Enhancing BCI Based Navigation by Adding Intention Detection and a Virtual Assistant

THESIS

submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

COMPUTER SCIENCE

by

Christian Vermorken
born in Nieuwegein, The Netherlands

Interactive Intelligence Group
Faculty EEMCS, Delft University of Technology
Delft, the Netherlands
http://ii.tudelft.nl/
Enhancing BCI Based Navigation by Adding Intention Detection and a Virtual Assistant

Author: Christian Vermorken
Student id: 1358499
Email: c.vermorken@eaglescience.nl

Abstract

If a person is severely disabled, he can be stuck in a wheelchair without being able to effectively control it. One method to regain control is through a Brain Computer Interface (BCI). However, using a BCI for direct input has several drawbacks, such as a low accuracy, low bitrate (causing a delay in input) and high required mental effort. As a result, navigation in such a way is unreliable and in need of improvement.

In order to reduce the limitations of a BCI used in a navigation task, the following paradigm was investigated. While navigating, the user can give direct input using a reactive BCI based on Steady-State Somatosensory Evoked Potentials (SSSEP). With this BCI, possible controls can be selected. To limit the required direct user input, a passive BCI has been investigated in the first part of this thesis. This BCI attempts to detect the user’s intention to change course. If such an intention is detected, we can stop at the next intersection and ask the user which direction to turn. If no intention is detected, it is possible to simply move straight ahead without the need for direct input by the user.

Since intention detection is not always effective, another passive BCI will be used to detect the errors made, which can then be corrected. Finally, a virtual assistant gathers knowledge about the environment in order to further automate the movement. The complete paradigm was tested in the second part of this project.

Offline classification for the passive BCI for intention detection resulted in a mean accuracy of 65%, which, for some subjects, was partly based on eye movement artefacts. However, in some subjects an Event Related Potential (ERP) was observed, unrelated to the eye movement artefacts. This suggest that it may be possible to further improve the classification, even when eye movement artefacts are not present.

The complete paradigm, including virtual assistant, was evaluated on effectiveness, efficiency and satisfaction with a small test group (n=2). Although still not rated sufficient for use in a real world setting, the results were an improvement compared to using only a reactive BCI for navigation. Further studies with disabled people are recommended in order to find at what BCI classification accuracies the navigation would be deemed satisfactory.
Thesis Committee:

Chair: Prof. Dr. M.A. Neerincx, Delft University of Technology
Supervisor: Dr. T.O. Zander, Technical University Berlin
External supervisor: Dr. L.J.P. van der Maaten, Delft University of Technology
This thesis was written for the Media and Knowledge Engineering track of the Master Computer Science at the Delft University of Technology. The project was performed at EagleScience as part of the Brainflight FP7 project, which investigates the use of BCI in the aerospace domain.

I would like to thank Mark Neerincx for his feedback during my thesis. Thanks to him, I was able to keep the project focussed and in line with the requirements for my Master.

I would also like to thank Thorsten Zander for his support and feedback, both while in Berlin, as well as after my visits. Not only did he provide valuable insights and ideas for my experiments, but he also made my stay in Berlin a welcome experience.

Thanks also goes out to Marc Grootjen, not only for providing me with the opportunity to do this project at EagleScience, but also for his continued support and feedback during the project.

I would also like to thank Klaus Gramann for the discussions on my experiments and his help during my stay in Berlin.

I also thank Franc Grootjen for his help. Apart from giving valuable feedback on the first part of my thesis document, he also created the algorithm for offline detection of eye-blinks, allowing me to further examine the results of my experiment on passive BCI.

I also thank Laurens van der Maaten for his willingness to be on the thesis committee of a student he had never met before.

Finally, I would like to thank the Technical University of Berlin for allowing me to prepare and perform my experiments there.

Christian Vermorken
Delft, The Netherlands
April 8, 2014
# Contents

I Introduction

1 A Need to Improve BCI Based Navigation 3

2 Domain Analysis 7
   2.1 Operational Demands 7
   2.2 Human Factors 9
   2.3 Envisioned Technology 9
   2.4 Conclusion 13
   2.5 Specification and Evaluation 14

II Experiment 1: Passive BCI for Intention Detection 17

3 Introduction 19

4 Method 20
   4.1 Participants 20
   4.2 Stimuli 20
   4.3 Initial Experimental Task and Procedure 21
   4.4 Modifications to the Experimental Task and Procedure 22
   4.5 Recording 23
   4.6 Virtual Environment Design 23

5 Results 26
   5.1 Feature Extraction and Classification 26
   5.2 Event Related Activity 27

6 Discussion 33
   6.1 Offline Classification 33
   6.2 Eye blinks 34
   6.3 Subjects 34

7 Conclusion and Recommendations 36
# CONTENTS

III Experiment 2: Virtual Assistant for Online BCI Use ................................................................. 39

8 Virtual Assistant Prototype ........................................................................................................ 41
   8.1 Prototype Functionality ........................................................................................................ 41
   8.2 Technical Design ................................................................................................................. 42

9 Virtual Assistant Evaluation ...................................................................................................... 44
   9.1 Participants .......................................................................................................................... 44
   9.2 Stimuli ................................................................................................................................ 44
   9.3 Experimental Task and Procedure ....................................................................................... 45
   9.4 Variables ............................................................................................................................. 48
   9.5 Hypotheses .......................................................................................................................... 50
   9.6 Measurement ...................................................................................................................... 50
   9.7 Recording ............................................................................................................................ 51

