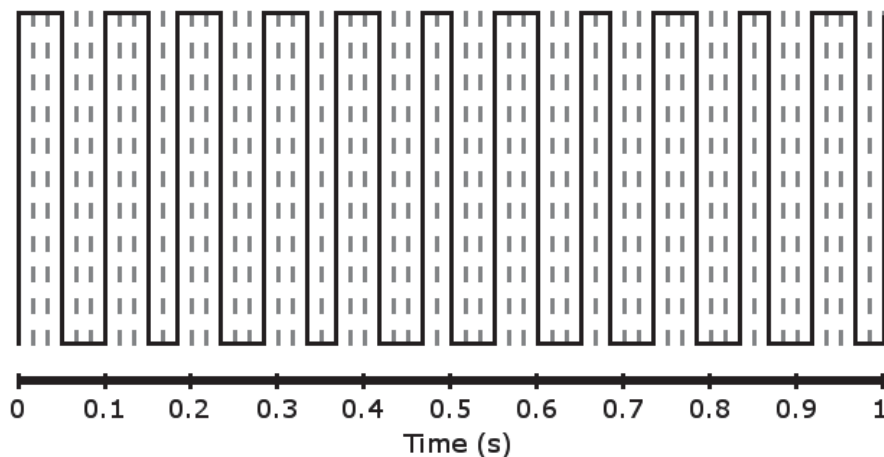


Figure 1.9: Frame sequence corresponding to an 11Hz stimulus

1.2.4 The Error Potential (ErrP)

Interaction Error-related potentials (ErrPs) are special features that can be detected in the EEG, after a wrong action selection by the BCI system or the user. After the onset of the feedback indicating the selected action, these features can be distinguished by first, a sharp negative peak after 250ms followed by a positive peak after 320ms and a second broader negative peak after 450ms [34].

In [35], a protocol is proposed to observe and use ErrP. The user performs motor imagery to control a cursor on the screen. Her/His goal is to move this cursor to a target area. When the motor imagery detects the wrong intention the cursor would move in the wrong direction. This normally triggers an ErrP, as seen in Figure 1.10, and the detection can be corrected. In [36], this is extended and the motor imagery is re-trained online when an error occurs.

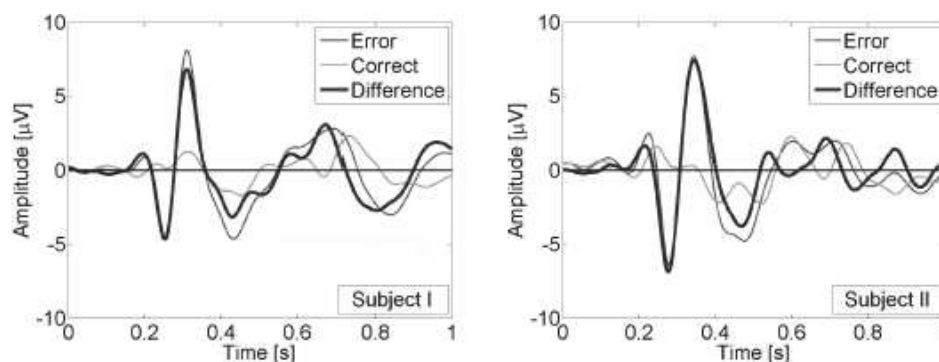


Figure 1.10: Results from [35]: grand averages of error trials, of correct trials and the difference error-minus-correct for channel FCz for both subjects. The ErrP is clearly visible in the difference (in black) when an error occurred.

The error potential is a very promising feature for the use of BCI as error recovery is very important in any application that may be developed using these systems. However, ErrP shall be 100% reliable to be effective and it does not appear to be the case in current trials. Finally, to the best of our knowledge, it has yet to be found when using P300 or SSVEP paradigms.

1.3 BCI Applications

As stated in the introduction, the final purpose of the feature extraction is to use this translation of the brain activity as a command input for a controlled system. A BCI has been used to control a very wide variety of applications [3]. In this section we present some of these applications. We focus on EEG-based BCI and specifically on three categories of applications: BCI-based spelling devices, games and virtual reality applications and finally robotic applications.

1.3.1 BCI-based spelling devices

Brain-Computer Interface research initially aims at allowing locked-in patients to regain motor and social capabilities through the power of their brain. As such, many systems have been devised to allow someone to communicate through a BCI. Here we introduce the three most popular BCI-based spelling devices.

1.3.1.1 “Thought Translation Device”

The “Thought Translation Device” (TTD) was introduced in 1999 by the university of Tübingen [18]. It is one of the first major BCI applications. The goal of this application is to allow a paralyzed user to spell words, which is done by allowing the user to select letters in a binary tree. Before using the system, the user learned to control her/his brain’s slow cortical potentials (SCP) so that s/he can trigger positive or negative variations.



Figure 1.11: The P300 speller interface used by [18]

In this application, the alphabet has been recursively divided into two parts and distributed as leaves of a binary tree. At each node of the tree, the user selects the left or right leaf by modulating its brain activity accordingly. At the end of the tree the user has selected a letter. The letter selection process took approximately 2 minutes.

1.3.1.2 P300 Speller

The P300 speller uses the P300 paradigm to allow the user to spell words. It was first introduced in 1988 by [37], see also [18]. In this application, the subject is shown a 6 by 6 matrix on a computer screen. The matrix contains all the Latin alphabet letters, digits (from 1 to 9) and a space character (see Figure 1.11).

During a trial, each row and column of the matrix is highlighted several times. The user's task is to focus attention on the letter s/he wishes to select and count the number of time this letter is highlighted. The highlight of this letter is an uncommon but expected event among the highlighting events and thus triggers a P300 response. Detecting the P300 allows to find which line and which column contains the letter that the user wants to select and thus the letter s/he wants to select. In this first iteration, the system allowed the user to spell up to 3–4 letters per minute.

This application being based on P300 it requires very little training to be used effectively [19]. This application has been very popular within the BCI community and is already widely used by paralyzed patients to communicate with other people [38].

1.3.1.3 Hex-o-Spell

The Hex-o-Spell interface [39] is a prime example of the integration of user application design and BCI technology. It can be considered as an evolution to the TDD system but

application is designed to overcome the bit rate issues of the BCI.

In this application, the user controls the rotation and length of an arrow using motor imagery. The rotation of the arrow is associated to one class of motor imagery while the length of the arrow is associated to another class. The interface displays a hexagon. Each cell contains either another hexagon or a letter. By rotating the arrow the user can point to a specific hexagon and by extending the arrow's length the user can select this specific hexagon as seen in Figure 1.12.

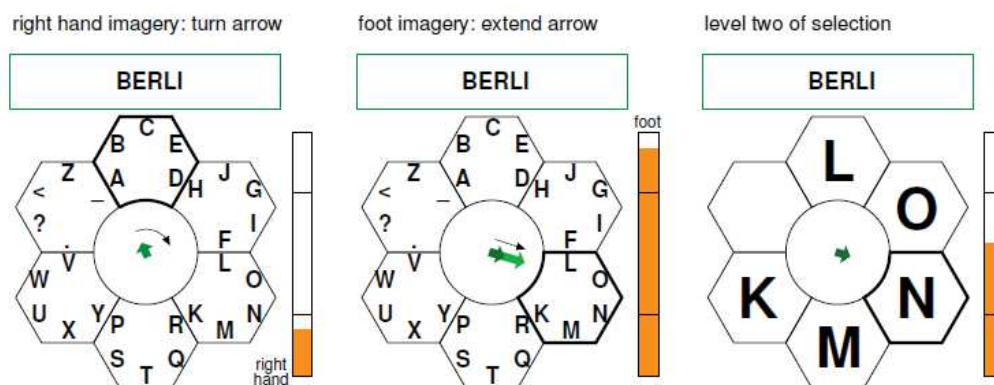


Figure 1.12: The interface of the Hex-o-Spell application (from [39])

Once a cell has been selected, the interface evolves:

- If the cell contains another hexagon, then the content of this hexagon is spread and the selection process restarts
- If the cell contains a letter, then this letter is selected by the system.

Using this system, the user can type up to 7.6 letters per minute.

1.3.2 Entertainment, Games and Virtual Reality Applications

Brain-Computer Interfaces were initially designed to help disabled people, however, the technology has recently been used in virtual reality applications and video games [40]. These applications are very interesting to researchers in the BCI community because they provide a safe, predictable and reliable control environments for applications but they also provide meaningful insight to the principles of BCI.

For example, in [41], Leeb *et al.* designed an application to allow a user to navigate a virtual apartment using the motor imagery paradigm. During this study, the user was first trained using the standard cue-based feedback; then the user would be put within the virtual environment. This team observed a clear improvement of the BCI performance during the virtual reality experiments and the subjects reported a motivational effect. This shows that virtual environments and gaming applications can be beneficial to the user of the BCI because of their highly engaging nature.

Furthermore, the successful use of visual stimuli-based paradigm, such as SSVEP, in rich and distracting environment show that these paradigms are well suited for use in real world application. For example, in [42], Legény *et al.* show that the SSVEP stimuli can be seamlessly integrated in the virtual environment without performance depreciation.

In the next paragraphs we present some games and virtual reality applications developed with BCI to illustrate the principles behind BCI driven applications. Finally, we present some recent successful commercial applications using BCI showing a future trend towards the integration of BCI powered devices in the real world.

1.3.2.1 MindBalance: an SSVEP-based game

In [43], Lalor *et al.* introduced an SSVEP-based game called “MindBalance”. In the “MindBalance” game, the user has to be able to control a virtual character walking along a tightrope. Two checkerboard patterns are shown on the left and the right hand of the character, flickering at 17 and 20Hz. A brief training period occurs at the beginning of the game where the user has to focus successively on the left and the right stimuli for 15 seconds (see Figure 1.13a), three times each.

Afterwards, the virtual character walks toward the user. Periodically, the character loses balance (see Figure 1.13b). The BCI user then has 3 seconds to shift her/his attention either to the left or the right stimulus to equilibrate the character. If s/he failed this time, the character loses his equilibrium even more, the user then has 3 more seconds to bring the character back. If s/he fails again, the character falls and the game is lost.

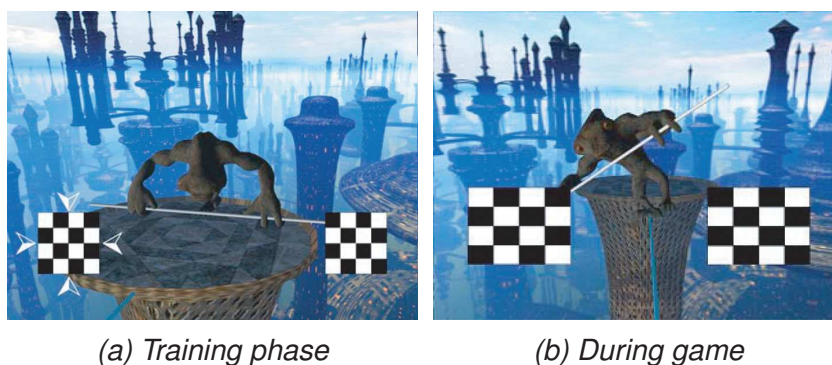


Figure 1.13: Screen captures, courtesy of [43], from the MindBalance video game: (a) Training phase (b) Game phase

1.3.2.2 “Use the Force”: an ERS-based game

In [44], Lotte *et al.* used a self-paced BCI system that can detect real or imagined foot movements by detecting a Beta Event Related Synchronization (ERS) that occurs after the real or imagined foot movements [45]. Based on this, they let the user control the “Use the Force” application.

In this application, the goal of the user is to lift the Tie-Fighter, a space vessel from the famous Star Wars franchise, using only the provided BCI. This application into trial, which timeline is shown in Figure 1.14. Each trial lasts 10 seconds. The first phase lasts 4 seconds and is a resting phase with no specific instructions given to the user. The second phase lasts 3 seconds. A green text saying “MOVE” appears on screen and the user is instructed to perform real foot movements to move the vessel. Finally, the last phase also lasts 3 seconds and this time a red text reading “STOP MOVING” appears and the user should stop moving the vessel. If the user was successful is score is increased as shown by the growing yellow gauge in the bottom-left corner on the screen.

A second version of the application exists which is identical except that in the second phase the vessel should be moved only by imagining foot movements.



Figure 1.14: Temporal structure of the game presented in [44]

This experiment was performed during the Laval Virtual 2008 VR exhibition with 21 novice subjects. It shows that despite a noisy environments and distracting conditions, the users were able to perform well in the application.

1.3.2.3 Navigation in virtual environments using the SSVEP paradigm

In [46] and [47], the user is placed within a virtual environment using a head-mounted display. Several applications were designed to show the BCI capacity of controlling an avatar using the SSVEP paradigm. In each of these scenarios, the stimuli are integrated interactively within the environment.

In the “Button Scenario”, shown in Figure 1.15a, the world is shown through the first person perspective. The avatar is facing down at her/his feet and two SSVEP stimuli are attached to the arms of the avatar. The goal for the user is to activate the button by three successive selections of the left or right stimulus and alternating the buttons selection as fast as possible.

In the “Slalom Scenario”, shown in Figure 1.15b, the avatar is shown in third person perspective and has three SSVEP stimuli attached to it. Each stimulus is associated to a control command for the avatar: the stimulus on the left side of the avatar is used to turn the avatar 45° in the left direction, the stimulus on the right side of the avatar is used to turn the avatar 45° to the right and the stimulus placed on the top of the avatar’s head is used to move

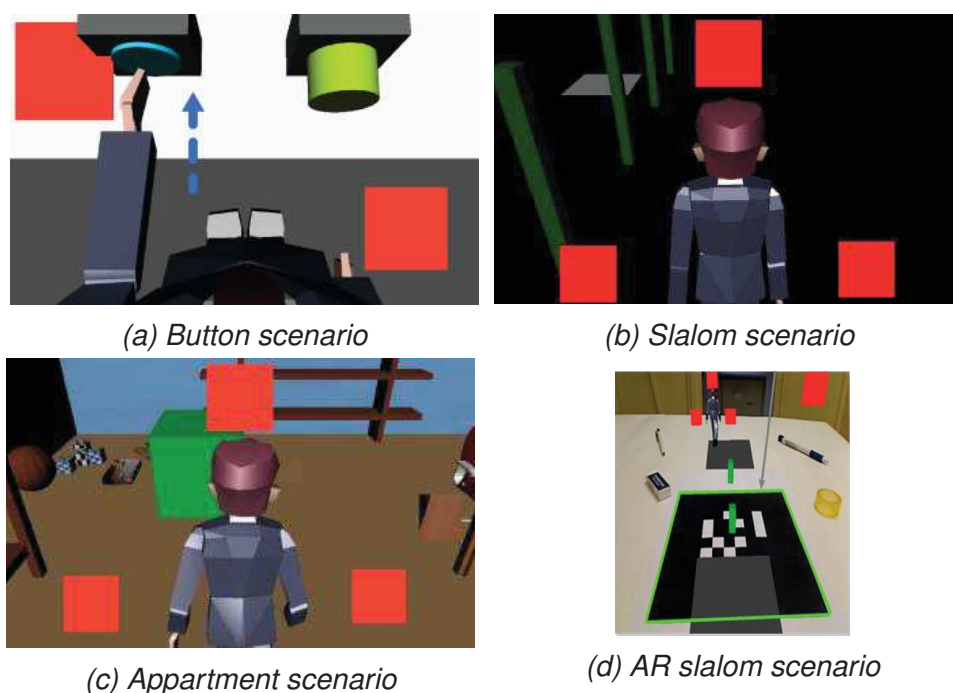


Figure 1.15: Virtual Reality Applications developed using the SSVEP paradigm: scenarios, courtesy of [46]

the avatar forward. The goal of the BCI user is to move the avatar through the slalom shown to him in less than 10 minutes.

In the “Apartment Scenario”, shown in Figure 1.15c, the control scheme is similar to the “Slalom Scenario”. The goal is similar too as the user has to reach a target in the apartment in less than 10 minutes. Two sessions were recorded, in the first one the user had to reach the center of the apartment while in the second one the user had to reach a target placed in the corner opposite of the avatar starting position in the house.

Finally, in the “Augmented Reality Slalom Scenario”, shown in Figure 1.15d, a fourth button and a secondary goal was added compared to the “Slalom Scenario” and “Apartment Scenario”. The fourth button is used to enable and disable the avatar’s control buttons. The secondary task of the user is to stop in two designated areas at the beginning and at the end of the slalom for 30 seconds.

While this work demonstrates the possibility of controlling an avatar using SSVEP to complete a variety of somewhat complex tasks, it does not consider the combination of these different scenarios to achieve more realistic and practical tasks in the virtual environment.

1.3.2.4 Commercial Applications

In the most recent years, some BCI systems have been commercialized for the general public. These applications are often simple compared to what can be achieved in laboratories but their success and popularity show the attraction of thought-powered devices for the public and the potential for BCI to come out of the laboratories.

The Mindflex [48] device, shown in Figure 1.16a and developed by NeuroSky [49], is one of this popular device. It was released in 2009. In this game the user can control a fan's speed using her/his brain activity. The goal is to use the fan to guide a ball through an obstacle course that can be customized to regulate the difficulty of the game.

The NecoMimi device [50], shown in Figure 1.16b, also relies on technology from NeuroSky. It was released in 2012. It measures the relaxation state of the user using EEG and the move fake ears attached to the device accordingly.



Figure 1.16: Two successful commercial BCI applications, courtesy of [48] and [50]

Finally, BCI have become more and more popular in the video game industry. In 2013, the first Neuro Gaming Conference and Expo was held [51], showing the increase in popularity of these technologies. As we have seen in previous examples, the BCI can be used as the sole mean of control for the game or, as shown in more recent examples such as the upcoming Son of Nor [52] video game, it can be used as a complementary input.

In the Son of Nor game, the control of the avatar's movements and gaze is done through the usual keyboard and mouse. However, casting magic spells and terraforming the world presented to the user is done using BCI [53].

1.3.3 Robotic Applications

In this last section we review some robotic applications in which various robots are controlled by a BCI system. BCI has not been a very popular device for robotic control as its main issues—high latency, low bit rate and unreliability mostly—make the control of robotic systems difficult. In this section we first focus on a very successful and notable application

of BCI technology in robotics: the brain-actuated wheelchair that was developed by Milàn *et al.* [54]. This application essentially uses the motor imagery paradigm. Therefore we also present two simple robotic applications: one that uses the SSVEP paradigm and another that uses the P300 paradigm. Finally, we present the only, to our best knowledge, BMI application preceding our work that focused on the control of a humanoid robot.

1.3.3.1 Brain-Actuated Wheelchair

The Brain-Actuated Wheelchair project takes its root from the work presented in [55]. This paper presents the first robotic device controlled through non-invasive brain-computer interface. Despite the low bit rate of BMI protocols available at that time—around one bit per second—the authors demonstrate the ability to control a simple mobile robot through a set of mental tasks recognized in the EEG activity of the user. The authors attribute their success to three key features present in their work:

1. The use of high-level commands to control the robot, such as “turn right at the next occasion”, that are autonomously executed by the robot.
2. The use of an asynchronous BMI protocol, allowing the user to pass command to the system without the need of an external cue and to let the robot executes the currently selected behavior until the user chooses to modify it.
3. The use of behavior-based control [56] to allow the robot to execute the high-level commands with obstacle avoidance and smooth turns.

The EEG protocol used in this work relies on the statistical analysis of the EEG spectrum of the user [57]. It is able to recognize 3 mental tasks with an accuracy of 70%. However, it also possesses the ability to recognize an “idle” state through the use of a statistical rejection criterion; therefore, while the accuracy of the system may seem low, it actually miss-recognize the user’s mental state only 5% of the time. It requires the users of the system to train over a few days to obtain satisfactory performance. Theoretically, the system is able to reach an information rate of 1 bit/s and 2 commands per second in a mode called “Mode I” by the authors or 0.85 bit/s and 1 command per second in a mode called “Mode II” that further reduce the error rate of the system by analyzing the EEG spectrum state over a longer period of time.

Regarding the particular case of [55], two subjects took part in the experiment. They trained for 5 and 3 days respectively. The first step of the training is to let the user decide which mental tasks they should perform among the following ones:

- “Relax”: the task of getting relaxed.
- “Left” and “Right” hand (or arm) movement: the task of imagining repetitive self-paced movements of the limb.
- “Cube rotation”: the task of visualizing a spinning cube.

- “Subtraction”: the task of performing successive elementary subtractions by a fixed number.
- “Word association”: the task of concatenating related words.

In the end, one subject chooses the “relax-left-cube” set of tasks while the other chooses the “relax-left-right” set. Afterwards, the user performs four consecutive training sessions of 5 minutes, separated by a 5 to 10 minutes break, during which the user randomly performs one of the mental tasks for 10 to 15 seconds and receives feedback on a computer screen. The feedback consists of three buttons: each button is associated with one mental task and the button starts flashing if the mental task is recognized. After each training session, the statistical classifier is optimized off-line. Finally, the user learns to control the robot for two days using “Mode II” in the BMI protocol.

The robot used in these experiments is the Khepera robot, shown in Figure 1.17a. It shares some features with a motorized wheelchair. It is a two-wheeled vehicle that moves at a maximum speed of one third of its diameter per second—same as a wheelchair in an office building. Furthermore, it is equipped with 8 infrared sensors to detect obstacles. However, the limitations of the sensors led the authors of the paper to use a multilayer perceptron mapping the 8 sensors into 6 classes of environmental states: wall to the left, wall to the right, obstacle to the left, obstacle to the right, wall or obstacle in front, and free space.

The robot is also equipped with three colored lights to mimic the feedback obtained on the computer screen during the training session. The lights are also used as a mean to communicate the behavior that the robot applies as it approaches a possible turn for example. Thus allowing the user to correct the robot’s behavior if necessary.

The robot’s controller uses both the user’s mental state and the robot’s perceptual state as inputs to a finite state machine with 6 states that dictate the robot’s behaviors, as seen in Figure 1.17b. In this figure, #1, #2 and #3 represent the 3 different mental states of the user recognized by the BMI, and, $|o, o|$ and \hat{o} represent 3 different perceptual state, respectively: left wall, right wall and wall or obstacle in front (the additional perceptual states of the robot are used internally to implement the obstacle avoidance routine). The different behaviors that are implemented are “forward movement”, “left turn”, “follow left wall”, “right turn”, “follow right wall” and “stop”. The interpretation of the user’s mental state by the robot controller thus depends on its current perceptual state. For example, mental state #2 is associated to the “left turn” behavior in open space but it is interpreted as “follow left wall” if the robot currently perceives a left wall. Finally, the robot automatically transits to the “stop” behavior if it senses an obstacle in front of it to avoid collision.

In the robot’s control sessions, the user’s task is to mentally drive the robot towards a randomly selected room. Once this room has been reached, another room is selected and the user should drive the robot to this new room. This initial experiment was very successful as both users were able to drive the robot from room to room at the end of the two days of training. Furthermore, the users reported that the control scheme adopted by the research team was easy to learn and intuitive to use. Finally, the experiment was repeated but the EEG input was replaced by a keyboard input. In this case, the performance of the users were better but by a fairly low factor—below 1.5.

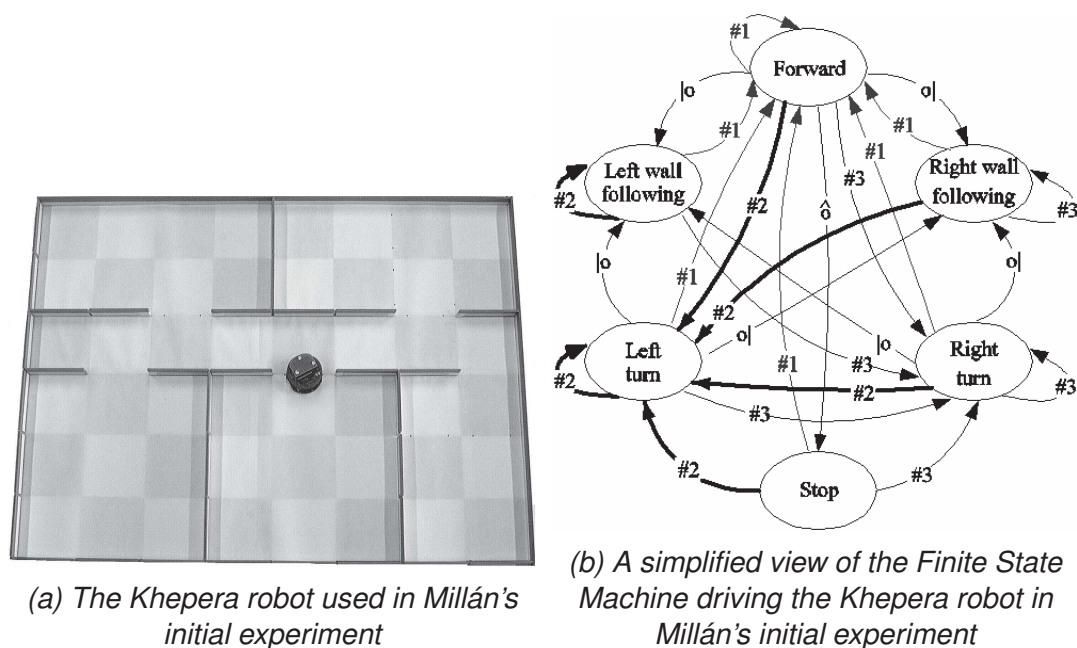


Figure 1.17: Illustrations used in the initial EEG-based robotic experiment by Millán et al. as seen in [55]

Following this initial robotic device, the next work focused on a simulated wheelchair [58]. In [58], the authors wished to establish the superiority of a shared-control approach, that is a system where the controlled device and the controller (in this case the simulated wheelchair and the human through the BMI) work together to achieve a task. Therefore they devised two different controllers. The first one, labeled “A0”, only allows the wheelchair to detect obstacles and stop to avoid collision. The second one, labeled “A1”, enables a higher intelligence level and avoids the obstacle when detected rather than stopping.

The authors performed an experiment with 4 novice subjects. The task performed is shown in Figure 1.18. They had to navigate the wheelchair through a corridor where two obstacles are present. Each subject performed the task 25 times using both controllers. When using the controller “A0”, the subjects were unable to guide the wheelchair to the goal. However, when using the “A1” controller, the users were able to drive the wheelchair to the destination despite their inexperience with the BMI system, reaching the target, respectively, 16, 9, 10 and 11 times out of 25.

The knowledge gathered through the previous experiments finally led to the development of an actual brain-controlled wheelchair that was first presented in [59] and in [54]. In [54], 3 users were trained to use the BMI system, two of them had previous experience with the simulated wheelchair and the third one had no experience neither with the wheelchair nor the use of BMI. The EEG protocol used by the authors is similar to the one presented in their previous work. Therefore, the BMI can only outputs three commands: **F**orward, **L**eft and **R**ight. Thus, as seen in previous works, a shared-control approach is necessary to allow both a better driving of the wheelchair and to recover from the BMI mistakes. However, the

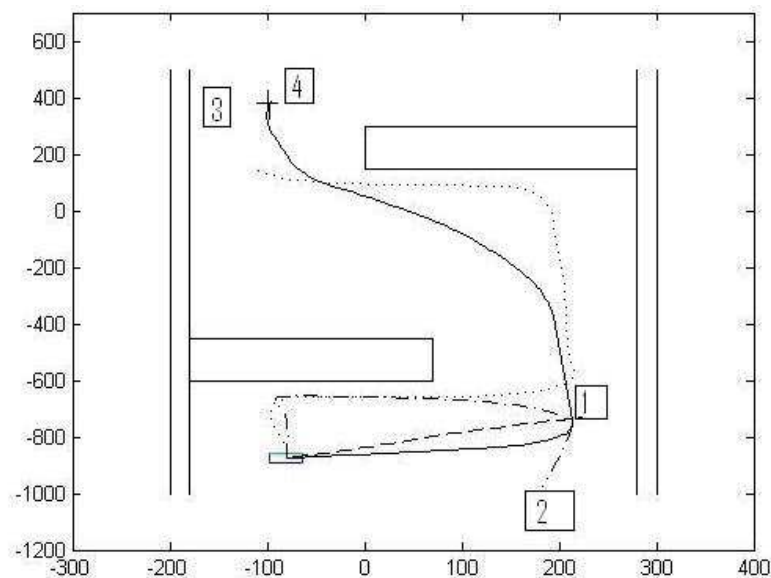


Figure 1.18: The task presented in [58]: the starting point is in the bottom left of the figure; the optimal trajectory is the one labeled “4”, a trial is considered successful if the wheelchair ends within 50cm from the target. “1” is a trajectory obtained with an “A0” controller. “2” is an unsuccessful trial with controller “A1” and “3” shows a successful trial with the same controller.

control scheme is more complex than before in this case and is briefly introduced here.

Figure 1.19 shows the approach developed by the authors. The inputs of the system are two-fold:

1. A probability distribution over the three mental commands generated from the EEG data of the user
2. The sensory readings from the wheelchair which is equipped with a laser range finder

The controllers generate the linear velocity v and the angular velocity ω of the wheelchair. First, (v, ω) is generated from the mental input. Then, this input is merged with the current velocity command to generate a smoother trajectory. Finally, this command is merged with the sensory readings of the wheelchair to select the proper assistive behavior and adjust the resulting velocity accordingly. Three different assistive behavior are proposed by the authors in [60]:

- A0** Collision avoidance: the linear velocity of the wheelchair is reduced until the wheelchair stops in order to avoid collision with an obstacle. This behavior is triggered if the system detects obstacles too close to it.
- A1** Obstacle avoidance: calculates (v, ω) to steer the wheelchair away from the obstacle. The user input and the sensory readings are taken into account to calculate proper values of v and ω .

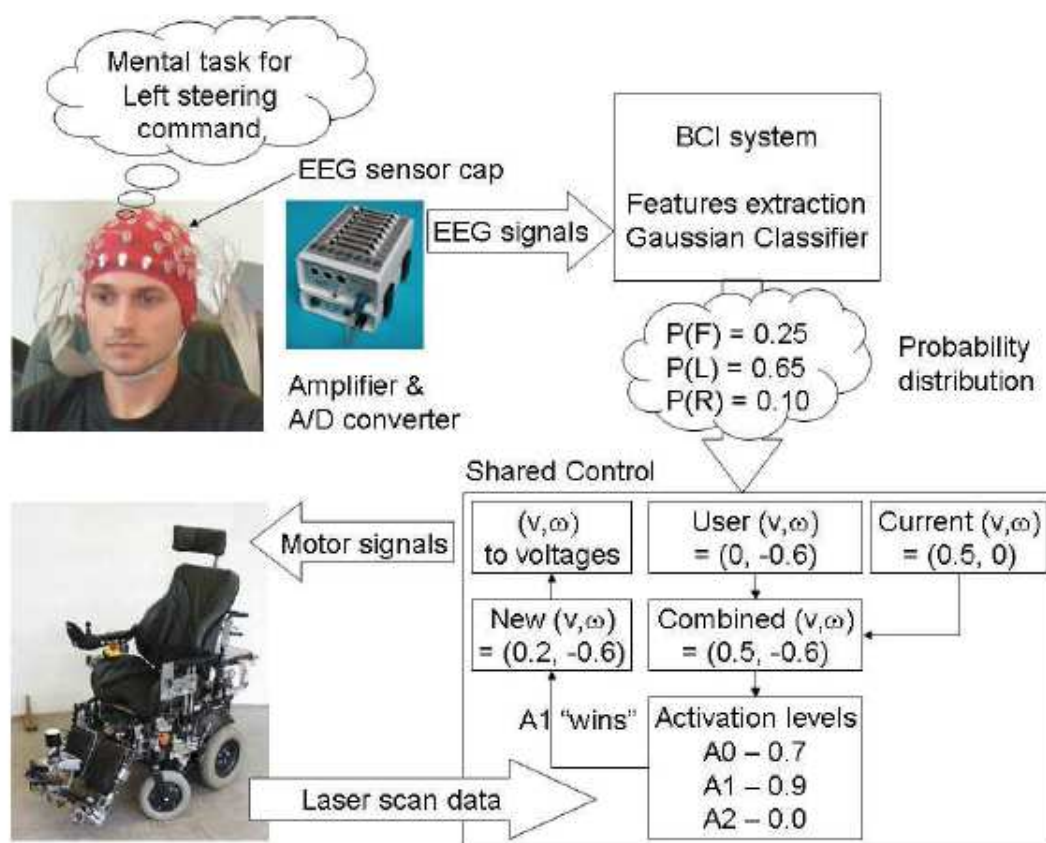


Figure 1.19: The shared-control scheme used in [54] to generate the linear and angular velocity (v, ω) of the wheelchair.

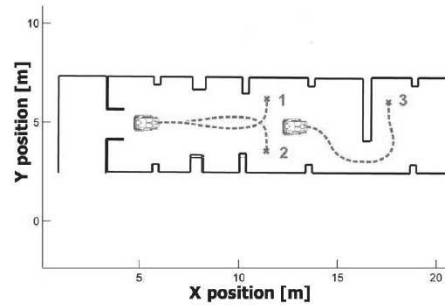
A2 Orientation recovery: corrects the orientation of the wheelchair if it is too misaligned with respect to a goal orientation.

Given this system, two tasks were performed by the subjects. The first task, executed only by the two experimented subjects, is illustrated in Figure 1.20a. The goal is for the user to drive the wheelchair through a corridor where obstacles are present. The second task, executed by all subjects, is illustrated in Figure 1.20b. The goal is for the user to drive the wheelchair to a target position (three targets are proposed) while avoiding obstacles in the scene. A trial is considered successful if the user stops the wheelchair less than 2 meters away from the target. The second task was also performed by a “random” BMI to be used as a baseline for performance comparison and assess the control of the wheelchair by the users.

These experiments were very successful in showing the ability of the users to drive the wheelchair in a real environment despite a lack of experience and BMI training. During the task 1, the users were able to drive the wheelchair from one extremity of the corridor to the other even at first trial. However, each trial would take about 6 minutes and the users reported fatigue due to the long usage of the BMI system. Nevertheless, control was mostly achieved by the user and not the assistive system as the assistance was active 20 to 35% of the time



(a) Task 1: the user goes from one extreme of the room (in the background on this picture) to another



(b) Task 2: the user has to drive the wheelchair to target 1, 2 and 3. In the figure, the starting position and the ideal trajectory are shown.

Figure 1.20: Tasks performed on the brain-controlled wheelchair in [54]

depending on the trial.

In task 2, the 37%, 53% and 27% of the targets while the “random” BMI only reached 13% of the targets. This shows a high degree of control of the wheelchair, even by an inexperienced user, in a more difficult task that requires fast and precise actions. Interestingly though, the users also had to adapt their strategy for reaching the targets to the wheelchair shared-control system as they first tried to reach targets 1 and 2 through a straight line which was then modified by the obstacle avoidance routine, thus leading to unsuccessful trials.

The brain-controlled wheelchair project is important within the context of our work because it was the first work demonstrating the ability to control a complex robotic device in a real environment with a non-invasive BMI. They also explicitly demonstrated the importance of the shared-control approach in such context as a way to work around the limitations of the BMI. Using similar principles, other projects have successfully allowed disabled people to control a wheelchair with a BCI. Notably, in [61], the user can select a pre-defined destination using P300 and the wheelchair automatically drives to this destination while ensuring its safety and the safety of its passenger. However the user can still stop the wheelchair using a faster BCI paradigm. This enables the user to move the wheelchair to the destination of her/his choice in an efficient way.

1.3.3.2 Using SSVEP to control a robot

While the SSVEP modality has been used in a wide variety of applications as we have seen in an earlier section, it was merely used in robotic applications. In this regard, the work presented in [31] is rather unique as in this work the SSVEP modality is used to drive a simple robotic device, as seen in Figure 1.21.

The system relies on the use of an external box—seen on the left side of Figure 1.21—to elicit the SSVEP in the user’s brain’s signals. The box has a LED mounted on each corner, each LED flickers at a different frequency that the system was trained to recognize. Each

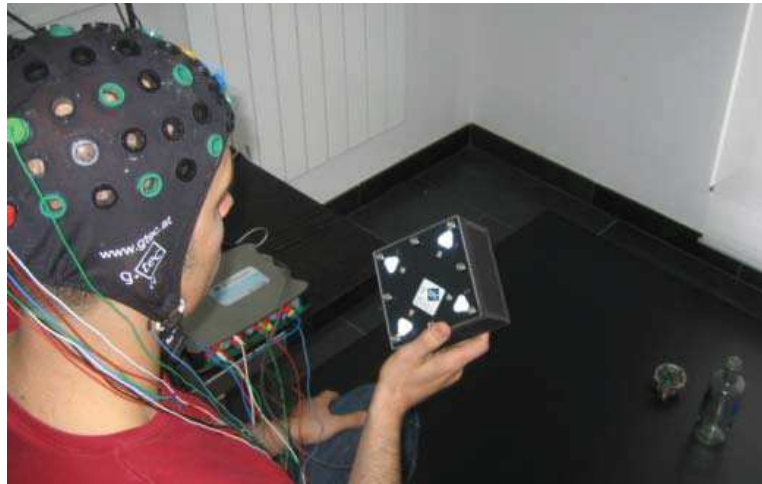


Figure 1.21: The robot and SSVEP control device used in [31]

LED is associated to a command that is sent to the robot: the upward LED moves the robot forward, the right LED turns the robot to the right, the downward LED moves the robot backward and finally the left LED turns the robot to the left. Using such a system 3 subjects were able to move the robot along a path designed by the experimenters with a high degree of accuracy.

This performance was made possible by two major features of the BMI system:

- A fast-pace (dubbed “quasi-continuous” by the authors)–4Hz–classification output with a low error-rate–below 10%.
- A “zero-class” implementation that allows the BMI to understand that the user wishes to do nothing at some point of the control.

1.3.3.3 Using P300 to control a robot

P300 has proven to be more popular for the control of robotic device through a BMI than the SSVEP modality as many teams tried to tackle this issue through a P300-based BMI [62] [63] [64].

In this section, we focus on the work presented by Escolano *et al.* in [63] as some of the principles employed in the design of the interface and the teleoperation architecture that was setup by this team have been inspirational to our work.

The robot controlled in [63] is a commercial Pioneer P3-DX, seen in the bottom part of Figure 1.23, a two-wheeled robot equipped with a laser sensor for obstacle detection and a video camera to provide video feedback to the robot’s user. The robot is steered through a high-level controller that allows it to go to a selected point in space while avoiding obstacles. The experiment involved 5 subjects who were given two tasks to execute. The first task was to go from a starting position to a goal position; however, an indicator present on the path of the robot conditions the attitude that should be observed regarding an obstacle on the road, moving around the object either through the left or right side. The second task displayed a similar scenario but with a different and longer room as well as a different cue

for the indication. These scenarios were designed to enforce the use of both control schemes offered to the user: navigation and camera control.

The two specificities of this work are:

- The “switching” interface that allows the system’s user to switch between two control schemes autonomously.
- The teleoperation aspect of the experiment as the user of the BMI and the robot were separated by roughly 300 km.

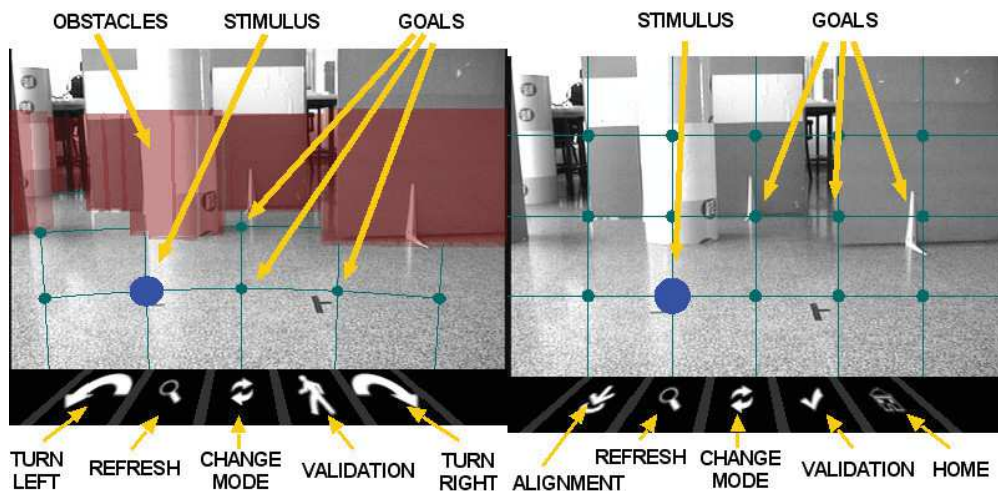


Figure 1.22: P300 interface devised in [63] to control a mobile robot. Left side shows robot navigation mode. Right side shows camera orientation mode.

One of the advantages of the P300 protocol is to allow the BMI to choose among a large number of possible outputs with a high degree of success. The work presented in [63] makes good use of this feature by offering a rich interface to the user of the system.

The interface that is displayed to the user is shown in Figure 1.22. On the left side, the interface is in “navigation mode” while on the right side it is in “camera control mode”. In both cases, the screen is divided into a 5×4 grid. The upper part of the screen is used to display the image coming from the robot’s camera. The grid-division allows the user to select one out of 15 points in the image. The bottom part is reserved for the control of the interface.

In navigation mode, selecting one of the points in the robot’s field of view moves the robot to this position. However, you can notice in Figure 1.22 that some parts of the grid are covered by a red layer. This red layer is there to indicate that the robot cannot navigate to this specific point because of an obstacle. The interface buttons have the following function, from left to right: (a) turn the robot left by 45° , (b) refresh the camera picture to get the current view as well as updated obstacles, (c) switch to camera control mode, (d) validate the previous selection, this is explained later, and (e) turn the robot right by 45° .

In camera control mode, selecting one of the points in the robot's field-of-view directs the robot's camera to this point. This is not limited by the presence or absence of obstacles; hence, all points are available for selection. The interface buttons have the following function, from left to right: (a) align the robot with the horizontal camera orientation and switch to navigation mode, (b) refresh the camera picture to get the current view, (c) switch to navigation mode, (d) validate the previous selection and (e) set the camera to its initial orientation.

In both modes, the execution of any action, except the "refresh" option, is conditioned by the validation of the action through the use of the validation button. The authors do this in order to ensure high fidelity between the intention of the user and the actions executed by the robot. Once an action has been validated, the interface waits for the execution of the action by the robot and performs another round of P300 stimulation to let the user select the next action.

The use of such a switching interface is an interesting feature that shows that a single BMI command does not have to be related to a single robotic command and thus can be dynamically modified to increase the range of actions that can be executed by the user. The integration of sensor information in a visual way done in navigation mode is also an interesting feature as it increases the degree of collaboration between the robot and the user. Finally, the confirmation is another important idea as it allows overcoming one of the limitations of the BMI—a relatively high error rate compared to traditional input interfaces—that could limit the interest of such a high-level controller.

The teleoperation scheme used in [63] is shown in Figure 1.23.

The interesting features to notice are:

- All interface logic, including the augmented reality overlay, and the signals treatment are executed on the BMI site
- All robot logic is executed on the robot site

The only information transferred between the two sites is the video and obstacle information coming from the robot to the BMI on the one hand and on the other hand, high-level commands issued by the BMI that transit from the BMI to the robot.

The main drawback with P300 control of robotic devices is the lack of reactivity that the interface provides. To elicit a P300, it is necessary to display the flashing stimuli a certain number of times so that the event expected by the user is indeed uncommon. Therefore, the BMI system usually reaches two to three decisions per minute [65]. This issue could be partly tackled through the development of asynchronous P300 paradigm such as the one proposed in [66]. However, such paradigms do not allow reactive behavior as the decision is still locked to the many repetitions of the P300 pattern necessary to elicit an answer.

1.3.3.4 Humanoid Robot Control Using P300

Finally, we present the only work prior to ours that focused on the control of a humanoid robot using a BMI system. In [64], Bell *et al.* introduce a P300-based BMI that is used to

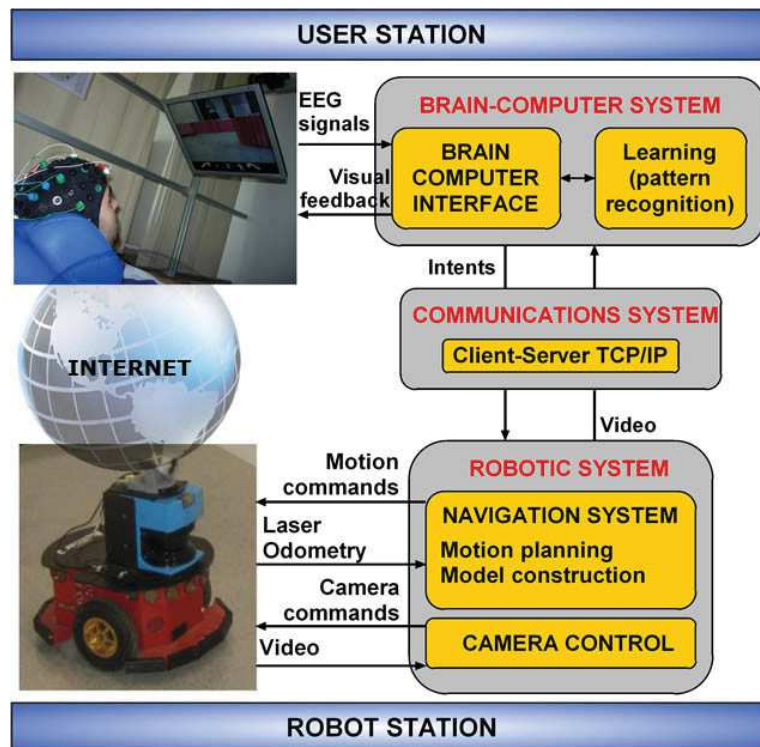


Figure 1.23: The BMI teleoperation architecture presented in [63].

control a humanoid robot. The goal of the experiment is fairly simple: pick up an object and place it on one of two possible destinations.