10 Results ...................................................................................................................................... 52
   10.1 Performance Measures ....................................................................................................... 52
   10.2 Task Load .......................................................................................................................... 52
   10.3 Interview Results .............................................................................................................. 54

11 Discussion ................................................................................................................................ 56

12 Conclusion and Recommendations .......................................................................................... 59

IV Conclusion and Recommendations .......................................................................................... 61

13 Recommendations to Enhance BCI Based Navigation ............................................................ 63

Appendices ..................................................................................................................................... 64

A Expected Navigation Accuracy ................................................................................................ 65

B Specification ................................................................................................................................ 73
   B.1 Use Cases ............................................................................................................................ 73
   B.2 Requirements ...................................................................................................................... 86
   B.3 Claims .................................................................................................................................. 92

C Results Experiment 1 .................................................................................................................. 95

D Experimental Procedure Experiment 1 ...................................................................................... 103

E Experimental Procedure Experiment 2 ..................................................................................... 104

F Interview Experiment 2 ............................................................................................................... 106
   F.1 General Questions ............................................................................................................... 106
   F.2 Experiment related questions .............................................................................................. 106
List of Figures

2.1 The sCE method ....................................................... 8
4.1 Virtual Environment Experiment 1 ................................ 21
4.2 International 10-20 system for electrode placement ........... 24
5.1 Eye Movement Classification ...................................... 27
5.2 ERP ................................................................. 28
5.3 ITC and ERSP Subject 4 .......................................... 30
5.4 ITC and ERSP Subject 8 .......................................... 31
8.1 Classes Virtual Assistant .......................................... 42
9.1 Virtual Environment Experiment 2 ................................. 46
9.2 Flowchart Condition One ......................................... 47
9.3 Flowchart Condition Two ......................................... 49
A.1 Condition One Straight Flowchart ................................. 67
A.2 Condition Two Straight Flowchart ................................. 68
A.3 Condition One Turn Flowchart ................................ 69
A.4 Condition Two Turn Flowchart ................................ 70
A.5 Error Correction Flowchart ...................................... 71
C.1 ERP .................................................................. 96
C.2 ITC and ERSP Subject 4 .......................................... 98
C.3 ITC and ERSP Subject 8 .......................................... 101
List of Tables

1.1 A schematic overview of the proposed navigation paradigm. 4
2.1 A schematic overview of the extended navigation paradigm 15
5.1 Classification Accuracy Experiment 1, Automatic Time Window 27
5.2 Classification Accuracy Experiment 1, Set Time Window 32
10.1 Navigation Errors Made 52
10.2 Decision Speed 53
10.3 Navigation Speed 53
10.4 Reactive BCI Error Rate 53
10.5 NASA TLX Scores 54
A.1 Comparison Error Detection and Correction 72
Part I

Introduction
Chapter 1

A Need to Improve BCI Based Navigation

If a person is severely disabled, it can become more difficult for him to interact with the environment, especially in the very severe cases with syndromes such as Amyotrophic Lateral Sclerosis (ALS). Although no statistics on the total number of people with such severe disabilities exist, ALS alone already has an estimated number of 30,000 people living with the disease in just the U.S. [1]. Because of diseases like this, patients are often unable to move autonomously and need daily care from a caretaker. In the later stages of such illnesses, patients tend to be stuck in a wheelchair without an effective method to control it. By assisting the patient with navigation in his wheelchair, he would be able to regain some autonomy and possibly lessen a caregiver’s burden.

Several assistive technologies exist that can aid severely disabled people, allowing them to control (among other things) a wheelchair. One technology that can do this is a Brain-Computer Interface (BCI). A BCI allows its user to interact with the world using only his brain signals. Several different BCI technologies have been developed. The most often used technologies are based on Electroencephalography (EEG), which are the focus of this thesis.

When discussing BCI, it is useful to make a distinction between different types of BCIs first. Zander et al. [42] proposed a categorisation of BCI techniques into three categories:

- **Active BCI**
  - An active BCI uses signals directly controlled by the user, for example through motor imagery (MI). This is independent from external events. An example of this is imagining a movement of the right leg in order to accelerate in a vehicle.

- **Reactive BCI**
  - A reactive BCI uses a person’s reaction to voluntarily perceived stimuli, to evoke specific brain activity which the user cannot natively generate. This is intended for direct control. An example of this is when a user has several tactile stimuli presented to him. He can then give a command by focussing his attention on one of these stimuli.

- **Passive BCI**
  - A passive BCI does not use any signal generated on purpose by the user. Instead, it uses spontaneous brain signals to gather implicit information on the user’s state. An example of this would be a BCI to detect when a user notices that he made an error in the task he is performing.

Since direct input is required in order to properly navigate in a wheelchair, currently only active and reactive BCIs are used for this purpose. However, these BCIs have several disadvantages when used in such a task. First of all, many active and reactive BCIs suffer from a low bitrate [8]. As a result, it can sometimes take several seconds to give a command,
A Need to Improve BCI Based Navigation

making immediate control impossible. Secondly, the classification accuracy of a BCI often leaves plenty to be desired as well [8]. An error rate of 20%, for example, is not uncommon. Because of this, errors will be frequent. Finally, the mental effort required to control a BCI can be high, since continuous focus may be required [44]. Therefore, before a BCI could realistically be used to control a wheelchair, improvements have to be made. One option is to increase the speed and reliability of the BCI, a subject on which most BCI research is conducted. Another option is to circumvent these disadvantages by providing additional support to automate some navigation tasks.