The robot used in this experiment is a Fujitsu HOAP-2 humanoid robot. It is equipped with a pan-tilt camera that is used to provide visual feedback to the user and locate objects in the environment. The robot is able to walk autonomously, navigate based on visual information and pick up and place objects. Furthermore, the robot has prior knowledge of the room layout. Finally, a second camera observes the scene from above and is used to locate the destinations available for the robot. The room layout can be seen in the upper picture of Figure 1.24. The red and green bricks that can be seen are the objects the robot should grasp and the blue zones are the “tables” where the robot should put the object.

The robot starts the experiment in front of the objects that the user should select. As the robot moves forward these objects, the image is being analyzed and segmented by the robot to detect the objects of interest. This information is then transferred to the interface and a P300 selection cycle begins. As seen in the bottom left picture in Figure 1.24, the detected segments are displayed with a fine red border around them. This border flashes—the timing of those flashes are discussed in [64]—and thus elicit a P300. Once the BMI has reached a conclusion about which object should be grasped, the robot autonomously grasps the object—bottom right picture in Figure 1.24. Afterwards, the interface switches to the overview camera. Then the user can select the dropping zone in the same way as with the objects. Once this selection has been made, the robot autonomously moves to the drop position and put the object on the table.

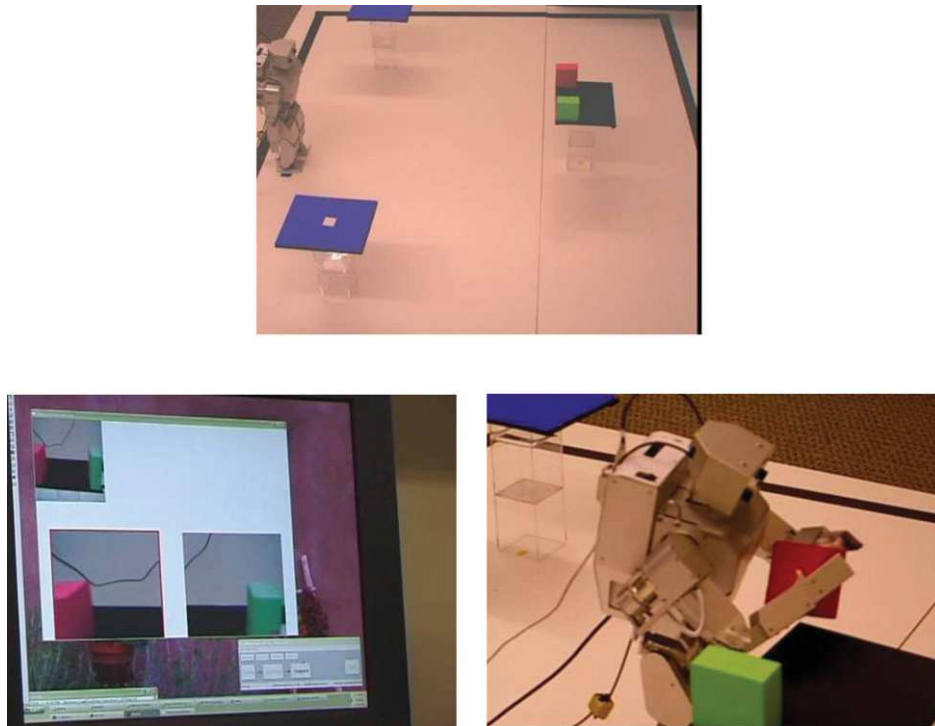


Figure 1.24: Overview of the humanoid control application (from [64]): (A) The robot in its arena: two objects can be chosen and two destination tables; (B) The interface shows the robot view and the detected objects; (C) The robot executes the action mandated by the user

The experiment conducted by Bell *et al.* can be seen in [67].

While this experiment is interesting for being the first work where a humanoid robot is controlled through a BMI, it is very far from exploiting further possibilities in using a humanoid robot. The interface offers very few choices to the user and the user has no control over the execution of tasks carried out by the robot. However, it shows that, using high-level commands, BMI can be a practical way to control a humanoid robot to perform complex tasks.

1.3.3.5 Robotic Arm Control using a brain implant

In a recent remarkable work [10], the motor activity of a subject's brain, acquired through an intra-cortical implant (BrainGate system), was decoded into movement inputs for a robotic arm, allowing the subject to move the arm freely in its operating space, grasp a bottle and bring it to the subject's mouth, as seen in Figure 1.25. This is made in a quasi-closed-loop fashion.

This study exposes the results of robotic arm control in 3D space performed by two subjects, later referred to as S3 and T2. Both of them have experience controlling a mouse cursor for point and click actions that they practiced weekly prior to the experiments for a long time—respectively 5.3 years for S3 and 5 months for T2. Two robotic arms were used during the experiments: the “DLR Light-Weight Robot III” and the “DEKA Arm System”.

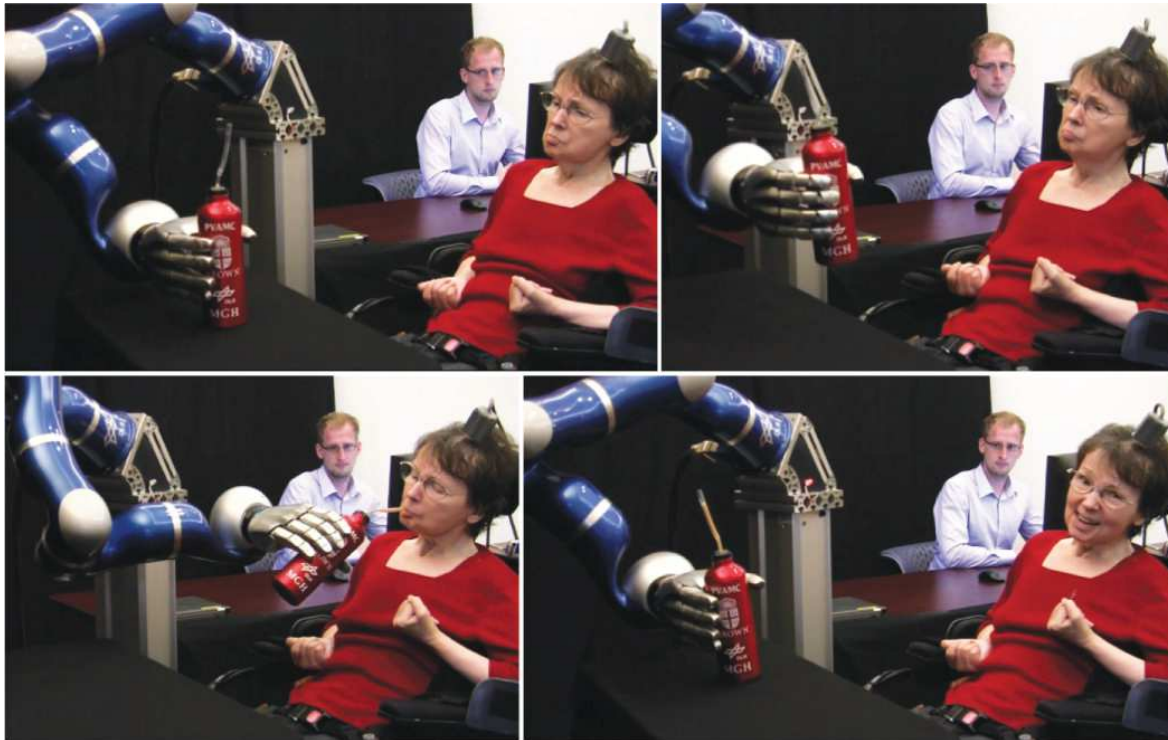


Figure 1.25: A subject controls a robotic arm through implanted electrodes to reach a bottle, grasp it, move it towards her mouth and put it back on the table (taken from [10]).

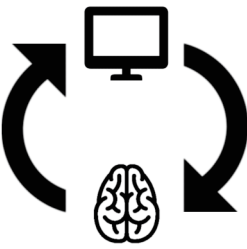
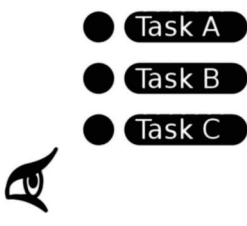
However, both subjects had little experiment with the robotic arms other than the assistance they provided during the development of the control algorithm and interface testing. The BMI system has to be calibrated before their use in robotic applications. On average, this procedure took about 30 minutes. The system allows the subjects to control the arm speed in 3D and trigger grasping/relaxing of the hand. Additional constraints are placed on the robotic arm movements to avoid collision with the table and the subject. Notably, the subjects cannot control the hand orientation.

The main experiment consisted in reaching and grasping targets—foam balls maintained in a fixed position in space through motorized levers—within the operational space of the robotic arms. Since the subjects were unable to control the orientation of the hand and since touching the rods maintaining the balls in the air would move the balls, the task was fairly complicated to achieve. A successful “touch” trial occurred when the subject touched the target with the hand, while a successful “grasp” trial occurred when the subject closed the hand while any part of the target or the top of support were in the hand. Given these conditions, S3 achieved 48.8% with the DLR arm and 69.2% with the DEKA arm of “touch” success—given 80 trials with the DLR arm and 78 with the DEKA arm. T3 achieved 95.6% “touch” success over 45 trials with the DEKA arm. Of these successful touches, S3 achieved successful grasps in 43.6% of DLR trials (21.3% of total DLR trials) and 66.7% of DEKA trials (46.2% of total DEKA trials). T2 achieved successful grasps in 66.7% of trials (62.2% of total trials).

The second experiment, shown in Figure 1.25, consisted of reaching and grasping a bottle with a straw inside, bring the bottle with the straw towards the mouth of the subject, let the subject drink from it and put the object back. Only Subject S3 performed this experiment. The control was restricted to the two-dimensional plan of the table and grasping control was used to trigger transitions between different hand actions that depended upon the phase of the task and the position of the hand. Out of six trials, the subject was able to perform the task four times. In the two failed attempts, the task was aborted, as the hand was about to push the bottle off the table because it was not properly aligned with it to allow the hand to grasp it.

1.4 Conclusion

Independently of the technology used to acquire the user’s brain activity, all BMI-based control application can be divided into two categories.

 <p>Neurofeedback based</p>	<p>This is the case of applications that try to understand the activity – often the motor activity – of the brain and translate it to the controlled device. This includes works that we have presented previously such as the intelligent wheelchair for EEG-based BMI [54] or arm control in ECoG-based neurofeedback [10], but also the control of a humanoid robot in fMRI-based BMI [68]. We later refer to this family of application as “NF-based” applications.</p>
 <p>Menu based</p>	<p>This is the case of applications that use stimuli, usually visual: P300 or Steady-State Visually Evoked Potentials (SSVEP), to map a behavior on the robot through the use of a “menu”. For example, among the works we presented, one can cite the work done in [31] where a mobile robot is controlled through SSVEP stimuli or [63] where a complex P300 interface allows the user to remotely control a mobile robot and its embedded camera. We later refer to this family of application as “menu-based” applications.</p>

The impressive works we introduced earlier illustrate a dominant trend in current BMI-based control. On the one hand, decoding the brain activity in order to map it onto the controlled device; this is constrained by the fact that controlling the Cartesian motion of a robotic arm by neuro-feedback requires a long and sustained training for the subject and usually perform with less than perfect accuracy and slowly compared to the capacity of the controlled robotic systems they use as substitutes or surrogates. Furthermore, we can question the actual need–beyond therapeutic benefits–of controlling an external device with such a degree of precision to perform simple tasks.

On the other hand, menu-based control, while closer to what we will introduce as object-centric BMI, still limit the capacities of the robot to a few pre-programmed behaviors.

1.4.1 Limitations of menu-based control

Menu-based BMI are easy to learn. In both cases we presented here—and other menu-based BMI studies—novice subjects are able to control the device with ease and perform well within the evaluation scenarios after a few minutes of training. However, through these examples, we can also see the two major drawbacks of menu-based control as they have been conceived until now.

On the one hand, the case of the SSVEP-based control of a mobile robot or a humanoid robot highlights the inability of the users to gain precise control over the movements of the robot. This is due to the input limitations brought on by the use of BMI technologies. The mapping between a discrete number of stimuli and the speed of the robot—a continuous value—has to limit the ability of the user to control the device to its full capacities. While in the case of a simple mobile device, such as the one presented in [31], the mapping can be satisfying enough, in order to allow more complex devices to be controlled, the interface has to become more complex.

On the other hand, complexity has a cost. In the fairly simple case of humanoid robot steering, the users struggled to understand and use the extra-capabilities offered by the interface, making it pointless compared to the simpler version. In the P300 interface we have presented, all the robot's capabilities can be controlled through the BMI interface. That is, the user can move the mobile platform almost everywhere it can go and the user can look almost everywhere the mobile platform can look. However, this is a fairly simple device with limited capacities. Especially, it is not capable of interacting with the environment in any way. Nevertheless, the interface to control the device is complex and tedious to use: moving somewhere requires to first select a point on the grid and then confirm this selection, this requires two accurate selections with P300 which can take up to a minute or even more if the BMI makes a mistake: the interface has to be simplified.

A menu-based BMI has to tackle a fundamental problem: it has to be simple to allow an efficient control of the device but it also has to be complex to offer all the capacities of the controlled device to the user.

1.4.2 Limitations of NF-based control

Within NF-based BMI, we include another common trend that is the use of motor intentions to guide a complex controller. For example, in [69], a humanoid robot is controlled by a NF-based BMI. In this case, a humanoid robot is controlled by an optimization-based controller that takes into account an initial configuration and a final configuration—given manually by the user—and autonomously plan the appropriate movements to go between those two configurations. The user of the BMI intervenes to adjust the position of the waypoints in the movement. In that case, the controller was designed first and the controller was adapted to include BMI inputs.

On the other hand, in [70], the authors designed a language to devise planar trajectory for a mobile aircraft with high-latency and low bit-rate input devices in mind, such as a BMI. As a result, the users of the BMI were able to guide a virtual airplane along fairly complex

paths. However, the controller is also quite complicated and some users were unable to guide the airplane. Moreover, the navigation did not take obstacles into account.

In the end, these systems are limited to simple locomotion task because EEG-based BMI and neurofeedback do not allow understanding the motor intentions with sufficient precision, and the BMI, whether the applications are adapted to it or designed for it, is used a binary input device. To obtain a better recording of the motor area, it is possible to record it through intra-cortical devices such as ECoG. Since it is an invasive acquisition technology, the use of ECoG in such applications has been rare.

The results presented in the ECoG studies are amazing: it shows the viability of such implants for rehabilitation, it allowed the people involved in this study to physically interact with the world from their own will for the first time since they became paralyzed and, it might be possible to use MI signals to reanimate the paralyzed muscles [71]. Yet, the results are also a bit unsatisfactory. Reaching and grasping an object is a simple task to achieve for a robot. The difficulty and the length of the task, as well as the eventual failure at realizing it, come from the difficulty to achieve an accurate decoding. Improvements are mostly expected from resolving all the issues revolving around the acquisition to achieve higher accuracy, faster decoding and natural control.

1.4.3 Introducing “object-centric” BMI

Nowadays, many teams have successfully achieved the control of a robotic device through BMI. However, the control has been limited to either simple devices that cannot achieve much or simple tasks that do not exploit the full capabilities of the controlled device. Our practical objective is to allow the BMI user to control the device as a surrogate to accomplish tasks on her/his behalf that s/he is not able to achieve by himself anymore. In this context, we believe the decoding of motor intentions to map it onto the operational space of the controlled device is cumbersome. As seen in [10], even using state-of-the-art method to decode the motor intentions, the results are still far from a full-proof and intuitive ways to control robotic devices for surrogate purposes. Without questioning the therapeutic benefits of this work, we can question the interest of achieving control through motor intentions detection as this is not such a conscious process that guides our actions but rather objectives. For example, assuming that we could understand the full movement intentions of a hand, i.e. decoding the full translation and orientation intentions, and assuming that we had a robotic hand capable of executing such movements, would this be practical to use?

This comes down to the general issue of goal selection versus process control strategies in BMI. In a goal selection strategy, the BMI is used to select the goal of the process while in a process control strategy (e.g. select an object that is grasped by a robotic hand), the BMI is used to control the details of the process (e.g. control the robotic hand trajectory to grasp an object). In [72], it was shown that the former leads to faster, more reliable and more accurate BMI applications. Given these considerations, we introduce “object-centric” BMI for robotic control.

“Object-centric” BMI aims at channeling the interaction between the BCI and the robot through the objects within the environment of the robot. This allows the user to perform

complex robotic tasks despite the intrinsic limitations of EEG-based BCI. Rather than mapping the motor intentions of the user to equivalent robotic actions or mapping external stimuli to single tasks, we chose to let the user interact directly the objects and infer the task s/he wishes to perform from the environment state perceived by the robot. It is achieved by a tight integration of artificial intelligence and computer vision tools and shared-autonomy approach such as those employed in previous BCI works that we have presented here. It is thoroughly described in Chapter 3.

Following this state-of-the-art (that is far from being exhaustive), we discuss in the next chapter the specific aspects of building rich BMI applications that we have developed. This includes a presentation of the interface framework to build complex and brain-controlled robotic applications and two studies we conducted that highlight particular aspects of BMI performance under typical conditions for robot control.

Building Interface for Robot Control with SSVEP

As discussed in the previous chapter, BCI have been used in many different applications including the control of robotic systems. However, early robotic systems controlled by BCI remained proof of concepts work (some advanced) and as such have not dived into all the problems raised by the use of BCI to perform complex robotic systems' control. In order to achieve this, we introduce the “object-centric” BMI, a novel approach of BCI for complex robotic systems' control. This approach, thoroughly discussed in Chapter 3, benefits from the task-function based approach and a tight integration between the environment perceived by the robot through its sensors and the interface that we propose to the user in order to obtain her/his intentions. In this chapter, we introduce the base of such an interface as well as BCI issues, which are specific to the robotic context (and peculiarly to humanoids) and the “object-centric” BMI context.

The interface that realizes the link between the robot, its environment and the user of the system should achieve the following objectives:

Multi-task capability

In order to achieve complex robotic missions, we have to be able to perform a succession of elementary tasks. For example, a task such as “fetch X object from the cupboard” requires to be able to (a) navigate the robot and (b) fetch an object. This means that the interface has to be able to chain multiple tasks within a single run.

Dynamic interface

Interacting with the objects in the environment is a key-feature of the “object-centric” BMI framework. In order to achieve this, we need to report in real-time the capacities of the robot within its environment. Therefore the interface should allow us to display stimuli or information in a dynamic way, that is, based on the current environment of

the robot.

Reliable goal selection

Since the “object-centric” BMI relies on the selection of object or goal to autonomously realize the intention of the user, it is important to ensure that this selection is accurate. While we may recover from a mistake, by using ErrP or a manual cancellation mechanism for example, it is obviously more desirable to initially understand the right intention to ensure a smooth experience for the user.

To achieve these goals, we introduce a novel framework to develop BCI applications in the first section of this chapter.

Although the later framework is agnostic on the underlying BCI paradigm that is used to understand the user’s intentions, we showed in Chapter 1 that, within the current EEG-based BCI paradigms, the SSVEP paradigm fits our needs better. Indeed, on the one hand, P300 does not allow to generate the kind of dynamic interfaces we envision and does not provide a reactive interaction due to the nature of the stimulation. On the other hand, motor imagery does not rely on visual stimuli and thus, while well adapted for locomotion tasks, does not fit well within the “object-centric” BMI. However, while SSVEP allows us to build a reactive and dynamic interface, it is also prone to interpretation errors which is in conflict with our reliability objective. Therefore, in the second and third section of this chapter, we address two important SSVEP performance issues related to the use of this paradigm in robotic control. First we look at the performance problem induced by having SSVEP stimuli displayed as an overlay on (often jerky) video feedback, e.g. video feedback from a robot’s embedded camera. We propose several solutions to limit the latter drawback. Then, and finally, we present a study that we conducted regarding the best way to display SSVEP stimuli for object selection, which is obviously of prime importance within the “object-centric” BMI framework.

2.1 Introducing the bci-interface: A Framework for Dynamic BCI Application

Several BCI frameworks, such as OpenViBE [4] or BCI2000 [5], are developed to produce high-quality BCI applications. The three layers model introduced earlier is implemented by almost any BCI framework. Yet, most of the development effort is put towards the signal acquisition and signal processing layers leaving the user application to the user. Therefore we developed a framework to address this issue and easily write BCI applications: the bci-interface.

This framework is developed with two important goals in mind.

First, it should be simple to use: the application developer should only worry about what s/he wishes the BCI user to accomplish. This is realized by providing simple bricks to the developer that s/he can arrange and interact with easily without considering the underlying mechanisms of the interface. We introduce our approach later in this section and showcase the framework simplicity through examples.

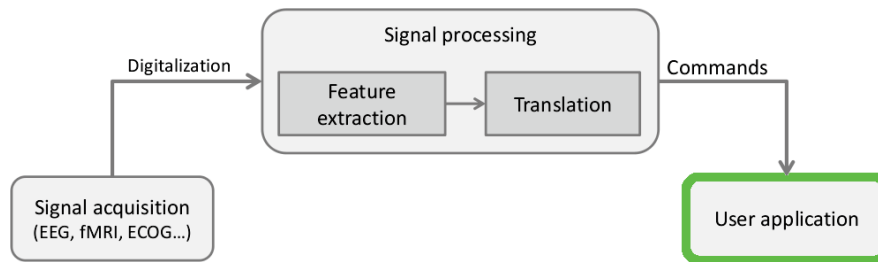


Figure 2.1: The *bci-interface* focuses on the user application layer in the BCI2000 model

Secondly, we want to be able to write dynamic BCI applications. Usual BCI applications are aimed at achieving one simple task: steering an avatar, controlling a cursor, selecting objects or bins. However, our goal is to allow a BCI user to take control of a humanoid robot and perform complex tasks. This implies using multiple paradigms as one interface cannot fit all of our needs. For example, displaying SSVEP arrows to control the humanoid steering may be satisfactory. Yet, these same arrows become cumbersome when we want to achieve grasping and another method has to be developed—as we see later—or these arrows could also be used to control the humanoid’s gaze or a limb. More importantly, we definitely want to be able to control all of these within a single application. It is then necessary to be able to dynamically switch between different BCI control schemes, a possibility offered by *bci-interface* that is also showcased in the examples.

The implementation details of the library are provided in Appendix A.

2.2 Dynamic Background and SSVEP Stimuli Interactions

In this section we present an experiment we carried out to determine the influence of the video feedback on the brain-computer interface performance of a system we designed to steer a humanoid robot. First, we introduce the interface used for the control and the motivation of the study. Then we present the experiment that was conducted, the results it brought and how we adapted our interface to reduce SSVEP false positives.

2.2.1 Motivation

At the very start of the project, our aim was to try out simple set-up to catch-up with the state-of-the-art. The first BCI application we designed [73] uses the *g.tec* system and is based on the SSVEP paradigm that we adapted to steer a humanoid robot. In this application, we made so that the SSVEP stimuli are blended into the video feedback from the robot in order to offer an immersive experience to the user. Indeed, the interface provided by *g.tec* consisted in a separate box composed of four flashing LEDs, which was nearly impossible to envision in an immersive set-up. Our interface can be seen in Figure 2.2. At that time we hypothesized that the SSVEP classification performance is not affected by the video feedback.

Initially, the user of the BCI is trained with a still image from the robot’s camera. Indeed, as the robot doesn’t move during the training, the background video is nearly static. To make

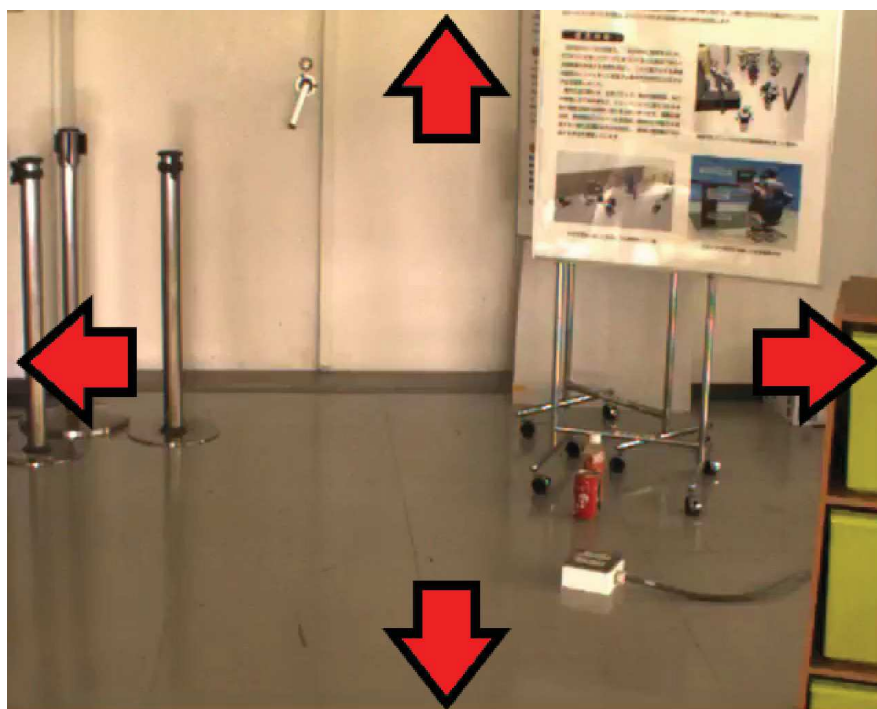


Figure 2.2: The interface used to steer the robot. Each arrow is rendered with own frequency that is different from the others. Stimuli are integrated within the video feedback of the robot's embedded camera. The robot moves to the direction pointed by the arrow the user is focusing onto. If no arrow is currently emphasized, the robot stops moving.

sure that the training is successful, the user tries to “activate” each stimulus successively by focusing on each designated—by another user—flickering arrow. If the user deems the performance of the interface satisfactory s/he can try to control the robot by thought, otherwise s/he undergoes another training session until success.

This demonstration was performed several times with different users. However, while the users might feel confident about the performance of the interface during this test run, they often complained as the interface was less responsive during the actual control of the robot. This led us to reconsider the hypothesis that the performances of the classifier were not affected by the video feedback provided during the control of the robot. We later refer to “video feedback” as the background video displayed to the user in the interface that is fed back from the robot's embedded cameras. The term “static video feedback” refers to a still image shown to the user, i.e. while the robot is not moving, while the term “dynamic video feedback” refers to moving images due to the robot movement.

The study we investigated aims at measuring the effect of dynamic video feedback on the performance of our BCI system; if this effect is indeed observable, we propose the “natural” solution of training the user with dynamic video feedback to account for the modification in the background that might affect the SSVEP response such as variation in luminosity and distraction induced by the *moving background*. This solution is tested in parallel with the initial study. In the following section we present the experimental protocol designed to

measure the effect of the dynamic feedback and our proposed solution to solve this issue. The results of this experiment are then presented. Finally, we discuss the outcomes of this experiment and introduce an alternative solution.

2.2.2 Material and Methods

The following section presents the experiment we conducted with five subjects regarding the impact of video feedback on BCI performance during steering control.

2.2.2.1 Subjects & Material

Five healthy volunteers participated in the experiments, four males and one female, aged 24 ± 1.4 y. One of them, later referred to as Subject 1, had previous experiences with the BCI steering interface, the others however had no experience of BCI usage.

We use a g.USBamp (24 Bit biosignal amplification unit, g.tec Medical Engineering GmbH, Austria) to acquire the EEG data from the user's brain at a sampling frequency of 256Hz, bandpass filtered between 0.5 and 60Hz with a notch filter at 50Hz to get rid of the power line noise. The electrodes positioning is shown in Figure 2.3. We use 8 Ag/AgCl active electrodes. The electrodes are placed on the POz, PO3, PO4, PO7, PO8, O1, O2 and Oz positions of the international 10-20 system [74], Fpz is used as the ground electrode and the earlobe as a reference.

The experiments are conducted with a HRP-2 humanoid robot running the control system described in the previous section.

During the experiment, the user is comfortably seated in an armchair about 1 meter away from a 17" LCD screen operating at a refresh rate of 60Hz upon which the user interface described in the previous section is displayed.

2.2.2.2 Experiment Protocol

These experiments have two goals:

- Assess the negative impact of video feedback on BCI performance in the experimental conditions described in [73], and
- explore the effect of training the user with dynamic video feedback.

Therefore we evaluate the performance of the BCI system under two conditions: (i) static video feedback and (ii) dynamic video feedback, when the user has been trained under two conditions: (a) train with static video feedback and (b) train with dynamic video feedback.

Training under static video feedback condition

In this condition, the robot is static and the user is showed the feedback from the robot's camera. The user sees the interface described earlier and s/he is given visual instructions to attend to one specific arrow. The user has to look to the arrow for 10 seconds, there is a

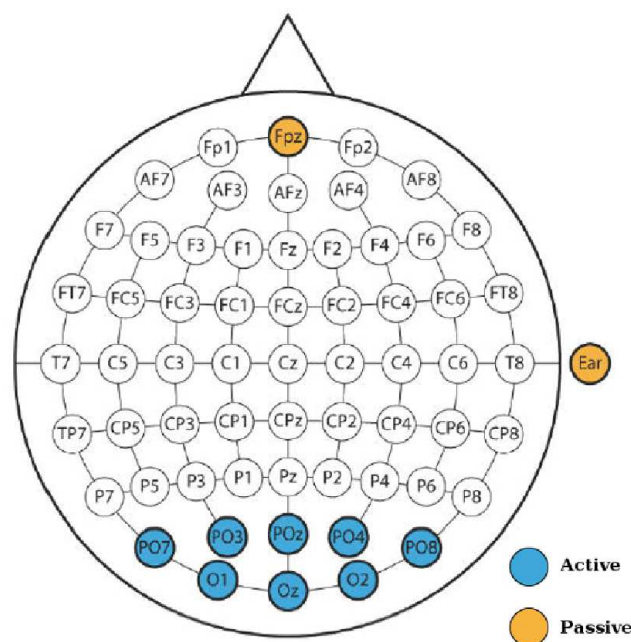


Figure 2.3: Electrodes positions for the experiment

3 seconds break between each arrow and each arrow has to be attended 5 times, therefore the whole training takes about 4 minutes and 30 seconds. The order of the indications is pre-defined (a cycle is always up, right, left and down) and known to the user. Even though the classifier—computed from previous experiments—is also running during this training, the user is shown no feedback.

Training under dynamic video feedback condition

This condition is similar to the training under static video feedback except that when the visual indication is given to the user the robot starts moving toward this direction to reproduce real dynamic conditions.

Performance evaluation under static condition

This condition is also similar to the training under static video feedback but the display order of the arrows is random and unknown to the user. The user is given feedback about his performance; if the classifier's output is different from zero, the corresponding arrow is highlighted.

Performance evaluation under dynamic condition

This condition is similar to the previous one but the output of the classifier is used to drive the robot.

As BCI performances tend to decrease over a unique session, the experimental conditions are shuffled between the subjects in order to give unbiased results. However, for practical reasons, performance evaluations under a similar training are done consecutively. Therefore,

one starts training with either static or dynamic video feedback then undergoes performance evaluation under static and dynamic condition (in whichever order), then trains with the other video feedback condition and finally undergoes performance evaluation under both conditions.

2.2.3 Experimental Results

We present the results of previously described experiments. In Figure 2.4, the individual results of each subject are given. For each training condition we give the average performance of the classifier for all classes. False positive rates, however, are not reported, as they are similar across all conditions and fall below the 10% rate.

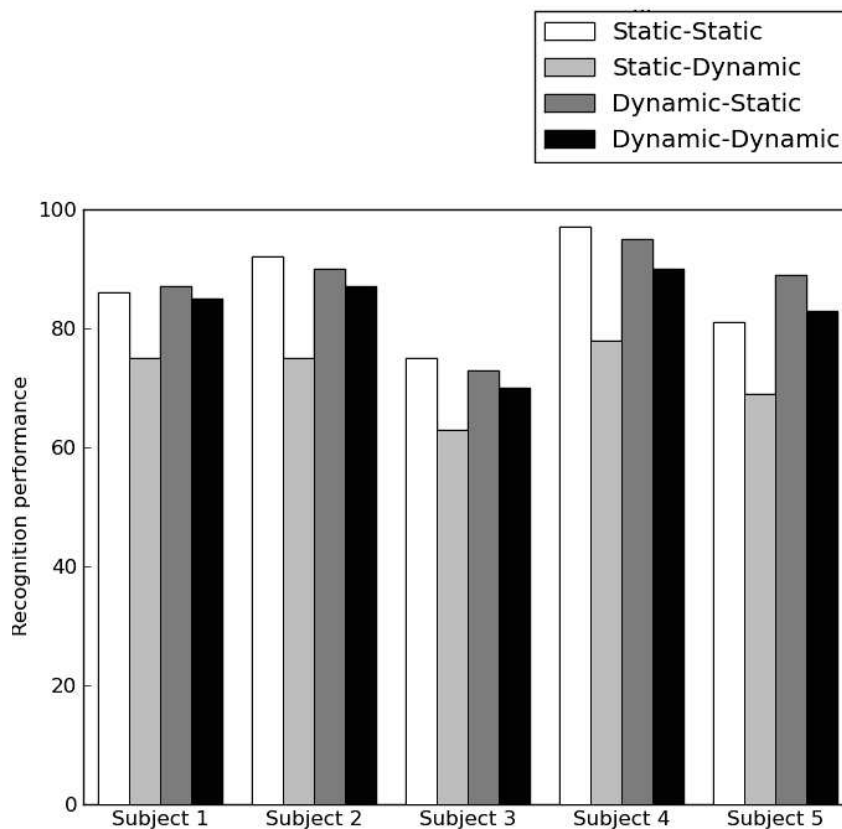


Figure 2.4: Individual results of the experiment

The first result that pops up is that, in accordance with our users' feedback, the classifier performances are indeed reduced when it is evaluated under dynamic feedback condition. In average we can observe a significant 14.2% (± 3.2) performance loss between static video feedback performance and dynamic video feedback performance. This clearly explains the difficulties our users encountered when steering the robot in spite of otherwise good performances.

On the other hand, while we still witness a performance loss between static and dynamic conditions, the dynamic training had the positive effect we expected. The average performance loss falls to 4% (± 1.5) which is much more manageable.

Finally, the performance of the classifier trained under dynamic conditions is on par with the performance of the classifier trained under static conditions with regard to the evaluations under static conditions. Thus, it seems recommendable to train under dynamic conditions to ensure maximum performance under all conditions.

2.2.4 Discussion

The previous investigation shows that the dynamic video feedback negatively impacts the performance of the BCI system. We identified, at least, three possible causes of this issue:

- Distraction due to what is happening in the video
- Distraction due to what is happening around the user
- Variation in luminosity below the stimuli shown to the user

The first hypothetical cause is unlikely. In recent works [47] [42] in virtual environment, SSVEP stimuli were integrated inside the environment in a way similar to what we did with live video feedback. In these works, no performance loss is reported to support this first hypothesis.

The second hypothesis is also unlikely. While the user might indeed be distracted by the robot's movements that are happening next to her/him and, while it may impact the performance, this would affect both training conditions in the same way and there would be no difference between the two as the dynamic training does not address this issue.

Finally, the luminosity variation in the scene is the most likely hypothetical candidate cause as this would affect the shape of the SSVEP generated by the brain. This also explains the difference we can observe from similar experiments in virtual reality environment as the lighting of the scene is much more controlled in those environments than in live environment. The gain obtained by training under dynamic conditions can then be explained by the fact that the classifier is trained with a wider variety of SSVEP shapes and therefore performs better at handling variations in the scene's lighting.

2.2.4.1 Hybrid solution

While training under dynamic conditions is very much feasible, the slight performance loss observed under static conditions for some subjects can be damageable, especially with subjects who are already performing below average.

For such cases, we propose a hybrid interface that isolates the SSVEP stimuli so that they are not affected by the video feedback. In this interface, the video feedback is not given in full screen but is embedded into a static interface as illustrated in Figure 2.5. The SSVEP controls are laid around the video feedback and therefore blink above a static background;

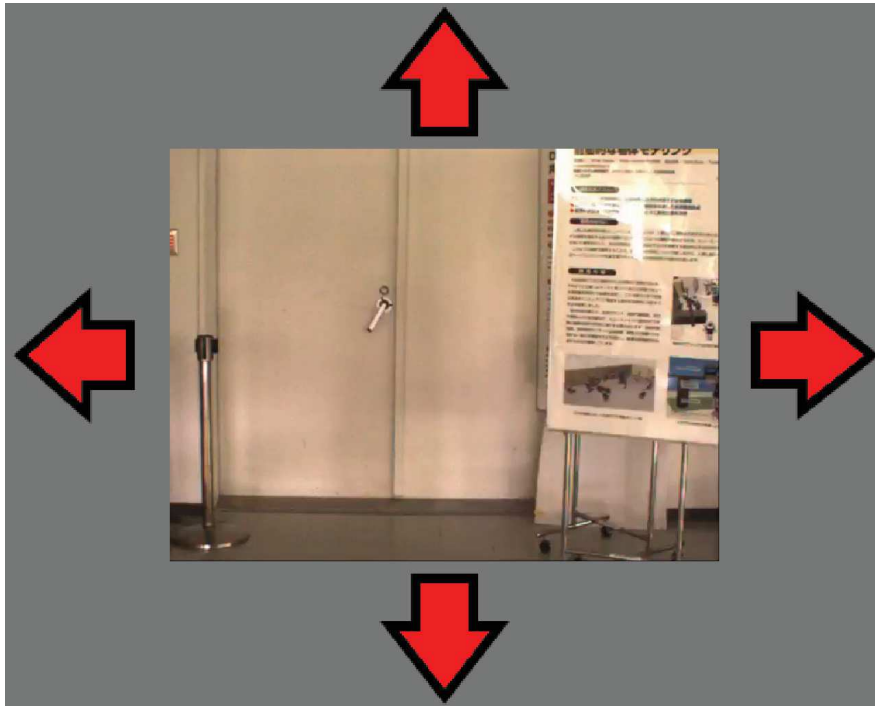


Figure 2.5: Proposed interface to compensate performance reduction with dynamic background; the video shown can be dynamic but the arrows are blinked on a static background

this removes the influence of the background luminosity on the SSVEP response. However, some users, especially those who obtain good performance, do not like this interface as much as the previous ones because they feel it is more difficult to realize what is happening in the robot environment.

2.2.5 Conclusion

We performed performance evaluations of the SSVEP-based BCI system during the steering of a humanoid robot based on negative feedback received from users during other experiments. This evaluation shows that dynamic video feedback induces a loss of performance but training under a similarly dynamic feedback can lessen this effect. An alternative solution is also proposed to render the SSVEP stimuli above static background while still showing live dynamic images from the robot. In future works, we hope to further investigate the causes of these performances loss and find novel solutions to address this issue.

It is important to consider this issue if the user interface is displayed upon an HMD since their resolution is usually rather low. Thus the “feed-in” solution is not applicable. For example, with the Oculus Rift SDK-1, that we used significantly, we have an 800×480 resolution for each eye which almost fits the resolution of the robot’s embedded camera. Thus, the interface shown in Figure 2.6 has to be used with the proper cautions that we highlighted in this section.

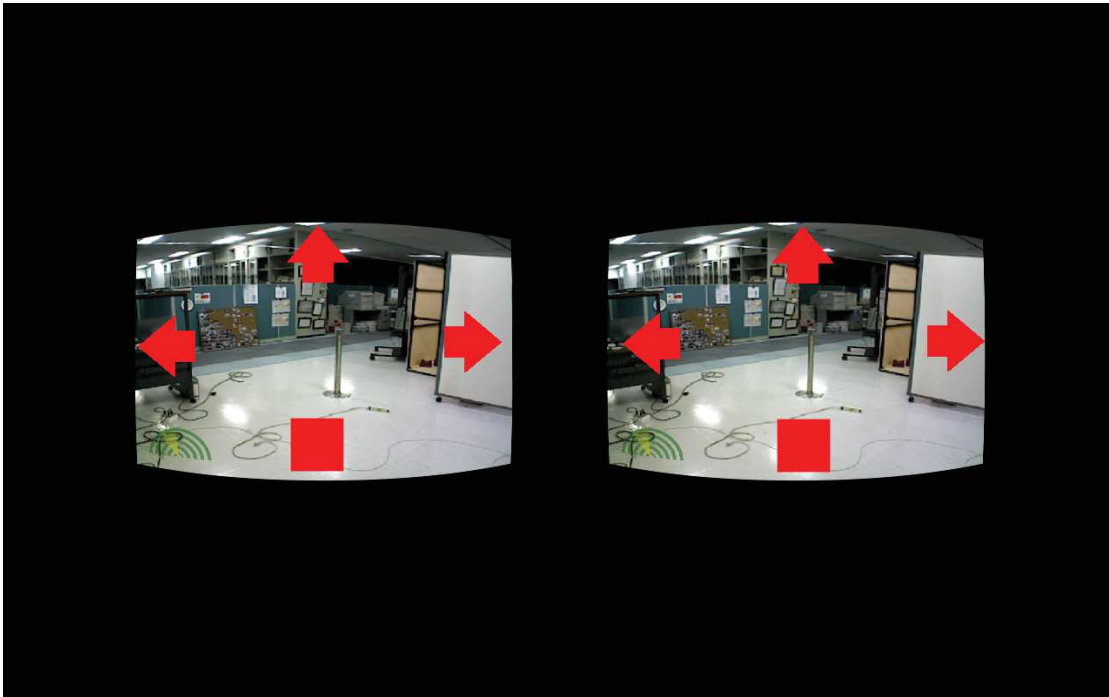


Figure 2.6: SSVEP stimuli displayed in the Oculus interface. All of the horizontal space and most of the vertical space that can be shown to the user is used by the camera feedback which prevents the use of a “feed-in” rendering solution.

2.3 SSVEP stimuli design for object-centric BCI

In Chapter 3, we propose a novel framework, the “object-centric” BMI framework, to perform whole-body humanoid robot control using a combination of techniques: task-oriented control, object recognition and SSVEP blending. When objects seen by the robot’s embedded camera are recognized, SSVEP stimuli are overlapped on those objects to allow the user to select the object of interest through SSVEP, and let the robot autonomously perform the appropriate task using affordance techniques. A similar technique is used to perform tasks that are not related to objects. For example, arrows are overlaid on the embedded’s camera feedback to allow the user to steer the robot freely. In order to overcome classification errors –that are unavoidable– we introduce an enforced selection scheme which requires the users to maintain their object’s selection for a *certain* amount of time in order to trigger a robotic action. In our first implementation [75], the *certain* time was set to 3 seconds and we reported a 100% successful objects’ selection.

We believe this approach can be promising to allow BCI users to perform complex robotic tasks despite the limitations of EEG-based BCI. However, in order to make the control experience smoother, the design of the SSVEP stimuli has to be improved to allow a faster selection of the objects and a better experience for the user. Indeed, various parameters can affect the strength of the SSVEP answer: the stimulus color, shape and size [24][25][26][27]. Improving the strength of the SSVEP answer can thus be done through a “better” presenta-

tion of the stimulus which should allow the recognition process to be more reliable and faster.

In this section, we investigate different designs for the SSVEP stimuli and evaluate them based on two criterions:

1. the stimulus design that provides the most accurate and fast selection;
2. the stimulus design that is most preferred by the users.

We report on two user studies we conducted:

1. The first study paves the way for the second. It considers 6 different stimuli design and involves 3 subjects. The results of this pilot study are presented together with the discussions which led to the design of the second experiment.
2. In the second study, 5 different stimuli design are proposed, 3 kept from the pilot study and the new 2 others being considered by this study alone. The design and results of an experiment performed over 10 subjects are presented. We could identify two stimuli as being relatively better for the purpose of object-centric BMI.

We believe that providing the best stimulus design allows one to build a more reliable interface and provide a more interactive experience to the user.

2.3.1 Material and methods

At first, we ran a set of pilot experiments to narrow the scope prior to an in-depth investigation. This is due to the high number of the conditions we considered and the higher number of factors involved in the results: frequency, color and inter-subjects variability are accounted in this pilot study. As a result, we qualified few promising conditions to be explored more deeply, and gained a better understanding of the experimental protocol. In this section, we introduce the methods we adopted in our work. The SSVEP display and extraction methods we employed in these studies were introduced in the previous chapter. When relevant, a distinction is made between the pilot study and the following in-depth one.