One method to partially automate the navigation is to introduce the notion of shared control [24]. In such a shared autonomy framework, the output of the BCI is usually combined with knowledge about the environment and the vehicle being controlled. Additionally, even knowledge about the user’s mental state can be used, which can be gathered using a passive BCI such as described above, in order to create a context-aware BCI [41]. Therefore, in order to improve BCI-based navigation, we propose the following paradigm, presented in Table 1.1. This paradigm consists of several navigation states, at each of which one or more possible controls are available. Which one of these controls is selected and at what time, is based on passive BCIs, a reactive BCI and a virtual assistant, as indicated by the table.

<table>
<thead>
<tr>
<th>Navigation State</th>
<th>Possible Controls</th>
<th>Passive BCI</th>
<th>Reactive BCI</th>
<th>Virtual Assistant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driving</td>
<td>Brake</td>
<td>Change course detected → Brake</td>
<td>Change course not detected → Brake</td>
<td>Detect blockade → Brake</td>
</tr>
<tr>
<td>Braking</td>
<td></td>
<td>Change course erroneously detected → Allow Drive Straight</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stationary</td>
<td>Drive Straight, Drive Left, Drive Right</td>
<td>Choose direction → Drive</td>
<td>Detect blockade → Limit available controls</td>
<td></td>
</tr>
</tbody>
</table>

Table 1.1: A schematic overview of the proposed navigation paradigm.

While in the driving state, there is only one possible control, which is to brake. In order to determine when to brake, a passive BCI for intention detection is used, indicated by the third column in the table. Whenever the user wants to change his current course, the intention detection BCI should detect this. When such an intention is detected, the braking control will be used and we will move to the braking state. However, to our knowledge, it has never been attempted to create such a passive BCI. Therefore, this leads to the following research questions, which will be the focus of the first part of this thesis:

1. How can a passive BCI be used to estimate the user’s intention to change course?
2. What is the accuracy of the passive BCI’s estimation?

Both while driving and while stationary, a virtual assistant will gather knowledge about the environment, such as the type of intersection ahead. This knowledge is then combined with direct input from the reactive BCI, which allows the user to select a possible control and choose a direction to move in. For example, when stationary after an intention to change course has been detected, the user is simply presented with the option to start driving to the left or to the right. He will then choose one of these options using the reactive BCI. However, if the virtual assistant has detected that it is not possible to make a left turn, a right turn can automatically be made, with requiring any direct input from the user. Additionally, the virtual assistant can provide feedback to the user.

As mentioned earlier, one inherent issue with BCIs is their lower accuracy rate. As a result, it is likely that the paradigm mentioned so far will regularly cause the user to make a turn in the wrong direction. One method to reduce the number of navigation errors made, is to introduce a second passive BCI for error detection. Whenever we incorrectly enter the braking state due to a misclassification of the intention detection BCI, the user will notice this. Whenever the user notices such an error in the task he is trying to perform, an Error Potential (ErrP) is generated automatically. This can then be classified with accuracies as high as 84% [40]. After such an error has been detected, the user can now also choose the option to continue straight ahead when stationary.

This leads to the second part of this thesis and its final research question, in which the entire navigation paradigm is investigated:

3. How can BCIs for intention detection, error detection and control selection be combined for an effective, efficient and satisfactory navigation?

Which type of BCI is used for direct input to a system is also of great importance to its design. In this case, the BCI used to give direct commands will be the SSSEP BCI developed in Berlin [43]. This is a reactive BCI based on Steady-State Evoked Potentials (SSEP), where a user perceives a stimulus at a certain frequency. Focussing his attention on this stimulus causes a change in the spectral domain of the EEG at that frequency, which can then be classified. These stimuli are mostly presented visually. More recently however, the use of tactile stimuli has also been shown to work [43], which is called Steady-State Somatosensory Evoked Potentials (SSSEP). With this BCI, the user is able to give a binary input to a system.

Tactile stimuli have several advantages over stimuli perceived in other modalities. First of all, when stimuli are presented in this fashion, the user can still look around freely. Additionally, users with impaired vision can also use the system. Another advantage of tactile stimuli is its intuitiveness in certain tasks. In navigation tasks, for example, a stimulus presented on one side of the user’s body can correspond with a movement in that direction [38]. Additionally, the tactile channel is typically not overloaded with information in real life applications, while the visual and auditory channels often can be. Finally, tactile stimuli can be delivered in a discreet way, by wearing the tactile display under your own clothes. These advantages make the SSSEP BCI particularly suited for the navigation task of this research.

There are however also limitations to the use of tactile stimuli. The main issue is that it is not possible to constantly present these stimuli and have the system active continuously, since this will result in the user no longer being able to differentiate between different tactile stimuli. Therefore, a user cannot provide continuous input and it is important to know when to present the stimuli and when not to. Although this is a serious drawback, this is
A Need to Improve BCI Based Navigation

another issue that, to some extent, can be solved by the proposed paradigm investigated in this thesis.

This document consists of four parts. In the second half of this part, a domain analysis is presented, on which the design of a virtual assistant is based. Then, Part II investigates research question 1 and 2, whether it is possible to estimate a user’s intention using a passive BCI. Next, Part III approaches the third research question, by implementing and testing a virtual assistant to support in the online use of a BCI. Finally Part IV presents the overall conclusion to the research questions.
Chapter 2

Domain Analysis

During this thesis, the Situated Cognitive Engineering method will be used for the design and testing of the virtual assistant. This is an iterative process, a visual representation of which is given in Figure 2.1. The first step of the sCE method, presented in this chapter, consist of a domain analysis, which will form the foundation for the requirements baseline. Here, three different aspects are investigated: Operational Demands, Human Factors Knowledge and Envisioned Technology.

After establishing a foundation, the requirements can be specified, along with the corresponding use cases and claims. The use cases contextualise the requirements, while the claims justify them. These are presented in Appendix B.

Finally, an evaluation of the designed system should be performed, which can be done in several different ways. A domain expert or future user could review the requirements and provide comments. Alternatively, a prototype could be built, which can be tested in a Human-in-the-Loop test, which results in an evaluation of the user experience (UX). Finally, the use of the designed system could be simulated, which can then be assessed. Based on the results of these different evaluation methods, the system specification should be refined.

The process of evaluating the system and refining the specification should be performed several times in order to obtain a properly tested specification. In this thesis, only the first iteration is performed, after which recommendations for future research and refinement of the specifications are given.

2.1 Operational Demands

2.1.1 Actors

Various actors are involved in the online use of a BCI to support a severely disabled person. The most important ones are as follows:

- **The User of the BCI.** Typically, the user is severely physically disabled, for example caused by a syndrome such as Amyotrophic Lateral Sclerosis (ALS). Because of diseases like this, he or she is unable to move autonomously and needs daily care from a caretaker. Assistive technologies are required if the user would want to do anything without help from other people.

- **The User’s Caregiver.** The caregiver assists the user on a daily basis. Mostly, he is the user’s spouse or a relative, but he can also be a professional carer. In cases of a
2.1 Operational Demands

Severe physical disability, his assistance will often be required throughout the day. This could be reduced by sharing this work with someone else.

- **The BCI Professional.** Currently, the BCI professional is typically a researcher involved in the development of the BCI system. However, once BCI would eventually become available commercially, this could potentially change to an employee of the company that sells the product. He or she can explain the setup and use of the system to a user and his or her caregiver. Additionally, he can train the user on how best to use the BCI. Ideally, commercially usable BCI systems may even eliminate the need for this type of assistance altogether, removing this actor from the picture.

- **Health Care Professional.** A health care professional can be the user’s doctor. He should be consulted on the effects that introduction of the BCI system can have on the user.

2.1.2 Analysis

The majority of severely disabled people live at home, with either a relative or a professional caregiver supporting them in their daily activities. As a result of the patient’s condition, they are dependent on their caregiver to a large extent. In most cases the carer is a spouse or other relative of the patient. Occasionally, a professional caregiver assists the patient with his activities of daily living. The amount of support required form a carer is related to the severity of the disability or illness. This support can, for example, be help to eat or get dressed etc., but also assistance to move from one location to another. The more severe a condition, the less the patient can do autonomously.

Even if a patient spends the majority of his time at home, this does not mean that he has plenty of time to learn to use a new technology. Typically, he is already involved in all sorts of therapies. As a result, it can be difficult to find enough time to train in the use of a BCI.
However, since the life expectancy with many disabilities is still very high, long-time use of a BCI is a reasonable option. Therefore, even if the BCI does not provide instant success, training to properly use it can still be worthwhile.

During training, it has been shown that the inter-personal relation between a trainer and a trainee can have a large impact on the learning process [32]. They state that a trainer’s warm and empathic attitude can be a powerful method of reinforcement towards a locked-in patient, helping to maintain the patient’s motivation and willingness to put effort into the training.

2.2 Human Factors

While a person may be severely physically disabled, his cognitive abilities can often be unaffected by this. Allain et al. [2] studied the cognitive abilities of two persons who have been living in a locked-in state for several years. Their only means of communication was based on some residual eye movement. In their study, they found that the cognitive abilities of neither of these patients had been diminished since the onset of their locked-in state. A patient’s disability can often be purely physical.

Even if a patient’s disability is purely physical, this will still have a significant effect on his mental well-being, as well as on the health status of his caregiver. Several studies have investigated the effect of disabilities on the quality of life of both groups. Jenkinson et al. [20] investigated the quality of life of ALS patients and their carers. They found that the health status of a patient can have a significant effect on the health of the carer. Not only does this include issues caused by the physical demands on the caregiver, but adverse emotional reactions of a patient also place an emotional demand on the carer. They suggest that any treatment that could reduce the patient’s burden will also have a positive effect on his caregiver. In another study, by Chió et al. [7], relation of an ALS patient’s neurobehavioral symptoms and a carer’s burden and quality of life were investigated. When comparing a patient’s neurobehavioral symptoms at the onset of the illness compared to the present, they found an increase in apathy, which was generally underestimated by the patient. This had a significant impact on the carer’s emotional state and was negatively correlated with their quality of life.

A problem for many patients suffering from illnesses such as ALS, is not only that they can no longer move certain muscles, but also that the residual movement they do have can be more difficult and exhausting to perform. In advanced stages, this is even the case for something as basic as gaze control. For some, it can be very tiring, or even impossible to focus their attention on one point, such as, for example, a computer screen. In this case, tactile or auditory stimuli and feedback can provide a solution, since these do not require the movement of any muscle. Kauhanen et al. [21] have compared tactile feedback with visual feedback and found no difference in training a BCI with either of these methods. This would suggest that haptic feedback is a viable alternative to visual feedback.