2.3.1.1 SSVEP “enforced-selection” paradigm

The “enforced-selection” paradigm is chosen in this experiment. This paradigm tackles possible interpretation errors (false positives) which are prone to happen during a BCI session. Indeed, the user can be distracted or introduce noise in the EEG data by moving involuntarily.

A typical SSVEP classification algorithm delivers a new classification a few times per second. In the “enforced-selection” paradigm, the user has to sustain her/his selection for a “given” time prior to a firm validation by the classifier. In other words, the classifier has to continuously provide the same output for a “given” time. For example, using our classifier and a “given” time of 2 seconds, the classifier has to give the same output 10 times in a row. This “given” time can be tuned to fit the user performance. In later sections, we refer to this time as $T_{\text{selection}}$.

2.3.1.2 About code-modulated visually evoked potentials (c-VEP)

Code-modulated visually evoked potentials (c-VEP) rely on a pseudorandom code to modulate visual stimuli. While they originate from the same neurophysiological background as SSVEP, they also outperform them in terms of accuracy and training times [76]. We had the opportunity to try a c-VEP system. However, in our —rather limited— experience, it did not perform exceptionally well in our robotic context. After training a user with the c-VEP system and assessing his ability to trigger stimuli selection, we asked him to perform objects' selection that he was not able to achieve unless we adopted a specific stimulation pattern for the object that is introduced as the “Strong hard-mask condition” in Section 2.3.1.3. Using this display method, the user was able to perform selections with the c-VEP system. This empirical observation suggests that, in order to use a c-VEP detection system in our experiment, we would have to train the subject for each pattern.

2.3.1.3 Considered SSVEP display methods

Pilot study The following display conditions are considered in this pilot study:

Reference

Simple square-shaped stimulus located at the center of the object recognized within the camera image flow.

Hard mask

The “mask”, i.e. the shape of the object, is flickered, the stimulus is a hard color.

Transparent mask

The mask is flickered, the stimulus is a transparent color, thus the user always sees a part of the actual background, the transparency value of the mask can be parameterized, in this study we considered a 0.5 composition value.

Inverted color

The mask is flickered, the stimulus inverts the color of the area covered by the mask thus creating high contrast.

Brightness modulation

The mask is flickered, the stimulus rises and lowers the luminosity of the area covered by the mask.

Brightness modulation with black and white background

The stimulus acts as in the previous condition, but the background is displayed in black and white to create more contrast.

The Figure 2.7 illustrates all the stimuli design that we implemented and tested in the pilot study.

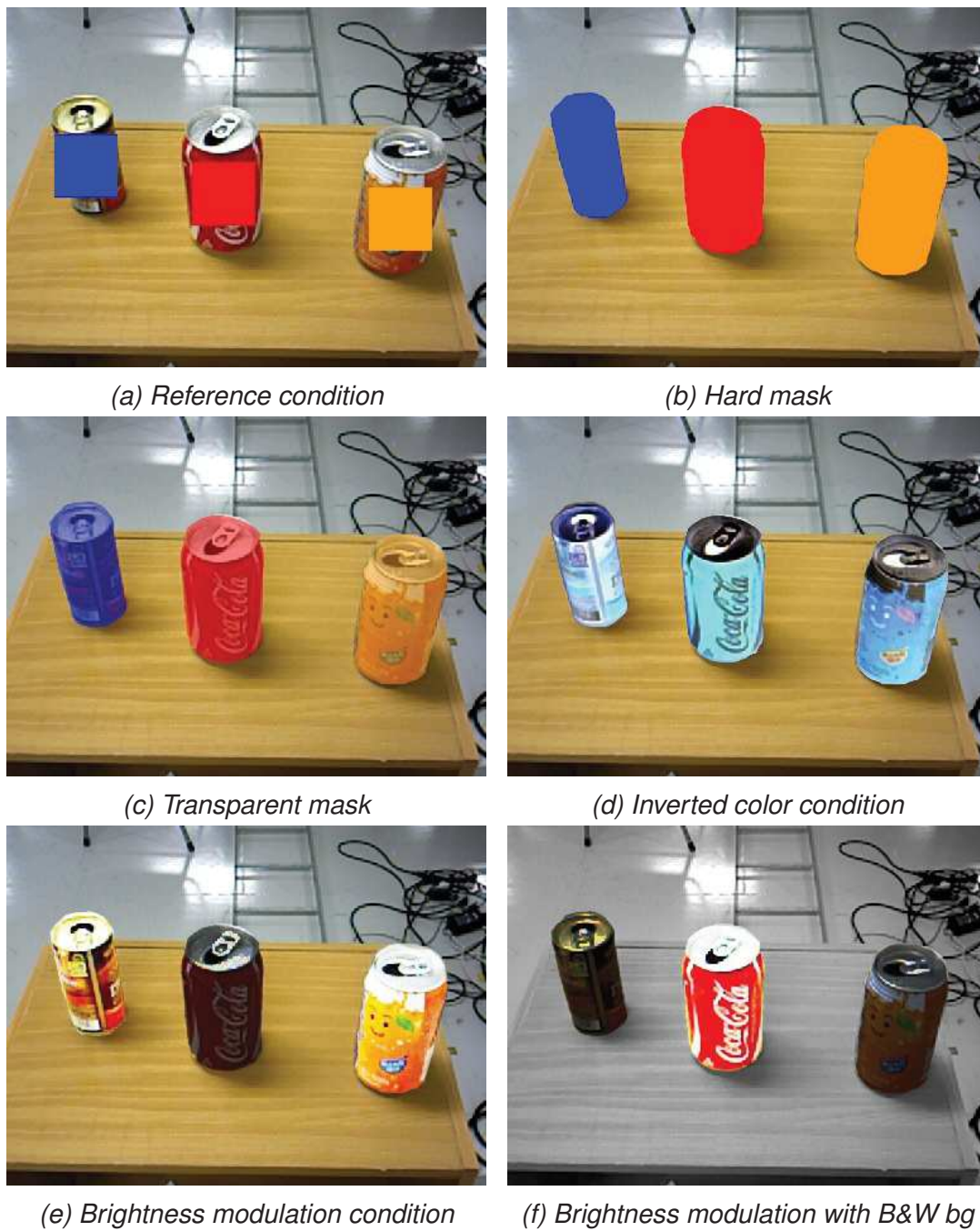


Figure 2.7: Stimuli conditions considered in the pilot study as seen by the users during the experiment.

SSVEP display optimization study Based on the results of the pilot study, including feedback from the users, we decided to restrain the stimuli presentation to few promising ones and exclude the least performing ones.

Excluded conditions

We decided to exclude two of the conditions that were initially considered:

- The “Transparent” condition: since despite being well received by the users, this condition performed poorly during the pilot study.
- The “Invert color” condition: as this condition performed poorly both in the quantitative and subjective parts of the pilot study.

Finally given the low difference between “Brightness modulation” and “Brightness modulation with black and white background”, we decided to also exclude the former one from this study.

New conditions

Since the design of the pilot experiment, we gained the ability to project the 3D graphical (mesh) model of the object onto the scene, offering a new presentation modality that we test in this study. The stimulus alternates between two states: the “model” state that can be seen in Figure 2.8 and the “hard mask” state. This condition is later referred to as the “hybrid” condition. It is important to notice that the “hybrid” condition totally hides the real object from the user. However, it shows a realistic representation of the object in place of the real one. We believe that this method helps us to present the object in an “interactive” way as the object stands out due to the difference in lighting conditions between the learned models and the objects presented to the user.



Figure 2.8: The model of the object is superposed on the actual image and flickered to trigger an SSVEP from the user.

We also introduce another alternating condition based on our observation with the c-VEP system reported in Section 2.3.1.2. In that condition, the stimulus alternates between two states: the “hard mask” state where the color of the mask is chosen according to the object

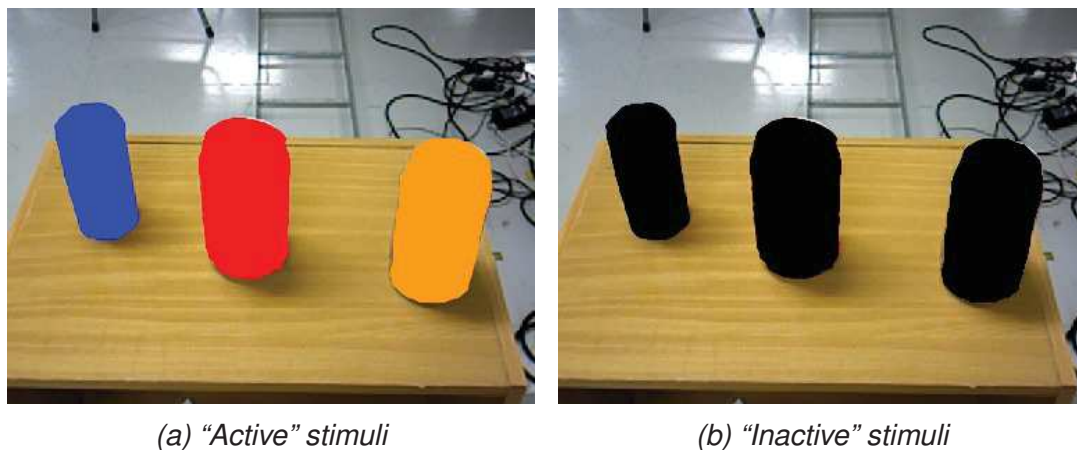


Figure 2.9: Strong hard mask condition: a condition based on the "hard mask" one but aimed at creating a higher contrast.

and another "hard mask" state with a black mask. It limits the differences that can exist between the training stimulus –usually red patches over a black screen– and the stimulus that can be recognized. This condition is later referred as "strong-hard mask" and similarly to the previous one, it totally hides the real object from the user. However, in that condition, no realistic representation of the object is shown to the user. This condition is shown in Figure 2.9.

In conclusion, the following conditions are studied in the SSVEP display optimization experiment: "Reference", "Hard mask", "Brightness modulation", "Hybrid" and "Strong-hard mask".

2.3.1.4 Material

We use a g.USBamp (24 Bit biosignal amplification unit, g.tec Medical Engineering GmbH, Austria) to acquire the EEG data from the user's brain at a sampling frequency of 256Hz, bandpass filtered between 0.5 and 30Hz with a notch filter at 50Hz to get rid of the power line noise. We use 8 Ag/AgCl active electrodes. The electrodes are placed on the POz, PO3, PO4, PO7, PO8, O1, O2 and Oz positions of the international 10-20 system [74], Fpz is used as the ground electrode and the right earlobe as a reference.

2.3.1.5 Subjects

Pilot study

Three male subjects with normal vision capabilities (i.e. without sight correction glasses or contact lenses), tagged A, B and C, aged 26 ± 1.2 yo, carried out this pilot study. During each experiment, each subject is comfortably seated in an armchair, about 1 meter away from a 17" LCD screen with a 60Hz refresh rate operating at a 1920×1080 resolution. In the pilot experiment, we only recorded the EEG signals without displaying any feedback on their choices, since all what we wanted at this stage, is to process their answer to the SSVEP

stimuli. Subject A had extensive experience of SSVEP-based BCI while the other two had none.

Additionally, a fourth subject, referred to as Subject D, participated in the pilot study. But he had difficulty to focus on the task and thus, his EEG records are discarded; yet his subjective answers are accounted for in the discussion.

SSVEP display optimization study

10 subjects, aged 26 ± 2.6 yo, 2 female, 1 left-handed, were involved in this study. They all had normal or corrected to normal vision. During the experiment, the user was comfortably seated in an armchair about one meter in front of a 17" LCD screen with a 60Hz refresh rate, upon which the testing interface was displayed at a 1920×1080 resolution. Five subjects had prior experience with SSVEP-based BCI and the five other subjects were complete novices. 6 of them play video games frequently or very frequently.

2.3.1.6 Experimental protocol

Pilot study

For each subject, experiments are divided into several trials: 3 for each condition ($\times 6$) and for each object ($\times 3$), that is $3 \times 6 \times 3 = 54$ trials. The following is an example of a single trial.

A scene is presented to the user with several objects in it. The experimenter tells the subject which objects are in the scene and in which order s/he should select by thought each object. The objects then start flickering using the "Reference" condition presented above. The flickering occurs for 10 seconds then stop for 5 seconds and restart with a different condition. When all the conditions for the current object are made, the subject switches her/his attention to the following object in a predefined order. The experiments ends when all the objects with all conditions are performed.

During a trial the frequency and color of the stimulus displayed on an object are known to the experimenter. We perform several trials with one subject to shuffle frequencies and colors. We also introduce a short break between each trial to allow the subject to rest as the experiments can be exhausting. The frequencies considered in this experiment are 6, 7, 8, 9, 10, 11 and 12Hz. The visual angle between the stimuli is about 10° .

SSVEP display optimization study

First, each subject performs a training session on BCI allowing her/him to use the SSVEP paradigm. The training is based on the presentation of rectangular shapes flickering at the different frequencies we wish to train, i.e. we are not using the shapes that will be used in the experiment. The reason of the latter choice is to reflect the variety of shapes that may be presented to them during actual robotic control scenarios. The training system uses 4 different frequencies: 6, 8, 10, and 14Hz. The frequencies were carefully selected to have neither common first or second harmonics and to be below 20Hz to minimize the risk of eliciting an epileptic crisis in healthy subjects as advised in [28] while remaining displayable on 60Hz monitors using the method we described above.

The user then undergoes an experimental trial that is presented to her/him as a game to reduce the "boredom" issue reported in the pilot study. A trial is limited in time: T_{trial} . During

the trial, the user is presented with three objects and is prompted to select one among the three within a certain time limit: T_{limit} . The objects are presented using one of the techniques discussed previously whereas flickering frequencies are chosen randomly among those used in the training phase. The objects, flickering frequencies and presentation conditions are chosen randomly to span all the conditions we wish to test. To select an object, the BCI uses the “enforced-selection” paradigm introduced earlier and $T_{\text{selection}}$ refers to the “given” time to trigger a selection. If the user selects an object successfully, s/he scores points. If s/he fails to select an object within T_{limit} , no point are awarded nor deducted. However, if the wrong object is selected, points are deducted. This highlights the fact that a wrong selection is more harmful than no selection. After either an object has been selected or T_{limit} is reached, a new target object is selected by the system and the “game” goes on. The goal for the subject is to achieve the best possible score and to improve her/his score from trial to trial. After each trial, we change the timing criteria: T_{trial} , T_{limit} and $T_{\text{selection}}$.

Each subject undergoes three trials with a 2 minutes break between each of them; the trials were performed in different order by different subject:

Trial A T_{trial} is 8 minutes, T_{limit} is 10 seconds and $T_{\text{selection}}$ is 3 seconds.

Trial B T_{trial} is 7 minutes, T_{limit} is 7 seconds and $T_{\text{selection}}$ is 2 seconds.

Trial C T_{trial} is 4 minutes, T_{limit} is 4 seconds and $T_{\text{selection}}$ is 1 seconds.

While the trials only span 3 possible values of $T_{\text{selection}}$, we also record the potential results under smaller values of $T_{\text{selection}}$. For example, in Trial 2, while $T_{\text{selection}}$ is set to 2 seconds, we have the ability to determine the capacity of the user to perform object selection with $T_{\text{selection}}$ below 2 seconds as well. In particular, $T_{\text{selection}} = 0s$ is similar to removing the “enforced-selection” paradigm.

After completing the third trial, the user is asked a series of questions in order to pool her/his subjective impressions on the different timing conditions and on the stimuli that are presented. This questionnaire is given in the Appendix C.1.

2.3.2 Pilot study – Results and discussion

2.3.2.1 Results computation

We define an observation as the 10 second period during a trial when the user is focusing on a particular object (and hence a particular frequency) in a given condition. After several trials with a subject we have a collection of observations with different frequencies and conditions.

For a given observation we compute the power spectrum density, in either an electrode channel or a combination channel, from 3 seconds after the beginning of the observation. Then we re-compute it every 200ms until 4 seconds before the end of the observation. Thus, an observation provides us with 15 values for the associated (frequency, condition) couple.

The computation was done on the following channels, which should deliver the best SSVEP response [31]:

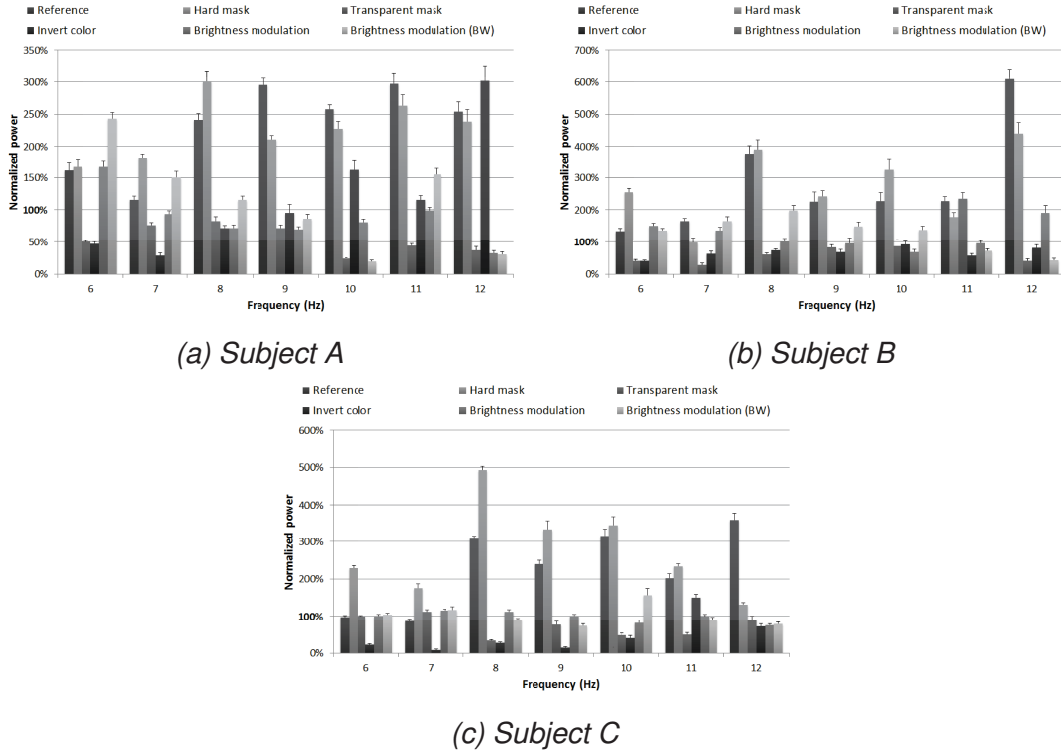


Figure 2.10: Normalized response intensity (power value at the L2 derivation) for subjects A, B and C, at the stimulation frequency for each condition and each frequency considered in the pilot.

- $L1 = 2 \times OZ - O1 - O2$
- $L2 = 4 \times OZ - O1 - O2 - PO7 - PO8$
- $L3 = 4 \times OZ - O1 - O2 - PO3 - PO4$
- $L4 = 2 \times OZ - PO7 - PO8$

2.3.2.2 Power analysis

We present the results of the pilot study based on the power analysis obtained previously. As the results are similar for the different channels we considered, we report only those processed from L2. Given the low number of participants in the pilot study and the issues we encountered with the experimental protocol, we only considered these results to identify conditions that performed poorly and exclude them from the final study. Therefore, we consider the amplitude of the SSVEP generated by each condition at the different frequencies we elected.

In Figure 2.10, we report the average intensity of the power spectrum over the L2 derivation at the stimulation frequency, computed as described previously, for Subject A, B and C. The intensity is normalized by the median intensity at the stimulation frequency for each subject.

For each subject and frequency considered, the difference in response intensity between the conditions is significant ($p < 0.001$). Although there is some variation among the subjects in the intensity of the answer to the stimuli and the ranking of those stimuli, the following trends emerged:

- The “Hard mask” and the “Reference” conditions performed well at all frequencies. It is expected that they would perform similarly since both are patches of color flickering on the screen.
- Both “Brightness Modulation” conditions performed well at low frequencies—below 10Hz— but poorly at higher frequencies.
- The “Transparent mask” and “Invert Color” conditions perform poorly at all frequencies, except 10 and 12Hz for the “Invert color” condition in Subject A’s case, and 6 and 7Hz for the “Transparent” condition in Subject C’s case.

These observations led to the exclusion of the “Transparent mask” and “Invert color” conditions from further study.

2.3.2.3 Subjects’ feedback

We present results of the pilot study based on subjective feedback obtained from the participants. These results guided our choices in the design of the next in-depth investigations. We first present the suggested rankings of the stimuli. We then address user’s remarks regarding the experimentation protocol.

Ranking of stimuli conditions Subjects A, B, C unanimously ranked the stimulus similarly, that is (from the most to least preferred ones):

1. Luminosity variation
2. Luminosity variation with black and white background
3. Transparent mask
4. Hard mask
5. Reference condition
6. Color inversion condition

The fourth subject D, however, suggested a different ranking only for the 4 first stimuli relatively to previous A, B, C subjects ranking, that is:

1. Transparent mask
2. Hard mask
3. Luminosity variation
4. Luminosity variation with black and white background

First, even if they did not always consider this criterion as the most important, all the users reported that the luminosity variation condition was the one they found less tiring. The “Transparent” and “Hard” color condition were always ranked in that order for that reason.

It is interesting to notice that all subjects had similar remarks about the different stimuli but choose to rank them differently. Subject D chooses to rank the luminosity conditions below because, while s/he preferred them esthetically, s/he felt it was easier to focus on the simpler masks that are the “Hard” and “Transparent” conditions.

Finally, the color inversion condition was unanimously pointed as being the worst condition; the color scheme resulting in the RGB components inversion was perceived as difficult to look at and aggressive for the eyes. Furthermore, as all the objects look similar in this condition, the subjects felt it was difficult to focus on one specific object during this condition because all objects tend to blend within their field-of-view.

Remarks concerning the experiment protocol Two issues concerning directly the experiment protocol are raised by the participants: (a) the duration, and (b) the boredom. However, it appears that both issues are related.

Regarding the timing during a trial: the subjects have to look at each object flickering with each condition. For each object and condition, each subject has to look for 10 seconds. Then, a 5 seconds break is allowed. That is 15 seconds per object and condition. For three objects and six conditions, as considered in this pilot study, this represents 270 seconds. We round up to 5 minutes to account for the breaks between each trial. To accumulate more data, we perform similar trials three times, hence 15 minutes. If we wish to study each object for each frequency –i.e. from 6Hz to 12Hz– one needs 105 minutes experiments, i.e. 1 hour and 45 minutes. This is indeed a very long experimental duration especially given that it is tiring to look at the stimuli for such a long time.

The boredom is a consequence of the long duration of the experiment and the lack of feedback during the experiment. Indeed, the only goal of the experimenter is to gather information about the effectiveness of the different stimuli conditions and thus no feedback was given to the subject while s/he is performing the experiment. The subject simply stares at the flickering stimuli for a long period of time without any visible results of her/his actions, which is certainly boring.

This was taken into account when designing the next experimental protocol as we tried to make it shorter and more engaging for the users.

2.3.3 SSVEP display optimization – Results and discussion

In this section, we first discuss the quantitative results of the in-depth experiments based on the selection accuracy and analysis of the EEG data. Then, we present the qualitative results regarding subjective appreciations of the stimuli obtained from analyzing the questionnaire answers and remarks. We finally present and discuss the most pertinent free answers to the questionnaire.

2.3.3.1 Selection accuracy based on stimuli design

The conditions are ranked depending on the amount of correct selections the subjects made and on the average time they needed to make them.

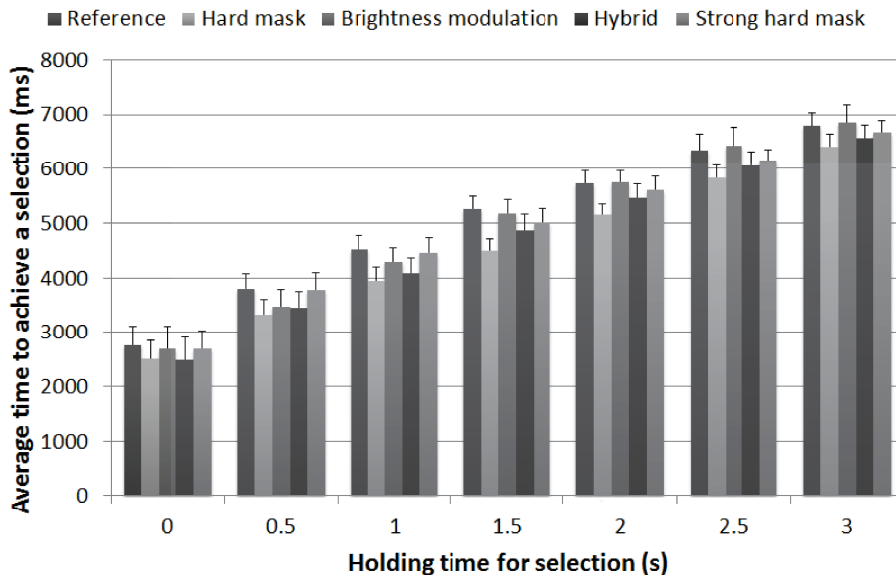


Figure 2.11: Average time to achieve a selection per condition and $T_{selection}$ for all the subjects.

We report the average selection time per condition in Figure 2.11. An analysis of variance reveals that the stimulus condition does not influence the selection time for successful selections (for all $T_{selection}$, $p > 0.2$). Hence, we will focus on the effect of the stimuli conditions on selection accuracy. The average selection accuracy for all the subjects is shown in Figure 2.12.

The accuracy that we obtained is below what is traditionally reported in the literature (for other contexts). This is mostly due to two factors we explain hereafter.

On the one hand, the existence of the zero-class and the introduction of the “enforced-selection” paradigm induce the possibility of false negatives. A false negative occurs when T_{limit} is reached before the user could hold a selection for $T_{selection}$, whereas a false positive occurs when the user holds a selection that is different from the assigned target for a time longer than $T_{selection}$. Figure 2.13 illustrates the effect of the “enforced-selection” paradigm. It shows that as $T_{selection}$ grows, it becomes more and more difficult to perform a wrong selection such that for $T_{selection} = 1s$, false positives account for about 68% of unsuccessful trials, whereas for $T_{selection} = 2s$ false positives account for about 10% of unsuccessful trials and 80% of subjects performed less than 1 wrong selection at $T_{selection} = 3s$.

This means that the difference in selection accuracy does not simply measure the user’s ability to trigger a selection with a given stimulus condition but also her/his ability to maintain her/his selection of that stimulus. This is shown in Figure 2.14 where one can see that as $T_{selection}$ increases, the selection accuracies of the “Reference”, “Brightness modulation”

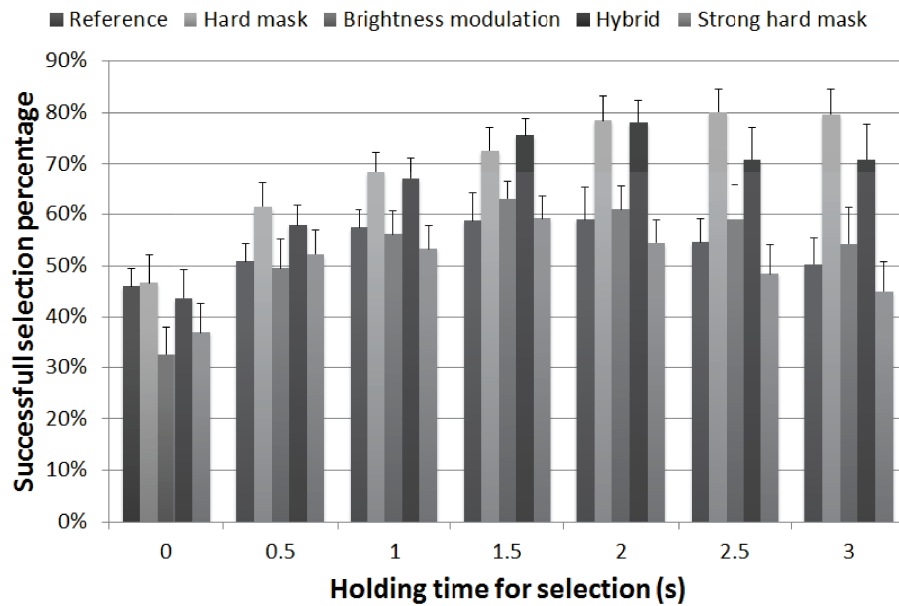


Figure 2.12: Average selection accuracy per condition and $T_{\text{selection}}$ for all the subjects.

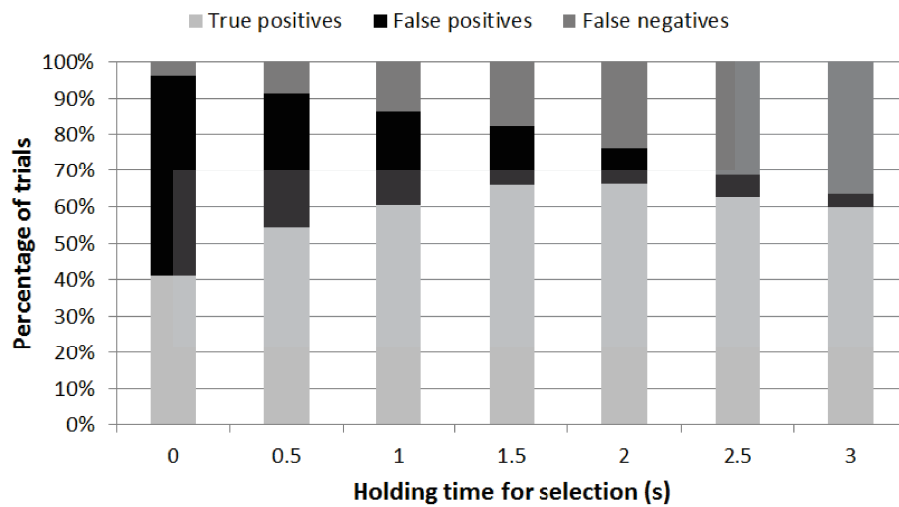


Figure 2.13: Overall repartition of true positive and false positive/negative depending on $T_{\text{selection}}$, s.e.m. for each category is below 5%.

and “Strong-hard mask” conditions are stable or decrease while the selection accuracies of the “Hard mask” and “Hybrid” conditions increase. This indicates that these two conditions allow the user to maintain a selection more easily.

On the other hand, the low accuracy at $T_{\text{selection}} \leq 1\text{s}$ can be explained by the SSVEP extraction method we use. In fact, we noted that, the classifier tends to favor one or two classes over the others as soon as the user starts observing an SSVEP stimulus. This leads

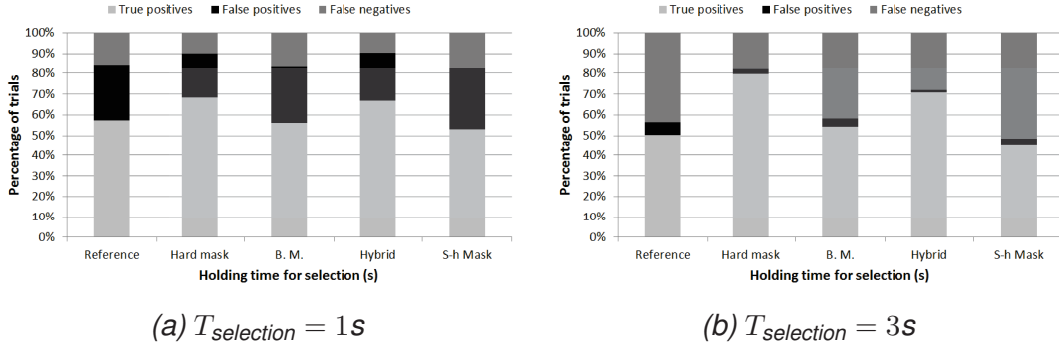


Figure 2.14: Repartition of true positive and false positive/negative per stimulus condition for two values of $T_{selection}$. In (a), all s.e.m. are below 5%. In (b), s.e.m. for false positives are below 3%, other s.e.m. are below 7%.

to a large number of false positives for $T_{selection} \leq 1s$. At $T_{selection} = 0s$, the first non-null outcome of the classifier is considered as the selection result, selections are thus confirmed quickly and errors are common. That is why the “enforced-selection” paradigm is introduced in the first place.

This effect can be seen in Figure 2.15 where one can see that false positives selection times correspond with the beginning of a round at $T_{selection} = 0s$. The favored classes vary from subject to subject hence we first ranked the classes depending on the number of bad selections that they triggered. For each subject, this ranking was the same for all values of $T_{selection}$. We then report in Figure 2.15 the proportion of false positives caused by the class ranked from 1 to 4. Rank 1 being the class that triggered the most false positives for a specific subject.

Note that, at $T_{selection} \geq 2s$, three subjects –30% of all subjects– account for 69% of the errors.

Other factors may also affect the performance such as subject’s fatigue, mistakes or frustration –e.g. following a sequence of unsuccessful trials. However, we believe those played a minor role in this study.

Finally, for every $T_{selection}$, we perform an analysis of variance on the proportion of good selections performed by the user depending on the stimulus conditions. We found no significant difference for $T_{selection} < 1s$ ($F_{(4,45)} < 1.5, p > 0.25$). However, we found a significant difference in selection accuracy for $T_{selection} \geq 1s$ (for $T_{selection} < 2s$, $F_{(4,45)} > 2.6, p < 0.05$ and for $T_{selection} \geq 2s$, $F_{(4,45)} > 5.5, p < 0.001$). A Tukey HSD test reveals three homogeneous subsets at $T_{selection} \geq 2s$ for $alpha = 0.05$. The first group includes the “Hard mask” and “Hybrid” conditions. The second group includes the “Hybrid”, “Brightness modulation” and “Reference” conditions. The third and last group includes the “Brightness modulation”, “Reference” and “Strong-hard mask” conditions. Pair-wise tests results are reported in Table 2.1.

We can, therefore, conclude the following results:

- The “Hard mask” and “Hybrid” stimuli conditions provide a better selection accuracy and fewer false positives. It seems that these conditions allow the user to better main-

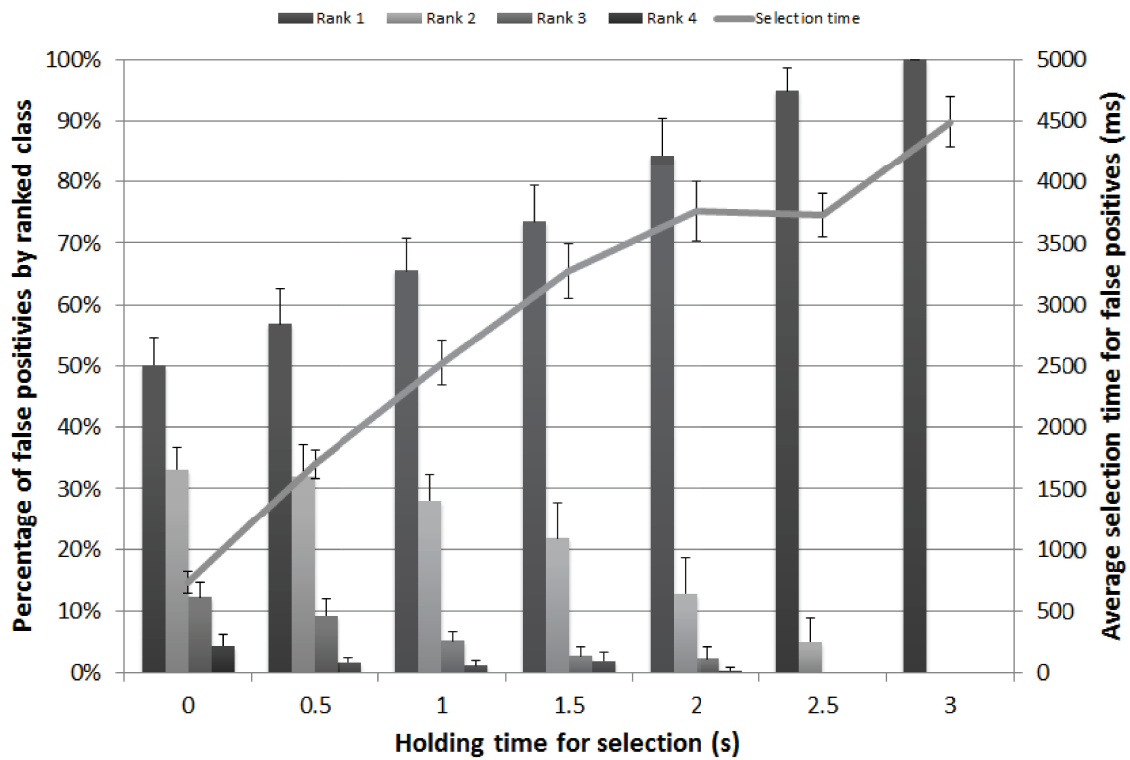


Figure 2.15: Average false positives selection time for all values of $T_{selection}$ and repartition of classes causing a false-positive.

Table 2.1: Statistical comparison of the studied stimuli methods based on selection accuracy. Green indicates that the stimuli in the current row is allowing more good selections than the stimuli in the current column, orange indicates an indecisive outcome and red indicates that the stimuli in the current row allows fewer good selections than the stimuli in the current column. B. M. refers to “Brightness Modulation”, S-h Mask refers to “Strong-hard Mask”.

	Reference	Hard Mask	B. M.	Hybrid	S-h Mask
Reference		Red	Orange	Red	Orange
Hard Mask	Green		Green	Orange	Green
B. M.	Orange	Red		Orange	Orange
Hybrid	Green	Orange	Orange		Green
S-h Mask	Orange	Red	Orange	Red	

tain her/his attention on the stimulus.

- Other conditions cannot be distinguished from each other.

2.3.3.2 EEG analysis

Method The goal of this analysis is to confirm that the conditions that performed the best during the selection “game” did so because they triggered better SSVEP responses. Since the analysis window of the EEG data used by the classifier lasts 2 seconds, we only use the trials where $T_{\text{selection}} \geq 2\text{s}$ and we only consider successful selections. We then define as an *observation* the 4 seconds period that starts two seconds before the command detection that triggered a successful selection. This is done for each frequency and condition. Thus, we gather a collection of observations.

An example of *observation* is given in Figure 2.16 where $T_{\text{selection}} = 3\text{s}$. In this case, the user was prompted with a target selection at $t = 0$ corresponding to the classifier output 3. The classifier starts recognizing the user’s intention from $t = 3\text{s}$ and the user was able to maintain that command for $T_{\text{selection}}$ thus triggering a selection. The observation window is then [1000 : 5000]ms.

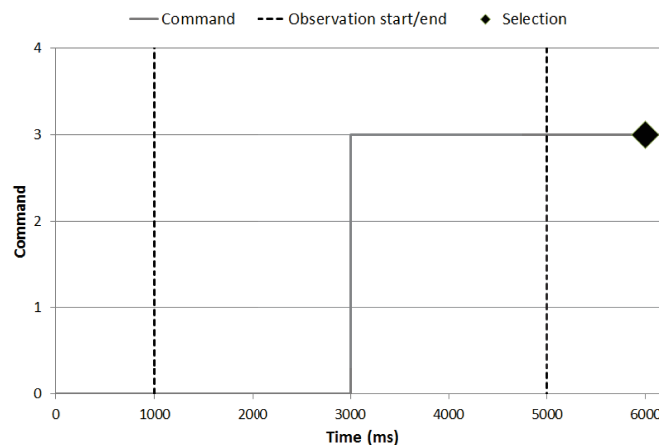


Figure 2.16: Example of an observation window.

Within an observation, we compute the spectral density on an interval of 2 seconds every 125 milliseconds, thus in the previous example the first window is [1000 : 3000]ms and the sixteenth and last window is [3000 : 5000]ms. The computation is made for each electrode as well as for the Laplacian combinations of electrodes that we introduced earlier: $L1$, $L2$, $L3$, and $L4$; we refer to those as *channels*. From this spectral density, we obtain the spectral power at the stimulation frequency. Hence, for each observation and channel we obtain 16 values. A graphical representation of an observation and of the sliding computation windows is given in Figure 2.16.

The results we present in the following section are obtained as follows:

First, for each frequency and channel, we observe the mean of the spectral density seeking for a general tendency. We then compute the average spectral density for each subject individually as well as the grand average. For each subject, frequency, channel and computation window, we put the spectral density power at the stimulation frequency on a scatter graph along with the mean value and standard deviation of this dataset. This allows us to observe

the tendency for each condition and channel. Finally, we use these values to perform two-sample z -tests and compare the conditions pairwise per channel and per frequency. This allows us to validate whether or not a particular condition is better than the other. The null hypothesis of this test is “*both conditions have the same average power value at the stimulation frequency*”. The two-sample z -test is performed with $\alpha = 0.01$.

Results In this section, we analyze the results of the power analysis described above. The results do not vary much between each frequency and channel therefore we will first present the general trend of the results rather than the results per channel and then discuss some specific differences at the end of the section.

Analyzing the grand average indicates that for each frequency, the “Hybrid” and “Hard mask” conditions perform better than the others. The third ranking is achieved by different conditions depending on the frequency that is considered. For 6Hz and 9Hz, it is “Reference”, for 8Hz it is “Brightness modulation” and for 10Hz it is “Strong-hard mask”. This outcome is coherent with the previous results. We then focused on the individual results to confirm this dominance.

The ranking of each condition is checked through a two-sample z -test. When the null hypothesis can be rejected, we consider that the condition that provided a higher stimulation “beats” the other. If the null hypothesis cannot be rejected, the two conditions are considered equivalent for this particular channel and frequency. For each frequency, we rank the conditions depending on the number of channels where they were able to outperform the others. If a condition does not outperform another by 3 channels, i.e. 25% of channels, we consider them as equivalent.

For 90% of the subjects, the order established as described previously is identical from one frequency to another. In Table 2.2, we report this ranking. “Hard mask ” and “Hybrid” perform better than the others but can be considered as equivalent. “Reference” performs better than “Strong-hard mask” and “Brightness modulation” but is worse than “Hard mask” and “Hybrid”. Finally, both “Strong-hard mask” and “Brightness modulation” perform worse than the others.

These results are coherent with those of the previous observations: the EEG ranking of the stimuli conditions we found is coherent with the ranking we obtained from selection accuracies.

Finally, one subject –10% of subjects– had different results as “Brightness modulation” came in the first or second place on every channel, i.e. as well as “Hard mask” and “Hybrid” conditions scores for all the other subjects. However, this is coherent with the score this subject obtained with the “Brightness modulation” condition since it is the one with which s/he scored the most for $T_{\text{selection}} \geq 2s$.

2.3.3.3 Questionnaire results

The first two questions address the subjective preference ranking of the stimuli. From each of these questions we compute a score: the preference score and the enjoyment score. We now explain how we compute these scores and we present the results we obtained.

Table 2.2: Statistical comparison of the studied stimuli methods based on the EEG analysis. Green indicates that the stimuli in the current row elicits a stronger SSVEP response than the stimuli in the current column, orange indicates an indecisive outcome and red indicates that the stimuli in the current row elicits a weaker SSVEP than the stimuli in the current column. B.M. refers to “Brightness Modulation”, S-h Mask refers to “Strong-hard Mask”.

	Reference	Hard Mask	B. M.	Hybrid	S-h Mask
Reference		Red	Green	Red	Green
Hard Mask	Green		Green	Orange	Green
B. M.	Red	Red		Red	Orange
Hybrid	Green	Orange	Green		Green
S-h Mask	Red	Red	Orange	Red	

The first question asks the user to rank the stimuli according to her/his preference for use: ‘1’ mark being the preferred method and the ‘5’ mark is the least liked method. Given this ranking we attribute a preference score of 5 to the stimulus that was ranked 1 by the user, 4 to the stimulus that was ranked 2, 3 to the stimulus that was ranked 3, 2 to the stimulus that was ranked 4 and 1 to the stimulus that was ranked 5. The second question asks the user to rank the stimuli according to their annoyance level. Hence, mark ‘1’ is the most annoying and the mark ‘5’ is the least annoying. The enjoyment score is equal to the rank given by the user. The average scores for these questions are shown in Figure 2.17.

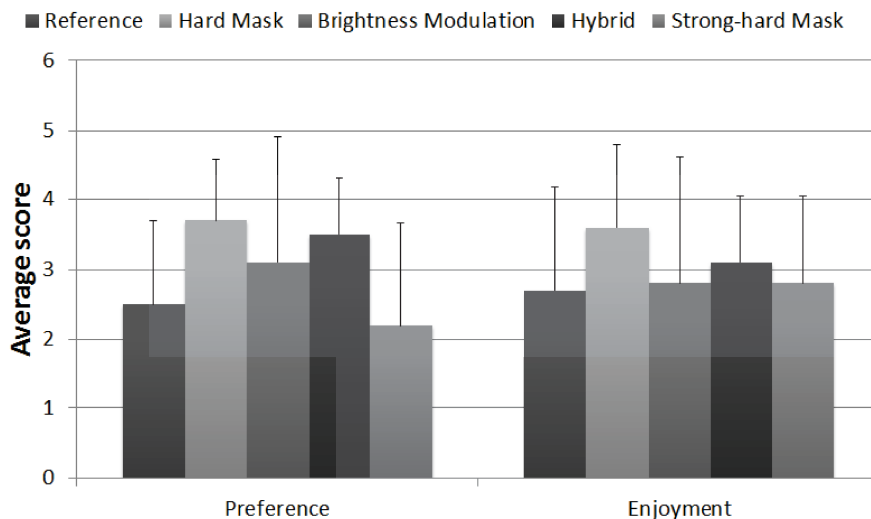


Figure 2.17: Average preference and enjoyment scores obtained by each stimulus condition. The higher the score, the better the users perceives the stimulus.