2.3 Envisioned Technology

2.3.1 Brain-Computer Interfaces

With EEG, it is possible to measure electrical activity on the user’s scalp. This electrical activity is mainly caused by brain activity, but also by muscle movement. Compared to other techniques to measure brain activity, EEG has several advantages. First of all it is non-invasive, and relatively cheap and portable. This makes it very suited for regular use,
Envisioned Technology Domain Analysis

2.3 Envisioned Technology Domain Analysis

Without any of the risks of the invasive methods. Additionally, EEG has a high temporal resolution, which makes it a good choice when timing or speed is important. There are however also drawbacks to EEG when compared to other techniques. The main drawback is its low spatial resolution, making it harder to determine the exact source of some detected activity. Although using a higher number of electrodes increases the spatial resolution, it will never achieve the same results as some other techniques. Additionally, since EEG measures brain activity at the surface of the user’s skull, it is poor at detecting neural activity below the outer layers of the brain.

By analysing the electrical activity measured with EEG, it is possible to detect certain patterns. These patterns can be actively generated by the user, or his response to perceived stimuli. A BCI will attempt to classify these patterns, making a distinction between different cases. By doing this, it becomes possible to interact with a computer directly through the user’s brain signals.

Passive BCI

One type of passive BCI that can be useful to improve the results of input methods with a low accuracy is based on Error Potentials (ErrP). Whenever the user notices an error in the task he is trying to perform, an ErrP is generated automatically. This can then be classified with accuracies as high as 85% [40]. Even initial research with general classifiers has been conducted, which do not require subject specific training data acquired at the start of each section. Instead, it may be possible to use data collected from a group of subjects to train the classifier for a specific task. This makes a passive BCI based on ErrPs a useful addition to many other BCIs suffering from a low accuracy.

Navigation

The first example of control of a wheelchair used by a tetraplegic is by Leeb et al. [23]. This study used an active BCI based on motor-imagery (MI). The subject traveled along a straight line in a virtual environment and had control over the speed of the wheelchair. His goal was to stop at several avatars among the line for couple of seconds. Another example of using MI to control a vehicle in a virtual environment is by Zhao et al. [45], where a car with behaviour somewhat similar to a wheelchair, is controlled. When using a reactive BCI instead of an active BCI for the control of a wheelchair, only visual stimuli have been used so far. Iturrate et al. [19] used several laser rangefinders to build a reconstruction of the environment, which is then displayed on a screen in front of the user. The user can then focus on one of several targets on the screen, representing the location he wants to move to. After a command has been given, the wheelchair moves fully automated to the position in space represented by the selected target. Although all participants were able to carry out the navigation tasks, the speed at which this happened does need improvement; subjects travelled 40 meters in about 11 minutes using 9 decision steps.

To reduce the limitations caused by the low accuracy and low bitrate of BCIs used for steering, shared control is introduced and investigated in [24, 28, 39]. While the user is still in direct control of the wheelchair, he is aided by some automation in the intelligent system. In [39], context information is combined with the BCI input signal to control a wheelchair, by combining a probability distribution of the possible steering actions with a probability distribution of the BCI commands. The result is a combined estimate of the user’s intent, which is then send as a command to the wheelchair. This has been further tested in [28]. Although shared control did increase performance, further improvements would be necessary for it to be useful outside a lab setting.

At a planning level, reactive BCIs are used more often than active BCIs. In this case, a visual P300 is usually used, showing a matrix with several options to choose from. Bell et al. [5] created a BCI for control of a humanoid robot, where the user is presented with four...
possible actions the robot can take. By basing these actions on segmented images made by the robot’s cameras, the robot could be commanded to pick up or drop off arbitrary objects. Most other P300 BCIs that operate at a planning level are based on use in a familiar environment, such as the system by Rebsamen et al. [34]. Here, a wheelchair automatically moves to a location selected by the user from a matrix. To stop the wheelchair when moving, a separate, faster, BCI is used, which is either a P300 with only one option or an active BCI based on MI. In addition to movement, other studies also add other options for environmental control, such as the lights in a home to the BCI, as in [11], where this is done in a virtual environment.

This research will mainly focus on the usage of the SSSEP BCI developed in Berlin [43] as control mechanism for a navigation task in a virtual environment. Although the basics of this system work, it does have some limitations. First of all, it takes several seconds to give a command to the system. This is caused by the somewhat low single-trial classification accuracy of 65%. In order to achieve a higher accuracy, the stimuli are best presented several times, allowing classification on multiple trials, which can then be combined. As a result of the slow commands, the BCI cannot be used to quickly give a command in situations where timing is important. Secondly, it is not possible to have the system active continuously, since this will result in the user no longer being able to differentiate between the tactile stimuli. Therefore, the right time to present the stimuli has to be determined.

**Personalisation**

Personalising a BCI to a specific user could improve its effectiveness. Mostly, this is done by training classifiers on subject-specific data, which usually leads to an increase in classification accuracy. Another way to personalise a BCI would be to look at other characteristics of the user, such as, for example, age, gender or education. By taking factors such as these into account, it may be possible to further enhance the performance of a BCI or to improve its user-friendliness. However, only a handful of studies on BCI investigate the individual characteristics of the participants. As a result, not much information is available that would allow one to personalise a BCI for a specific user.