The “Reference”, “Brightness Modulation” and “Strong-hard Mask” are consistently

ranked among the three most annoying conditions without any particular order. However, there is a clear preference for “Brightness Modulation” over the other two, as well as for “Reference” over “Strong-hard Mask”. Therefore, we propose the following “subjective” ranking of the stimuli methods:

1. “Hard Mask”
2. “Hybrid”
3. “Brightness Modulation”
4. “Reference”
5. “Strong-hard Mask”

The other parts of the questionnaire focused on different criteria with respect to each stimulus. These questions allow us to compare a stimulus with the other stimuli, but they also provide an objective result. The scores are simply deduced from the marks given by the user. The results are shown in the Figure 2.18. The categories in this figure follow the order of the questionnaire but may have been reworded so that the “higher is better” precept appears more naturally.

We can see from the results that the “Reference” and “Strong-hard Mask” conditions are mildly appreciated with an average score around three in each category. Meanwhile, the other three conditions scored sensibly higher especially regarding the ability, for the users, to focus on these stimuli that are consistently rated higher up to five. These results are also coherent with the objective ones, i.e. the stimuli that “worked” better scored higher on the questionnaire. However, since the questionnaire was carried out after the experiment, the objective performance might have influenced some of the subjects’ answers (which is normal). If these studies are to be reproduced by the reader, we advice taking care of conducting an additional round of such questions priori to the objective measurements.

While there is a certain degree of variation among the subjects, some trends can be deduced from the results of the questionnaire. A very important issue noted from the users’ feedback is that *most of the subjects tried to focus on a feature of an object to make a selection*. Which explains that the “Strong-hard Mask” condition is not easy to be focused since it totally hides the original appearance. Similarly, since the square used in the “Reference” condition does not cover the entirety of the object made it difficult for the subject to focus on their chosen object’s feature (namely if it is outside the graphical square). Finally, some subjects chose “Brightness Modulation” over other conditions because it was *softer* to their sight. However, the selection performance and the appreciation of the stimuli by the subject did not match the users’ preference.

We now discuss the most unique and instructive answer to the questionnaire free remarks parts. There were several different issues raised. However, almost all of them do not relate to a specific SSVEP stimuli. We will now discuss the two most relevant remarks.

“Keep an outline of the object to help maintaining the focus”

This remark is relevant to all stimuli methods we tested. Indeed, when an object is selected, the user gets feedback in the form of an outline of the object to indicate which object is being selected. This users’ comment suggests that we keep the outline active at all time and change its color when it is activated in order to ease the attention focus on the object

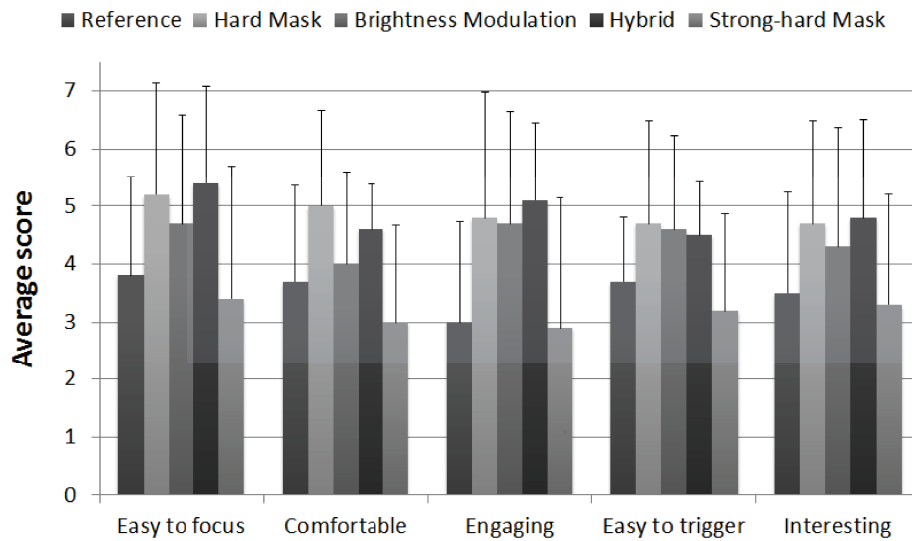


Figure 2.18: Average score obtained by each stimulus condition in each category of the questionnaire. The higher the score, the better the users perceive the stimulus.

of interest. This seems like a good idea and may help the stimuli to stand out even more. If we were to detect stimuli that flicker above the perception limit [29] this would even be necessary in order to differentiate “interactive” objects from the scene.

However, it is questionable whether this change would impact performance. In a sense, this outline is already available when the SSVEP selection starts thus it is already here to help “maintaining” the focus. It may be interesting to study the effect of such a permanent outline on the selection speed in a future study specific to this issue.

“It was frustrating when the selection disappeared then came back”

This remark refers, in layman terms, to the false-negative phenomenon. For example, at $T_{selection} = 2s$, the typical timeline of such an event is: (1) the user focuses on an object, (2) it is recognized by the system for $t_S = 1.5s$, (3) the system does not recognize it; or it recognizes the wrong object for a short time $t_{FN} = 0.4s$, (4) the system goes back to selecting the object that is the focus of the user. Two things can happen at this point. Either the user is able to finish his selection. Or, there is not enough time left to confirm the selection and we reach T_{limit} before the user can hold the selection for $T_{selection}$, resulting in a null trial.

Intuitively, it might seem possible to counteract this effect by handling short switches in the classification output using a “tolerant” enforced-SSVEP selection that we introduce now.

In the enforced SSVEP-selection algorithm that was used in the study, if the classifier output goes to zero while then the timer restarts and the user has to redo the entire selection. For example, with $T_{selection} = 2s$, the user holds a selection for $1s$ but the classification drops to zero, e.g. because the user lost his focus, for $t_{drop} = 200ms$, then s/he would have to hold the selected command for another $2s$ in order to achieve the right selection. This algorithm, dubbed the “tolerant” enforced SSVEP-selection algorithm, tolerates such short falls to a zero class output. Going back to the example, if the selection drops to zero after the user was focused for $1s$ then s/he only has to focus for $1s$ to trigger a selection. A possible

implementation for this algorithm is shown in Algorithm 1, the following new notations are introduced in the algorithm pseudo-code:

- T_{FN} is the maximum time we allow for a zero-class drop
- $t_{command}$ is the current holding time of a non-zero command
- t_{zero} is the current holding time for the zero-class
- $command$ is the current output of the classifier
- $previous_command$ is the previous output of the classifier

Algorithm 1 The “tolerant” enforced-SSVEP selection algorithm

```

1: function INTERPRETCOMMAND(command)
2:                                     ▷ Called by the interface to interpret the classifier output
3:                                     ▷ The function returns True when a selection is triggered
4:   if command = 0 then
5:     if previous_command ≠ 0 then
6:       UPDATE( $t_{zero}$ )
7:       if  $t_{zero} \geq T_{FN}$  then
8:         ▷ 0 was hold for  $T_{FN}$  discard previously hold command
9:         previous_command ← 0
10:        t_command ← 0
11:         $t_{zero}$  ← 0
12:      end if
13:    end if
14:  else
15:    if previous_command = command then
16:      UPDATE( $t_{command}$ )
17:      if  $t_{command} \geq T_{selection}$  then
18:        return True
19:      end if
20:    else
21:      previous_command ← command
22:      t_command ← 0
23:       $t_{zero}$  ← 0
24:    end if
25:  end if
26:  return False
27: end function

```

We applied this algorithm to the classification data we recorded during the experiment for all subjects. The results on the overall accuracy, at $T_{selection} = 2s$, are shown in Figure 2.19. We can see that it generally has a null or negative effect for all $T_{FN} \neq 0$ and all conditions that we considered, with the exception of the “Reference” condition for which we observe a marginally positive effect for $250ms \leq T_{FN} \leq 750ms$.

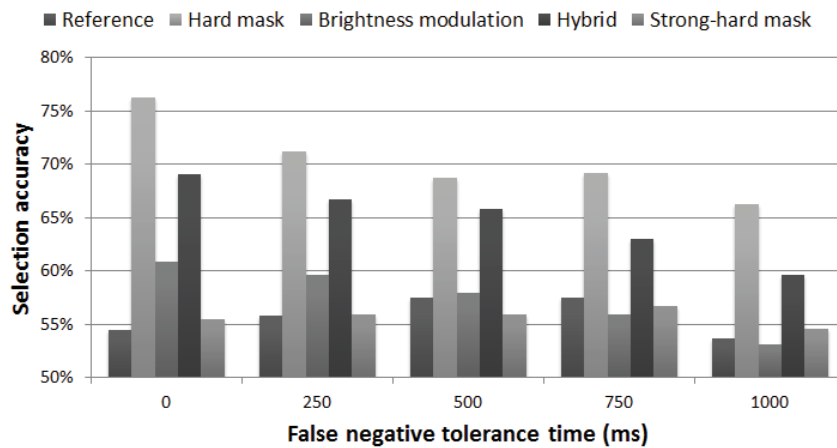


Figure 2.19: Overall effect of T_{FN} on accuracy.

Figure 2.20 shows the effect of different values of T_{FN} on two subjects who both reported that they felt the selection was going away and coming back quickly. Yet we can observe distinct effects of T_{FN} on each subject. For Subject A, shown in Figure 2.20a, who has rather poor performance initially—below 70%, the addition of a tolerance time significantly improves the selection performance for all conditions. However, for Subject B, shown in Figure 2.20b, who has a good level of selection performance initially and whose questionnaire had the commented answer, the addition of a tolerance time has a completely opposite effect. This may due to an important rate of false positive selections that may not be visible to the subject since s/he is focused on one particular object at the time of the experiment and thus may miss short feedback on other objects. Finally, since the study was conducted without tolerance time for false negatives, the feedback on the object of interest would drop during the experiment which may have led to a drop of attention and thus make it difficult to come back to a proper selection.

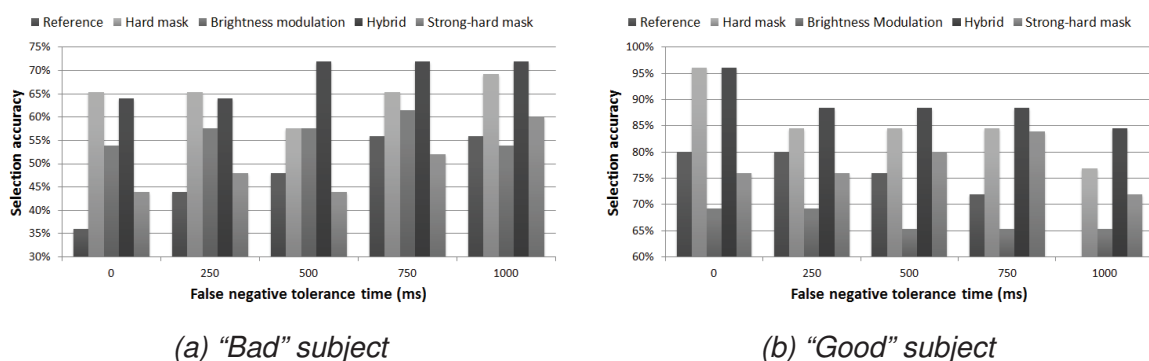


Figure 2.20: Effects of different T_{FN} values on selection accuracy for individual subjects.

In the end, this opens up a new issue to find a good balance on this topic since an optimal

choice of T_{FN} would be guided by the rate of false positives and false negatives for each particular subject. A thorough study on this topic may be of interest for future research.

2.3.4 SSVEP Display Optimization: Conclusion

This study brings two major results regarding the use of SSVEP stimuli for object selection. On the one hand, we have shown which display models allow better performances both objectively, accuracy-wise, and subjectively, at the subject's preference level. On the other hand, we have shown both the effectiveness of the enforced-selection approach and the existence of performance's "plateau".

Regarding the design of stimuli, we have shown that:

- The "hard-mask" and "hybrid" performed better on both the objective and subjective criteria;
- The conditions that slightly altered the properties of the object to create a flickering, i.e. the "transparent mask" condition and the "brightness modulation" conditions, performed well at the subjective level but poorly functionally;
- The conditions that strongly affected the perception of the object, i.e. the "invert color" condition and the "strong hard-mask" condition performed poorly on both criteria.

Although we tested only a limited number of stimuli methods, we can assess some properties that effective stimuli design should follow for optimal selection accuracy and approval by the BCI users: the stimuli should cover the entire object but allow the user to see the entirety of it so that s/he can focus on particular elements when performing the selection. Furthermore, the stimulus design should keep a simple and bright design to maintain the SSVEP properties.

Regarding the selection time, it seems that we reach a "plateau" in the selection performance around $T_{selection} = 2s$ as longer selections are more difficult to maintain due to false negatives and shorter selections are unreliable due to false positives. However, having isolated two sensibly better stimuli, it may be interesting to measure the information transfer rate (ITR) of those two conditions under the "enforced-selection" paradigm and different values of $T_{selection}$ in order to better estimate an optimal $T_{selection}$ regarding the speed-accuracy trade-off.

Future research on this topic could focus on the issues raised by the questionnaire analysis. In particular, the impact of "sustained" objects' highlighting should be carefully evaluated since it was only requested by one subject of the experiment. The most important issue to face in the future would be those of false negatives and false positives which are responsible for unresponsive selections and false selections. We have shown by looking back at the data we acquired during the second experiment that those were difficult issues to tackle yet we did not have the opportunity to also address them at the user feedback level, which may be beneficial.

2.4 Conclusion

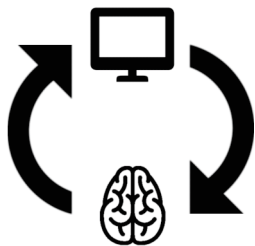
We have shown in this chapter several improvements that we brought to the BCI for robot control and to the use of SSVEP for control purpose in general.

- We introduced a novel framework to design BCI applications with frequent paradigm shifts at the core. This framework shows its value in the upcoming chapters where it is extensively used;
- We demonstrated the negative effects of dynamic visual feedback in SSVEP applications and proposed a simple solution to counteract this effect. This solution is applied throughout the experiments we conducted after its introduction except when using an HMD due to technical limitations that we have addressed;
- We determined appropriate time criterion and an effective way to display SSVEP stimuli on top of real-world object to allow selection with both a high degree of accuracy and a good user experience.

In the following chapter, we bring the use of SSVEP to perform whole-body control of a humanoid control. The presented work takes advantage of all the issues we have raised and resolved in this chapter.

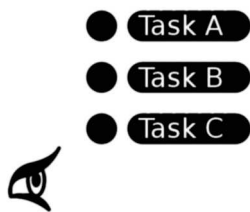
Whole-Body Control with Steady-State Visually Evoked Potentials (SSVEP)

As we explained in Chapter 1, independently of the technology used to acquire the user's brain activity, all BMI-based control application can be divided into two categories:



Neurofeedback based

This is the case of applications that try to understand the activity – often the motor activity – of the brain and translate it to the controlled device. This includes works that we have presented previously such as the intelligent wheelchair for EEG-based BMI [54] or arm control in ECoG-based neurofeedback [10], but also the control of a humanoid robot in fMRI-based BMI [68]. We later refer to this family of application as “NF-based” applications.



Menu based

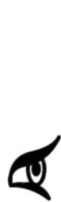
This is the case of applications that use stimuli, usually visual: P300 or Steady-State Visually Evoked Potentials (SSVEP), to map a behavior on the robot through the use of a “menu”. For example, among the works we presented, one can cite the work done in [31] where a mobile robot is controlled through SSVEP stimuli or [63] where a complex P300 interface allows the user to remotely control a mobile robot and its embedded camera. We later refer to this family of application as “menu-based” applications.

The impressive works we introduced earlier illustrate a dominant trend in current BMI-based control. On the one hand, decoding the brain activity in order to map it onto the controlled device; this is constrained by the fact that controlling the Cartesian motion of a robotic arm by neuro-feedback requires a long and sustained training for the subject and

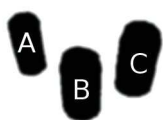
usually perform with less than perfect accuracy and slowly compared to the capacity of the controlled robotic systems they use as substitutes or surrogates. Furthermore, we can question the actual need—beyond therapeutic benefits—of controlling an external device with such a degree of precision to perform simple tasks. On the other hand, menu-based control, while closer to what we are going to introduce as object-centric BMI in this chapter, still limit the capacities of the robot to a few pre-programmed behaviors.

In this chapter, we introduce a novel approach to control a (humanoid) robot using BMI. Moving away from the classical approaches—menu/icons selection or understanding the motor intentions of the human to translate it to a device—we propose to move towards an object-centric methodology. This is motivated by current development in neuroscience and task-space control that will lead to such a possibility. We also assessed the viability of our approach in an EEG-based demonstrator. We call this approach “object-centric” BMI.

3.1 Introducing Object-Centric BMI



Object-centric BMI



Object-centric BMI designs the extension of menu-based control to the use of objects as a central point in the control of a robotic device through BMI. The development of this framework originates from the well-known task-space control paradigm employed in robotics, the improvements of environment perception techniques in robotics and recent developments in stimuli-based BMI. Those are introduced here along possible limitations of object-centric BMI that we foresee.

3.1.1 From Cartesian-Motion to Task-Space Control

The task-function based control is a powerful control paradigm to design complex behaviors for robots without explicit trajectory planning at the joint or Cartesian level. A task can be seen as motion primitives or constraints that can be defined directly or indirectly in the robot’s sensory space. The desired task can be defined simply as a state or a state error vector in the sensory space, which is mapped into the robot’s motor space (joint actuators) using an appropriate projection operator, e.g. the robot’s Jacobian in kinematics. A complex behavior can then be planned as a succession of tasks, which can in turn be structured into a hierarchical way: a stack of tasks. This formalism proved to be particularly suited for the control of highly redundant robots; several complex scenarios have been setup using this powerful control tool with a humanoid robot [77]. It has been extended to address issues such as tasks scheduling and the fundamental issue of control continuity under discrete tasks scheduling operations in the stack, such as tasks insertion, removal and swapping. Such a controller is well suited to enable and disable behaviors based on the limited inputs of a BMI.

Indeed, contrary to other interfaces used to control a robot, a BMI is a rather unreliable and very slow interface. As we have seen before, it is very difficult for the user of a BMI

to gain accurate control of a robotic device. However, selecting a stimulus among many can be reliably achieved. This observation leads to considering tasks as affordances on the surrounding objects/environment. Thus, controlling a robot with a BMI becomes a problem of selecting an (object, task) pair, which is simpler to achieve than closed-loop brain-motor control.

3.1.2 From menu-based BMI to object-centric BMI

Object-centric BMI premises can be seen in works such as [64] and [75]. In those works, the BMI is menu-based ([64] relies on the P300 paradigm while we chose to rely on SSVEPs). These works originate from the menu-based applications but deviate in the sense that the stimuli are not mapped to a single robotic command anymore but are adapted along the course of the demonstration and actions. We discuss the pioneer experiment of [64] to illustrate this point.

The robot used in this experiment is a *Fujitsu HOAP-2* humanoid robot. It is equipped with a pan-tilt camera that is used to provide visual feedback to the user and locate objects in the environment. The robot is able to walk autonomously, navigate based on visual information and pick up and place objects. Furthermore, the robot has prior knowledge of the room layout. Finally, a second camera observes the scene from above and is used to locate the destinations available for the robot.

Two important features distinguish this work from other menu-based BMI:

1. Stimuli are associated to different actions depending on the state of the robot and the global task that the user wishes to accomplish.
2. The stimuli are not associated to a single motor action but rather trigger tasks selection for the robot. The selected task is then executed independently of the user actions—here merely observes the results of his selection—based on the sensory inputs of the robot.

The work we produced in [75] adopts similar principles but extends the capacities offered to the user and improves the embodiment experience. In this present work, we formalize this initial work into the object-centric BMI framework.

Object-centric BMI moves away from the standard association between a brain command and a robotic command. It is centered around the interaction between the robot and its environment through the objects available within this environment based on the robot's capacities. In the end, we wish to extract an (object, task) pair that can be executed by the robot and realizes the user's intention. Ultimately, this allows the user to perform complex tasks through her/his robotic avatar while maintaining the complexity of the control minimal.

3.1.2.1 Limitations of object-centric control

The principle of object-centric BMI applications is to move away from the mapping of stimuli to robotic actions towards the selection of (object, task) couples to provide better functionalities to the user of such systems. However, not all tasks are suitable to this formalism.

The simplest example that comes to mind is navigation. Yet, navigation can be thought of as an object-oriented task. For example, in the work of [64], the robot autonomously navigates towards a selected object without input from the user. Moreover, if no object is available to guide the navigation, we can resort to using a shared-control approach that has proven successful in previous works to navigate the robot until an object is available to switch back to object-centric BMI control.

3.2 A framework for object-centric BMI

In this section, we introduce the main elements that compose the object-centric BMI framework that we envision. An important goal of each of these elements is to be as independent of each other—and most importantly of the underlying techniques and technologies that support them—as possible. Indeed, the control of a robot through BMI is inherently a cross-disciplinary problem and thus advances in one field should be adoptable as soon as possible.

We first give a general overview of the framework and then go into the guiding intentions of each module that compose the framework. For each module, we give a description of its function and discuss how they relate to other modules.

3.2.1 Overview

An overview of the framework can be seen in Figure 3.1. Some purely technical elements or trivial links are not displayed for the sake of clarity, though they are mentioned in the text when relevant. The modules are voluntarily presented in a non-technical fashion as the implementation of these modules is discussed later in this chapter.

The modules that are discussed in the following sections are shown in a bold and underlined font. The inputs and outputs of these modules are also introduced. The other parts of this framework—the robot sensors and the robot controller—are out of the scope of this chapter.

3.2.2 BCI Module

The *BCI Module* inherits from the general BCI framework introduced for example by OpenVIBE or BCI2000 that we introduced in Chapter 1.

In those frameworks, the BCI is divided within three layers:

1. **Signal Acquisition**, where the signals are acquired from the user’s brain through various technologies: EEG, fMRI, ECoG...
2. **Signal Processing**, where the brain’s signals that were acquired in the previous step are transformed from their raw values to the user’s “intentions”.
3. **User Interface**, where the intentions obtained in the previous step is used to drive any kind of user interface, including a robot.

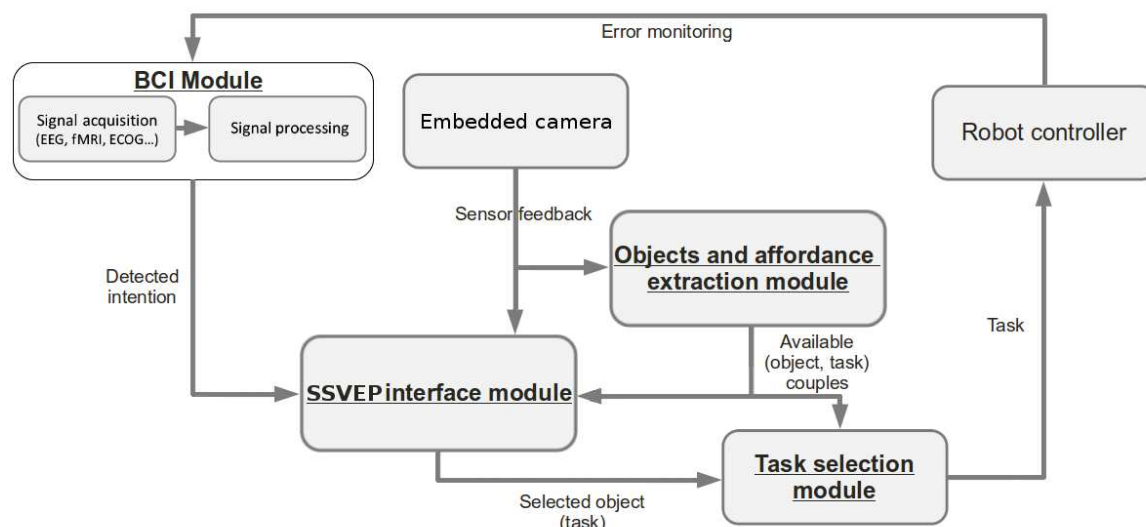


Figure 3.1: Overview of the object-centric BMI framework. The modules which are in bold and underlined are presented in this section.

However, within this work, we have only encapsulated the first two within the *BCI Module* in order to focus the attention on the core of our contribution, the *SSVEP Interface Module*, that we introduce next. The most important aspect here is that the *BCI Module* is unaware of the true “intentions” of the user and is merely a black box that can extract information related to the stimulation of the brain provided by the *SSVEP Interface Module*.

An optional feedback link also exists between the *SSVEP Interface Module* and the *BCI Module* as some paradigms, e.g. P300 or code-based VEP, require a synchronization between the signal processing unit and the process that is in charge of displaying the stimuli. This is a purely technical/implementation point, therefore it is not represented on Figure 3.1.

A link between this module and the *Robot Controller* is shown on Figure 3.1. This is related to error monitoring. Other than cancelling an action based on a user’s input, it is also possible to rely on a brain phenomenon to detect errors made by the user or the robot. This phenomenon is the Error Potential (ErrP) [34] that we introduced in Chapter 1. However, there are different types of ErrP which depend on the trigger of the error:

- An erroneous interpretation of the user’s intention by the BCI, i.e. a false positive;
- The user made an error and made a wrong selection; for example, the user may select to steer the controlled device to the left and then realize that s/he needs to steer it to the right;
- An error in the execution of the process triggered by the correct interpretation of the intention by the BCI; for example, if the user selected an object to grasp but the robot then fails to grasp the object.

The first kind of ErrP can be easily monitored and only requires a link between the *BCI Module* and the *SSVEP Interface Module* that we evoked previously. On the contrary, the second type of ErrP is much more difficult to detect automatically as in this case, the error

has a much higher semantic sense. In our work, we opted for a “manual” solution in these cases where the user has to either deal with the consequences of his error, e.g. in the case of steering, or trigger an action cancellation, e.g. in the case of grasping. Finally, the last kind of ErrP can be detected as long as we can monitor the action executed by the robot to detect failures to perform the current task.

Finally, the module outputs the user’s “raw” intentions. We call “raw” intentions, the outcome of a BCI classifier that does not necessarily translate to an actual intention. For example, it may be the output of a SSVEP classifier that detects that the user is focused on a 6Hz stimulus or a “left-hand” output in motor-imagery paradigms. The *BCI Module* itself does not know what actual intentions those outcomes map. This is the role of the *SSVEP Interface Module* that we introduce now.

3.2.3 SSVEP Interface Module

The *SSVEP Interface Module* is in charge of gathering sensors’ data from the *Robot Sensors* and environment perception data from the *Objects and Affordance Extraction Module* and synthesize the appropriate user interface that let the user performs as much actions as possible through the robot. Furthermore, the *SSVEP Interface Module* holds the state of the robot and the previous actions done by the robot, offering a way to cancel past actions and execute appropriate behaviors on the robot. In the interface, we distinguish two types of tasks: (i) affordable tasks that are related to objects and (ii) “free tasks” that cannot be related to an object – a typical example is the control of the robot’s gaze. We now give a “high-level” example of interface building that illustrates the role of the *SSVEP Interface Module*.

If the *Objects and Affordance Extraction Module* has already detected some object and did not detect any obstacle yet, then, the role of the *SSVEP Interface Module* is to build an interface where:

- Appropriate stimuli are displayed upon the objects;
- “Free tasks” related stimuli, e.g. looking around or steering, are displayed;
- Additional information on the environment are provided.

In that example, the *SSVEP Interface Module* takes care of filtering out some of the stimuli if necessary. For example, if the BCI is only able to recognize a few stimuli—typically around 5. It then only displays a limited number of objects and leave one stimulus for a “cycle” switch that allows the user to get more possibilities. Furthermore, this filtering can be refined by including user preferences into the *SSVEP Interface Module*. For example, when selecting products within a cupboard, the stimuli could be placed upon the user’s favorite products initially.

Finally, on a technical note, the interface itself relies on the *bci-interface* library that we introduced in Chapter 2. The link between the *SSVEP Interface Module* and the *BCI Module* that may exist for synchronization purpose is therefore hidden within the implementation and interactable objects can be displayed independently of the underlying BCI paradigm. Moreover, the library is designed to facilitate switches between different control modalities which is obviously a fundamental capability in this framework.

3.2.4 Objects and Affordance Extraction Module

The role of the *Objects and Affordance Extraction Module* is to provide information about environment to both the robot and the *SSVEP Interface Module*. It operates in two major phases:

1. Extract objects from the observed scene, this can be done through different computer vision methods that give different levels of detail regarding the objects in the scene;
2. From knowledge acquired offline and the observation of the scene, synthesize a list of affordable objects and affordable tasks related to this object;

The vision algorithms that were used in our demonstrators and fall within the scope of the *Objects and Affordance Extraction Module* are presented in Section 3.4. For example, the Blocks World Robotic Vision Toolbox (BLORT), introduced in Section 3.4.3, is able to detect several objects in the scene, label them and even detect if an object is partially hidden. In a case where two objects are present within the scene, if one of them is visible and another is seen by the system but partially hidden by the first one, the *Objects and Affordance Extraction Module* provides a set of two objects, one of which has many tasks associated with it—including a task to move it in order to access the second one—while the other has no task associated.

3.2.5 Task selection module

The role of the *Task Selection Module* is to select the most appropriate task considering the object that was selected by the user and the affordable tasks associated to this object. In our initial implementation of multi-task humanoid robot control with BCI the role of this module was limited to executing the behavior associated to a fixed stimulus, e.g. move the robot towards the arrow selected by the user or grasp the object that was associated. However, the addition of probabilistic tools within this module allows to relieve the user from most of the tedious control involved by the use of BCI in this context.

Moreover, the *Task Selection Module* can persist throughout the experiments involving the same user which allows learning strategies to be applied and thus a better experience for the user of the system.

A proof-of-concept implementation of this module based on a Bayesian network is proposed in Section 3.5 along with an experiment that highlights the benefits of this method.

In the following sections we focus on the technical elements that allow to achieve the object-centric BMI framework we introduce in this section. The final section of this chapter is dedicated to user experiments conducted at different implementation stage of the framework that highlights the benefits of our approach and the modularity of the framework.

3.3 SSVEP Interface Module for Object-Centric BMI

In the object-centric BMI, the user selects an object and the system automatically infers the task that corresponds to the user's intention based on the current state of the robot and its

environment. The selection of the object by the user should be as certain as possible in order to ensure a pleasant interaction. Furthermore, the interface should provide relevant information based on the task that is being executed. In the object-centric BMI framework, this is ensured by the *SSVEP Interface Module*. In this section, we introduce the elements that provide these two features.

3.3.1 Recursive and Enforced SSVEP Selection

The brain features we decided to exploit are the steady-state visually evoked potentials (SSVEP). The SSVEP describe the activities that the brain generates when the user observes a flickering stimulus. The method relies uniquely on the user's attention to the stimulus. It also allows detecting that the user is maintaining his attention on a given stimulus and to detect a shift of attention in a few seconds. The process we used to extract the SSVEP is based upon the minimum energy classifier approach introduced in [30] and detailed in Chapter 1. It provides a zero-class implementation that allows to detect that the user is not interested in interacting with the system. After a short training, about 6 minutes, it is able to operate at an 80% recognition rate [31] for 4 classes and can provide a new command every 200ms. This is a satisfactory performance for an SSVEP-based BCI system [32].

As mentioned previously, our SSVEP extraction process can reach an 80% successful recognition rate with a short training phase. However, given the nature of the SSVEP stimuli, errors are bound to happen over the course of the experiment due to distraction or fatigue [24]. Therefore, the 100% mark would be difficult to reach. This unreliability of the decision outcome becomes an important problem when using the BCI to control an avatar, especially if the decision that was taken cannot be reversed. Moreover we aim at minimizing the number of frequencies that we can detect to ensure high accuracy while keeping the training time as short as possible. Therefore we devised an SSVEP-based selection paradigm that allows for a large number of commands and puts an emphasis on accuracy.

To increase the number of commands we used a simple recursive selection algorithm. For example, if we have sixteen commands available but only trained the system to detect four different frequencies we split each command into four groups. The user then selects one of these groups and finally selects a command among the four commands in this group. This recursive selection is illustrated in Figure 3.2.

To enforce the selection we ask the user to maintain her/his attention on the command s/he wants to select for a certain time. We consider this command as the actual intention of the user only if s/he was able to maintain her/his attention "long enough". For example, it could mean three seconds, i.e. fifteen successive classifications.

3.3.2 State Transition in BMI

The interface relies on the library we introduced in Chapter 2. A finite state machine is used to switch between the different control schemes involved in a particular scenario. To overcome the limitations of BCI, object-centric BMI relies heavily on the capacity of the robot to realize the intention of the user autonomously while s/he monitors the proper execution

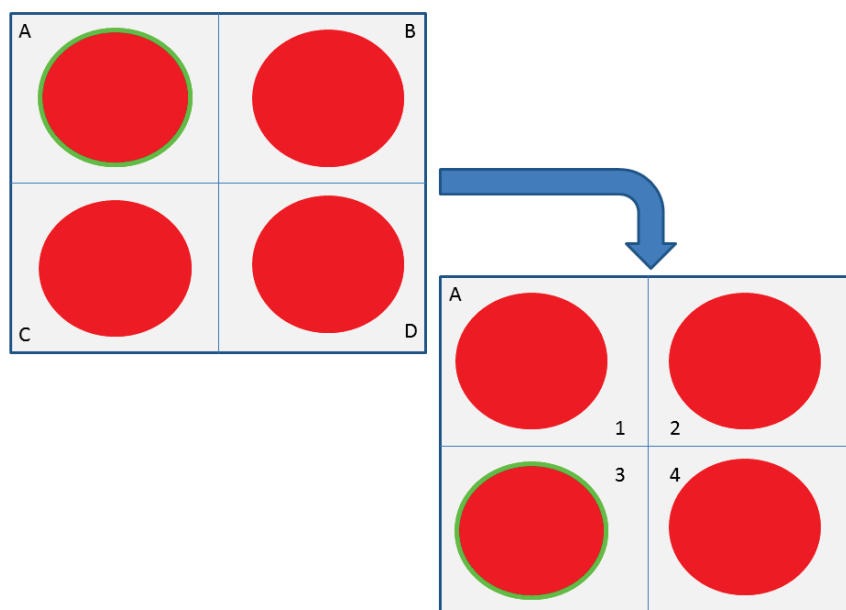


Figure 3.2: Principle of the recursive selection with SSVEP: here the user first selects the “A” quarter which is zoomed in, then s/he selects the “3” quarter, therefore the final selection is “A.3”.

of her/his intention. Therefore the transitions in the FSM are triggered either by the user, through the selection of a stimulus, or by the robot, following the completion of a task.

For example, to realize a pick-and-place task, we distinguished three main phases which are of different natures, i.e. “free tasks” or object-oriented tasks:

- Grasping phase: the user selects an object with the BCI and the robot grasps it;
- Walking phase: the user steers the robot to the destination table;
- Drop phase: the user selects a spot where to drop the object and the robot drops the object.

These phases and their natures are reflected in the FSM presented in Figure 3.3. Yet, they can be divided into multiple states.

3.4 Objects and Affordance Extraction Module

The *Objects and Affordance Extraction Module* is built around the well-known Robot Operating System (ROS) middleware that is used to acquire the image from the robot embedded camera and diffuse it on the network for further treatment. Within the context of the object-oriented BCI framework, the images are streamed to the *SSVEP Interface Module* for a display within an immersive interface on the one hand and on the other hand, the images are streamed to various vision algorithms that compose the *Objects and Affordance Extraction Module*. We present here the most used vision methods that compose this module and that we have employed throughout our demonstrators.

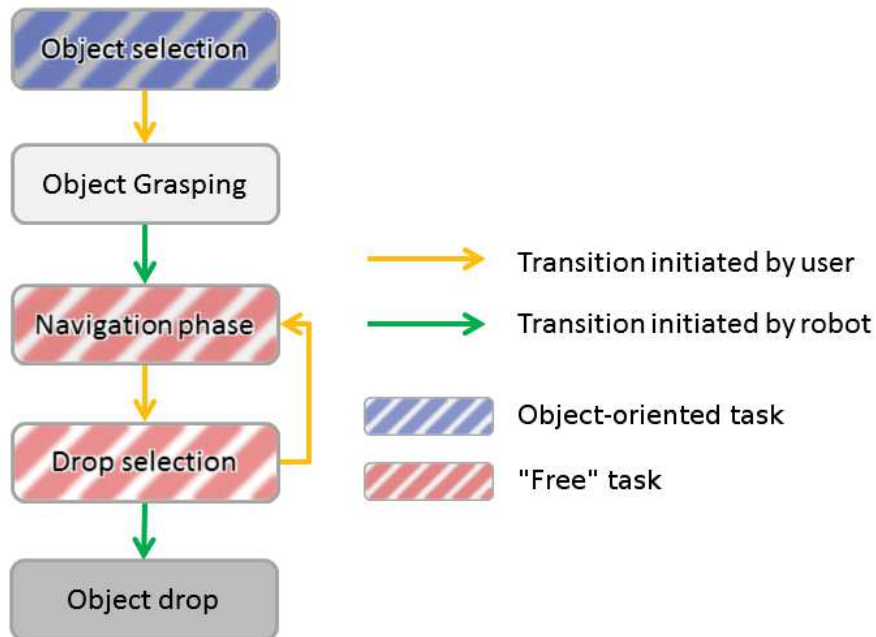


Figure 3.3: Finite State Machine of a pick-and-place demonstrator in object-centric BMI

3.4.1 Objects Recognition - 2D Recognition: Algorithm

The object recognition method is based on the work presented in [78] and its extension to account for color properties of objects in [79]. This method relies on the construction of a vocabulary set of texture-related features and color-related features. These features are trimmed down to a restricted set through a k-means clustering algorithm, associated to the relevant objects and organized in a kd-tree for efficient closest neighbor research needed by the recognition algorithm. This allows this method to scale very well as when the objects database grows, the vocabulary itself does not grow but evolves and enriches itself.

The recognition algorithm then consists in (i) extracting interest points in the scene, (ii) computing color and texture features at those interest points, (iii) match those features with the ones from the vocabulary, (iv) each feature from the vocabulary then casts a vote for the relevant object - this voting mechanism is further explained afterward - (v) the object presence is decided based upon its best score value. The score each object gives is determined from a training set where the algorithm knows the objects present in the scene. The votes are computed so that the more specific to an object a feature is the bigger vote it casts. For example, if the same feature is extracted from 10 images with 10 different objects in the scene it contributes a 0.1 vote to each object. However, among the same set, if a feature is found only twice, it casts a 0.5 vote for the two objects involved in these two scenes. Finally, a second pass over the training set allows us to define threshold score above which the object presence is assumed. Thanks to the sharing of features among different objects, the recognition algorithm can operate very efficiently, above 15Hz even in rich scenes, and the features selected for classification permit a consistent detection of the objects.

3.4.2 Objects Recognition - 2D Recognition: Shape Extraction

In order to allow a smooth integration of the detected objects within the user interface, we extended this method to detect the objects' shapes.

To do so, we need to collect more data when we build the vocabulary as well as during the course of the algorithm. During the training run, when we create a link between a feature and the object, we register two information from the training image: the shape of the object and the position of the feature point. During the recognition process, we maintain a list of all voters for each object. For each voter, we register the point of interest position in the image being processed, the link it is associated with and where the vote is casted.

Once an object and its center have been detected, we extract the voters that voted for the object around the object's center. Each voter is ultimately linked to an object's mask from the vocabulary set through the link it holds. For each of these masks, we compute a homography from the points in the vocabulary set to the matched points in the processed image. The points are filtered before the homography computation. If multiple points in the processed image match the same point in the vocabulary set, we keep the one for which the distance between the point and the center of the object is closest to the same distance for the vocabulary point. The same selection criterion is applied when the same point in the processed image matches different points in the vocabulary set, which can happen when the texture match and the color match are different points from the same image of the vocabulary set. The vocabulary object that retains the most matching points in the end is selected and its shape is deformed thanks to the computed homography which allows us to match the shape of the object in the processed image.

3.4.3 Objects Recognition - 3D Recognition

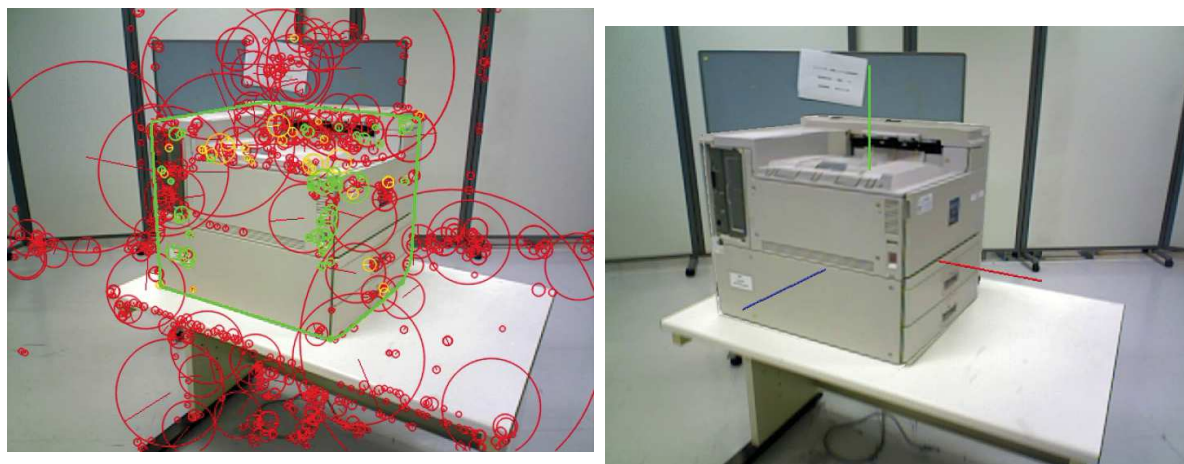
BLORT, the Blocks World Robotic Vision Toolbox, is proposed in [80]. It is an open source software that aims at providing a set of tool for robotics to:

- recognize known objects in the environment, and
- track those objects over a sequence of images.

An object is learned by providing a 3D model of the object, semi-manually matching this model to a real view of the object and acquiring features from the object. The object recognition and tracking mechanism then operates in two phases. First, using Scale-Invariant Feature Transforms (SIFT) [81] associated to the objects, and learned prior to operation, it tries to recognize the known objects. Then, the object is tracked over a sequence of images, using a bootstrap filter method [82]. A sample from the outcome of the first phase can be seen in Figure 3.4a, while the Figure 3.4b illustrates the result of the tracking phase.

3.4.4 Robot localization in indoor environments

SLAM (Simultaneous Localization and Mapping) [83] provides recently a very mature technology [84] and allows the robot to simultaneously and continually build the map of the



(a) Recognition result for a single object in the robot's field-of-view. The circles represent the SIFT features. Green circles represent match with the cookbook of the object, yellow circles represent partial matches and red circles represent non-matching features.

(b) Tracking result for a single object in the robot's field-of-view. The blue, red and green lines represent the object's frame. Thus reflecting the object's position and orientation in the current view.

Figure 3.4: Example of BLORT usage on a printer.

environments. Recent experiments conducted on our humanoid HRP-2 and HRP-4 robots with our colleagues' software in [85], demonstrate an amazing precision in localization with real-time performances. SLAM however builds a "rigid model" of the surrounding without semantics nor the possibility to distinguish objects that composes the environment. SLAM can also be used in closed-loop navigation to reach a given target defined as desired 3D coordinate point in the model. If previously mapped objects composing the entire scene as moved w.r.t to their mapped position, they are considered as outliers, yet the rigid map can be corrected to integrate this change in the position, see recent work in [86].

We adopted the SLAM software D6DSLAM that is presented in [85] and outlined in Figure 3.5. D6DSLAM unifies volumetric 3D modeling and image-based key-frame modeling, to provide and update a rich 3D map of an unknown environment as well as provide accurate localization of the robot in its surrounding environment. D6DSLAM proved to be very robust and accurate in robotic scenarios, and in particular, it handles very well the typical sway motion of a humanoid robot during walking, that is usually problematic for visual servoing applications [87].

3.5 Implementation of the Task Selection Module

The *Task Selection Module* implementation we present here is based on a Bayesian network. Therefore we first introduce the mathematical concept that is a Bayesian network and how we can use it to serve our purpose.

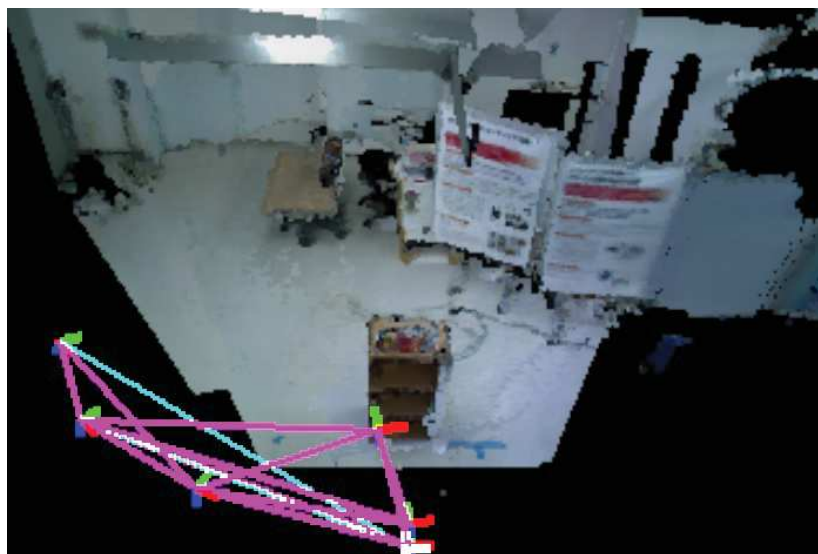


Figure 3.5: A view of the environment map generated by the D6DSLAM software and the robot's localization within this map. The nodes of the purple graph represents key-frames while the blue line indicates the current dislocation of the robot compared to the key-frame that is used as a reference for localization.

3.5.1 Definition

A Bayesian network is a directed acyclic graph whose nodes are random variables X_i and edges are conditional dependencies. Nodes that are not connected represent conditionally independent variables. A probability function can be associated to each node. This function takes as input a particular set of values for the node's parents and gives the probability of the variable represented by the node. A simple example to consider is a case where the node has n parents which are Boolean node, i.e. the values they can take are simply true or false, then the probability function is a 2^n table that give the value of the node for each combination of the parents' variables.

Given this graph, the probabilities of a given node can be inferred. Furthermore reinforcement learning methods can be used to improve the Bayesian network's prediction capabilities from failed and successful predictions.

3.5.2 Using a Bayesian network as a Task Selection Module

We have previously discussed how a user can easily and reliably select an object to interact with through a BCI. The next step, to allow a smoother control of the robot through BCI, is to autonomously determine and execute the task that the user wants to execute with this object through the robot. In this iteration of the network we include the following tasks:

- Move the object to reach other (hidden) objects;
- Move the object to a new location;
- Bring the object to a person;

- Throw the object;
- Pour the content of the object into a glass;

Some of these actions obviously share similarities. For example, many require the robot to first grasp the object and then navigate towards an object, a location or a person. The availability of these actions also depends on the context, e.g. if no hidden objects are detected then the first action is very unlikely, on the nature of the object, e.g. not all objects can be “poured” into a glass, or on their status, e.g. an empty can is likely to be thrown out. This additional information can be modeled within the Bayesian network. Therefore, when an object has been selected, the Bayesian network can compute the most probable action to perform depending on the current observations that determine the value of certain nodes. Furthermore, these nodes’ values can be updated in real-time to reflect new information. For example, it may be impossible to know from the vision only if a bottle is empty or not, e.g. if it is an opaque container, however, once the robot grasps the object it is able to evaluate its weight from the force sensors equipped in its hands. Finally, if the Bayesian network does not select the right action, the system can propose corrections to the user and “plug-in” the new task upon the one that was being executed. For example, if the selected task was to bring a can to a person but the actual intention of the user is to move the can to a new location then only the destination of the “moving” task needs to be updated. Such a network is illustrated in Figure 3.6.

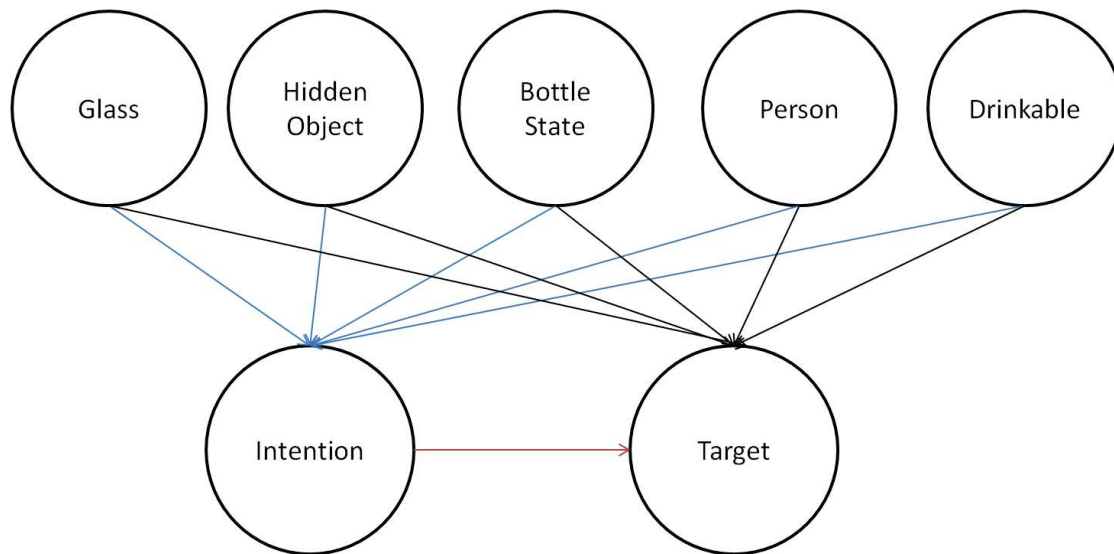


Figure 3.6: Bayesian network used to infer the intention of the user from the object’s and environment’s properties.

3.5.3 Improving the network

The Bayesian network can be updated upon wrong and right predictions of the user’s intention so that the user’s preferences can be taken into account within the model in a natural

fashion. However, this requires the execution of a large number of experiments to finely tune the network.

3.6 Object-oriented Robot Control

We discuss here some specific aspect of the stack-of-tasks (SoT) controller that we introduced earlier. In particular, we present the modifications to the controller that permit object-oriented control.

3.6.1 Walking with a Humanoid Robot in BMI

The walking control part of this work relies on the pattern generator presented in [88]. In this work, the robot is modeled as a linear inverse pendulum and a Linear Model Predictive Control scheme is applied to compute the footsteps which minimize the input, a reference and the previewed center of mass (CoM) velocity. These footsteps are then passed onto the robot controller that executes them as illustrated in Figure 3.7.

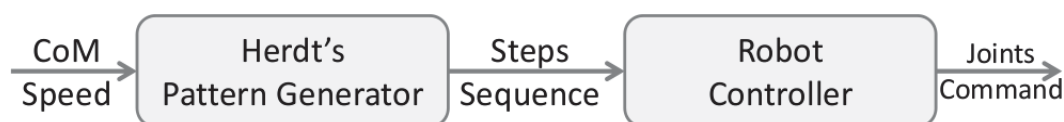


Figure 3.7: Pattern generator overview

The advantages of this pattern generator specific to the context of BCI control are twofold. First, it takes a simple input: the CoM speed; this makes the mapping between BCI commands and robotic commands very simple as we simply have to associate a static speed to each available BCI command; yet it also permits more complex schemes [73]. The other advantage is that it is highly reactive; it can handle rapid switches between speeds and “burst” commands, e.g. the command is set to a null speed, then for 200ms (less than the time for a step which is 800ms) it is set to 0.2m/s forward and finally set back to zero, without compromising the equilibrium of the robot; those are both desirable qualities for BCI control as the interpretation of the user’s intention can be mistaken and result in this kind of input to the pattern generator. On the HRP-2 robot, this pattern generator is implemented as part of the stack-of-tasks controller [77], therefore we can remotely interact with it through the network.

However, while controlling the speed of the robot through BCI using the CoM speed as an input is possible [73], it is a difficult task because (a) precision is difficult to achieve using the SSVEP paradigm as lag and errors are difficult to account for and (b) the user only gets feedback from a camera on a 2D screen so depth perception is very poor. This results in an important training effect during the control of the robot as the user gets better and better at both handling the SSVEP steering controls and locating her/himself within the experimental space.

3.6.2 Navigation assistance for humanoid control through BMI

The difficulty of manually steering the robot through a BCI naturally leads—as in previous works in the field, particularly, regarding the control of a wheelchair—to the addition of navigation assistance mechanisms. In this section we present two methods that we used in our experiments: a simple depth information provider and a more fleshed-out object-oriented navigation scheme.

3.6.2.1 Providing depth information to the user

In order to improve the performance in this phase we mounted an Asus Xtion Pro Live on top of the robot’s head to add depth perception to the robot. The robot is walking with the head directed towards the ground to allow the user to see upcoming obstacles. As the robot approaches the table it can be clearly seen in the depth map of the robot’s camera as seen in Figure 3.8a. We can then apply a Canny edge detection algorithm and a Hough transformation on the image to locate the corner of the table. Given the intrinsic and extrinsic parameters of the camera we are then able to locate the position of the table within the robot’s frame. This information is used to restrain the robot’s movements when it is near the table.

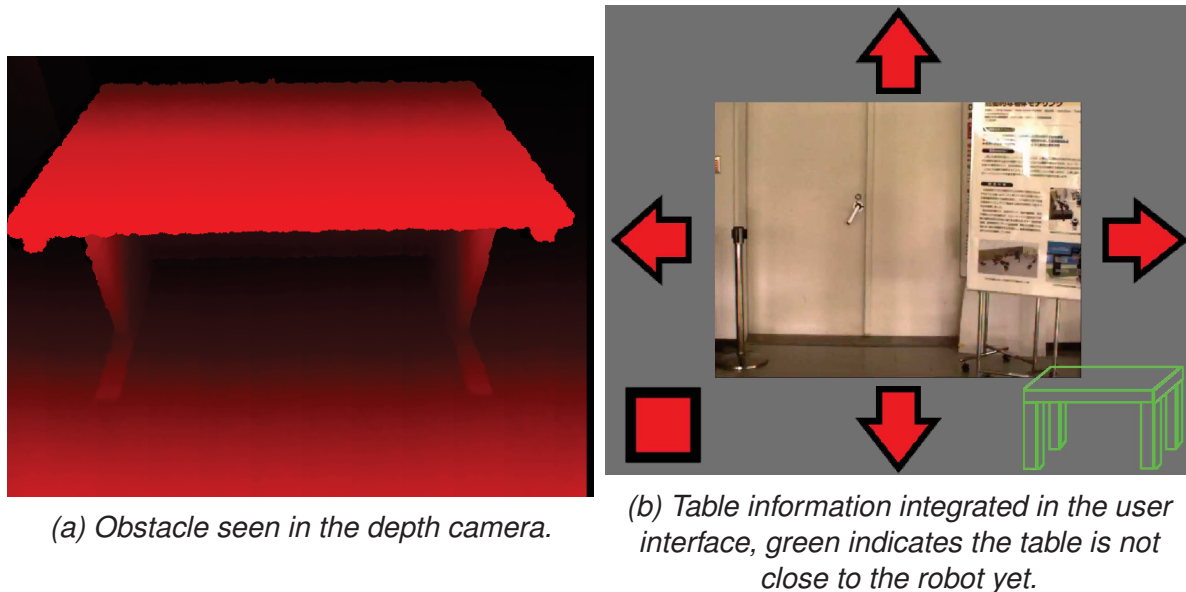


Figure 3.8: Providing depth information to the user for navigation assistance.

Finally, the information about the table position can be figuratively fed back to the user: displaying a simple icon - as shown in Figure 3.8b - the user can be informed about the proximity which allows her/him to know when to stop the robot.

3.6.2.2 Object-oriented navigation in indoor environments

Recent advances in robotic visual tracking [89] [90] is demonstrated in complex tasks that are achieved in robust closed-loop fashion among which reaching, manipulation, navigation,

when targets of interest are totally or partially in the field-of-view of the robot's embedded camera(s). However, closed-loop visual servoing is still difficult to solve with humanoid robots due to many aspects: weak odometry, important blur caused by rapid movements or the sway motion and the feet impacts generated during walking movements [91] [92] [93]. In order to overcome these limitations, we combine the 3D SLAM developed in [85] and the visual object recognition and tracking provided by BLORT [80] to allow the robot to autonomously retrieve and reach known objects in indoor environment which model is built from SLAM. This is an important step to enable robots to competently perform everyday manipulation activities [94].

In order to provide information for navigation, the SLAM software was augmented with a pixel tracker. It is possible to select a pixel in the current image obtained from the camera. Once this pixel has been selected then its 3D position is tracked and streamed on the network continuously as the robot progresses and the software builds a larger and more complex map.

Limitations of object recognition for navigation

A toolbox such as BLORT is not limited to object recognition, but can provide tracking capabilities, as we have mentioned previously. Thus, we might also rely on the tracking capability of BLORT to navigate and reach the object or spot of interest. This is however limited for mainly two reasons:

1. On one hand, the tracker obviously requires the object or the image of interest to be in the field-of-view of the camera in order to operate. Unfortunately, in some cases, e.g. fulfill a robotic or task constraints such as obstacle avoidance, it may be necessary to have the object temporarily out of the camera field-of-view. It is also possible to recognize the object of interest when it re-enters the image, by reinitializing the BLORT tracker. However, this may require additional object learning, since the recognition accuracy varies tremendously with the distance to the object. Therefore, the robot should navigate towards the object by relaxing the field-of-view inclusion constraints whenever needed without jeopardizing the navigation robustness.
2. On the other hand, an object tracker provides information only about the object of interest. This is not sufficient to provide a safe route in a changing environment comprised of obstacles, walls, doors, humans, etc.

Previously described causes, suggest that we use object recognition and tracking to search and first retrieve an object of interest in the environment, and eventually updating this position if the latter changes, but we rely solely on SLAM for vision-based navigation.

Combining object tracking and SLAM

In order to retrieve known objects in the humanoid surrounding environment, we propose to follow a two-phase strategy:

1. The *research phase* consists in trying to first locate the object within the environment
2. The *object-oriented navigation phase* where we autonomously go towards the object of interest that was previously detected.

We now describe each of these phases: how the recognition information is transmitted to the localization software and finally, how the localization information is provided to the autonomous navigation scheme.

Search phase

The goal of this phase is to locate the object. The first step is to query BLORT about the presence of the object within the current field of view of the robot. If the object has been detected by BLORT then we can retrieve the position of the object and dispatch it to the SLAM software. If the object has not been detected, we have to search for the object in another location. This constitutes the core behavior of this ‘Search phase’.

At the moment, two cases are to be considered:

1. either the robot has not “seen” everything around it. Then it should rotate on itself and try to find the object in its field-of-view; or
2. the robot has “seen” everything around it, in which case it goes towards other unexplored locations or request help from the human user. This location can be determined either randomly or provided by the SLAM—for example, an unknown part of the map behind a wall.

However, in our opinion the search phase shall be seriously considered under a semantic SLAM search. Indeed, if we are able to structure SLAM memory in a way where objects are labeled and associated with SLAM data structure, then the object search can boil into a search within the data structure which then provides (at least) a preliminary guess of the location that BLORT would confirm. This would then require the possibility for the robot to plan and reach the presupposed location containing the object of interest using only SLAM data structure. For example, if we are starting at the entry of a corridor with multiple rooms, the current implementation checks in the room on the left, then the room on the right then continue forwards into the corridor to check the following rooms. However, this is definitely not the optimal way. In that case, a semantical approach would allow the robot to first search the rooms that are more likely to contain the object thus saving time in the achievement of its mission [95].

From recognition to autonomous navigation

Once the object has been recognized, we determine which pixel corresponds to the centroid of the object. This pixel is then transmitted to the localization software which then starts to track this pixel within the egocentric frame of the robot, in real-time. This tracking information is then used by the robot for autonomous navigation.

Note that if the object position is not conform to the current SLAM memory –for example, in the case where the robot had already processed the object of interest in a previous mission as part of its current SLAM and meanwhile, the object has changed its position–, it is detected as an outlier and the SLAM updated comfortably to its current arrangement in the updated SLAM data. It is this update that is used for tracking.

Object-oriented navigation phase

Given the localization information, the robot starts walking towards the object, using the navigation algorithm described in [96]. This navigation scheme allows us to set a navigation goal, but also to specify way-points that should be reached prior to the destination, and are

used to avoid obstacles. Once the object is reached, the task is achieved, and other tasks can be performed to interact with the reached object.

Implementation results

In [97], we demonstrated the feasibility of this approach through simple demonstrations that did not include the BCI. However, since it only requires the selection of an objective, we believe this method is particularly well-suited for BCI control of humanoid robots, in particular within the context of object-oriented BCI.

3.6.3 Object soft-drop with guarded motion

In our introductory work [75], the robot dropped the selected object on the spot selected by the user in the final phase of the demonstrator. In this initial version, the robot would simply put the gripper holding the object over the spot selected by the user and then open the gripper. This solution was not very elegant and we propose a friendlier strategy to softly drop the object using the force sensors within the robot's wrists. In the remainder of this section, the term "drop spot" refers to the point selected by the user where the grasped object is to be released.

3.6.3.1 Drop Spot Selection and Drop Height Estimation

The selection of the drop spot is done through the recursive selection presented earlier. However, this is done through the image provided by the RGB-D camera mounted on the robot. This means that we are able to obtain an estimation of the height of the drop spot.

3.6.3.2 Soft-drop Implementation

First, we go to the situation of the initial implementation of the demonstration: the robot's gripper is put above the spot. The soft-drop strategy then consists in moving the arm towards the drop surface, e.g. a table; the arm is speed-controlled in this case, using the algorithm presented below. In the following algorithm, \hat{z} is the estimated height from the vision, z_c is the current height of the robot's hand, z_{\min} is the minimum allowed height, corresponding to the robot's physical limitations, \dot{z} is the velocity command for the robot's hand, \dot{z}_d is the desired reference velocity given before-hand, F_s is the force read from the wrist's force sensor and F_σ is a force threshold defined before-hand.

While the hand goes down it starts to decelerate as it approaches the estimated contact height \hat{z} . The deceleration starts as the height of the hand, z_c , reaches certain thresholds which are the following:

- $\hat{z} + 0.1$; the 10cm constant was chosen because, while we control the gripper height, we have to take into account the fact that the gripper is holding an object, therefore the contact height might be lower than expected and so we start to decelerate early;
- $\hat{z} + 0.02$; if the contact has not occurred yet 2cm above the estimated height, we considerably decelerate to avoid a strong shock with the table.

Algorithm 2 The soft-drop algorithm

```

1: if  $F_s > F_\sigma$  or  $z_c < z_{\min}$  then
2:      $\triangleright$  Drop the object if a contact is detected or the hand has reached a low position
3:     Open the gripper
4:      $\dot{z} = -\dot{z}$ 
5: else
6:      $\triangleright$  Lower the speed command as the hand approaches the estimated contact height
7:     if  $z_c > \hat{z} + 0.1$  then
8:          $\dot{z} = \dot{z}_d$ 
9:     else if  $z_c > \hat{z} + 0.02$  then
10:         $\dot{z} = \frac{\dot{z}_d}{2}$ 
11:    else
12:         $\dot{z} = \frac{\dot{z}_d}{10}$ 
13:    end if
14: end if

```

Using this approach, we can ensure that the object is softly put on its destination spot. On the one hand, this is a more natural behavior and therefore the robot's actions can be better perceived as their own by the users of the system. On the other hand, it is also a necessary step for more complex scenarios, for example, a serving scenario where the robot would serve drinks to guest in which case it is essential to be able to put drinks on the table softly.

3.6.4 Action Cancellation

To ensure that the action the robot performs reflects the intention of the user, we make use of the enforced selection introduced earlier. However, this might still prove to be wrong if the user mistakenly focuses on the wrong stimulus, or the user might simply change his mind. Another issue is that the enforced selection method artificially slows down the process when the SSVEP detection algorithm performs well for a subject, in such cases we would like to reduce the “given” time of the enforced selection but this might result in wrong decisions (false positives).

In such situations it is important to be able to recover a wrong grasp and that is why we introduce here a grasp cancel capability. To illustrate this issue, we discuss the specific case of grasping in this section.

During the grasping phase, the objects within the field of view of the robot are recognized using computer vision pattern recognition techniques. The information provided by the vision are then merged within the interface to display the SSVEP stimuli upon the objects that are graspable by the robot. Then, the user focuses her/his attention on one of the objects and the robot autonomously grasps the selected object.

3.6.4.1 Grasping Strategy

Before introducing our approach to canceling a wrong grasp, we have to introduce our approach to grasping. First, the arm that performs the grasping movement is automatically selected depending on the current position of the robot relatively to the selected object. Then, the grasping movement executed by the robot can be decomposed into 5 phases:

1. Replan the robot's footprints respectively to the selected object to prepare the arm motion for the grasp;
2. Put the selected arm at the position of the selected object through a checkpoint trajectory, the robot first lifts its arm close to its body—to avoid collisions with the table where the objects are—and then moves the arm towards the object;
3. Close the correct gripper;
4. Lift-up the object;
5. Put the arm in its original position through a checkpoint trajectory, the robot first moves back its arm close to its body and then moves the arm back to its initial position.

3.6.4.2 Canceling a grasping task

Given the strategy for grasping, canceling the grasping task is simply a matter of rewinding the different phases of the movement. For example, if the grasp was canceled during the last phase of the grasping movement: the return of the arm to the initial position, then the canceling follows this sequence:

1. Go back through the checkpoints that led to the current position of the arm;
2. Put the arm down;
3. Open the gripper;
4. Go back through the first checkpoint trajectory: moving the arm back to the robot's body then put the arm in its original position;
5. Execute the opposite steps to get back to the original position of the robot.

The final issue is to be able to trigger such a cancellation. Currently, this is implemented in the user interface by displaying a flickering cross during the grasping executed by the robot to let the user cancel the grasp movement as seen in Figure 3.9. However, in the future, we would like to be able to use the error-related potential. Error-related potentials (ErrPs) are special features that can be detected in the EEG, after a wrong action selection by the BCI system or the user. After the onset of the feedback indicating the selected action, these features can be distinguished by first, a sharp negative peak after 250ms followed by a positive peak after 320ms and a second broader negative peak after 450ms [34]. Using this brain pattern in the frame of this work would allow a much smoother integration of task canceling strategies and offer premises to closing the loop in action realization and perception.

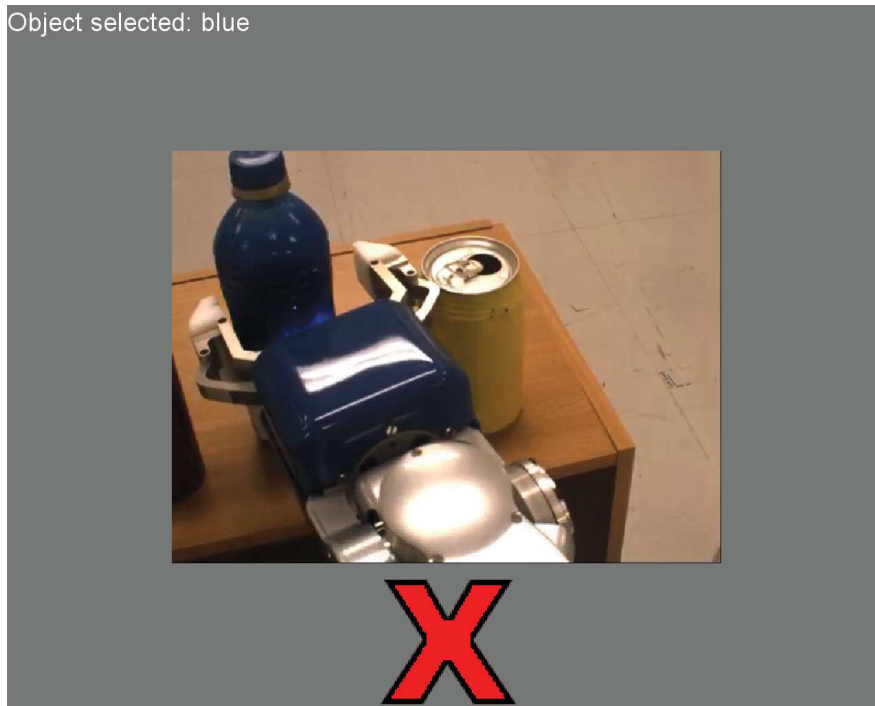


Figure 3.9: Interface displayed to the user during the grasping phase executed by the robot. If the user focuses on the cross in the bottom of the screen, the grasp is canceled.

3.6.4.3 Reverting Issue

While the proposed canceling strategy works well in the work presented here, it does not address the general issue of canceling a movement. For example, in the case of grasping, it does not take into account the dynamic properties of the grasped object, especially when those properties affect the dynamic properties of the robot's arm. The primary problem is that this method assumes that the trajectory to grasp the object and to put the object back is the same. However, this assumption is wrong. For example, if the object is particularly heavy then these two movements are different. However, the general principle behind this canceling strategy remains valid: decompose the task that has to be canceled into elementary tasks and then rewind these elementary tasks. In our case, this means that we add a memory to our stack-of-task controller.

3.7 User Experiments

In this section we present and discuss three user experiments that were conducted at different stage of the implementation of the object-centric BMI framework. We refer to the above technical sections when relevant.

3.7.1 Initial User Experiment

3.7.1.1 Experiment Scenario

The scenario is designed to illustrate multi-task control, i.e. locomotion and manipulation, via BCI.

At first, the user is presented, through the robot's eyes, with multiple objects - known from the recognition system - put on a cupboard. In this phase, the user selects the object s/he wishes to interact with. The selection relies on an SSVEP paradigm: the objects are blinked as discussed in the previous section. Once this selection happens, the robot grasps the object and the second phase of the experiment begins.

In the second phase, the user steers the robot freely in its environment to a location of his choice. The steering is done through SSVEP. Three stimuli allow the user to control the robot orientation and make it move forward, a fourth one allows her/him to stop the robot. Once the robot is stopped, the final phase begins.

In the final phase, the user recursively selects a position within the visual field of the robot. The selection of a position makes the robot drop the object above this position thus achieving the experiment.

To confirm the usability of the interface, the user is given two instructions that define her/his mission: (a) which object to pick up from the cupboard and (b) which sixteenth of the screen to select at the end. To evaluate the objective performances of the interface we measure the time taken to achieve each phase and the success rate of the missions.

3.7.1.2 Experiments

Material and System Setup

We use a g.USBamp (24 Bit biosignal amplification unit, g.tec Medical Engineering GmbH, Austria) to acquire the EEG data from the user's brain at a sampling frequency of 256Hz, bandpass filtered between 0.5 and 30Hz with a notch filter at 50Hz to get rid of the power line noise. The electrodes positioning is shown in Figure 3.10. We use 8 Ag/AgCl active electrodes. The electrodes are placed on the POz, PO3, PO4, PO7, PO8, O1, O2 and Oz positions of the international 10-20 system [74], Fpz is used as the ground electrode and the earlobe as a reference.

The experiment was carried out using the HRP-2 humanoid robot. The subject, equipped with an EEG cap, is comfortably seated in an armchair, about 1 meter away from of a 17" LCD screen. In such setup the accurate display of SSVEP stimuli is ensured thanks to the method proposed in [33] and detailed in Chapter 1. The SSVEP stimuli frequencies that were used in this work are: 6, 8, 9 and 10Hz. Those were carefully selected to have neither common first or second harmonics and are below 20Hz to minimize the risk of eliciting an epileptic crisis in healthy subjects as advised in [28].

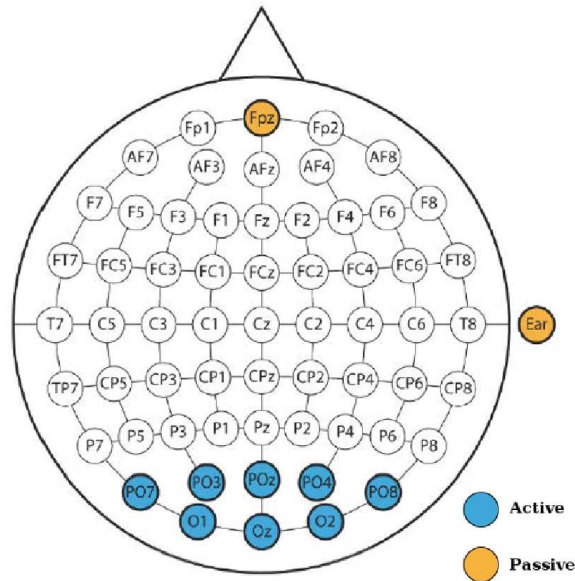


Figure 3.10: Electrodes positions for the experiment

3.7.1.3 Results

A video showing the interface in action as well as the robotic counterpart can be retrieved at the following URL: <http://dinauz.org/~pbs/Videos/ISER-2012.avi>.

Four users performed the scenario that we described. Each of them performed the scenario five times with different conditions, i.e. different objects and different drop locations. Over the 20 trials, the users' selections were consistently accurate thanks to the enforcing of SSVEP selection we setup. In Figure 3.11 we report their average performance over the multiple trials.

The phase 1 and phase 3 times are consistent with the performance of our SSVEP classification method and the adoption of the enforced SSVEP selection process. The system operates on a 3 seconds window of EEG data and we require the user to maintain her/his attention on the stimulus for 3 seconds before we make a conclusion about her/his intention. The phase 3 involves the recursive SSVEP paradigm described earlier and thus the time needed to reach a conclusion in phase 3 is about twice the time needed in phase 1 – due to a technical issue, the time reported includes the time the robot took to stop its walking and orient its head toward the table, hence about 10 seconds should be discarded.

The enforced SSVEP selection paradigm also proved its usefulness during the trials as no misinterpretation of the user's intentions occurred during these phases for all subjects across all trials. However, in this specific experiment, we chose to enforce the selection a 3 seconds period which covers 15 decisions by the SSVEP extraction process. To achieve a more reactive experience, this activation time could be tuned down according to the classifier

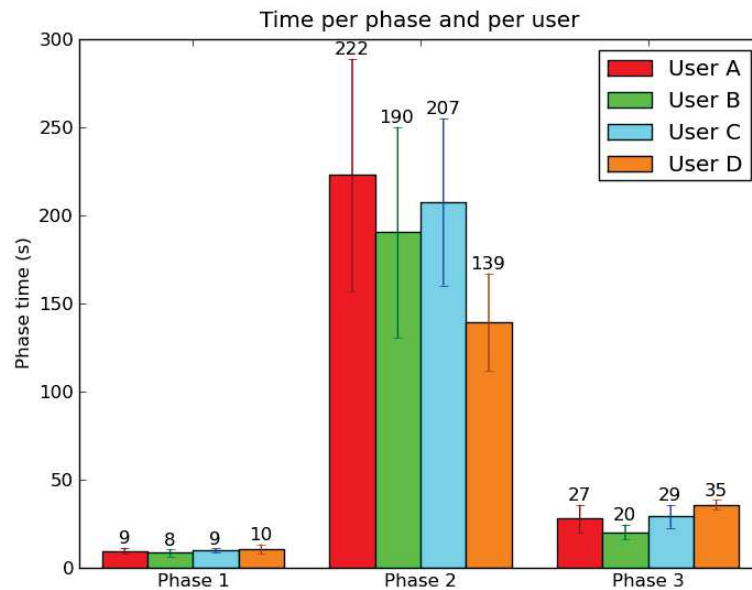


Figure 3.11: Average performance of each user over the experiment trials

performance with the user. This would allow the system to reach a conclusion more rapidly while keeping a very high-level of accuracy.

Finally, the phase 2 times illustrate interesting effects of training and motivation on the scenario performance. The performance of each subject over each trial can be seen in Figure 3.12. It shows that the user's performance in the navigation phase improve after each trial. Two factors, reported by the users, can explain this phenomenon. On the one hand, the users felt more and more comfortable with the interface. They progressively learned how to compensate for the lag of the SSVEP extraction process and they also acquainted themselves with the environment and the way it is perceived through the robot's camera. Indeed, this camera has a rather constrained field of view which is well suited for the stereo application it is used for but makes it particularly difficult for the user to understand the robot's location in space. On the other hand, the user also used more 'aggressive' strategy to reach their destination because they felt more comfortable steering the robot but most importantly because they wished to 'beat' their previous time which illustrates the importance of motivation in the context of BCI applications [41].

3.7.2 Assessing the benefits of Navigation Assistance

This experiment focuses on the addition of a navigation assistance mechanism to the previously described experiment implementation and highlight its benefits towards the user's ability to control the robot.

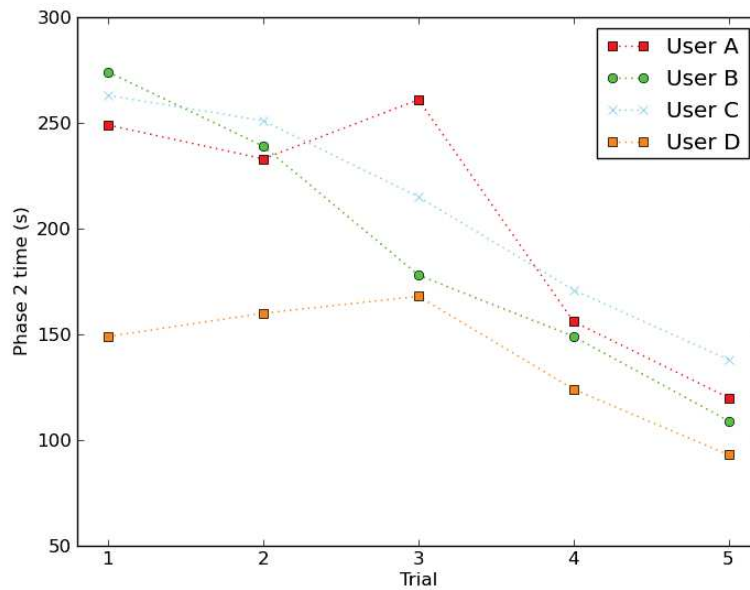


Figure 3.12: Walking phase performance for each subject and trial

In the user study we presented previously, the most difficult phase is the navigation phase. In this phase the user has to reach a table where s/he drops the object. The steering is done using flickering arrows that allow steering the robot left, right, backward and forward. A fifth stimulus lets the user stop the robot. The user has to get close enough to the table to allow the robot to drop the object on it afterwards.

3.7.2.1 Expectations

In this experiment, we reproduce the same setting as in the previous one, however, we use the simple navigation assistance provided by depth perception and introduced in Section 3.6.2.1. We expect the benefits of this detection to be twofold:

Safety and reliability

The user instructions cannot make the robot collide with its environment anymore (that was the case when we let the humanoid robot freely steered by the BCI), and no training is required for the user to steer the humanoid robot into the proper position.

Performance

The improved reliability may allow us to speed up the robot during the walking phase, it also eases the learning of the robot control through BCI and thus the overall performance is increased.

3.7.2.2 User Experiments

To assess our expectations, we performed a user experiment using the technical elements presented in the previous section.

Material and Methods

Eight healthy volunteers participated in the experiments, six males and two females, aged 21.5 ± 1.07 y. None of them had previous experience regarding BCI control or robotics.

We use a g.USBamp (24 Bit biosignal amplification unit, g.tec Medical Engineering GmbH, Austria) to acquire the EEG data from the user's brain at a sampling frequency of 256Hz, bandpass filtered between 0.5 and 30Hz with a notch filter at 50Hz to get rid of the power line noise. We use 8 Ag/AgCl active electrodes. The electrodes are placed on the POz, PO3, PO4, PO7, PO8, O1, O2 and Oz positions of the international 10-20 system, Fpz is used as the ground electrode and the right earlobe as a reference.

The experiments are conducted with a HRP-2 humanoid robot running the control system described in the previous section.

During the experiment, the users are located in Rome (Italy) while the robot is located in Tsukuba (Japan). The user is comfortably seated in an armchair, about 1 meter away from a 19" LCD screen operating at a refresh rate of 60Hz upon which the user interface is displayed.

Experimental Conditions

First, the SSVEP classifier is trained for the user. Then, a short video is presented to the user to explain the task and the goal of the task. In this experiment, the distance between the grasped object and the destination table is about 4 meters, much shorter than in the initial user experiment. A target is positioned on the final table to give a clear goal to the user.

A trial consists of the successful execution of the entire demonstration: from the selection of the object to the drop on the destination table. Each subject performed the task 4 times. These experiments were conducted within the scope of the embodiment study that is described later in Chapter 4.

For each trial, we measure the time taken by each phase and the distance between the object and the target once it has been dropped. In this section, we only report the results regarding the navigation phase time, the error between the target and the final object's position are discussed in a later section.

Results

Some trials failed due to network connection problems during the experiments, in such cases the trial was restarted and thus was not included in the results. In the end, all users successfully completed the task 4 times with varying degree of success regarding the accurate drop of the object on the target.

In Figure 3.13, we report the average time taken by the subject to perform the navigation phase.

We can notice that the walking phase time is much less variable than in the previous user

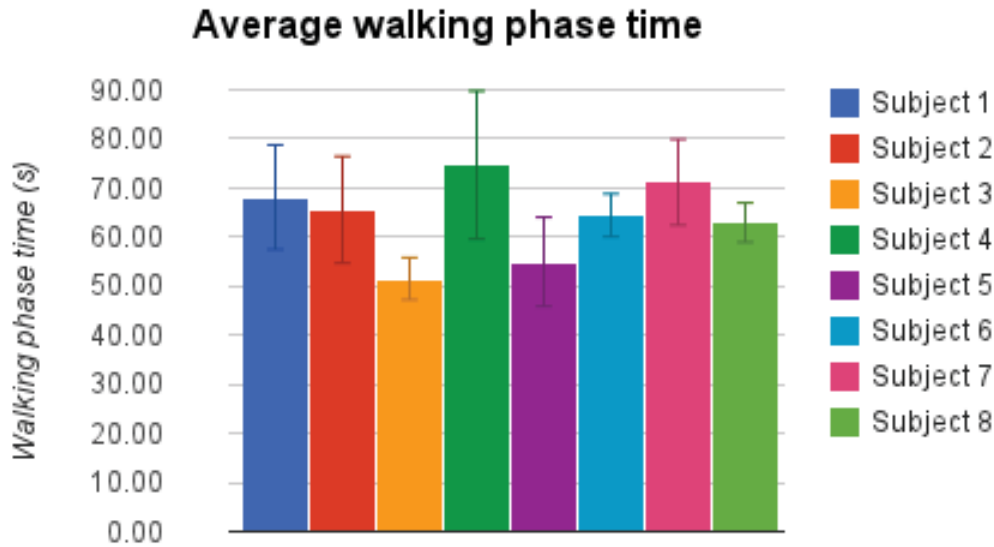


Figure 3.13: Average walking phase time for each subject, the vertical bar represents standard deviation.

experiment. This shows that the addition of the obstacle detection brought confidence to the user as s/he was told there were no risks of collision between the robot and the destination. The users all reported that they felt in control of the robot’s actions and that the control of the robot through the BCI was easy.

3.7.3 A Bayesian-network assisted BCI for Humanoid Robot Control

In the experiments we presented in this chapter so far we gradually introduced more complexity within the demonstrator while improving the control capabilities of the user. However, the *Task Selection Module* remained fairly simple as only one task—“Grasp”—was known to the *Objects and Affordance Extraction Module* and thus each object was associated with a single task.

In this section, we introduce a complete implementation of the object-oriented BCI framework and show the benefits it can introduce within the context of BCI controlled humanoid robots.

3.7.3.1 User Experiments

We expect the addition of a complete *Task Selection Module* to significantly simplify the control of humanoid robot using a BCI in a complex multi-task scenario. Therefore, the experimental protocol we introduce assess the difference in performance between a complete and non-complete object-oriented BCI in such a scenario. The Bayesian network that is employed in this scenario was introduced in the previous section. We now focus on the experimental protocol and the results of the experiments.

Experimental Protocol

The objective of this experiment is to evaluate the impact of adding a decision tool within the “pick-and-place” scenario that we introduced earlier. Therefore, the experiment scenario is based on this one, then:

- The user selects an object;
- The Bayesian network infers the most probable task based on the object properties;
- The task starts being executed while the user monitors its progress;
- If necessary, the user can select where the object s/he carried to a location should be dropped.

Each user is given a set of two tasks to realize and s/he should realize them two times: one time with the Bayesian network assistance and the object-oriented navigation scheme and another time without the Bayesian network or the object-oriented navigation scheme. For each task, we observe whether it was successfully accomplished or not and we measure the time that was taken to complete it. This allows us to compare the two methods objectively. The first task—Task 1—is to grasp an object from a shelf and put it on another shelf in the room. The second task—Task 2—is to grasp an object and throw it in a garbage can that is also in the room.

Finally, we ask the users to fill out a questionnaire following the experiment. This questionnaire is presented thereafter.

User Questionnaire

The questionnaire presented in Table C.2 in Appendix C.2 is proposed to the subjects after the experiment. It aims at assessing the preferred control scheme as well as receiving meaningful feedback regarding our approach.

Results

Three subjects, aged 32.3 ± 7.57 y, all male, took part in this experiment. They performed the same tasks but the tasks and the experimental conditions—with or without Bayesian network assistance—were shuffled. The first two subjects, Subject A and Subject B, had previous experience with BCI while Subject C had none.

Task 1 was achieved successfully for each subject and condition. Task 2 was failed by Subject A and Subject B in the “Manual” condition because they stopped the robot too close to the garbage can and could not select it.

During the experiments, we recorded the time they took to complete each task. In Figure 3.14, we report the time they took to complete the navigation phase of those tasks depending on the user and the experimental condition. In the “Assisted” condition, the robot uses the autonomous navigation scheme described earlier to reach the place chosen by the Bayesian network. In the “Manual” condition, the user steers the robot freely to reach the place where the task should be executed.

The following observations can be drawn from this data:

- The assisted scheme performed similarly across the subjects (Task 1: 58.50 ± 0.88 s, Task 2: 84.66 ± 10.89 s).

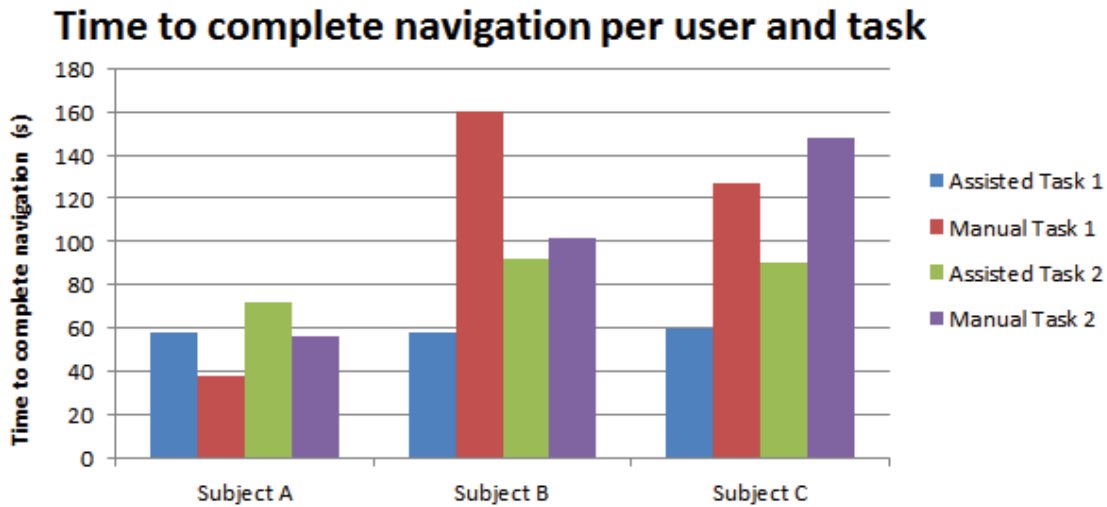


Figure 3.14: Time to complete the navigation phase of the task per subject, task and experimental condition. Subject B took a long time to stop the robot when performing Manual Task 1, hence this time is abnormally high. Before attempting to stop the robot, Subject B's performance in manual navigation was close the assisted navigation scheme.

- Subject A outperformed the assisted scheme in navigation. This is explained by both excellent BCI performances—no false-positive or false-negative during navigation—and assistive navigation limitations that are discussed later.
- Subject B performed equally well using either scheme. The stopping issue that happened when performing Task 1 did however lead to some frustration that was not present in other trials.
- Subject C, a BCI novice, could not reach the same level of performance using the manual navigation scheme.

The questionnaire answers do not provide an interesting metric to compare the two schemes as they were equally positively received.

Discussion and Perspectives

The addition of the Bayesian network and the assisted navigation scheme generally allowed the task to be performed efficiently and accurately. However, time-wise, a BCI user with good performance could outperform the assisted navigation scheme using manual navigation. This is explained by two limitations of this work that need to be investigated.

First, the assistive navigation scheme we used is slow. Most of that time is spent trying to find the object of interest. This time could be cut down by either using the semantic SLAM approach we described earlier or relying on user's inputs—through BCI or through head's movements for example—to guide the search for the object. The latter solution would also help to keep the user engaged into the control of the robot and thus hopefully improve the embodiment property.