The first large scale study investigating how many people are able to use a BCI was conducted in 2003 by Guger et al. [14]. In this study, an active BCI was tested with 99 healthy people. After two training sessions, about 93% of subjects was able to use the BCI with an accuracy above 60%. Unfortunately, no demographic information was gathered. Lately, a couple of studies examining reactive BCIs with a large group of participants have been conducted. In 2009, Guger et al. tested a P300 based BCI with visual stimuli on 100 subjects [13]. Several stimuli were presented in a random order, while the user focussed on the target stimulus. Each time this target stimulus was presented, the elicited P300 Event Related Potential (ERP) is much stronger, which is measurable from an EEG signal. In this study, personal information was gathered. They found that sex, education, working duration, and cigarette and coffee consumption did not affect the subjects’ performance. However, people who slept less than 8 hours the night before the experiment did show a significantly better performance than subjects who slept more. In a study by Allison et al., examining the demographics of people using a SSVEP BCI, a better performance for both young and female participants was found [3]. This difference, however, was not found to be statistically significant when conducting an ANOVA. Their recommendation is to ensure a more equal distribution of age, gender, and other key characteristics, in order to better evaluate its effect. They also argue that a lot more work on BCI demographics should be performed, mainly to find out what causes illiteracy for certain types of BCI and which type would be best for a certain person.
2.3 Envisioned Technology Domain Analysis

**BCI Training**  On first time use of most BCIs, performance is often not that good. However, after training a subject in several sessions, performance can sometimes be increased significantly, mainly in the case of an active BCI. When using a reactive BCI, much less training is required to achieve decent classification results, since the signals used for classification are not generated natively by the user, but in reaction to perceived stimuli. However, as shown by Muller-Putz et al. [29], it may still be possible to enhance the performance of an SSSEP BCI by training the participant. In this case, he or she is not trained to generate specific signals, but instead to focus on a stimulus at the right time.

One important element of training a BCI is proper feedback from the system. Without this, it is impossible for a subject to determine whether or not he is generating an effective EEG signal or focussing on a stimulus at the right time. Ron-Angevin and Diaz-Estrella [35] compared feedback from a virtual reality environment with basic feedback of a bar moving left or right on a computer screen when training an active BCI based on motor-imagery. They found that subjects using the virtual reality feedback achieved a significantly lower error rate than those using the more basic type of feedback. This suggests that the type of feedback given to a subject can influence the effectiveness of a BCI system.

Another factor in training to use BCIs is to gradually introduce the application in small pre-defined steps [32]. This could prevent the user from being overwhelmed by the extensive system.

Several studies have also shown the importance of motivation when training to use a BCI system. A higher motivation leads to increased BCI performance in many cases. Maintaining a high motivation can therefore be an important factor in successful BCI usage. Although motivation can be increased with extrinsic rewards, ideally, this motivation would be intrinsic. Deci et al. [9] have shown that extrinsic rewards will reduce intrinsic motivation.

**Virtual Environments**  The use of BCIs in a virtual environment has potential at two levels, according to Lotte et al. [26]. On one hand, BCI can be used as a new input method in a virtual environment. On the other hand, a virtual environment can be a useful research tool to test a BCI. They identify several challenges, which include that the user should be able to give several different commands to the environment, that these commands can be sent at any time and that there exists an intuitive mapping between the user’s mental state and the command. One more thing that should be taken into account when using a BCI in a virtual environment, according to Groenegress et al. [12] is the user’s sense of presence. In their experiment, a visual P300 BCI was used, located on a different screen than the virtual environment and compared with a more direct method of interaction. They investigated the presence score the users gave, which was significantly lower when using the BCI, since commands could be given without the need to even look at the other screen, but just by looking at a matrix with a set of commands. Therefore, they recommend that the user is forced to make inferences about the environment.

2.3.2  Virtual Assistant

There are many roles a virtual agent can adopt in order to assist the user, depending on the task at hand. For their task and time management assistant, Myers et al. [30] identified three models of assistance. The first of these, is to watch over the user’s shoulder and intervene when necessary. This is mainly applicable when the user has limited problem solving skills, possible due to some cognitive disability. The second model of assistance is to relieve the user of routine task, so his workload does not become too high and he can...
focus on more important tasks. Finally, the third model is of a collaborative agent, assisting the user to accomplish a shared task.

Virtual assistants can come in many different forms, each with their own advantages and disadvantages. One important factor in deciding the form of the agent, is the interaction modality used. Blanson Henkemans described three different modalities and their advantages and disadvantages [6]. First of all, when using text, it is easy to clearly communicate without any ambiguity. However, the interaction will not be fast, especially if there is a large amount of text for the user to read. Secondly, images can also be used to convey information, which can be helpful to explain certain concepts. It is highly dependant on good graphic design though, and if done badly, can have a significant negative impact on user satisfaction. Finally, sound and speech can be used to interact with the user, which is a more natural way of doing this. The disadvantage of using this method, is that it can be time consuming to convey the right information and therefore may not be suitable when timing is critical.