Finally, the Bayesian network that we implemented for this experiment is still rudimentary and the experiment's environment is too small to allow it to surpass the manual approach.

In future works, we would like to expand the scope of these experiments to realistic settings, for example, performing complex tasks in a house-like environment—multiple rooms, many objects to interact with, many tasks to perform—where this decision tool could truly provide assistance.

3.8 Conclusion

We demonstrated that brain signals can be used to command a sophisticated mobile device such as a humanoid robot to perform useful tasks like navigating to a desired location and interacting with different objects through grasping/pressing etc. We used computer vision to discover objects of interest in the environment, a dynamic immersive interface that communicates the corresponding actions to the user and a humanoid robot controlled by the user in an immersive environment using BCI. In principle, the robot could execute a wide range of actions instructed by the user in a multi-task scenario.

This problem was solved by the introduction of the object-centric BCI framework that allows the user to perform complex interactions with the environment without the need to gain low-level control of the robotic device which is an impractical solution to achieve with BCI. We implemented increasingly complex demonstrators based upon this framework. This was made possible by incremental improvements to the framework's modules which show its extensibility. However, we wish to continue to extend the capabilities of this system to provide a more immersive experience. In particular, some issues have still to be figured out and would tremendously improve the capacities of the interface and the embodiment sensations:

Improving the field of view feedback

One of the most detriment limitation of the current system is the limited field of view proposed to the user. That is a consequence of two things: (i) the limited FoV of the HMD we use and (ii) the limited FoV of the camera. While there is not much that can be done regarding the first issue, we can virtually extend the field of view by integrating the actual image from the robot's sensors into a SLAM map that we are building online during the demonstration [83] [85].

Improving the Bayesian network capabilities

The Bayesian network experiment we presented remains at an early stage and could be improved. In particular, the states that compose the nodes of the network should be discovered from the perception system rather than pre-programmed as in the experiment we presented.

Merging multiple BCI paradigms

To improve the embodiment capacity of the interface some of the tasks parameters should be decided by the user. Using multiple BCI paradigms, for example motor-imagery, could allow additional inputs that could then be used to customize the task execution based on user preferences.

Closing the framework

At the moment, no satisfying error potential implementation that work concurrently with SSVEP has been made available by g.tec. Hence we relied on alternative ways to close the loop in the framework such as the user inputs presented in Section 3.6.4. However, the use of error potentials would lead to a much more immersive experience.

In the following chapter, we show how the object-oriented BCI framework that we have introduced in this chapter can be used to study different aspects of embodiment in the case of auditory feedback.

Humanoid Control with a Brain-Computer Interface: Applications

The tools we developed along the course of the work presented in this thesis gave us a good experience on both the robotic aspect and the BCI aspect of the control of a humanoid robot with a brain-computer interface. This experience was used to perform several studies that were not directly related to solving the problem of controlling a humanoid robot through a BCI but rather used the BCI and the humanoid robot as tools to perform studies that goes beyond this scope.

In this chapter, we present the most thorough application of our work: the study of the impact of sound on the embodiment quality. This study is conducted through two experiments that are both presented here. In the first study, we determined the most appropriate auditory feedback we could provide and we proposed a scenario to test different feedback modalities and their effect on the robotic control within a simple “pick-and-place” scenario with an intentionally limited visual feedback. This study led to significant results regarding the role of sound in the embodiment process [98] which are reported here. Following these results, we designed a new study with two major modifications. On the first hand, the subjects have to perform a significantly more complex navigation task. On the other hand, visual feedback is provided through an head-mounted display (HMD) which provides a more immersive setting. A thorough study was conducted within this setting involving both healthy subjects and patients suffering from spinal cord injury (SCI). Although this study has been inconclusive so far, we discuss further analyses that may lead to significant results.

4.1 Studying the Impact of Sound in Embodiment Experience

One main goal of the VERE project is to allow people with limited mobility to interact with the environment using a robot remotely controlled through BCI technologies. One important point of novelty of VERE is the possibility to make people with motor and sensory disabilities perceive the body of an external agent (physical or virtual) as if it were their own body. In this section, we describe a novel experimental task aimed at evaluating how auditory feedback can increase the sense of presence and the performance of a user in Rome controlling via BCI the actions of a humanoid robot located in Japan. We also report preliminary data collected on a small sample of healthy subjects. The experiment is an upgrade of the previously designed scenarios. In the original scenario, a BCI user steers a robot from a starting position to a final target location while hearing a footstep sound either synchronously or asynchronously with the real footsteps of the robot. In the present experiment, to increase the participants' engagement we added to the original plan an additional phase in which participants instruct the robot via BCI to grasp a bottle, walk for some meters and place the bottle on a table. From a robotic point of view, the experiment is very similar to the one we introduced in [75] and in Chapter 3 of this document hence the technical details of the experiment are not further discussed in this chapter.



Figure 4.1: A BCI user controlling the robot from Italy.

4.1.1 Material and Methods

4.1.1.1 Participants

A total of 28 healthy subjects took part in the study. Nine subjects (3 women; age range, 20–26 years) participated in the main experiment and 19 subjects (12 women; age range, 21–33 years) were tested in one of the two pilot studies. All the subjects were right handed according to a standard handedness inventory [99] had normal or corrected-to-normal visual acuity in both eyes, and were naive as to the purposes of the experiments. None of the participants had any contraindication for the BCI study [28]. Participants provided written informed consent and the procedures were approved by the ethics committee at the Fondazione Santa Lucia and were in accordance with the ethical standards of the 1964 Declaration of Helsinki. Participants received reimbursement for their participation and were debriefed on the purpose of the study at the end of the experimental procedure. In the SSVEPs-BCI experiment no discomfort or adverse effects were reported or noticed.

4.1.1.2 Pilot Studies - Footsteps Sounds Selection

Two independent groups of subjects have been tested in different pilot studies (Group 1, 12 subjects, 7 female, range 21–33 years; Group 2, 7 subjects, 5 female, range 20–23 years). In Pilot 1, eight different human footsteps audio files were interleaved by a variable number (min 5, max 10) of pure tones to avoid habituation and randomly heard by participants. Subjects were asked to guess what the sound was and type the answer within a text box. Participants listened to each sound only once. The sound represented two “hits” with the ground (e.g., right-left foot). In Pilot 2 participants rated on a 0–100 Visual Analog Scale (VAS) how much the sound they heard was reproducible by the human body. The sounds were the same as in Pilot 1 and interleaved by pure tones. The selected sound was freely categorized as “footstep” by 91% of the sample from Pilot 1 and rated as reproducible by the human body at the 93.7 ± 2.57 (mean \pm s.e.m.) on the 0–100 VAS scale from Pilot 2. In this way we chose the most recognizable footstep sound that was judged as highly reproducible by the human body.

4.1.1.3 Main Experiment - Task Description

Participants located in Rome (Italy) controlled a HRP-2 humanoid robot located in Tsukuba (Japan, see Figure 4.1) by a SSVEPs-BCI system. The task had four sub-goals (SGs). First the BCI user had to steer the robot from a starting position to a table (SG1) and command HRP-2 to grasp a bottle (SG2). Then, the participant guided the robot as close as possible to a second table (SG3) and tried to drop the bottle as close as possible to a target location marked with two concentric circles (SG4, see Figure 4.2A). Participants were asked to complete the task as fast and accurate as possible. An enforced- (for all SGs) and a recursive-selection (SG4) were adopted to allow the user to complete the task. In half experimental conditions, we placed a mirror behind the second table to provide additional information about the spatial proximity between the table and the robot. During the task the BCI user heard a footstep

sound either synchronous or asynchronous with the real footsteps of the robot (during SG1 and SG3) and could see or not the robot's body reflected in a mirror placed behind the second table (SG3 and SG4). In this way we had a 2×2 design with Footstep (Synchronous, Asynchronous) and Mirror (Present, Absent) as within subjects factors for a total of four experimental conditions. After each condition the subject answered questions concerning the experience they had (see the questionnaire presented below). Importantly auditory feedback was delivered in Italy. Data regarding robot's feet contact were streamed from Japan along with the video stream. Hence when a foot touched the ground we could deliver an auditory feedback to the user located in Italy synchronously or asynchronously with the video feedback.

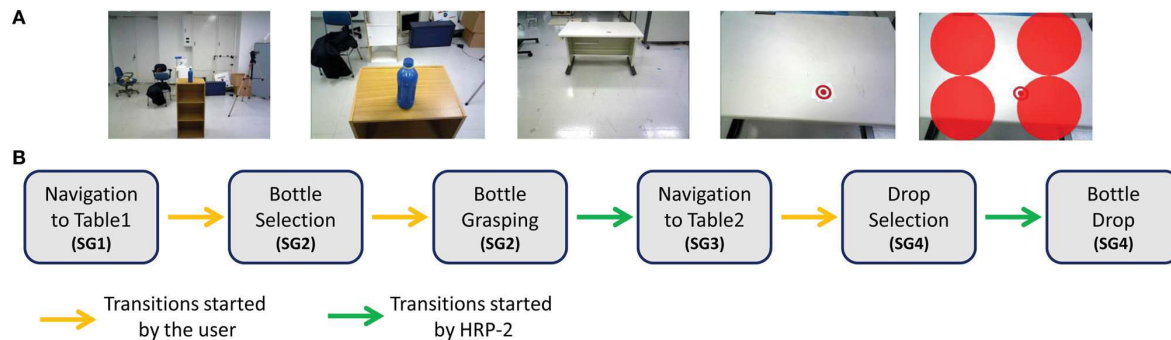


Figure 4.2: **(A)** A sequence of images depicting the different sub-goals (SGs). **(B)** A state-flow diagram showing the Finite State Machine (FSM). Yellow arrows represent transitions initiated by the user while green arrows represent transitions initiated by the robot.

4.1.1.4 Procedure

The user was comfortably sitting in an armchair about 75cm away from a 19" LCD screen operating at a refresh rate of 60Hz upon which the user interface was displayed. The SSVEP classifier was trained on individual EEG data and a short video was presented to explain the goals to accomplish during the experiment. A trial consisted of the execution of the entire demonstration: from grasping to dropping the bottle on the second table. Two concentric circles indicated the target position on the second table. At the end of the trial, participants answered 4 questions to assess the quality of their interaction with the robot. An initial practice trial was performed before the experimental conditions. Note that we have been able to set the classifier error rate at 0% for all subjects. After training the error rate was null after 7.52 ± 0.22 s (mean \pm s.e.m.). All participants completed successfully the training and were able to use the SSVEP.

4.1.1.5 Data Analysis

The Total time to complete the task, Walking time (i.e., time to steer the robot from the first table to the second one; SG3) and Place Accuracy (i.e., displacement between the target location and dropped bottle position; SG4) were used as measures of behavioral performance.

We discarded from the main analysis one participant who did not follow task instructions and presented a Walking time 202% higher relative to the other participants. Thus, the final sample was of 8 subjects (2 female, 21.5 ± 1.06 , range 20–23 years). An ANOVA with Mirror (present, absent) and Footstep sound (Synchronous, Asynchronous) as between subjects' factors was performed after checking for normality distribution (using the Shapiro-Wilk test). Subjective answers to the quality of the interaction were analyzed by means of non-parametric tests Friedman ANOVA and Wilcoxon test for within-group comparisons.

4.1.2 Hardware and software integration

Brain computer interface

EEG signals were acquired at 256Hz by means of a g.USBamp (24 Bit biosignal amplification unit, g.tec Medical Engineering GmbH, Austria). We applied a band-pass (0.5–30Hz) and a notch filter at 50Hz. Eight Ag/AgCl active electrodes were placed on the POz, PO3, PO4, PO7, PO8, O1, O2, and Oz positions of the international 10–20 system. Fpz was used as ground electrode and the right earlobe as a reference.

Feature extraction

SSVEPs were extracted from the EEG signal using a minimum energy classifier [30]. The system categorized 6 different classes with 100% accuracy in all participants after a short training (6 min). Five classes were implemented to steer the robot (forward, 6Hz; backward, 9Hz; turn right, 8Hz; turn left, 14Hz; stop, 10Hz) and an additional zero-class detected when the user was not attending to any flashing stimuli. The frequencies were selected to avoid first or second harmonics and to minimize the risk of eliciting seizures in healthy subjects (Fisher et al., 2005). Critical frequencies fall in the range of 15–2Hz. It is worth noting that some individuals may result sensitive also to higher rapidity of the flashing stimuli [28]. Moreover, frequencies higher than 25Hz are difficult to render properly on a 60Hz monitor. Thus, the selected frequencies of the flashing stimuli were 6, 8, 9, 10, and 14Hz.

Graphical user interface

During walking phases, four flickering arrows allowed the participant to send the robot instructions to move leftward, rightward, backward, or forward. A fifth flashing square was added in the bottom-left of the interface and was meant to stop the robot. When the BCI recognized one of these possible commands, the corresponding arrow or square's border changed color (from black to green, see Figure 4.3).

From that moment, participants observed a change in robot's video feedback after about 800ms (400ms to send information to Japan, 400ms to receive robot's cameras feedback in Italy). Moreover, color-change signaled to the BCI users whether the system was correctly or incorrectly categorizing their intentions. This may be relevant for the experienced sense of control over robot's actions in addition to the audio-visual temporal matching. It is also worth noting that distal intention increases the sense of agency as measured by intentional binding [100] [101]. Finally a simplified table-icon changed color (from green to red) when

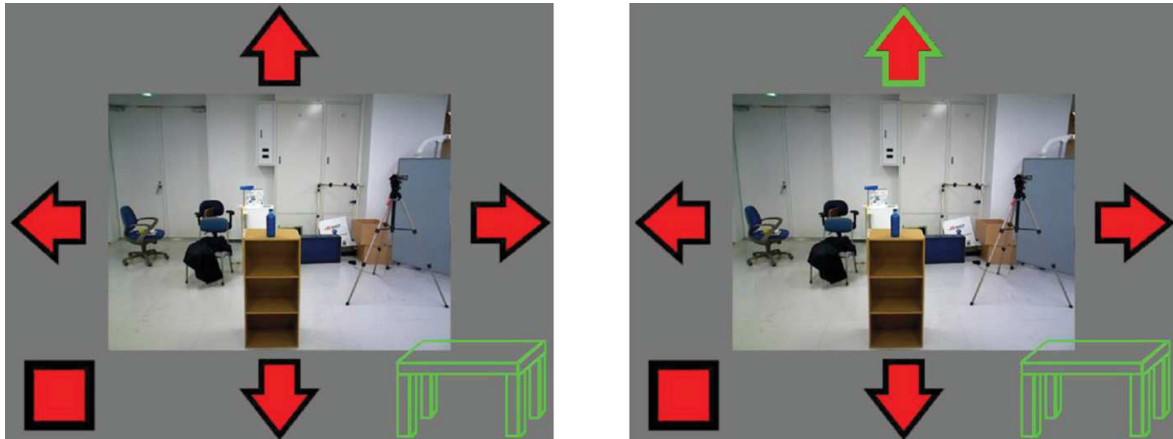


Figure 4.3: Within the implemented graphical user interface whenever the BCI recognized a command (e.g., top arrow, forward-command), the corresponding icon's border changed color (from black to green).

the proximity of the robot to the table was too close and potentially dangerous.

Sub-goals transitions

Transitions between different phases of the SGs were achieved by the implementation of a finite state machine (FSM, Figure 4.2B). In this way we combined both user' and robot's autonomy. The BCI user freely decided for the robot's walking directions, the moment to grasp the bottle and the place for dropping it. HRP-2 followed BCI-users' intention and autonomously implemented the correct movements for walking (SGs1-3), grasping (SG2) and dropping (SG4). Therefore, transitions between different SGs were triggered either by the user or the robot with a FSM as shown in Figure 4.2B.

4.1.3 Results

4.1.3.1 Performance Assessment

Total time

Data were normally distributed (Shapiro-Wilk test for all conditions: $p > 0.20$). The ANOVA did not reveal any main effect [all $F_{(1,7)} < 0.62$, $p > 0.45$, $\eta^2 < 0.08$] or interaction [$F_{(1,7)} = 4.53$, $p = 0.07$, $\eta^2 = 0.39$]. We performed an additional ANOVA to check any learning effect with the Order of the conditions as 4 level factor (Trial1, Trial2, Trial3, Trial4). We did not observed any learning effect [$F_{(3,21)} = 1.38$, $p = 0.27$, $\eta^2 = 0.16$].

Walking time

Data were normally distributed (Shapiro-Wilk test for all conditions: $p > 0.15$). The ANOVA revealed a main effect of Footstep [$F_{(1,7)} = 10.10$, $p = 0.01$, $\eta^2 = 0.59$] with faster time in the Synchronous (mean \pm S.E., $60.00s \pm 2.62$) relative to the Asynchronous condition ($68.36s \pm 3.42$; Figure 4.4). No main effect of Mirror [$F_{(1,7)} = 3.96$, $p = 0.09$,

$\eta^2 = 0.36$] or interaction [$F_{(1,7)} = 0.01, p = 0.97, \eta^2 < 0.01$] was found. We additionally checked for any learning effect. The ANOVA performed with Order as a 4 level factor (Trial1, Trial2, Trial3, Trial4) did not reveal any effect [$F_{(3,21)} = 0.86, p = 0.47, \eta^2 = 0.11$]. To rule out any role of distraction or drop of attention during the asynchronous condition we checked the total commands sent to the robot and the times the robot stopped (i.e., the BCI classifier categorized a “zero class”) as index of participants’ uncertainty that may have explained the result.

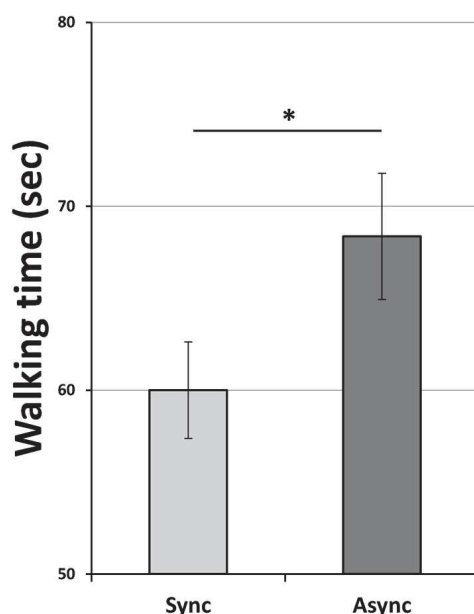


Figure 4.4: Mean walking time to drive the robot from the first to the second table and drop the bottle. Light-gray and dark-gray columns represent Synchronous and Asynchronous footstep sound heard by participants. Error bars represent s.e.m. Asterisk indicate significant comparisons ($p < 0.05$)

Due to technical failure for one subject the number of commands and stops sent to the robot were missing in the Sound Asynchronous—Mirror Absent condition. The missing cell was substituted by the mean of commands and stops the subjects ($n = 7$) sent to the robot in that condition. Neither the number of sent commands in the Synchronous (25.87 ± 4.00) and Asynchronous (29.55 ± 3.00) condition [$t_{(7)} = -1.53, p = 0.16, r = 0.06$] nor the times the robot stopped in the Synchronous (4.44 ± 0.89) and Asynchronous (4.29 ± 0.67) condition [$t_{(7)} = 0.27, p = 0.79, r = 0.28$] did differ.

Place accuracy

Data were not normally distributed (Shapiro-Wilk test in two out of four conditions: $p < 0.05$). Non-parametric Friedman ANOVA did not reveal any differences in the Place accuracy ($\chi_3 = 5.07, p = 0.16$).

4.1.3.2 Quality of the Interaction

Table 4.1 describes the questions participants answered after each experimental condition. Participants were asked to answer on a 0-100 scale (where 0 indicates “nothing” and 100 indicates “very much”).

Questions aimed at investigating	Item
Sense of Presence	
1	I had the sensation to be where the robot was. Where 100 represents normal experience of being in a place.
2	There were moments that the images I saw were the reality.
3	There were moments that the laboratory was the reality.
4	During the control of the robot, there were moments when I actually thought I was where the robot was.
5	<i>(eyes closed)</i> When you think back to the experience, do you recall it as you personally did it.
6	<i>(eyes closed)</i> Mentally rehearsing the experience makes me feel like I actually were where the robot was.
Sense of Embodiment	
1	<i>(mirror condition only)</i> When I saw the robot in the mirror I was surprised not to see my body.
2	It was as if the robot were my body.
3	It was as if parts of the robot (legs, limbs, hands...) were part of my body.
4	It was as if I had two bodies (one here, one there).
5	It was as if my body were becoming robotic.
Sense of Agency	
1	The robot executed my commands.
2	I was in control of robot’s actions.
BCI Interface	
1	I paid attention to the images displayed.

2	It was easy to instruct the robot about the direction where to move.
3	Looking at the flashing arrows was difficult .
Plausibility	
1	Displayed images conveyed a sense of realism/plausibility.
2	The actions (walking, grasping, placing) conveyed a sense of being really executed (it was not a pre-recorded movie).
Illusionary Movement	
1	It was like walking (as if my legs were actually moving).
2	I had the feeling of swaying along with the robot.

Table 4.1: List of the questions participants answered after each experimental condition.

The results are summed up in Table 4.2. We assessed the felt agency (Q1) over robot's actions and no differences between experimental conditions (Friedman ANOVA, $\chi_3 = 3.08$, $p = 0.38$) were found. Throughout all the experiment, participants did pay attention to the images while they were controlling the robot (Q2, $\chi_3 = 1.19$, $p = 0.75$) and found easy to steer HRP-2 (Q3, $\chi_3 = 3.65$, $p = 0.30$). Importantly, participants reported no discomfort due to the flashing stimuli (Q4, $\chi_3 = 1.26$, $p = 0.73$) in any condition.

	Sound Sync		Sound Async	
	Mirror	no-Mirror	Mirror	no-Mirror
Q1 I was in control of robot's actions	66.88±6.74	73.13±5.08	67.50±7.07	71.88±5.50
Q2 I paid attention to the images displayed	71.88±6.40	71.88±6.54	74.38±8.10	77.50±7.91
Q3 It was easy to instruct the robot about the direction where to move	64.38±7.99	68.75±5.49	65.63±10.41	62.50±5.00

Q4 Looking at the flashing arrows was difficult

25.99±8.02	27.50±7.96	28.13±9.54	28.13±7.44
------------	------------	------------	------------

Numbers represent values comprised between 0 and 100 (mean ± s.e.m.)

Table 4.2: Participants answers to questions assessing the quality of the experience.

4.1.4 Discussion

Since the SSVEPs BCI system is a rather stable one, we adopted it to test the role of visual and auditory feedback during the remote control of a humanoid robot (HRP-2). More specifically, we explored whether temporal congruency of seen and heard input can modify the ability to steer a robot and the perceived quality of the human-robot interaction. We designed an easy way to use graphical interface sending to participants information about obstacles' proximity. Participants continuously controlled the walking directions of the robot by means of the SSVEPs flashing commands. The manipulation of audio-visual feedback was effective in modifying the participants' performance. More specifically, footstep auditory feedback delivered synchronously with the seen robot's steps allowed the participant to use less time for correctly driving the robot from the first to the second table. This effect was limited to the walking phase and did not generalize to the total time required to complete the task. It is worth noting that the overall time is composed of transitions autonomously initiated by the robot and that only the walking phase required an accurate control to turn and drive the robot to the second table. These factors may have flattened the overall time required to accomplish the task resulting in a non-statistical difference of the total time between the experimental manipulations. Importantly participants did not differ in the total number of commands and stops sent to the robot. This indicates that participants were not more imprecise or erroneous in sending commands to the robot in the asynchronous relative to the synchronous condition. The ability to efficiently decide when an action has to be performed within a given environment may be affected by the reliability of sensory information. Feedback uncertainty may indeed affect this decision making process [102]. The sound of synchronous footsteps may have improved the ability to decide "when" changing command. In other words, the use of audio-visual synchrony may have helped BCI-controllers' decisions to better coordinate the robot. This result is relevant for current research in BCI the aim of which is to improve the human control over artificial devices either physical (e.g., humanoids robots) or virtual (e.g., avatars). Furthermore we showed that participants could maximize the performance (faster walking) taking advantage of audio-visual feedback. The observed advantage did not parallel any change in the perceived effort as indicated by the absence of differences in the questionnaire concerning the synchronous versus asynchronous experimental conditions.

Our data expand a previous study that showed that background noise did not affect the

users' BCI performance [103]. Indeed the overall performance (total time and place accuracy) in the asynchronous conditions did not differ from synchronous conditions. Importantly, the increased ability to move the robot did not affect the dropping accuracy. Moreover the mirror did influence neither the speed nor the drop accuracy suggesting that spatial information from the mirror did not facilitate dropping ability. It is worth noting, however, that participants may need additional training or more time to actually learn to use mirrored images for a better control of robot's orientation respect to the external environment. Related to this, we also note that the relative narrow field-of-view (FoV) may indeed represent a limiting aspect of teleoperation [75]. FoV can affect the ability to discriminate distances and consequently the ability of subjects to avoid obstacles and stop the robot at the right distance to perform an action. Moreover we applied recursive and enforced selection, a camera depth perception and combined user and robot's autonomy to facilitate the interaction with the environment. All these advancements were implemented in a graphical interface that allowed participants to pay attention to the environment while controlling the robot.

Overall we obtained two main results: (i) maintaining a good level of users' experience and (ii) improving the navigational performance. More specifically, the performance and quality of the interaction were assessed in this study involving eight healthy people who successfully teleoperated the robot through BCI. Importantly it has been reported the automatic tendency of the central nervous system to integrate tactile and auditory feedback even when presented with a slight temporal asynchrony [104]. This may be in keeping with the result that in our setup asynchronous auditory feedback did not affect the perceived sense of agency. It is also worth noting that this result might also have been positively influenced by the visual feedback provided by coloring the selected command. The combination of these two factors may explain why our participants maintained a good sense of agency throughout the task and experimental conditions.

Finally we did not observe a learning effect as previously reported [75] although the setups were different. Participants were not faster in the last trials relative to the initial ones. This may indicate that the asynchronous auditory feedback may have disrupted a possible learning curve. Our results are relevant for current BCI research in that they highlight how congruent sensory information can improve human-machine interaction without affecting the quality of the experience.

4.1.5 Conclusion

The continuous technological advancement in the field of controlling external devices introduced several possibilities to actually act on the environment through a surrogate. The advancement of peripheral devices made possible the combination of BCI and eye tracking technologies [105] [106]. Here we designed and tested an application that used SSVEPs BCI system to decode user's intentions. Future works should combine EEG and eye tracker systems to integrate robot's navigation and action to interact in the world. An example may be the use of eye gaze for navigation and object-integrated SSVEPs for action. This would increase the options to perform different actions on the same object. Eye tracker might indeed perform a rough recognition of user's navigational purposes and SSVEPs, through recursive

and enforced selections, might categorize the action to be executed on a specific object. This line of improvement should also parallel the development of the same application scenarios with more immersive systems like head mounted displays and very immersive virtual reality experience of the type one can experience in CAVE systems. Moreover, more detailed questionnaire may shed new light about the feeling of control and comfort that remote BCI users may experience during similar human-robot interface. All in all, our data are important for designing user-friendly interfaces that allow people who cannot control their body anymore (e.g., spinal cord injured patients) to re-enter to the world in the most efficient possible way.

4.2 Impact of Sound in Embodiment Experience with a Strong Visual Component

4.2.1 Introduction

This second study represents an upgrade compared to the one we presented previously. From a technical point of view, all but one of the elements that compose this study remain the same than in the previous study. Thus we only highlight the changes regarding the scenario and the experimental method. However, its scope is different. In the previous scenario, a BCI user steered a robot from a starting position to grasp a bottle, then walked for some meters and placed the bottle on a table. In the present scenario, the user starts by grasping a bottle but the difficulty of the walking task has been increased by putting a gate in the robot's path, as seen in Figure 4.5. Furthermore, the interface is now displayed on a head-mounted display (HMD), the Oculus Rift, instead of a LCD screen. This creates a much more immersive experience for the user.



Figure 4.5: Setup of the second embodiment experience. The path to the table is “blocked” by obstacles and the user has to navigate through it. Marks are placed on the ground to help the user navigate and on the table to indicate the drop location.

As in the previous study, we provide sound feedback during the walking task but in this study we oppose four conditions and we wish to understand the impact, if any, on the

embodiment and control performance of the system. The four conditions that we study are:

1. Biological sound synchronized with the walking;
2. Biological sound unsynchronized with the walking;
3. Non-biological sound synchronized with the walking;
4. Non-biological sound unsynchronized with the walking.

Finally, a group of spinal cord injury (SCI) patients also participated in this study and performed the same experiment as a group of healthy subjects, allowing us to compare each group performance as well as the impact of sound feedback on each group.

4.2.2 Participants

Two groups of participants are involved in this study. A group composed of healthy subjects and a group composed of SCI patients.

4.2.2.1 Healthy participants

Participants ($n = 14$, mean age 24.66 ± 2.96 , 6 females) seated comfortably in a chair in front of a monitor. A g.tec EEG-cap was mounted (8 channels with the same configuration than the previous study) and the task explained. All the participants had normal or corrected to normal vision.

4.2.2.2 SCI patients

At the moment of writing, we tested two patients with SCI that we refer to as Patient1 and Patient2. Patient1 had a lesion level C4-C5 (age 22), Patient2 had a lesion level of C3-C4 (age 31).

4.2.3 Methods

We trained the g.tec-SSVEP classifier and we have been able to select 0% error rate for each participant. We presented a short video to explain the task and the goal of the task which consisted in grasping a bottle placed on a table and then place the very same bottle as close as possible to a target position on another table located about 4 meters from the first. The BCI user during the remote control could see through HMD in Rome the video streamed images from the Japan robot's first person perspective. Further, we allowed BCI user to observe the robot's body by placing a mirror close to the second table.

We manipulated the audio-visual synchronization and the type of sound participants listened through earphones. Thus, the experimental design consisted in 4 conditions with sound Onset (Synchronous, Asynchronous) and Sound (Footsteps, Beep) as within-subjects Factors.

As dependent variables we collected:

1. the total time to complete the task;

2. the time to steer the robot from the first table to the second one;
3. the distance between the target position and the actual position of the bottle at the end of the task.

The experiment consisted of 5 sessions namely:

1. one SSVEP training phase;
2. a trial to learn the task;
3. a baseline phase where participant completed the task and no auditory feedback was provided;
4. the 4 experimental sessions;
5. a final baseline phase.

The experimental conditions involved in step 4 were shuffled among subjects. And, at the end of each experimental condition, participants answered questions concerning his/her experience interacting with the robot as seen in Table 4.3.

4.2.4 Data Analysis: Current Results

All performance data were baseline corrected (ratio between actual value and the mean of the first and second Baseline). Patients' performance was compared with the control sample.

4.2.4.1 Total time, walking time and placing accuracy

We now introduce the definition of the different metrics that we report here.

Total time

The time from beginning of the experiment to the end of the experiment.

Walking time

The time from the moment the user gain control of the robot's steering until the user stops the robot in front of the table.

Placing accuracy

The distance between the final object's center position and the target on the table.

Healthy Participants (HP) - Total time

We observed no differences between conditions [$F_{1, 13} < 2.02$; $P > 0.19$].

HP - Walking time

We observed no difference between conditions [$F_{1, 13} < 3.36$; $P > 0.09$].

HP - Placing accuracy

We observe a better accuracy, see Figure 4.6, during the Foot sound ($mean = 1.481$) relative to the Beep sound ($mean = 1.975$) [$F_{1, 13} = 6.71$; $P = 0.022$].

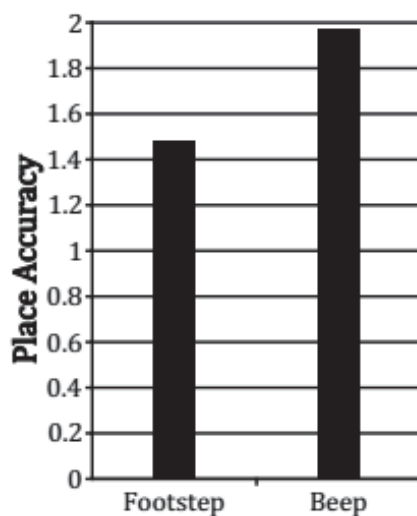


Figure 4.6: A better placement accuracy (lower values) is obtained with Footstep sound feedback during the control of the humanoid robot.

Patient1 (P1)

Patient1 was able to conclude only the conditions with the biological sound. We found that during the Foot Asynchronous condition Patient1 took more time during the walking phase relative to controls ($p = 0.043$).

No other comparisons (Total Time, Place Accuracy) with the control sample were close to significance (all $t < 1.70$, $p > 0.112$).

Patient2 (P2)

We found a tendency to have longer time to complete the task in the Foot- ($t = 1.15$, $p = 0.054$) and Beep-Asynchronous conditions ($t = 1.86$, $p = 0.085$).

No other comparisons (Total Time, Place Accuracy) with the control sample were close to significance (all $t < 1.48$, $p > 0.165$).

4.2.4.2 Subjective Measures

Table 4.3 describes the questions participants answered after each experimental condition. Participants were asked to answer on a 0 – 100 scale (where 0 indicates “nothing” and 100 indicates “very much”; see also Table 4.4 for Patients’ raw data).

Questions aimed at investigating	Item
Place illusion	
1	I had the sensation to be where the robot was.

2	There were moments that the images I saw were the reality.
Embodiment	
1	It was as if the robot were my body.
Embodiment Control	
1	It was as if I had two bodies (here and there).
Agency	
1	The robot executed my commands.
2	I felt in control of robot's / avatar's actions.
BCI Interface	
1	I could pay attention to the environment displayed.
2	It was easy commanding through the BCI interface.
3	It was hard looking at the flashing stimuli.
Illusory Movement	
1	It was like walking (as if my legs were actually moving).
2	I had the feeling of swaying along with the robot.

Table 4.3: List of the questions participants answered after each experimental condition.

We found no difference between the experimental conditions ($F_{1,13} < 3.82$, $p > 0.072$).

Overall participants perceived low level of Place Illusion ($meanPI_{1,2} = 35.67 \pm 3.80$), low level of Embodiment ($mean = 25.44 \pm 3.42$) and Illusionary Movement ($meanIM_{1,2} = 28.25 \pm 3.12$), high level of Agency ($meanA_{1,2} = 74.96 \pm 2.29$).

Importantly subjects were able to pay attention to the environment ($mean = 64.95 \pm 2.91$), reported that commanding the robot was relatively easy ($mean = 67.83 \pm 3.13$) and that looking at the flashing arrows was not hard ($mean = 34.12 \pm 4.10$).

The same trend was observed for Patient1 and Patient2 as seen in Table 4.4.

Patient1

Patient2

Place Illusion	62.50 ± 7.50	28.75 ± 10.48
Embodiment	60.00 ± 0.00	12.50 ± 2.50
Embodiment Control	60.00 ± 0.00	20.00 ± 10.00
Agency	70.00 ± 0.00	68.75 ± 3.15
BCI Interface	53.33 ± 3.33	58.33 ± 2.89
Illusory Movement	52.50 ± 2.50	13.75 ± 2.39

Numbers represent values comprised between 0 and 100 (mean \pm s.e.m.)

Table 4.4: Patients' answers to questions.

4.2.5 Discussion and Perspectives

Data suggest that SSVEP-BCI does not limit the possibility to observe the environment and that the designed interface allows participants to easily control robot's movements. Further, the SSVEP reliability allows participants to experience high levels agency on the robot's actions. The low level of place illusion and embodiment may be driven by the absence of sensorimotor congruency [107] for head movements as the robot's head remained fixed and subjects were asked to remain still with the head.

Importantly, comparison of patients' answers may suggest the higher the spinal lesion the lower is the perceived place illusion and embodiment. Moreover data from healthy subjects suggest the role of human-related sounds to control a robot. Moreover the observed interference effect of the asynchronous footstep sound condition in SCI people may suggest a reduced ability of patients to manage multi-sensory conflicts when related to lower body parts.

While no notable effect is seen on the entirety of the subjects, we are starting to consider sub-groups of subjects in order to observe different effect based on the individual levels of performance. Indeed, we have seen during our experiments that some subjects were performing the task "well" and some were performing it "badly" based on how well the subject was able to steer the robot to reach its destination.

An example of one "bad" and one "good" subjects is shown in Figure 4.7. In those figures:

- the dotted line links the start at position $(x, y) = (0, 0)$ and the arrival position;
- a single blue point and a red line represent the robot and its facing orientation;
- the transparency of the red line indicates time progression: it starts almost transparent at $t = 0$ and ends opaque at the arrival;
- to build this figure, a point was recorded every time the speed input "dropped", i.e. each time the BCI selection dropped to zero;

- the “ $n = N$ ” indicator in the top-right corner of the image is the number of recorded points in the figure, due to the scale of the picture and short drops, this number is higher than the number of points that can be actually seen.

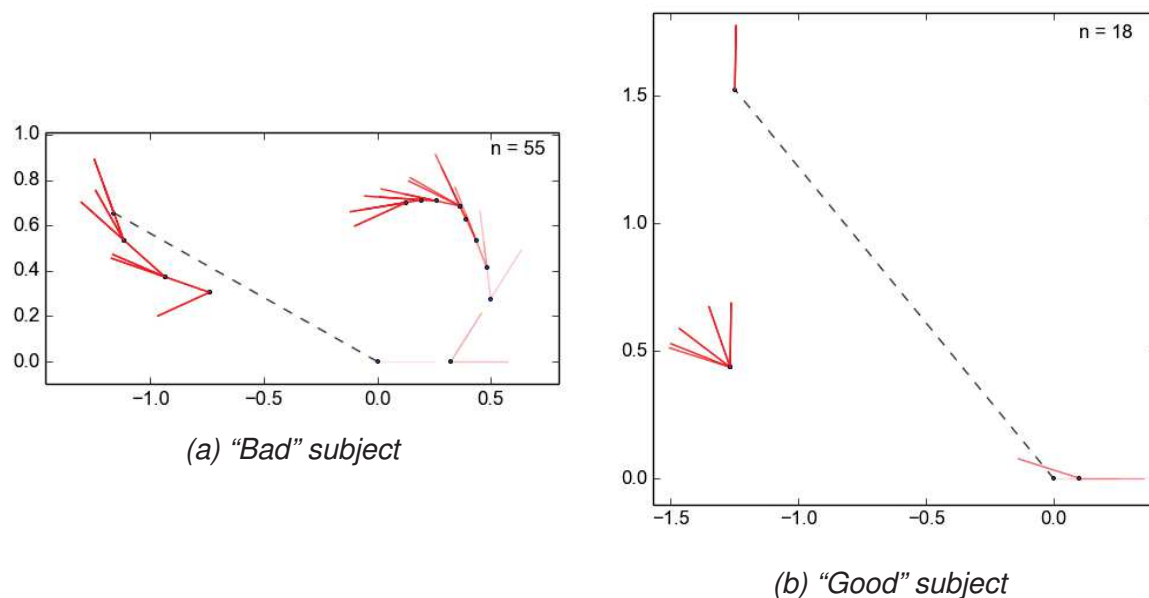


Figure 4.7: Comparison of trajectory for two different subjects with very different performance levels.

A “bad” subject, such as the one seen in Figure 4.7a, is characterized by a large number of points that indicate a difficulty to control the robot during the steering. In the example we show, the subject first moves forward, i.e. s/he goes toward the table where the object s/he just grasped was, then turn left and moves forward. However, s/he is at this point too close to the barrier that we setup so s/he need to turn around and get the robot in a position that allows it to cross the gate. Once s/he has crossed this gate and s/he can see the table, s/he stops the robot.

On the other hand, the “good” subject we show also wrongly walk a bit forward in the beginning but s/he is then able to steer the robot left, move it toward a position where s/he can cross the gate, cross the gate and stop near the table and finally adjust the robot’s orientation and stop it.

At the moment, we are still figuring an objective criterion that reflects this control quality. We expect one of the following effects:

1. The sounds have no effect on “good” subjects. For “bad subjects”, it has a similar effect to what we demonstrated in our previous study, i.e. synchronized footstep sounds help the subject to control the robot compared to the other conditions;
2. Similar effects than the previous but the “good” and “bad” groups are reversed.

Finally, we plan to perform more experiments with SCI patients in order to confirm, or infirm, the initial observations we have made.

4.3 Conclusion

In this chapter we have shown that, thanks to the framework introduced in Chapter 3 and to the low-level BCI work shown in Chapter 2, we can implement rich paradigms that allow to test embodiment parameters involving multiple sites. Indeed, in those cases, the robot was operated in Japan but controlled from Italy.

Overall both studies we have introduced in this chapter suggest the importance of an embodied-BCI perspective where sensorimotor-related information may help BCI users to control an external devices. However, we have to pursue the analysis of the second experiment data in order to understand the impact of the sound feedback when immersive visual feedback strategies are involved.

Furthermore, the same framework can be applied to study other embodiment parameters. In particular, we are planning to observe the effect of tactile feedback in SCI patients. For healthy patients, it has already been shown that stimulating the tendons through a vibro-stimulator can lead to illusory movements [108]. However, for SCI patients who lack sensations below the neck, this is not a practical solution. Yet, a similar observation was made in a tetraplegic patient when stimulating his face [109]. A similar stimulation for walking could be engineered but it is unknown whether this would be beneficial or not.

Finally, the fairly important number of people involved in these studies, $n = 23(9 + 14)$, and the generally positive reception of the demonstrators involved show that robots can be practically controlled using EEG-based BCI. In the next chapter, we steer away from EEG-based BCI and present some preliminary results obtained in controlling a humanoid robot using an ECoG-based BCI.

ECoG-based BCI using a humanoid robot

The previous chapter closes our contribution in SSVEP object-centric BMI and its application to humanoid robot for the purpose of embodiment. We discussed thoroughly the limitations of the EEG technology and the difficulty to bypass the poor signal-to-noise ratio. This chapter presents the (still ongoing) work that has been started regarding the use of a humanoid robot in ECoG-based BCI. This was made possible because we had the opportunity to pair with a hospital at Hokkaido and access to patients on which ECoG is implanted as part of an investigation of potential brain disorder. We first give a more descriptive explanation than what we introduced in Chapter 1 concerning the ECoG technology/ We also present the method that is most commonly used to exploit the data obtained from the brain using this ECoG. We then discuss the few experiments that we performed with two patients: a one degree of freedom neurofeedback experiment and a two degrees of freedom neurofeedback experiment. These two latter experiments were conducted in collaboration with g.tec and Dr. Kamada from Asahikawa University Hospital in Hokkaido, Japan. Finally, we discuss future works in this field as well as some of the issues and limitations of such experiments. Please note that having such an opportunity is rather rare and we had to wait for the timing and the consent and the ethical procedure to establish. We had only two short windows in time where it is possible to conduct such kind of experiment. We cannot choose this window opportunity and also we cannot “reuse” the patients. The latter has to be ready and in the mood on the D day otherwise the opportunity is lost. It is really difficult to organize such experimentations.

5.1 An introduction to ECoG-based BCI

In this section, we discuss the ECoG technology and a commonly used method to perform BCI based on this technique.

5.1.1 ECoG technology

The EEG technology has many advantages: it is cheap, widely available and researched, easy to setup and provides a very high temporal resolution. However, it has two important drawbacks: (i) a very low spatial resolution, and (ii) a very low signal-to-noise ratio. These drawbacks are detrimental to most BCI applications that require a good spatial resolution such as motor-imagery paradigms.

To overcome these limitations one can use when possible the more invasive ECoG technique. Contrary to non-invasive EEG that can only provide information in the range of 0–40Hz; whereas ECoG can provide information up to 500Hz [110]. This broader bandwidth allows to retrieve the movement related–and motor-imagery related–increases in bandpower that occurs above 40Hz, that is the high-gamma activation (HGA) [111]. Furthermore, the HGA is strongly focused at specific cortical regions related to the executed–or imagined–task.

5.1.2 ECoG-based BCI: the CSP method

The method we described here is reported in [112], g.tec since then improved it to perform the classification online by reducing the dimensions of the features space and by automating the classifier selection. This allows trials to be chained effectively with improved classifier's and user's performances. We report here the information relative to the common spatial pattern (CSP) method in ECoG-based BCI for hand poses classification.

5.1.2.1 Data Acquisition

The ECoG data is recorded at bedside using a g.HIamp biosignal amplifier (g.tec medical engineering GmbH, Schiedlberg, Austria) with a sampling rate of 1200Hz and its built-in band-pass filter set to the range 0.5–500Hz.

5.1.2.2 Pre-processing

The data acquired from the subject is band-pass filtered from 100 to 500Hz (high gamma range) with additional notch filters at all harmonics h_i of the 50Hz power line frequency. All filters were designed as Butterworth filter of order 5 with a notch width of $h_i \pm 5$ Hz.

5.1.2.3 CSP Feature Extraction and Classification

Common Spatial Patterns (CSPs) are a standard method for EEG data to extract optimal discriminant features in movement (or movement imagination) tasks. In the case of hand

poses classifications, a two-stage process is employed where, in a first step, the movement state is told apart from the idle state and in a second step the three different hand poses are decoded in a “one versus all” comparison [113].

To prevent influence of visual or auditory stimuli onto the classification, the CSPs are run exclusively on the electrode grids covering the motor cortex. Error rate based search algorithms allow to select optimal parameters regarding the CSP window size.

The CSP weight matrix calculated with the optimal window size is then used to spatially filter the ECoG signal, and the four most discriminant feature channels are selected per decision pair (two largest eigenvalues from each side of the spectrum). Then the signal variances are computed and the resulting channels are normalized and have their logarithm taken for numeric stability.

Based on those features a two-class linear discriminant analysis (LDA) for the case “movement” versus “idle”, or a multi-class LDA (MLDA) for the hand pose description is calculated using 10×10 fold cross-validation [114].

5.2 One degree of freedom ECoG-based BCI

The first ECoG-based BCI experiment we performed allowed a patient (female) implanted with an ECoG array over the left-hand motor area to control a robotic arm by using thoughts alone. The implementation location and the CSP map for hand motor imagination are shown in Figure 5.1. This experiment is illustrated in Figure 5.2 and is described thereafter.

Please note that for practical reasons, the number and duration of these experiments were limited. The number of possible experiments is limited by the therapeutic process within which the patient is involved. Furthermore, in the case of this patient, she had suffered some postoperative complications. Thus, we first had to wait for her to fully recover from the surgery, and then, following the doctors’ direction, we were able to work with her only one hour per day—including the time required to set-up the recording—over a period of three days. These practical issues are further discussed in Section 5.4.

The patient is placed in front of a computer screen where the video from the robot’s head-mounted camera is shown. The robot is placed near a cupboard where an object is standing. The robot has its hand raised, ready to grasp the object. A cue is shown to the user. The cue has two states as seen in Figure 5.2:

Rest

A red hexagon with the Japanese character for rest (left side).

Move

A green circle with the Japanese character for move (right side).

The BCI system is able to detect these two state: *rest* and *move*. For the patient, the mental strategy for *rest* is to keep a relaxed state in her mind while the *move* state corresponds to the imagination of a grasping movement with the left hand.

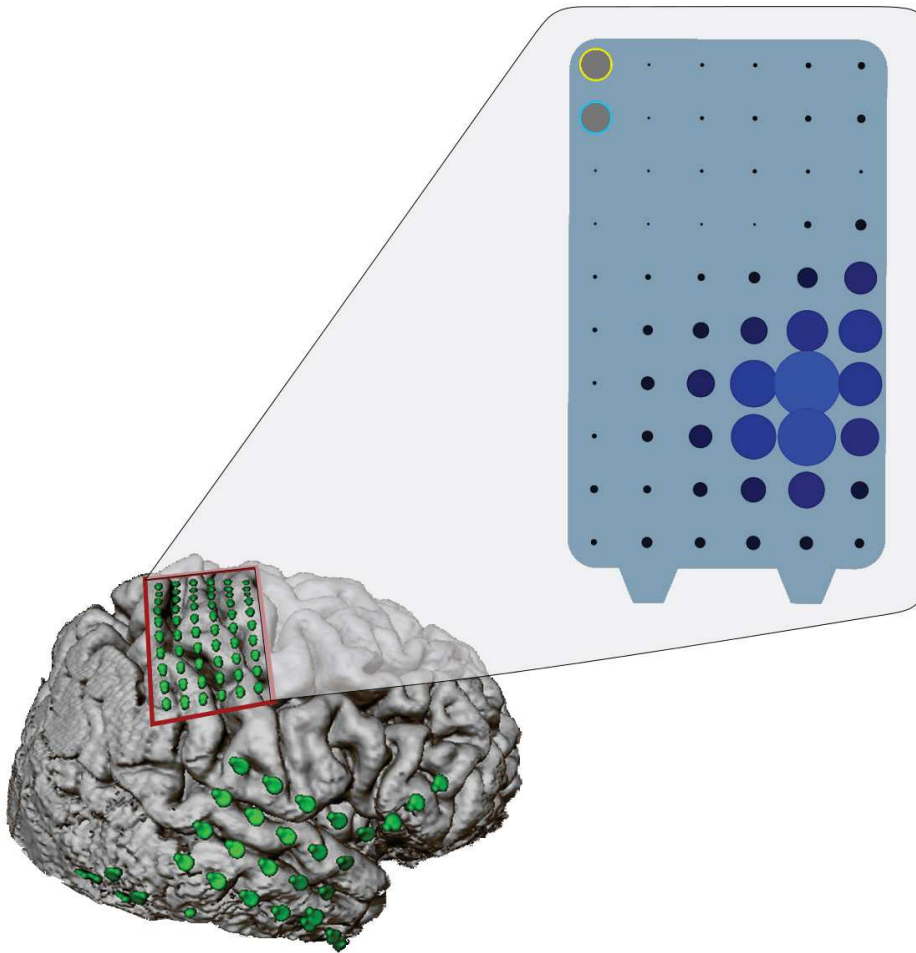


Figure 5.1: ECoG grids implemented on the right-hemisphere of the patient's brain and CSP map for hand motor imagination in Patient 1. The main grid used in this experiment, framed in red here, is placed over the left-hand motor area and is composed of 60 electrodes.

When the cue is set to *rest*, the patient should stay relaxed. When it is set to *move*, the patient should imagine a grasping action with the left hand. The robot transits between the following states:

1. Rest
2. Move the hand towards the object
3. Close the gripper to grasp the object
4. Lift the object
5. Maintain the object in the air

Once the BCI has detected the *move* intention for 500 ms, the robot transits from its current state to the next state or stays in the final state. The execution of each state takes roughly 500 ms, therefore, if the patient is able to maintain the *move* state for 2 seconds, the robot executes a smooth “grasp and lift” movement. If the BCI detects a *relax* state then the

current movement is immediately canceled—the robot moves backward and transits from the current state to the previous state—then transits from its current state to the previous state as long as the *relax* state is maintained.

The patient was able to perform such control over 4 neurofeedback sessions in total: 3 of those were performed consecutively over one day and the last one was performed two days afterwards. After each session, the distance between the features associated to each mental state increased and thus the recognition performance improved.

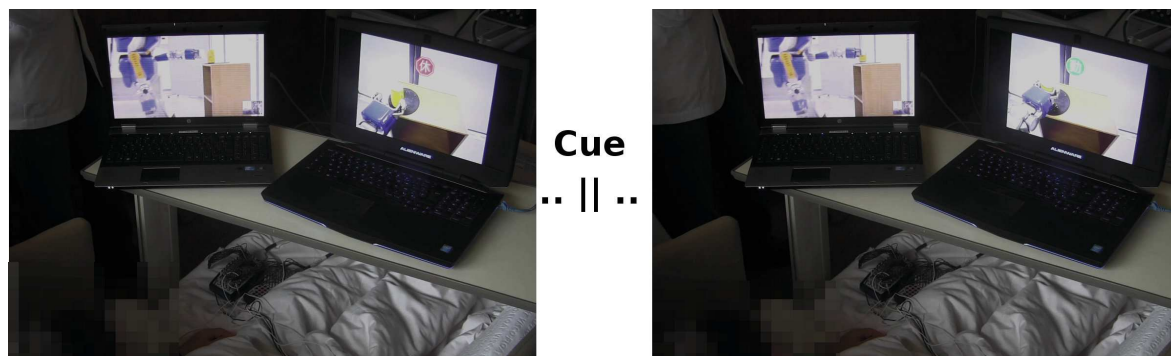


Figure 5.2: ECoG patient during the neurofeedback experiment we conducted. On the left side, the patient is in relax state. As soon as the cue comes, she imagines hand grasping movement and the robot starts moving as seen on the right side.

This experiment showed that ECoG-based motor imagery performed very well despite a very short training period compared to similar paradigms in EEG-based BCI: a few sessions were enough to reach a good level of performance (> 80%) whereas EEG-based MI requires a long period of training before reaching such performances [16]. The patient was also able to maintain a motor imagery state for extended period of times. Finally, the use of the humanoid robot in the neurofeedback was beneficial to the training of the BCI.

5.3 Two degrees of freedom ECoG-based BCI

The second patient's (also female) implantation was much larger than the first one. We hoped that this would allow us to perform multiple kind of motor imagery simultaneously with a short training period and a high degree of accuracy compared to EEG-based BCI [16]. Indeed, while multi-class motor imagery is possible in EEG, it is usually performing at a low-level of performance—around 60%—and only after lengthy periods of training. Furthermore, the higher accuracy of ECoG recordings should allow us to perform multi-class simultaneous detection. Therefore, we proposed a synchronized motor imagery paradigm that illustrates these capabilities. We will see in this section that the patient was ultimately unsuccessful in the realization of this task but the experience was nonetheless insightful with regards to the ECoG-based BCI.

5.3.1 Patient implantation

The second patient was implanted with a large number of ECoG electrodes, as seen in Figure 5.3, we initially aimed to understand the following intentions:

- Left-hand motor imagery
- Right-hand motor imagery
- Tongue motor imagery

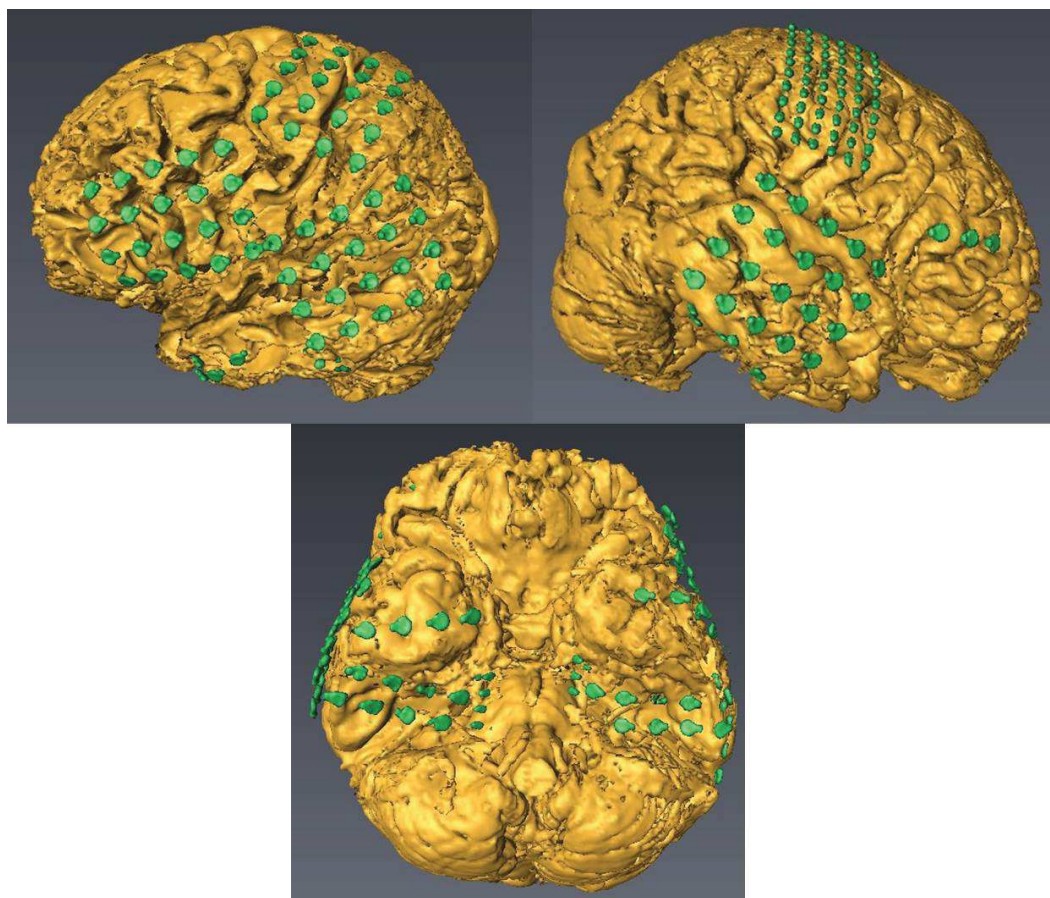


Figure 5.3: ECoG grids implemented on the left hemisphere, right hemisphere and sub-parietal region—respectively top left, top right and bottom part of the figure.

5.3.2 Results of Brain Mapping

All figures in this section reports the activation maps of the brain during performance of the **real** reported movement when compared to a resting condition, referred to as “Rest” later.

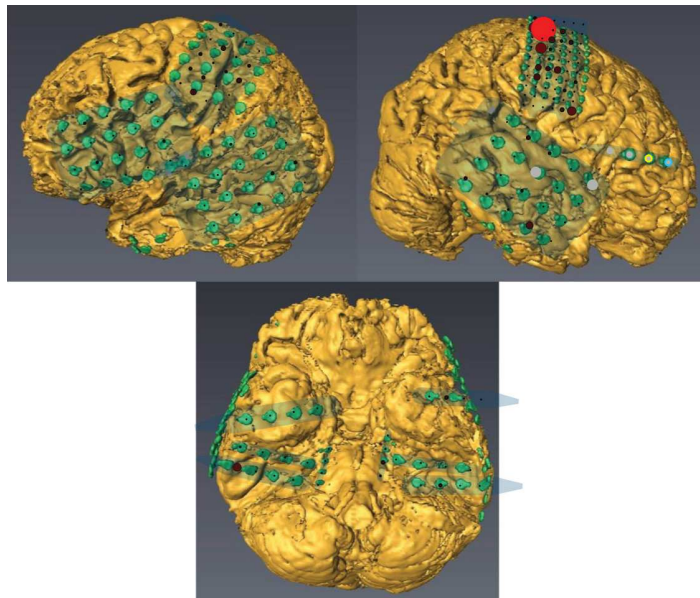


Figure 5.4: Left-hand versus Rest activation patterns. Most of the activity clearly originates from the motor cortex in the right hemisphere.

In Figure 5.4, we report the activation of left-hand movement versus “Rest”. As expected, most of the activity is retrieved in the right motor cortex area. Motor imagery of this movement follows a similar pattern yet activation is weaker.

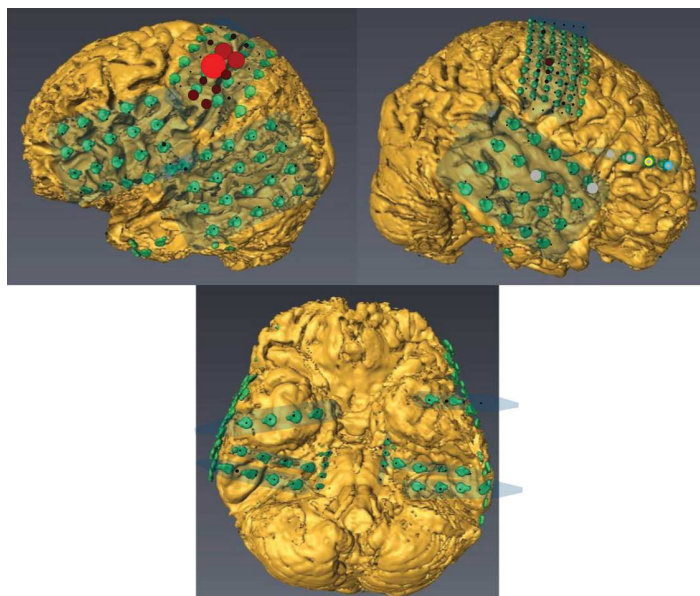


Figure 5.5: Right-hand versus Rest activation patterns. Most of the activity clearly originates from the motor cortex in the left hemisphere.

In Figure 5.5, we report the activation of right-hand movement versus rest. As expected, most of the activity is retrieved in the left motor cortex area. Motor imagery of this movement follows a similar pattern yet activation is weaker.

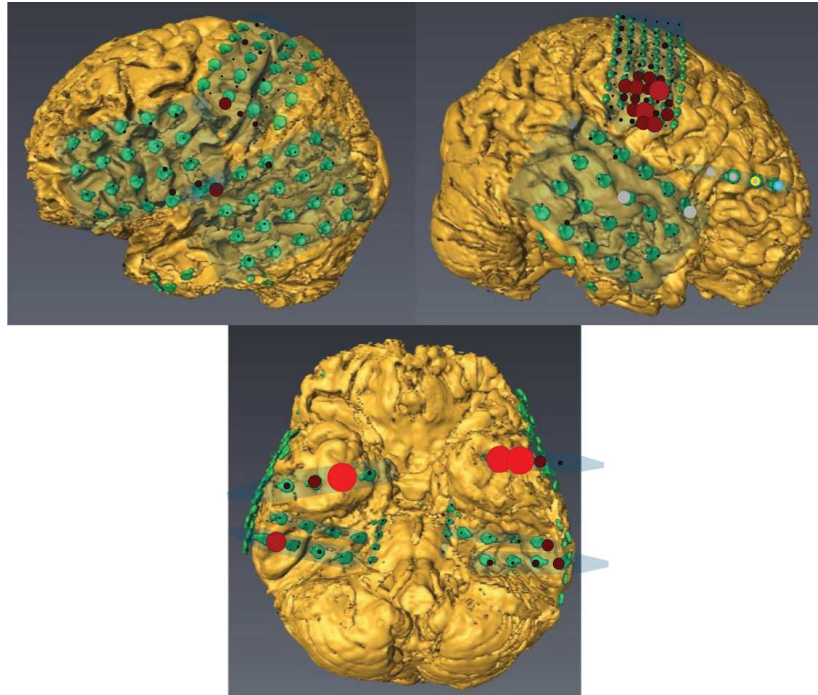


Figure 5.6: Tongue versus Rest activation patterns. The activity is mostly located in the right motor cortex area and the sub-parietal areas.

In Figure 5.6, we report the activation of tongue movement versus rest. Most of the activity is retrieved in the right motor cortex area and the sub-parietal areas. Motor imagery of tongue movement is notably difficult to perform hence activation in that case is less noticeable.

5.3.3 A synchronized motor-imagery paradigm

The paradigm we present here investigates the possibility to use the features intensities as an input to the system rather than the classification output. It also tries to exploit coordinated motor imagery. Hence, we propose to let the patient perform a two hands coordination-oriented task. We now discuss this task while the following section details the schedule we proposed to achieve this task.

The task to perform is illustrated in Figure 5.7: it is an assembly task. The goal of the robot is to insert a peg, held by one hand, into a hole, held by the opposite hand. The full task is composed of three sub-tasks:

1. The robot grasps each object **independently** and hold them at a random and different height (but otherwise properly aligned).
2. The robot aligns the object **coordinately**.
3. The robot assembles the object (i.e. move them towards each other).

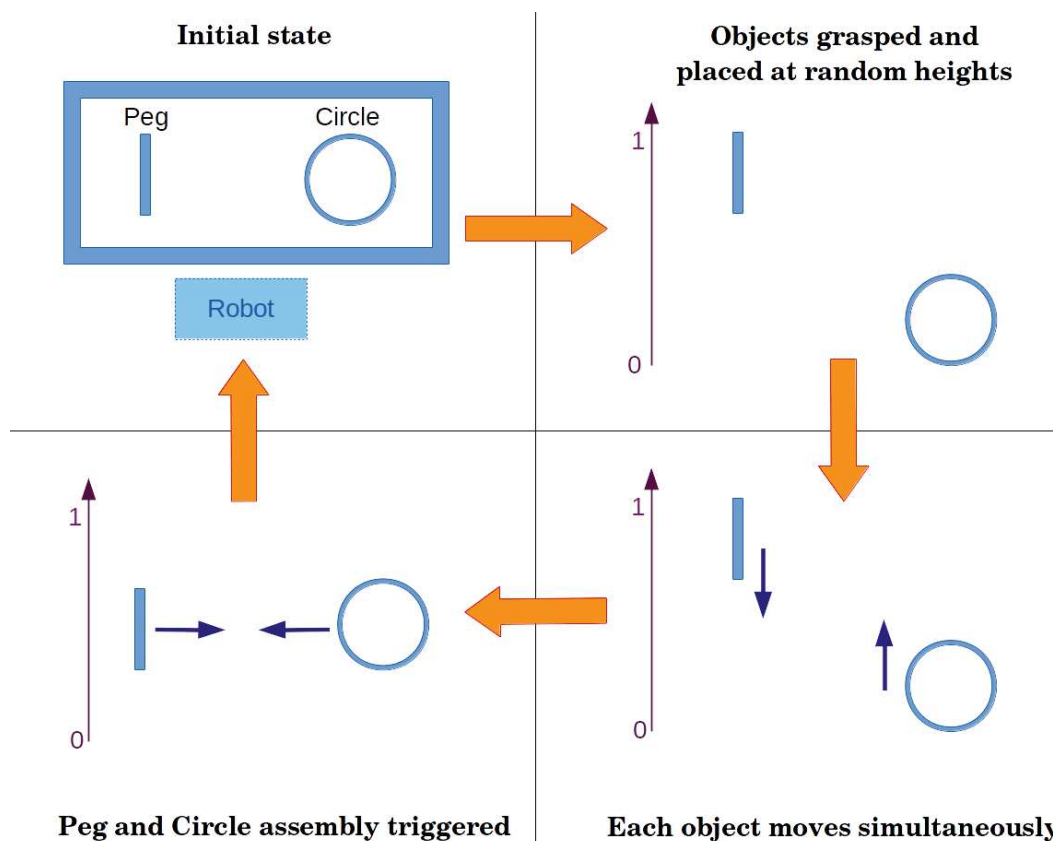


Figure 5.7: An illustration of the assembly task we propose to realize through ECoG-based motor imagery.

For each sub-task we propose the following mental strategies of control:

1. Left-hand motor imagery triggers the grasping and lifting of the peg, right-hand motor imagery triggers the grasping and lifting of the circle (similarly to the first experiment).
2. Simultaneous, and **only** simultaneous, activation of left- and right-hand motor imagery move the hand in opposite direction. Initially, simultaneous activation triggers a symmetric movement of both hands. If the performance of the patient allow it, we could increase the independence of each hand control through the use of the features' values.
3. Tongue motor-imagery triggers the assembling, this step is optional and assembly can be triggered autonomously once the objects are properly aligned.

5.3.4 Experiments

We initially planned to follow the following schedule:

1. Training on one degree of freedom motor-imagery paradigms using the robot's arms as feedback for left-hand and right-hand in a fashion similar to the first experiment we presented earlier;
2. Training on two degrees of freedom motor-imagery with left-hand and right-hand motor imagery, feedback is provided using the robot's arms;
3. Training on two degrees of freedom motor-imagery with left-hand and right-hand motor imagery as well as simultaneous activation of the two classes, feedback is provided using the robot's arms;
4. Free session on the assembly task described earlier where the assembly is automatically triggered when the objects are properly aligned, the subject's goal is to perform the assembly task as many times as possible over the trial duration;
5. Free session on the assembly task described earlier where the assembly task is triggered by tongue imagination, the subject's goal is the same as in the previous step.

5.3.5 Results

The patient was able to perform the Step 1 well after the first session with a high degree of accuracy as in the case of the first patient. We were very confident at this point that the patient would be able to tackle the following steps since she was very responsive and enthusiast with respect to the experiments with the robot.

Unfortunately, a setup issue during the training recordings for the Step 2 led to a mix-up between the different ECoG channels. Because of this issue, the system was unable to build a good enough classifier to discriminate between the three states—(i) rest, (ii) left-hand imagination and (iii) right-hand imagination—as the classifier would systematically favor either the left or the right hand class. As a result we could not meet the proposed schedule.

In fact, as in the previous experiment, we had, once again, limited time to work with the patient. In this case, the patient recovered very well from the surgery and we had the chance to work with her over four days during lengthier—about 2 hours—sessions. However, following the “failed” session, the patient seemed disheartened by the lack of results. In the following session, after noticing our mistake of the previous day, we repeated the experiments corresponding to the Step 2 in order to obtain a suitable two degrees of freedom motor-imagery classifier. However, the patient was not able to remain as focused as previously during these sessions. Thus, the classification accuracy was not very satisfying and the resulting feedback was jerky. This led to a more frustrating experiment and did not help to improve the performance of the classifier over the training sessions.

We have since envisioned different feedback conditions when performing this kind of experiments to lead to a smoother feedback. In the experiment we presented, the “rest” activity is favored over other movement activities since the user does not need to maintain the “rest” activity to trigger the return of the robot to a neutral position. We may adapt the paradigm to remove this favoritism. We would also like to consider using the features' value

to drive the robot rather than the outcome of the classifier. We hope to be able to test these new modalities with future patients.

5.4 Challenges of ECoG-based BCI

In this section we discuss some of the challenges posed by ECoG-based BCI.

5.4.1 Problems induced by the surgery

The first step of ECoG-based BCI is to implant the electrode grids within the patient's scalp. This obviously requires surgery however this surgery is *relatively* simple and safe to perform. It may be difficult for the surgeon to place the grids with a very high accuracy during the procedure but the localization of the electrodes is checked in a postoperative procedure. Thus we can obtain a ground-truth regarding this localization issue, though, in some cases, it may not be sufficient.

For example, the first patient had been implanted with a grid on the occipital lobe in order to record the signals from the visual cortex and possibly perform ECoG-based SSVEP experiments, similar to those we introduced in Chapter 3. Since the main grids were inserted over the motor cortex area, the surgeon had to slide the grid towards the occipital lobe, which is a difficult surgical act. As a result it was nearly impossible to record SSVEP activities for the patient and the insertion resulted in various complications.

Because of these complications, the first patient's cognitive abilities were reduced during the first week following the surgery. In particular, it was initially difficult for her to understand the instructions she was given and as a result the experiments did not progress as fast as they could have.

5.4.2 Experimenting within the therapeutic process

To implement a patient with ECoG electrodes, a surgical operation has to be performed. We have seen previously that this already induces multiple risks. However, it also means that the patient who is undergoing the operation is following a therapeutic process and this raises two additional issues.

5.4.2.1 A restrained time-frame

The patients we had the opportunity to work with suffer from severe cases of epilepsy that could not be properly managed by medication alone. In order to—hopefully—get rid of their seizures they need to undergo surgery [115] [116] to resect the parts of the brain that are triggering the seizures. Prior to this resection, the incriminated zones have to be identified which is done through ECoG monitoring.

The patient then keeps the ECoG grids implanted for about two weeks. However, during the first week, the patient is constantly monitored so that when a seizure occurs the surgeons are able to identify the responsible regions of the brain. It usually requires two to three

seizures in order to get an accurate location. Once this localization has happened, the patient is able to participate in BCI studies until the removal. Furthermore, ECoG BCI is quite demanding and thus recording sessions can only last about one hour per day.

Those factors result in a fairly short time-frame to conduct experiments which may limit the success we are able to obtain. In another case, unrelated to our work, a voluntary tetraplegic patient underwent surgery to get ECoG grids implanted for a duration of 30 days—the legal limit in Pennsylvania—and he was able to gain control of the 3D movements of a robotic arm [117].

5.4.2.2 Working in a hospital environment

Experimenting within the therapeutic process also means that all the experiments have to be done within the hospital environment. This brings up a number of technical issues that are difficult to take into account: network availability or connection quality, rooms that have high levels of electrical noise due to the presence of heavy machinery in nearby rooms or rooms that are electronically shielded. While these issues can be worked around this sometimes requires time, for example, to move to a different room. Yet, as mentioned above time is a precious resource in this context.

5.5 Perspectives and Conclusion

The ECoG experiment was very interesting in showing the possibilities of such devices. Unfortunately, the limited time allowed to perform experiments on human patients has reduced the extent of the success we could achieve. However, it also raises the question of the relevance of this type of interface given its inherent flaws. Is ECoG worth it?

The ECoG technology is impressive and, as of today, has provided the experience of “mind-control” that is the closest to what we might envision. However, its success has been limited to the 3D control of devices. Early works conducted with “truly invasive” devices, such as the BrainGate system, suggest that these devices surpass ECoG-based BCI for control applications. Furthermore, patients have had these devices for extended period of time, i.e. several years, whereas this is currently not possible with ECoG grids.

Furthermore, it remains unclear whether evoked potentials, such as P300 or SSVEP, can be retrieved from the ECoG signals. This would be a significant downgrade to the object-centric BCI control of humanoid robot framework that we introduced in Chapter 3 as the control of position and orientation of a robotic limb in a one-to-one way is not, in our opinion, a relevant way to control a robotic avatar. However, we could envision the use of such paradigms as an additional input within the framework to refine the robotic behaviors that are executed. Of course, the benefits of ECoG-based BCI for rehabilitation and the positive affect of a robotic device for feedback in this context should also be considered.

In conclusion, these initial experiments were quite successful. Although we are now only envisioning ECoG as a supplementary input or as a therapeutic tool, the experience we gained during these experiments may be later levered into following experiments with this kind of devices. We are still on-going the research with Dr. Kamada and we are in

expectation for additional patients to redo some experiments as we gained enough expertise to optimize their set-up.

Conclusion and Perspectives

This thesis has allowed us to explore the use of BCI to control a humanoid robot. Along this work we have been able to solve several problems specific to our context and to introduce novel concepts within the field.

We have shown several improvements that we brought to the BCI for robot control and to the use of SSVEP for control purpose in general. In particular:

- We have introduced a novel framework to design BCI applications with frequent paradigm shifts at the core;
- We have demonstrated the negative effects of dynamic visual feedback in SSVEP applications and proposed a simple solution to counteract this effect;
- We have determined appropriate time criterion and an effective way to display SSVEP stimuli on top of real-world object to allow selection with both a high degree of accuracy and a good user experience.

Based on those elements we proposed in the previous chapters, we demonstrated that brain signals can be used to command a sophisticated mobile device such as a humanoid robot to perform useful tasks like navigating to a desired location and interacting with different objects through grasping/pressing etc. We used computer vision to discover objects of interest in the environment, a dynamic immersive interface that communicates the corresponding actions to the user and a humanoid robot controlled by the user in an immersive environment using BCI. In principle, the robot could execute a wider range of actions instructed by the user in a multi-task scenario.

This problem was solved by the introduction of the object-centric BCI framework that allows the user to perform complex interactions with the environment without the need to gain low-level control of the robotic device which is an impractical solution to achieve with BCI. The power of this framework was illustrated through various demonstrators that illustrate its extendibility since each demonstrator systematically improved on the previous one by enhancing the capacities of one of the framework's modules. However, we wish to continue to extend the capabilities of this system to provide a more immersive experience.

Thanks to this framework, we can implement rich paradigms that allow to test embodiment parameters such as the effect of sound feedback that we studied in both healthy subjects and CSI-afflicted patients.

Finally, we were able to present preliminary results using ECoG, an invasive brain signals recording technique, to control a humanoid robot. Those experiments were very interesting in showing the possibilities of such devices. Unfortunately, the limited time allowed to perform experiments has reduced the extent of the success we can achieve. However, the experience gained from these experiments can be levered into the next ones and thus allow us to gradually improve over those initial success.

Perspectives

There is still a lot of work to achieve within this topic at different levels. We organize these perspectives around three axes: the technical improvements to pursue, the embodiment questions to study and finally some novel signals to involve.

Technical improvements

Those improvements were mostly discussed in Chapter 3, we recall the most prominent here:

Improving the field of view feedback

One of the most detriment limitation of the current system is the limited field of view proposed to the user. That is a consequence of two things: (i) the limited FoV of the HMD we use and (ii) the limited FoV of the camera. While there is not much that can be done regarding the first issue, we can virtually extend the field of view by integrating the actual image from the robot's sensors into a SLAM map that we are building online during the demonstration [83] [85].

Improving the task selection module capabilities

We have introduced an experiment that uses a Bayesian network to infer the affordable task associated to an object. This leads to a BMI that is convenient for the user as s/he only has to make a few choices to trigger complex behaviors. However, this can be significantly improved to (a) automatically generate affordable tasks and (b) allow more personalization within the realization of the task.

Closing the framework

At the moment, no satisfying error potential implementation that work concurrently with SSVEP has been made available by g.tec. Hence we relied on alternative ways to close the loop in the framework. However, the use of error potentials would lead to a much more immersive experience and open new ways for the user to interact with the controlled device.

Embodiment Question

We had the opportunity to study the effect of sound on embodiment and the robot's control performance. However, the framework we have developed and introduced throughout this

work is well suited to pursue other questions within this field. Some of these questions are currently addressed by my colleagues within the Joint-Robotic Laboratories and future experiments are also planned with our colleagues within the VERE project. However, given that these studies require a large amount of subjects and time, they can only be tackled successively.

In the near future, we would particularly like to study the effects of tactile feedback during robotic embodiment.

Addition of other signals

In this thesis we focused our effort on the use of the BCI technology to achieve the control of the robot. However, many other physiological signals can be acquired and processed in real-time to bring additional information to the system. Unfortunately we did not have the time to incorporate such signals within our studies.

In particular, physiological signals, such as the skin conductance or the heart-rate, could be used to measure the “emotions” of the user and affect the way the robot realizes its movements based on this information.

Finally, using an eye-tracking technology would be a logical step to improve the SSVEP performance as it would allow to significantly reduce the false-positive rate which has been the source of many problems through this thesis.

Bibliography

- [1] Jacques Vidal. Toward Direct Brain-Computer Communication. *Annual Review of Biophysics and Bioengineering*, 2:157–180, 1973.
- [2] Jonathan R. Wolpaw, Niels Birbaumer, Dennis J. McFarland, Gert Pfurtscheller, and Theresa M. Vaughan. Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, 113:767–791, 2002.
- [3] José del R. Millán, Rudieger Rupp, Gernot R. Müller-Putz, Roderick Murray-Smith, Claudio Giugliemma, Michael Tangermann, Carmen Vidaurre, Febo Cincotti, Andrea Kübler, Robert Leeb, Christa Neuper, Klaus-Robert Müller, and Donatella Mattia. Combining Brain-Computer Interfaces and Assistive Technologies: State-of-the-Art and Challenges. *Frontiers in Neuroprosthetics*, 4:161, 2010.
- [4] Yann Renard, Fabien Lotte, Guillaum Gibert, Marco Congedo, Emmanuel Maby, Vincent Delannoy, Olivier Bertrand, and Anatole Lécuyer. OpenViBE: An Open-Source Software Platform to Design, Test and Use Brain-Computer Interfaces in Real and Virtual Environments. *Presence Teleoperators & Virtual Environments*, 19(1):35–53, 2010.
- [5] Gerwin Schalk, Dennis J. McFarland, Thilo Hinterberger, Niels Birbaumer, and Jonathan R. Wolpaw. BCI2000: A General-Purpose Brain-Computer Interface (BCI) System. *IEEE Transactions on Biomedical Engineering*, 51:1034, 2004.
- [6] Gernot R. Müller-Putz, Christian Breitwieser, Febo Cincotti, Robert Leeb, Martijn Schreuder, Francesco Leotta, Michele Tavella, Luigi Bianchi, Alex Kreiling, Andrew Ramsay, Martin Rohm, Max Sagebaum, Luca Tonin, Christa Neuper, and José del. R. Millán. Tools for brain-computer interaction: a general concept for a hybrid BCI. *Frontiers in Neuroinformatics*, 5, 2011.
- [7] E. Asano, C. Juhász, A. Shah, O. Muzik, DC. Chugani, J. Shah, S. Sood, and H. Chugani. Origin and propagation of epileptic spasms delineated on electrocorticography. *Epilepsia*, 46:1086–1097, July 2005.

- [8] Jörn Vogel, Sami Haddadin, John D. Simeral, Sergej D. Stavisky, Dirk Bacher, Leigh R. Hochberg, John P. Donoghue, and Patrick van der Smagt. Continuous Control of the DLR Light-weight Robot III by a human with tetraplegia using the BrainGate2 Neural Interface System. In *International Symposium on Experimental Robotics (ISER2010)*, 2010.
- [9] BrainGate - project website. <http://braingate2.org/>.
- [10] Leigh R Hochberg, Daniel Bacher, Beata Jarosiewicz, Nicolas Y. Masse, John D Simeral, Joern Vogel, Sami Haddadin, Jie Liu, Sydney S. Cash, Patrick van der Smagt, and John P. Donoghue. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485:372–375, 2012.
- [11] Baolei Xu, Yunfa Fu, Lei Miao, Zhidong Wang, and Hongyi Li. Classification of fNIRS Data Using Wavelets and Support Vector Machine during Speed and Force imagination. *Proceedings of the 2011 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1224–1229, 2011.
- [12] S. A. Huettel, A. W. Song, and G. McCarthy. *Functional Magnetic Resonance Imaging (2 ed.)*. Massachusetts: Sinauer, 2009.
- [13] Honda announce regarding fmri-based bci for robot control. http://techon.nikkeibp.co.jp/english/NEWS_EN/20060525/117493/.
- [14] Honda press release regarding eeg/nirs-based bci for robot control. <http://www.honda-ri.com/news033109.php>.
- [15] G. Pfurtscheller and A. Aranibar. Event-related cortical desynchronization detected by power measurements of scalp EEG. *Clinical Neurophysiology*, pages 817–826, 1977.
- [16] G. Pfurtscheller, C. Brunner, A. Schlögl, and F.H. Lopes da Silva. Mu rhythm (de)synchronization and EEG single-trial classification of different motor imagery tasks. *NeuroImage*, 31(1):153–159, May 2006.
- [17] S. Sutton, M. Braren, J. Zubin, and E. R. John. Evoked-Potential Correlates of Stimulus Uncertainty. *Science*, pages 1187–1188, 1965.
- [18] Emanuel Donchin, Kevin M. Spencer, and Ranjith Wijesinghe. The Mental Prosthesis: Assessing the Speed of a P300-Based Brain-Computer Interface. *IEEE Transactions on Rehabilitation Engineering*, 8:174–179, 2000.
- [19] Christoph Guger, Shahab Daban, Eric Sellers, Clemens Holzner, Gunther Krausz, Roberta Carabalon, Furio Gramatic, and Guenter Edlinger. How many people are able to control a P300-based brain–computer interface (BCI)? *Neuroscience Letters*, 462:94–98, 2009.

- [20] Piyush Goel and David Brown. Time-invariant Neural Nets for a Brain-Computer Interface using EEG signal. <http://www.port.ac.uk/research/iir/research/currentresearch/dataanalysis/braincomputerinterface/>, 2006.
- [21] Brigitte Röder, Frank Rösler, Erwin Hennighausen, and Fritz Näcker. Event-related potentials during auditory and somatosensory discrimination in sighted and blind human subjects. *Cognitive Brain Research*, 4:77–93, 1996.
- [22] Fabio Aloisea, I. Lasorsaa, F. Schettini, AM Brouwer, D. Mattiaa, F. Babilonia, S. Salinaric, M.G. Marciania, and F. Cincottia. Multimodal stimulation for a P300-based BCI. *International Journal of Bioelectromagnetism*, 9:128, 2007.
- [23] An Luo and Thomas J Sullivan. A user-friendly SSVEP-based brain–computer interface using a time-domain classifier. *Journal of Neural Engineering*, 7:1–10, 2010.
- [24] Danhua Zhu, Jordi Bieger, Gary Garcia Molina, and Ronald M. Aarts. A Survey of Stimulation Methods Used in SSVEP-Based BCIs. *Computational Intelligence and Neuroscience*, 2010:12, 2010.
- [25] D. Regan. An effect of stimulus colour on average steady-state potentials evoked in man. *Nature*, 210:1056–1057, 1966.
- [26] Hovagim Bakardjian. *Optimization of Steady-State Visual Responses for Robust Brain-Computer Interfaces*. PhD thesis, Department of Electronic and Information Engineering, Tokyo University of Agriculture and Technology, 2010.
- [27] M. Byczuk, P. Poryzala, and A. Materka. *Human – Computer Systems Interaction: Backgrounds and Applications 2*, chapter SSVEP-Based Brain-Computer Interface: On the Effect of Stimulus Parameters on VEPs Spectral Characteristics, pages 3–14. Springer Berlin Heidelberg, 2012.
- [28] Robert S. Fisher, Graham Harding, Giuseppe Erba, Gregory L. Barkley, and Arnold Wilkins. Photic- and pattern-induced seizures: a Review for the Epilepsy Foundation of America Working Group. *Epilepsia*, 46:1426–1441, 2005.
- [29] Christoph S. Herrmann. Human EEG responses to 1–100 Hz flicker: resonance phenomena in visual cortex and their potential correlation to cognitive phenomena. *Experimental Brain Research*, 137(3-4):346–353, 2001.
- [30] Ola Friman, Ivan Volosyak, and Axel Gräser. Multiple Channel Detection of Steady-State Visual Evoked Potentials for Brain-Computer Interfaces. *IEEE Transactions on Biomedical Engineering*, 54(4):742–750, April 2007.
- [31] Robert Prueckl and Christoph Guger. A brain-computer interface based on steady state visual evoked potentials for controlling a robot. *Lecture Notes in Computer Science*, 5517:690–697, 2009.

- [32] François-Benoît Vialatte, Monique Maurice, Justin Dauwels, and Andrzej Cichockia. Steady-state visually evoked potentials: Focus on essential paradigms and future perspectives. *Progress in Neurobiology*, 90:418–438, 2010.
- [33] Yijun Wang, Yu-Te Wang, and Tzyy-Ping Jung. Visual stimulus design for high-rate SSVEP BCI. *Electronic Letters*, 46:1057–1058, 2010.
- [34] P. W. Ferrez and J. del R. Millán. Error-Related EEG Potentials Generated During Simulated Brain–Computer Interaction. *IEEE Transactions on Biomedical Engineering*, 55:923–929, 2008.
- [35] P. W. Ferrez and J. del R. Millán. Simultaneous real-time detection of motor imagery and error-related potentials for improved BCI accuracy. In *Proceedings of the 4th International Brain-Computer Interface Workshop & Training Course*, 2008.
- [36] Ricardo Chavarriaga and José del R. Millán. Learning From EEG Error-Related Potentials in Noninvasive Brain-Computer Interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(4):381–388, August 2010.
- [37] L.A. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*, 70:510–523, 1988.
- [38] T.M. Vaughan, D.J. McFarland, G. Schalk, W.A. Sarnacki, D.J. Krusienski, E.W. Sellers, and J.R. Wolpaw. The wadsworth BCI research and development program: at home with BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14:229–233, 2006.
- [39] B. Blankertz, M. Krauledat, G. Dornhege, J. Williamson, R. Murray-Smith, and K.-R. Müller. A Note on Brain Actuated Spelling with the berlin Brain-Computer Interface. *Universal Access in Human-Computer Interaction. Ambient Interaction*, pages 759–768, 2007.
- [40] Anatole Lécuyer, Fabien Lotte, Richard B. Reilly, Robert Leeb, Michitaka Hirose, and Mel Slater. Brain-computer interfaces, virtual reality, and videogames. *IEEE Computer*, 41:66–72, 2008.
- [41] Robert Leeb, Felix Lee, Claudia Keinrath, Reinhold Scherer, Horst Bischof, and Gert Pfurtscheller. Brain-Computer Communication: Motivation, aim and impact of exploring a virtual apartment. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15:473–482, 2007.
- [42] Jozef Legény, Raquel Viciano Abad, and Anatole Lécuyer. Navigating in Virtual Worlds Using a Self-Paced SSVEP-Based Brain–Computer Interface with Integrated Stimulation and Real-Time Feedback. *Presence: Teleoperators and Virtual Environments*, 20(6):529–544, December 2011.

- [43] E. C. Lalor, S. P. Kelly, C. Finucane, R. Burke, R. Smith, R. B. Reilly, and G. McDarby. Steady-State VEP-Based Brain-Computer Interface Control in an Immersive 3D Gaming Environment. *EURASIP Journal on Advances in Signal Processing*, 19:3156–3164, 2005.
- [44] Fabien Lotte, Yann Renard, and Anatole Lécuyer. Self-Paced Brain-Computer Interaction with Virtual Worlds: A Quantitative and Qualitative Study “Out of the Lab”. In *4th International Brain Computer Interface Workshop and Training Course*, 2008.
- [45] G. Pfurtscheller and F. H. Lopes da Silva. Event-related EEG/MEG synchronization and desynchronization: basic principles. *Clinical Neurophysiology*, 110:1842–1857, 1999.
- [46] Josef Faller, Robert Leeb, Gert Pfurtscheller, and Reinhold Scherer. Avatar navigation in virtual and augmented reality environments using an SSVEP BCI. In *1st International Conference on Applied Bionics and Biomechanics ICABB-2010*, 2010.
- [47] Josef Faller Gernot Müller-Putz, Dieter Schmalstieg, and Gert Pfurtscheller. An Application Framework for Controlling an Avatar in a Desktop-Based Virtual Environment via a Software SSVEP Brain–Computer Interface. *Presence: Teleoperators and Virtual Environments*, 19:25–34, 2010.
- [48] Mindflex. <http://company.neurosky.com/products/mattel-mindflex/>.
- [49] Neurosky. <http://www.neurosky.com/>.
- [50] Necomimi. <http://necomimi.com/>.
- [51] Neurogaming 2013 Conference and Expo. <http://www.neurogamingconf.com/>.
- [52] Son of nor. <http://stillalive-studios.com/portfolio-item/son-of-nor/>.
- [53] Son of nor - gameplay video. http://www.youtube.com/watch?feature=player_embedded&v=NbCtu_yt4kk.
- [54] José del R. Millán, Ferran Galán, Dirk Vanhooydonck, Eileen Lew, Johan Philips, and Marnix Nuttin. Asynchronous non-invasive brain-actuated control of an intelligent wheelchair. In *31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Minneapolis, USA, September 3-6, 2009*, 2009.
- [55] José del R. Millán, F. Renkens, J. Mouri no, and W. Gerstner. Noninvasive brain-actuated control of a mobile robot by human EEG. *IEEE Transactions on Bio-Medical Engineering*, 51:1026–1033, 2004.