The visual representation of the assistant can have a large impact on the way it is perceived. Looije et al. [25] investigated three different agents; a text interface, a virtual character and a physical character. The latter two had both a social and a non-social version. They found that people preferred both the social characters and the text interface over the non-social characters. For half of the participant the text interface was the preferred assistant, while the other half would rather use the social agent. From this, we can conclude that in order to create an effective, embodied, virtual assistant, it should have social behaviour. If not, a text interface would have been a better solution. The physical characters were found to be more trustworthy, but less empathic than the virtual ones.

Linked to the model of assistance used by the agent, is his feedback style. According to Blanson Henkemans, two different styles can be used to support the interaction between the assistant and the user [6]. On the one hand, the assistant can have a cooperative feedback style. This style is appropriate for an educating and advising agent, which requires a high level of participation for the user. It is more oriented towards user satisfaction and development of long-term skills. Although this feedback method will help the user to learn new competencies and can eventually result in very good performance, it can also be experienced at patronising and become tedious to work with. On the other hand, a directive feedback style can be used. In this case, the assistant will take more decisions on its own, without consulting the user. This style is oriented towards the quick solving of problems and does not require the user to be competent. However, although it can achieve better short-term performance, it is also more likely to make mistakes and will cause the user to experience a loss of control. For some applications, it may be better to change the interaction style of the assistant according to the current situation and needs of the user.

2.4 Conclusion

From the domain analysis presented in this chapter, we can derive the following conclusions:

- The system should reduce patient’s burden, thereby also reducing caregiver’s burden.

- In order to reduce the load on the visual channel, the system should provide tactile and/or auditory feedback and stimuli.

- Using shared control based on passive BCI and a virtual assistant can reduce the limitations caused by a BCI’s low accuracy and bitrate.
The system should not require knowledge about the environment other than what can be determined in real time.

Forcing user to make inferences about the virtual environment will increase his presence.

The user should be relieved of as many tasks as possible and the system should collaborate with the user when input is required.

The virtual assistant does not have to be embodied.

The virtual assistant should have a directive feedback style, taking more decisions on its own, without consulting the user.

2.5 Specification and Evaluation

Based on the conclusions presented in Section 2.4, we can conclude that the virtual assistant and passive BCIs should assist the user in navigating with his wheelchair by automating most tasks. Whenever a change in direction is possible, the assistant will first attempt to decide where to go on its own, based on knowledge about the environment and input from a passive BCI for intention detection. When additional information is required, the assistant will stop the movement of the wheelchair and ask the user which direction to go in using a reactive BCI.

To further enhance the reliability of the BCIs used, a second passive BCI for error detection is also used by the assistant. Whenever either the user’s intention or his input using the reactive BCI is incorrectly classified, the assistant will be notified of this automatically. The agent can then stop the wheelchair and ask for additional input. The extended functionality of the error detection BCI is indicated by $^1$ in Table 2.1.

A more detailed specification of the system is presented in appendix B. This specification consists of use cases, requirements and claims, conforming to the sCE method.

The evaluation of parts of the designed system are presented in the next two parts of this thesis. First, in Part II, the first steps to create a passive BCI for intention detection are taken. Next, Part III examines the use of a partial implementation of the navigation paradigm.
### Navigation

**Passive BCI**

<table>
<thead>
<tr>
<th>Navigation State</th>
<th>Possible Controls</th>
<th>Intention Detection</th>
<th>Error Detection</th>
<th>Direct Input</th>
<th>Observe Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driving</td>
<td>Brake</td>
<td>Change course detected → Brake</td>
<td>Change course not detected → Brake</td>
<td>Detect blockade → Brake</td>
<td></td>
</tr>
<tr>
<td>braking</td>
<td></td>
<td></td>
<td>Change course erroneously detected → Allow Drive Straight</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stationary</td>
<td>Drive Straight</td>
<td>Direct input misclassified → Ask for new input</td>
<td>Choose direction → Drive</td>
<td>Detect blockade → Limit available controls</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Drive Left</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Drive Right</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.1: A schematic overview of the extended navigation paradigm.

1. Compared to the paradigm presented in table 1.1, the error detection BCI is also used to recover from errors made on classification of the reactive BCI.
Part II

Experiment 1: Passive BCI for Intention Detection
As specified in Chapter 1, this part of this thesis will attempt to answer research questions 1 and 2:

1. *How can a passive BCI be used to estimate the user’s intention to change course?*

2. *What is the accuracy of the passive BCI’s estimation?*

To answer these questions, an experiment is conducted where EEG data, of a participant watching automatic movement through a virtual environment, is being collected and analysed. Using this data, an attempt will be made to create a passive BCI to detect the user’s intention to make a change in direction.

To our knowledge, this has not been attempted before, making this approach a new direction in the field of passive BCIs.

This part of the document is set up as follows. First, Chapter 4 will describe the method used. Then, the results will be presented in Chapter 5, which will be discussed in Chapter 6. Finally, Chapter 7 will present the conclusion.
Chapter 4

Method

The goal of this experiment is to find out whether it is possible to predict the user’s intention to change direction while navigating, using a passive BCI. In order to do this, participants will watch automatic movement through a virtual environment, regularly encountering intersections. At each intersection, several spheres are shown which can be collected when moving over them. Since the goal is to collect multiple spheres, these serve as an indication of the direction to move in. While watching this movement on a computer monitor, EEG data is being collected. This data is later classified offline, in order to discriminate between two different situations:

1. A sphere is presented straight ahead of the participant, while no spheres are presented towards the left or right.
2. A sphere is presented on the left and/or the right of the participant, while no sphere is presented straight ahead.