- [56] Ronald C. Arkin. *Behavior-Based Robotics*. MIT Press, 1998.
- [57] J. del R. Millán, J. Mourino, M. Franzé, F. Cincotti, M. Varsta, J. Heikkonen, and F. Babiloni. A local neural classifier for the recognition of EEG patterns associated to mental tasks. *IEEE Transactions on Neural Networks*, 13:678–686, 2002.
- [58] Eileen Lew, Marnix Nuttin, Pierre W. Ferrez, Alexandra Degeest, Anna Buttfeld, Gerolf Vanacker, and José del R. Millán. Noninvasive brain-computer interface for mental control of a simulated wheelchair. In *Proc. 3rd Int. Brain-Computer Interface-Workshop & Training Course*, 2006.
- [59] Ferran Galán, Marnix Nuttin, Eileen Lew, Pierre W. Ferrez, Gerolf Vanacker, Johan Philips, and José del R. Millán. A Brain-Actuated Wheelchair: Asynchronous and Non-Invasive Brain-Computer Interfaces for Continuous Control of Robots. *Clinical Neurophysiology*, 119(2007):2159–2169, January 2008.
- [60] Johan Philips, José del R. Millán, Gerolf Vanacker, Eileen Lew, Ferran Galán, Pierre W. Ferrez, Hendrik Van Brussel, and Marnix Nuttin. Adaptive Share Control of a Brain-Actuated Simulated Wheelchair. In *IEEE 10th International Conference on Rehabilitation Robotics, 2007. ICORR 2007.*, 2007.
- [61] B. Rebsamen, Cuntai Guan, Haihong Zhang, Chuanchu Wang, Cheeleong Teo, V.M.H. Ang, and E. Burdet. A brain controlled wheelchair to navigate in familiar environments. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 18(6):590–598, Dec 2010.
- [62] Luca Tonin, Emanuele Menegatti, Marianna Cavinato, Costanza D’Avanzo, Marco Pirini, Antonio Merico, Lamberto Piron, Konstantinos Priftis, Stefano Silvoni, Chiara Volpato, and Francesco Piccione. Evaluation of a Robot as Embodied Interface for Brain Computer Interface Systems. *International Journal of Bioelectromagnetism*, 11:97, 2009.
- [63] Carlos Escolano, J. Antelis, and Javier Minguez. Human Brain-Teleoperated Robot between Remote Places. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2009.
- [64] Christian J Bell, Pradeep Shenoy, Rawichote Chalodhorn, and Rajesh P N Ra. Control of a humanoid robot by a noninvasive brain-computer interface in human. *Journal of Neural Engineering*, 5:214–220, 2008.
- [65] Joris Elshou and Gary Garcia Molina. Review of brain-computer interfaces based on the P300 evoked potential. *Technical Report PR-TN 2009/00066, Koninklijke Philips Electronics*, 2009.
- [66] Haihong Zhang, Cuntai Guan, and Chuanchu Wang. Asynchronous P300-based brain-computer interfaces: A computational approach with statistical models. *IEEE Transactions on Biomedical Engineering*, 55:1754, 2008.

- [67] Direct brain control of a humanoid robot - video of bell et al. experiment. <http://www.youtube.com/watch?v=TQ7EOpPNQyw>.
- [68] Ori Cohen, Sébastien Druon, Sébastien Lengagne, Avi Mendelsohn, Rafael Malach, Abderrahmane Kheddar, and Doron Friedman. fMRI based robotic embodiment: a pilot study. In *IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pages 314–319, Roma, Italy, 24–28 June 2012.
- [69] K. Bouyarmane, J. Vaillant, N. Sugimoto, F. Keith, J. Furukawa, and J. Morimoto. BCI Control of Whole-Body Humanoid by Combining Motor Imagery Detection and Autonomous Motion Planning. In *Proceedings of 2013 International Conference on Neural Information Processing*, 2013.
- [70] Abdullah Akce, Miles Johnson, Or Dantsker, and Timothy Bretl. A Brain–Machine Interface to Navigate a Mobile Robot in a Planar Workspace: Enabling Humans to Fly Simulated Aircraft with EEG. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 21:306–318, 2013.
- [71] Chet T. Moritz, Steve I. Perlmutter, and Eberhard E. Fetz. Direct control of paralysed muscles by cortical neurons. *Nature*, 456:639–642, 2008.
- [72] Audrey S. Royer and Bin He. Goal Selection vs. Process Control in a Brain-Computer Interface based on Sensorimotor Rhythms. *Journal of Neural Engineering*, 6, 2009.
- [73] Pierre Gergondet, Sebastien Druon, Abderrahmane Kheddar, Christoph Hintermüller, Christoph Guger, and Mel Slater. Using Brain-Computer Interface to Steer a Humanoid Robot. In *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2011.
- [74] F. Sharbrough, G.-E. Chatrian, RP. Lesser, H. Lüders, M. Nuwer, and TW. Picton. American Electroencephalographic Society guidelines for standard electrode position nomenclature. *Journal of Clinical Neurophysiology*, 8:200–202, 1991.
- [75] Pierre Gergondet, Abderrahmane Kheddar, Christoph Hintermüller, Christoph Guger, and Mel Slater. Multitask Humanoid Control with a Brain-Computer Interface: user experiment with HRP-2. *International Symposium on Experimental Robotics (ISER)*, 2012.
- [76] Martin Spüler, Wolfgang Rosenstiel, and Martin Bogdan. Online adaptation of a c-vep brain-computer interface(bci) based on error-related potentials and unsupervised learning. *PLoS ONE*, 7(12):e51077, 12 2012.
- [77] Nicolas Mansard, Olivier Stasse, Paul Evrard, and Abderahmane Kheddar. A versatile generalized inverted kinematics implementation for collaborative working humanoid robots: The stack of tasks. In *Int. Conf. on Advanced Robotics (ICAR)*, 2009.

- [78] Erik Murphy-Chutorian and Jochen Triesch. Shared Features for Scalable Appearance-Based Object Recognition. In *IEEE Workshop on Applications of Computer Vision*, 2005.
- [79] Erik Murphy-Chutorian, Sarah Aboutalib, and Jochen Triesch. Analysis of a Biologically-Inspired System for Real-time Object Recognition. *Cognitive Science Online*, 3.2:1–14, 2005.
- [80] T. Mörwald, J. Prankl, A. Richtsfeld, M. Zillich, and M. Vincze. BLORT - The Blocks World Robotic Vision Toolbox. *Best Practice in 3D Perception and Modeling for Mobile Manipulation (in conjunction with ICRA 2010)*, 2010.
- [81] D. G. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157. IEEE, 1999.
- [82] Arnaud Doucet, Nando de Freitas, and Neil Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, 2001.
- [83] H. Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *Robotics Automation Magazine, IEEE*, 13(2):99–110, June 2006.
- [84] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136, Basel, Switzerland, 26-29 October 2011. IEEE Computer Society.
- [85] Maxime Meilland and Andrew I. Comport. On unifying key-frame and voxel-based dense visual SLAM at large scales. In *International Conference on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2013.
- [86] Renato F. Salas-Moreno, Richard A. Newcombe, Hauke Strasdat, Paul H. J. Kelly, and Andrew J. Davison. SLAM++: Simultaneous localisation and mapping at the level of objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1352–1359, Portland, Oregon, 23-28 June 2013.
- [87] Claire Dune, Andrei Herdt, Olivier Stasse, Pierre-Pierre-Brice Wieber, K. Yokoi, and Eiichi Yoshida. Cancelling the sway motion of dynamic walking in visual servoing. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 3175–3180. IEEE, 2010.
- [88] Andrei Herdt, Nicolas Perrin, and Pierre-Brice Wieber. Walking without thinking about it. In *IEEE International Conference on Robotics Systems (IROS)*, 2010.
- [89] François Chaumette and Seth Hutchinson. Visual servo control, Part I: basic approaches. *IEEE Robotics and Automation Magazine*, 13(4):82–90, December 2006.

- [90] François Chaumette and Seth Hutchinson. Visual servo control, Part II: advanced approaches. *IEEE Robotics and Automation Magazine*, 14(1):109–118, 2007.
- [91] O. Stasse, B. Verrelst, A. Davison, N. Mansard, F. Saidi, B. Vanderborght, C. Esteves, and K. Yokoi. Integrating walking and vision to increase humanoid autonomy. *International Journal of Humanoid Robotics, special issue on Cognitive Humanoid Robots*, 5(2):287–310, 2008.
- [92] P. Alcantarilla, O. Stasse, S. Druon, L. M. Bergasa, and F. Dellaert. How to localize humanoids with a single camera ? *Autonomous Robot*, 34(1-2):47–71, 2013.
- [93] Don Joven Agravante, Andrea Cherubini, Antoine Bussy, Pierre Gergondet, and Abderahmane Kheddar. Collaborative human-humanoid carrying using vision and haptic sensing. In *IEEE International Conference on Robotics and Automation*, pages 607–612, Hong Kong, China, 31 May - 7 June 2014.
- [94] Miao Li, Hang Yin, Kenji Tahara, and Aude Billard. Learning object-level impedance control for robust grasping and dexterous manipulation. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2014. Accepted for publication.
- [95] M. Tenorth, L. Kunze, D. Jain, and M. Beetz. Knowrob-map - knowledge-linked semantic object maps. In *Humanoid Robots (Humanoids), 2010 10th IEEE-RAS International Conference on*, pages 430–435, Dec 2010.
- [96] Damien Petit, Pierre Gergondet, Andrea Cherubini, Maxime Meilland, Andrew I. Comport, and Abderahmane Kheddar. Navigation assistance for a bci-controlled humanoid robot. In *IEEE International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (IEEE-CYBER 2014)*, 2014.
- [97] Pierre Gergondet, Damien Petit, Maxime Meilland, Abderahmane Kheddar, Andrew I. Comport, and Andrea Cherubini. Combining 3D SLAM and Visual Tracking to Reach and Retrieve Objects in Daily-Life Indoor Environments. In *International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 2014.
- [98] Emmanuele Tidoni, Pierre Gergondet, Abderrahmane Kheddar, and Salvatore M Aglioti. Audio-visual feedback improves the bci performance in the navigational control of a humanoid robot. *Frontiers in Neurobotics*, 8(20), 2014.
- [99] Gary G. Briggs and Robert D. Nebes. Patterns of Hand Preference in a Student Population. *Cortex*, 11(3):230–238, September 1975.
- [100] P. Haggard, S. Clark, and J. Kalogeras. Voluntary action and conscious awareness. *Nature Neuroscience*, 5(4):382–385, April 2002.

- [101] M. C. Vinding, M. N. Pedersen, and M. Overgaard. Unravelling intention: distal intentions increase the subjective sense of agency. *Consciousness and cognition*, 22(3):810–815, September 2013.
- [102] D. M. Wolpert and M. S. Landy. Motor control is decision-making. *Current opinion in neurobiology*, 22(6):996–1003, December 2012.
- [103] E. V. Friedrich, R. SchR. Scherer, K. Sonnleitner, and C. Neuper. Impact of auditory distraction on user performance in a brain-computer interface driven by different mental tasks. *Clinical neurophysiology : official journal of the International Federation of Clinical Neurophysiology*, 122(10):2003–2009, October 2011.
- [104] J. P. Bresciani, M. O. Ernst, K. Drewing, G. Bouyer, V. Maury, and A. Kheddar. Experimental brain research. *Feeling what you hear: auditory signals can modulate tactile tap perception*, 162(2):172–180, April 2005.
- [105] G. Onose, C. Grozea, A. Anghelescu, C. Daia, C. J. Sinescu, A. V. Ciurea, T. Spiricu, A. Mirea, I. Andone, A. Spânu, C. Popescu, A. .S. Mihaescu, S. FazS. Fazli. Danóczy, and F. Popescu. On the feasibility of using motor imagery EEG-based brain-computer interface in chronic tetraplegics for assistive robotic arm control: a clinical test and long-term post-trial follow-up. *Spinal Cord*, 50(8):599–608, August 2012.
- [106] Thorsten O. Zandera, Matti Gaertner, Christian Kothe, and Roman Vilimel. Combining Eye Gaze Input With a Brain-Computer Interface for Touchless Human-Computer Interaction. *International Journal of Human-Computer Interaction*, 27(1):38–51, December 2010.
- [107] Mel Slater, B. Spanlang, and D. Corominas. Simulating virtual environments within virtual environments as the basis for a psychophysics of presence. *ACM Transactions on Graphics*, 29(4), 2010.
- [108] Emmanuele Tidoni, Gabriele Fusco, Daniele Leonardis, Antonio Frisoli, Massimo Bergamasco, and Salvatore Maria Aglioti. Illusory movements induced by tendon vibration in right- and left-handed people. *Experimental Brain Research*, pages 1–9, 2014.
- [109] Emmanuele Tidoni, Luigi Grisoni, Marco Tullio-Liuzza, and Salvatore-Maria Aglioti. Rubber hand illusion highlights massive visual capture and sensorimotor face-hand remapping in a tetraplegic man. *Restorative Neurology and Neuroscience*, 32:611–622, 2014.
- [110] Richard Staba, Charles L. Wilson, Anatol J. Bragin, Itzhak Fried, and Jerome Engel. Quantitative Analysis of High-Frequency Oscillations (80-500 Hz) Recorded in Human Epileptic Hippocampus and Entorhinal Cortex. *Journal of Neurophysiology*, 88(4):1743–1752, 10 2002.

-
- [111] Nathan Earl Crone, Diana Miglioretti, Barry Gordon, and Ronald Peter Lesser. Functional mapping of human sensorimotor cortex with electrocorticographic spectral analysis. II. Event-related synchronization in the gamma band. *Brain : a journal of neurology*, 121(12):2301–2315, 12 1998.
- [112] Christoph Kapeller, Christoph Schneider, Kyousuke Kamada, Hiroshi Ogawa, Naoto Kunii, Rupert Ortner, Robert Prueckl, and Christoph Guger. Single Trial Detection of Hand Poses in Human ECoG Using CSP Based Feature Extraction. In *IEEE Engineering in Medicine and Biology Society (EMBS)*, 2014.
- [113] Yonghui Fang, Minyou Chen, Xufei Zheng, and Robert F. Harrison. Extending CSP to Detect Motor Imagery in a Four-class BCI. *Journal of Information and Computational Science*, 9(1):143–151, 2012.
- [114] Christopher M. Bishop. *Neural networks for pattern recognition*. Clarendon Press, 1995.
- [115] John S Duncan. Epilepsy surgery. *Clinical Medicine*, 7(2):137–142, 2007.
- [116] Gretchen L. Birbeck, Ron D. Hays, Xinping Cui, and Barbara G. Vickrey. Seizure reduction and quality of life improvements in people with epilepsy. *Epilepsia*, 43(5):535–538, 2002.
- [117] Wei Wang, Jennifer L. Collinger, Alan D. Degenhart, Elizabeth C. Tyler-Kabara, Andrew B. Schwartz, Daniel W. Moran, Douglas J. Weber, Brian Wodlinger, Ramana K. Vinjamuri, Robin C. Ashmore, John W. Kelly, and Michael L. Boninger. An electrocorticographic brain interface in an individual with tetraplegia. *PLoS ONE*, 8(2):e55344, 02 2013.

Appendices

A Framework for Dynamic BCI

Application: bci-interface

This appendix completes the presentation of bci-interface that was introduced in Chapter 2. We review the motivation and goals that were presented in this earlier chapter and dive into the internals of the library. Finally, examples are introduced in Appendix B.

A.1 Motivation and Goals

Several frameworks, such as OpenViBE [4] or BCI2000 [5], have already been developed to produce high-quality BCI applications. The three layer model introduced earlier is implemented by these, however, most of the development effort is put towards the signal acquisition and signal processing layers leaving the user application to the user. Therefore we developed a framework to tackle this issue and easily write BCI applications: bci-interface.

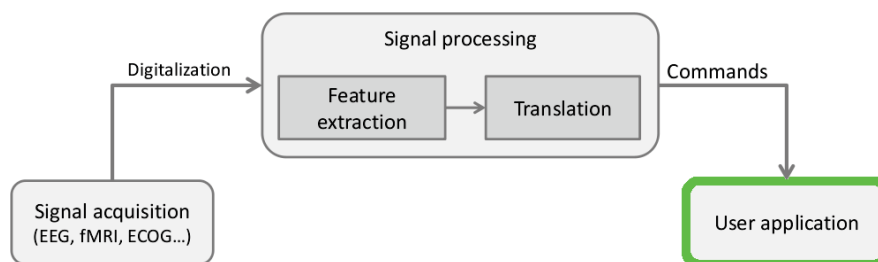


Figure A.1: bci-interface focuses on the user application layer in the BCI2000 model

This framework is developed with two important goals in mind.

First, it should be simple to use: the application writer should only worry about what s/he wishes the BCI user to accomplish. This is realized by providing simple elements to the de-

veloper that s/he can arrange and interact with easily without worrying about the underlying mechanisms of the interface. We introduce this elements later in this section and showcase the framework simplicity through examples.

Secondly, we want to be able to write dynamic BCI applications. Usual BCI applications are aimed at achieving one simple task: steering an avatar, controlling a cursor, selecting objects or bins. However, our goal is to allow a BCI user to take control of a humanoid robot and perform complex tasks. This implies the use of multiple paradigms as one interface cannot fit all of our needs. For example, displaying SSVEP arrows to control the humanoid steering may be satisfactory. Yet, these same arrows become cumbersome when we want to achieve grasping and another method has to be developed. These arrows could also be used to control the humanoid's gaze or a limb. More importantly, we definitely want to be able to control all of these within a single application. It is then necessary to be able to dynamically switch between different BCI control scheme, a possibility offered by bci-interface that is also showcased in the examples.

A.2 Command Model in bci-interface

In bci-interface the commands are represented by integer values. So for examples, if a BCI system is setup to detect 4 different SSVEP frequencies and has implemented the zero-class principle—a no-command implementation. Then the following mapping occurs:

- 0 is no command
- 1 is the first frequency setup in the BCI system
- 2 is the second frequency setup in the BCI system
- 3 is the third frequency setup in the BCI system
- 4 is the fourth frequency setup in the BCI system

A.3 Framework Elements

The library relies on a set of basic elements that each participate in a single part of the process. These elements are coordinated by a core class: *BCIInterface*. In this section we introduce the core class and the other elements of the interface and their respective roles. Except for *BCIInterface* which is fully implemented and provided as-is, all these elements are virtual classes that should be implemented to meet the developer needs. The specific functions that should be implemented by each element and, when relevant, some sample implementations are presented here.

A.3.1 BCIInterface

BCIInterface is the core of the bci-interface library. Its role is to create the graphical interface, handle drawings, distribute events and manage the other objects. It is only in charge of managing the threads that may be initiated by other elements of the library. We focus more specifically on this class later to discuss its workflow.

A.3.2 EventHandler

An *EventHandler* is in charge of handling events received by the windows, e.g. keyboard inputs or mouse movements. This is mostly used for debugging as the actual inputs that matter for the interface are those from the BCI.

The developer only has to implemented one function in this case:

```
void EventHandler::Process(sf::Event & event);
```

Where *sf::Event* is a class provided by the SFML library that stores the event type and properties.

An *EventHandler* is added to the interface by calling the following function:

```
/* Add an event handler,  
   the interface assumes ownership of the object */  
void BCIInterface::AddEventHandler(EventHandler * handler);
```

A.3.3 Background

A *Background* implementation has two goals:

- Draw the interface background
- Get updates of the background, e.g. if the background is fed from a network source

Therefore, three functions should be implemented by the developer to use a *Background* in a bci-interface application.

```
virtual void Background::Draw(sf::RenderWindow * app);
```

As we explain it later, this function is called at each frame to draw the background on screen. It is possible to use either SFML calls to draw to the screens or to use OpenGL. In the later case the following function should be implemented to return *true* when one wishes to use OpenGL to draw the background.

```
virtual bool Background::DrawWithGL();
```

The second role of a *Background* implementation is ensured by two functions:

```
virtual void Background::UpdateLoop();
```

```
virtual void Background::Close();
```

The *UpdateLoop* function is executed in its own thread. Its role is to update the image that is drawn to the background. For example, it can be a function receiving images streamed on the network by a remote camera. The *Close* function is called by *BCIInterface* to signal the end of life of the *Background* implementation.

A *Background* is added to the interface by calling the following function:

```
void BCIInterface::SetBackground(Background * background);
```

A.3.4 DisplayObject

The *DisplayObject* designates an object that is displayed on screen in the interface. This covers both the stimuli and the “decoration” objects in the interface such as speed or proximity indicators.

Two categories of objects are considered by the interface:

- Active objects are stimuli used by the interface, this is the objects the user interacts with
- Inactive objects are used for decoration purpose

Therefore there is two functions to add a *DisplayObject*:

```
/* This function is used to add an active object */
void BCIInterface::AddObject(DisplayObject * object);

/* This function is used to add an inactive object */
void BCIInterface::AddNonOwnedObject(DisplayObject * object);
```

The following function should be implemented to draw the object:

```
virtual void DisplayObject::Display(sf::RenderWindow * app,
    unsigned int frameCount, sf::Clock & clock);
```

As for *Background*, the display function can be written using either SFML calls to draw to the screens or using OpenGL. In the later case the following function should be implemented to return *true* when one wishes to use OpenGL to draw the object.

```
virtual bool DisplayObject::DrawWithGL();
```

The following functions can also be re-implemented, if necessary, to let the *Command-Interpreter*—or another external mechanism—control the object:

```
virtual void DisplayObject::SetPosition(float X, float Y);

virtual void DisplayObject::SetRotation(float deg_angle);

virtual void DisplayObject::Resize(float size_x, float size_y);
```

The *DisplayObject* can also provide *Highlight* and *Unhighlight* functions. These can be particularly useful for active objects as these functions can be used to provide feedback to the user about the command selected by the BCI.

Finally, *DisplayObject* can also react to windows events, e.g. keyboard actions or mouse movements. To do so, one simply has to implement the same function as for an *EventHandler*:

```
virtual void DisplayObject::Process(sf::Event & event,
    const BCIInterface & iface);
```

Two examples of *DisplayObject* that are provided with the distributed code are:

- *TextObject*
- *SSVEPStimulus*

A *TextObject* is designed to be an inactive object. Its only purpose is to display text within the interface.

A *SSVEPStimulus* on the other hand is specifically designed to be an active object. It can display an SSVEP stimulus of any frequency on the screen, using the method introduced in the first chapter. Using different constructors it is possible to use a rectangle shape, a circle or a specific sprite.

A.3.5 CommandReceiver

A *CommandReceiver* is used to receive commands from the BCI system. Like the *Background*, the *CommandReceiver* can receive its commands asynchronously. To do so, the developer should implement the following functions:

```
virtual void CommandReceiver::CommandLoop();
```

```
virtual void CommandReceiver::Close();
```

The *CommandLoop* function is launched in its own thread by *BCIInterface* and the *Close* function is called when the interface is closing or the interface wishes to switch the *CommandReceiver*. As the receiving occurs asynchronously, the *BCIInterface* calls the following function during the execution to receive the current command of the BCI system, note that this can also be used to receive the command synchronously:

```
virtual int CommandReceiver::GetCommand();
```

A *CommandReceiver* is added to the interface by calling the following function:

```
void BCIInterface::SetCommandReceiver(CommandReceiver * receiver);
```

Two *CommandReceiver* are implemented in the distributed code:

- *UDPReceiver* is an UDP server that receives commands asynchronously
- *TCPReceiver* is a TCP server that receives commands asynchronously

A.3.6 CommandInterpreter

A *CommandInterpreter* is used to interpret the commands sent by the BCI and received through a *CommandReceiver*. To do so, the following function should be implemented:

```
virtual bool CommandInterpreter::InterpretCommand(int command,  
                                                  const std::vector<DisplayObject*> & objects);
```


The first argument of these functions correspond to the key that should be pushed to trigger the command while the `command` corresponds to the command that is sent to the *CommandInterpreter*. When the interface is running, if the interface is receiving an event that is handled by the *CommandOverrider* then it sends the command attached to this event to the current *CommandInterpreter* independently of the command currently received by the *CommandReceiver*.

A *CommandOverrider* is added to the interface by calling the following function:

```
void SetCommandOverrider(CommandOverrider * receiver);
```

A.4 Workflow

The workflow of *BCIInterface*, the core of the library, is shown in Figure A.2. This Figure shows what happens each time a frame is being drawn on the screen. The library was conceived so that this workflow would run efficiently, i.e. at least at the frequency of the screen upon which the interface is displayed. However, this may run faster but the core provides the current number of frame actually displayed on screen in case it is relevant for the object being treated by the interface, e.g. in the case of the SSVEP stimuli.

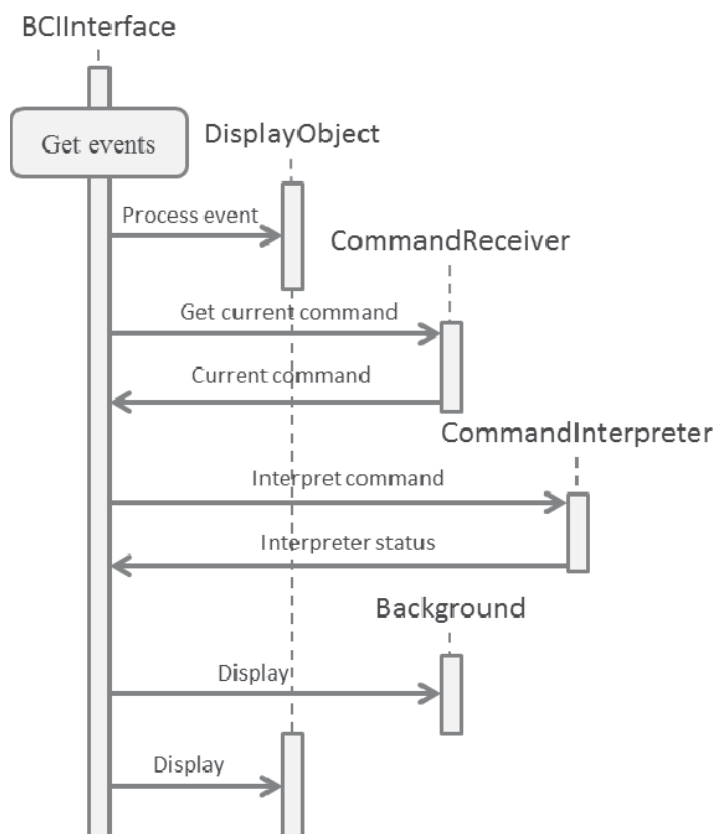


Figure A.2: *bci-interface workflow: all of this occurs in a single iteration*

During a frame, the workflow of *BCIInterface* can be described as follow:

1. Get all events that occurred since the last drawing.
2. Pass these events to: the *EventHandlers*, the *DisplayObjects*, the *CommandOverrider* and finally the *CommandInterpreter*.
3. Get the command from the BCI system through the *CommandReceiver*.
4. Check the *CommandOverrider* status, if it is currently overriding the command, i.e. one of the registered inputs is active, then override the command received from the *CommandReceiver*.
5. Pass the command to the *CommandInterpreter*, if the *CommandInterpreter* returns `true` then change the paradigm status—as explained later. Note that Steps 3 through 5 are only executed if both a *CommandReceiver* and a *CommandInterpreter* have been setup with the interface. Identically, Step 4 is only executed if a *CommandOverrider* has been set in the interface.
6. Draw the *Background* if any is present.
7. Draw the *DisplayObjects*. The order of drawing for the different kind of *DisplayObject* is: active objects drawn with SFML calls, inactive objects drawn with SFML calls, active objects drawn with OpenGL and inactive objects drawn with OpenGL; objects are also drawn in the order they were added to the interface.

This is repeated until the application is exited, which can be triggered by calling the following function:

```
void BCIInterface::Close();
```

A.4.1 The paradigm workflow approach

In bci-interface, a paradigm is represented by the combination of two elements:

- A set of active *DisplayObjects* that are used to elicit the correct answer in the subject's brain.
- A *CommandInterpreter* that interprets the commands sent by the BCI into commands for the controlled system, e.g. a robot.

A switch between two paradigms is triggered by the end of life of a *CommandInterpreter*. This is signaled by the *CommandInterpreter* itself when it returns `true` as it is interpreting a command or by an external stimulus that calls the *BCIInterface::StopParadigm()* function. After this is done, the interface continues to display the objects; however, no command is interpreted until a new *CommandInterpreter* is set and the *BCIInterface::StartParadigm()* function is called.

Algorithm 3 High-level approach of the paradigm workflow in bci-interface

```

1: while iface not closed do
2:                                     ▷ While the interface has not been closed
3:   if iface has an interpreter then
4:
5:     Start the display loop with the interpreter until it is ended
6:   else
7:
8:     Start the display loop with no interpreter until a new paradigm is started.
9:   end if
10: end while

```

The way the paradigm workflow is done in bci-interface at the highest level is shown in Algorithm 3. In this algorithm, *iface* designates an instance of *BCIInterface*.

From outside, the paradigm state is manipulated using three functions:

```

void BCIInterface::StartParadigm();

void BCIInterface::StopParadigm();

bool BCIInterface::ParadigmStatus();

```

BCIInterface::StartParadigm() is used to start a new paradigm after setting a new *CommandInterpreter*. *BCIInterface::StopParadigm()* is used to stop a paradigm even if it is not finished. For example, if a paradigm switch should be triggered by an external stimulus rather than a decision of the user. Finally, *BCIInterface::ParadigmStatus()* is used to get the current status of the paradigm. This is particularly useful to wait for the end of a paradigm before launching a new one as we show in the examples.

A.5 Examples

Finally, we give two simple examples of the usage of the library. The first one is a very simple application used to steer a humanoid robot while the other one is used to control both steering and gazing. The first example illustrates the basic principles of the library while the second one focus on the paradigm workflow approach. These example can be find in Appendix B.

Example uses of the bci-interface library

In this chapter, we give two simple examples of the usage of the library. The first one is a very simple application used to steer a humanoid robot while the other one is used to control both steering and gazing. The first example illustrates the basic principles of the library while the second one focus on the paradigm workflow approach.

B.1 Minimal example: an application for steering

In this example, we show how to build a simple BCI interface to control a robot steering. The only element that has not been introduced is the *WalkingInterpreter*. A *WalkingInterpreter* is a *CommandInterpreter* that does two things:

1. Highlight the object selected by the BCI
2. Send a speed to the robot, assuming the stimuli correspond to: 1–forward, 2–right, 3–backward and 4–right.

The video streaming part of the interface is excluded from this example to simplify it. This is typically done by using a specialized instance of *Background* that receives the video through a network connection.

```
#include <SFML/Graphics.hpp>
#include <bci-interface/BCIInterface.h>
#include <bci-interface/CommandReceiver/UDPReceiver.h>
#include <bci-interface/CommandInterpreter/WalkInterpreter.h>
#include <bci-interface/DisplayObject/SSVEPStimulus.h>

/* Everything in the lib lives in the bciinterface namespace,
   this makes this example more readable */
```

```
using namespace bciinterface;

int main(int argc, char * argv[])
{

/* First you need some information about window size
   and fullscreen mode */
int width  = 1024;
int height = 768;
bool fullscreen = false;

/* N.B: we create pointer here to show an important aspect
   of objects ownership in the library; this is not mandatory
   and this example could be written with normal objects */

/* We create a new BCIInterface with the width/height specified */
/* Note that this line alone does not start any display */
BCIInterface * bciinterface = new BCIInterface(width, height);

/* We setup a UDP server on port 1111 to receive commands */
UDPReceiver * receiver = new UDPReceiver(1111);
/* And set this as the receiver for the interface */
bciinterface->SetCommandReceiver(receiver);

/* We create a WalkingInterpreter to handle BCI commands */
WalkingInterpreter * interpreter = new WalkingInterpreter();
/* And set this as the interpreter for the interface */
bciinterface->SetCommandInterpreter(interpreter);

/* Create a few SSVEP stimuli on the screen, we show only
   one of the different constructors that can be used */
/* Sprite stimulus:
   args(freq, screenfreq, center_x, center_y, size_x, size_y,
        sprite_img, sprite_hl_img) */
bciinterface->AddObject(new SSVEPStimulus(6, 60, width/2, 100,
        200,200, "UP.png", "UP_HL.png"));
/* [snipped] Repeat to add the other three arrows */

/* And finally start the interface */
bciinterface->DisplayLoop(fullscreen);
/* This exits when the user exits the interface */

delete bciinterface;
delete interpreter;
delete receiver;
}
```

B.2 Switching example: control steering and gazing

In this example, we show how to build a simple BCI interface to control a robot steering and gazing. Another *CommandInterpreter* is introduced by this example the *GazeInterpreter*. A *GazeInterpreter* is a *CommandInterpreter* that does two things:

1. Highlight the object selected by the BCI
2. Send a pan/tilt command to the robot's head to control it, assuming the following correspondance between stimuli and command: 1–look up, 2–look right, 3–look down and 4–look right.

Both *WalkingInterpreter* and *GazeInterpreter* interpret the command “5” as the intention to switch between the two control paradigms.

```
#include <SFML/Graphics.hpp>
#include <bci-interface/BCIInterface.h>
#include <bci-interface/CommandReceiver/UDPReceiver.h>
#include <bci-interface/CommandInterpreter/WalkInterpreter.h>
#include <bci-interface/CommandInterpreter/GazeInterpreter.h>
#include <bci-interface/DisplayObject/SSVEPStimulus.h>
/* Additional includes needed for paradigm switch */
#include <boost/thread.hpp>

/* Everything in the lib lives in the bciinterface namespace,
   this makes this example more readable */
using namespace bciinterface;

/* This function is called to setup the interface */
void SetupInterface(BCIInterface * iface)
{
    /* Add the arrow objects */
    iface->AddObject(new SSVEPStimulus(6, 60, width/2, 100,
                                       200,200, "UP.png", "UP_HL.png"));
    /* [snipped] Repeat to add the other three arrow and switch */
}

int main(int argc, char * argv[])
{
    /* Same as the first example to begin with */
    int width  = 1024;
    int height = 768;
    bool fullscreen = false;

    BCIInterface * bciinterface = new BCIInterface(width, height);
```

```
UDPReceiver * receiver = new UDPReceiver(1111);
bciinterface->SetCommandReceiver(receiver);

WalkingInterpreter * wInterpreter = new WalkingInterpreter();
GazeInterpreter * gInterpreter = new GazeInterpreter();
/* This boolean holds whether current mode is walking or gazing */
bool walking = true;

/* And now things change to allow paradigm switch */
/* out_cmd is the command retained by the interface
   after a paradigm switch was triggered */
int out_cmd = -1;
/* Time during which commands are ignored after a switch */
float timeout = 2;
/* And we launch the interface in its own thread */
boost::thread th(boost::bind(
    &bciinterface::BCIInterface::DisplayLoop, bciinterface,
    (sf::RenderWindow*)0, fullscreen, &out_cmd, timeout));

/* Launch the paradigm loop */
while(out_cmd != 0)
{
    /* out_cmd will be set to 0 if we exit manually */

    /* Prepare the interface display */
    bciinterface->Clean();
    SetupInterface(bciinterface);

    /* Setup the command interpreter according to the
       current paradigm */
    if(walking)
    {
        bciinterface->SetCommandInterpreter(wInterpreter);
    }
    else
    {
        bciinterface->SetCommandInterpreter(gInterpreter);
    }
    walking = !walking;

    /* Launch paradigm */
    bciinterface->StartParadigm();
    /* Wait for the current paradigm to be over */
    while(bciinterface->ParadigmStatus())
    {
```

```
        usleep(100000);  
    }  
}
```

```
delete bciinterface;  
delete wInterpreter;  
delete gInterpreter;  
delete receiver;  
}
```


Users' questionnaires employed

C.1 SSVEP display optimization – user experience questionnaire

The questionnaire presented in Table C.1 is proposed to the subjects after the SSVEP display optimization experience presented in Chapter 2 – Section 2.3.

Table C.1: Questionnaire proposed to the user after the SSVEP display optimization experience.

Sex	Male / Female
Age	
Left handed / Right handed / Ambidextrous	
How often do you play videogames?	
Very frequently	Frequently Occasionally Rarely Very Rarely Never
If you had to use this method to select objects very often, which condition would you prefer to use? Why?	
<hr/> <hr/> <hr/>	
Please rank the five conditions regarding preference for its use: 1 is first, 5 is last.	
<input type="checkbox"/> Reference	<input type="checkbox"/> Hard mask <input type="checkbox"/> Brightness <input type="checkbox"/> Hybrid <input type="checkbox"/> Strong-hard
Which condition was the most annoying? Why? Please describe as much as possible.	

<hr/> <hr/> <hr/>									
Please rank the five conditions									
1 being the most annoying, 5 being the least annoying									
<input type="checkbox"/> Reference <input type="checkbox"/> Hard mask <input type="checkbox"/> Brightness <input type="checkbox"/> Hybrid <input type="checkbox"/> Strong-hard									
Please rate how easy it was to stay focused on the target in each condition:									
Reference	Very hard	1	2	3	4	5	6	7	Very easy
Hard color	Very hard	1	2	3	4	5	6	7	Very easy
Brightness	Very hard	1	2	3	4	5	6	7	Very easy
Hybrid	Very hard	1	2	3	4	5	6	7	Very easy
Strong-hard	Very hard	1	2	3	4	5	6	7	Very easy
Please rate how comfortable was each condition:									
Reference	Not at all	1	2	3	4	5	6	7	Very much
Hard color	Not at all	1	2	3	4	5	6	7	Very much
Brightness	Not at all	1	2	3	4	5	6	7	Very much
Hybrid	Not at all	1	2	3	4	5	6	7	Very much
Strong-hard	Not at all	1	2	3	4	5	6	7	Very much
Please rate how engaging was each condition:									
Reference	Not at all	1	2	3	4	5	6	7	Very much
Hard color	Not at all	1	2	3	4	5	6	7	Very much
Brightness	Not at all	1	2	3	4	5	6	7	Very much
Hybrid	Not at all	1	2	3	4	5	6	7	Very much
Strong-hard	Not at all	1	2	3	4	5	6	7	Very much
Please rate how challenging was each condition:									
Reference	Very much	1	2	3	4	5	6	7	Not at all
Hard color	Very much	1	2	3	4	5	6	7	Not at all
Brightness	Very much	1	2	3	4	5	6	7	Not at all
Hybrid	Very much	1	2	3	4	5	6	7	Not at all
Strong-hard	Very much	1	2	3	4	5	6	7	Not at all
Please rate how boring was each condition:									
Reference	Very boring	1	2	3	4	5	6	7	Very enjoyable
Hard color	Very boring	1	2	3	4	5	6	7	Very enjoyable
Brightness	Very boring	1	2	3	4	5	6	7	Very enjoyable

Hybrid	Very boring	1	2	3	4	5	6	7	Very enjoyable
Strong-hard	Very boring	1	2	3	4	5	6	7	Very enjoyable
Overall, how difficult did you find to perform the task?									
	Very difficult	1	2	3	4	5	6	7	Very easy
Please describe how you felt during the experiment:									
<hr/> <hr/> <hr/>									
What did bother you the most during the achievement of the task?									
<hr/> <hr/> <hr/>									
Do you have previous experience in BCI? If yes, please explain									
<hr/> <hr/> <hr/>									
Do you have any other comment?									
<hr/> <hr/> <hr/>									

C.2 – user experience questionnaire

The questionnaire presented in Table C.2 is proposed to the subjects after the Bayesian-network assisted BCI for humanoid robot control experience presented in Chapter 3 – Section 3.7.3.

Table C.2: Questionnaire proposed to the user after the Bayesian-network assisted BCI control experience.

Sex	Male / Female
Age	
Left handed / Right handed / Ambidextrous	
How often do you play videogames?	

Very frequently	Frequently	Occasionally	Rarely	Very Rarely	Never				
Between Assisted and Manual, which condition would you prefer to use? Why?									
<hr/> <hr/> <hr/>									
Please rate how easy it was to complete the task in each condition:									
Assisted	Very hard	1	2	3	4	5	6	7	Very easy
Manual	Very hard	1	2	3	4	5	6	7	Very easy
Please rate how comfortable was each condition:									
Assisted	Not at all	1	2	3	4	5	6	7	Very much
Manual	Not at all	1	2	3	4	5	6	7	Very much
Please rate how tiring was each condition:									
Assisted	Very much	1	2	3	4	5	6	7	Not at all
Manual	Very much	1	2	3	4	5	6	7	Not at all
Please rate how frustrating was each condition:									
Assisted	Very much	1	2	3	4	5	6	7	Not at all
Manual	Very much	1	2	3	4	5	6	7	Not at all
Please describe how you felt during the experiment:									
<hr/> <hr/> <hr/>									
What did bother you the most during the performance of the task?									
<hr/> <hr/> <hr/>									
Do you have previous experience in BCI? If yes, please explain									
<hr/> <hr/> <hr/>									
Do you have any other comment?									

TITRE : Commande d'humanoïdes robotiques ou avatars à partir d'interface cerveau-ordinateur

RÉSUMÉ : Cette thèse s'inscrit dans le cadre du projet Européen intégré VERE (Virtual Embodiment and Robotics re-Embodiment). Il s'agit de proposer une architecture logicielle intégrant un ensemble de stratégies de contrôle et de retours informationnels basés sur la "fonction tâche" pour incorporer un opérateur humain dans un humanoïde robotique ou un avatar par la pensée. Les problèmes sous-jacents peuvent se révéler par le démonstrateur suivant (auquel on souhaite aboutir à l'issue de cette thèse). Imaginons un opérateur doté d'une interface cerveau-ordinateur et d'un ensemble de traqueurs physiologiques (ex. respiration, activités musculaires, rythme cardiaque, etc.); le but est d'arriver à extraire de ces signaux la pensée de l'opérateur humain, de la traduire en commandes robotique et de faire un retour sensoriel afin que l'opérateur s'approprie le "corps" robotique ou virtuel de son "avatar". Une illustration cinématographique de cet objectif est le film récent "Avatar" ou encore "Surrogates". Dans cette thèse, on s'intéressera tout d'abord à certains problèmes que l'on a rencontré en travaillant sur l'utilisation des interfaces cerveau-ordinateur pour le contrôle de robots ou d'avatars, par exemple, la nécessité de multiplier les comportements ou les particularités liées aux retours sensoriels du robot. Dans un second temps, nous aborderons le cœur de notre contribution en introduisant le concept d'interface cerveau-ordinateur orienté objet pour le contrôle de robots humanoïdes. Nous présenterons ensuite les résultats d'une étude concernant le rôle du son dans le processus d'embodiment. Enfin, nous montrerons les premières expériences concernant le contrôle d'un robot humanoïde en interface cerveau-ordinateur utilisant l'électrocorticographie, une technologie d'acquisition des signaux cérébraux implanté dans la boîte crânienne.

MOTS-CLEFS : Robotique, Interface Cerveau-Ordinateur, Humanoïdes, Embodiment, Teleopération, Contrôle orienté vision.

TITLE : Humanoids robots' and virtual avatars' control through brain-computer interface

ABSTRACT : This thesis is part of the European project VERE (Virtual Embodiment and Robotics re-Embodiment). The goal is to propose a software framework integrating a set of control strategies and information feedback based on the "task function" in order to embody a human operator within a humanoid robot or a virtual avatar using his thoughts. The underlying problems can be shown by considering the following demonstrator. Let us imagine an operator equipped with a brain-computer interface and a set of physiological sensors; the goal is to extract the thoughts of the human operator from these signals, then translate it into robotic commands and finally to give an appropriate sensory feedback to the operator so that he can appropriate the "body", robotic or virtual, of his avatar. A cinematographic illustration of this objective can be seen in recent movies such as "Avatar" or "Surrogates". In this thesis, we start by discussing specific problems that we encountered while using a brain-computer interface for the control of robots or avatars, e.g. the arising need for multiple behaviours or the specific problems induced by the sensory feedback provided by the robot. We then introduce our main contribution which is the concept of object-oriented brain-computer interface for the control of humanoid robot. We then present the results of a study regarding the role of sound in the embodiment process. Finally, we show some preliminary experiments where we used electrocorticography (ECoG)—a technology used to acquire signals from the brain that is implanted within the cranium—to control a humanoid robot.

KEYWORDS : Robotic, Brain-Computer Interface, Humanoids, Embodiment, Teleoperation, Vision-based control.

DISCIPLINE : Génie Informatique, Automatique et Traitement du Signal

Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier
UMR 5506 CNRS/Université de Montpellier 2
161 rue Ada - 34392 Montpellier Cedex 5 - FRANCE