If these two different situations can be identified with a BCI, we would gain some knowledge on the intention of a participant, which could then be used to partially automate steering tasks.

Initially only an offline analysis is performed, allowing the collection of data without it being influenced by feedback from the system. The collected data will then be analysed using various different algorithms.

4.1 Participants

Although the eventual BCI system created is aimed at people suffering from severe physical disabilities, due to time limitations this study uses 8 able-bodied participants.

4.2 Stimuli

Participants will have a tactor attached to the palm of each hand. This tactor is made from a loudspeaker and vibrations are produced by playing an audio file on it. These tactors will simulate the use of the reactive BCI system that will eventually be used in the online system. This should make the experiment conditions more similar to the intended online use, reducing the chance of a decreased performance.

Apart from the tactile stimuli, the most important stimuli are visual. Each participant will watch a 27 inch Asus VE278H computer monitor running the virtual environment at
a resolution of 1440 x 900. In this environment, yellow spheres will indicate the direction a user is assumed to want to move in. Additionally, a sound effect is played whenever one of the spheres has been collected.

4.3 Initial Experimental Task and Procedure

The experimental task of this research merely consist of watching a computer monitor, while the user moves through the virtual environment automatically, as shown in Figure 4.1. The user is not able to control the movement through the environment himself. Although it will be harder to keep subjects interested when they are not required to take any action, the choice for automatic movement was made to prevent artefacts in the EEG signal caused by muscle movement.

Figure 4.1: A typical scene of the virtual environment, where the subject is approaching an intersection. The sphere to be collected can be seen on the left, indicating the desired direction.

Since automatic movement through the environment will be quite boring, a game element is introduced in order to keep the participant interested to a certain extent. Similar to the 1980 game Pacman, the goal is to collect all the yellow spheres that are placed all around the map, one every few meters. Because of this, it is likely that the participant would also have a desire to move in the direction where there are still spheres left to collect, even though he is not able to control the movement himself. When approaching an intersection for the first time, there will be spheres in every direction, in which case a participant should be indifferent about the direction he moves in. However, when visiting the same intersection again, some of the spheres will have been collected already, leaving spheres in only a few directions, similar to the situation in Figure 4.1. Naturally, the subject should then want to move in the direction where there are still objects left to collect (left in the case of Figure 4.1).
Alternatives to the spheres for the indication of direction have also been considered, such as an arrow at the centre of the screen. The main advantage of using an arrow is the fact that the user’s sight will be focussed on a single point. This should limit the amount of eye movement required by the subject, thereby limiting the artifacts in the EEG from these movements. However, by using such a method, where the user focusses on a single point on the screen, the user will no longer be forced to look at the environment and make inferences about it, but can simply stay focussed on the appearance of an arrow at the centre of the screen. As suggested by [12], this can lead to a lower feeling presence, making the experiment more different from one set in a real world environment. Additionally, in a real world setting, eye movements will also occur and can hardly be prevented. Creating a passive BCI that can handle these artifacts, will therefore make the BCI more usable in a real world environment.

Movement through the environment is done randomly. When arriving at an intersection for the first time, all directions have an equal chance of being chosen. However, each time a path is chosen, the chance to visit it again will be reduced. By doing this, it is more likely to move in a direction in which there are still spheres left to collect. As a result, there is a high chance to have an efficient path through the environment, while simulating the errors introduced by control with a BCI.

When the decision to turn at the next intersection is made by the system, the camera will slowly come to a halt there and the tactors on the palm of the participants hands will be activated, to simulate the reactive BCI used in the next experiment. If, on the other hand, a decision to go straight is made, a constant speed is maintained and the camera will not stop, nor will the tactors be activated. This, again, simulates the workings of the eventual online system controlled by a BCI.

Participants will sit through several relatively basic maps, while data is collected for each of these maps. These maps are kept small in order to limit the time it takes to collect all spheres. Ideally, it would take at most 10 minutes to collect all spheres in an entire map. These maps do not have an equal distance between all intersections.

### 4.4 Modifications to the Experimental Task and Procedure

After a small initial pilot test without the collection of EEG data, one major issue issue with the method described in Section 4.3 was identified. The initial setup turned out to be too time consuming, not generating enough trials where a clear indication of the direction would be present (about 1 useful trial per minute). Several changes have been made to prevent this problem.

The first change made to prevent this problem, is in the design of the map of the virtual environment. While initially the plan was to use several different maps, with shorter and longer stretches of road between the intersections, a large grid-like map is now used all the time. This map has an equal distance between all intersections and with the participant always starting in the centre. Additionally, the intersections are now all very close together and as a result will be encountered in a much higher frequency.

The second change is to limit the number of spheres that appear. At each of the four directions of a regular intersection, the spheres each have a chance of only 0.4 to appear. Because of this, at a large number of intersections there will already be a clear direction to move in when approaching it for the first time, since only one or two spheres will appear in total, instead of one in every direction.

Since the larger map size and randomised placements of the spheres could result in a huge decrease in the frequency of useful trials towards the end of the experiment, when
only a few spheres are still left to be picked up, another change was made. In the new
setup, only a predetermined amount of spheres, 25 in this case, have to be collected, instead
of collecting all spheres on the entire map.

Another change was required due to the reduction of the distance between intersec-
tions. As a result of this, the spheres close to and in-between the other intersections were
already visible when approaching an intersection, making it much harder to control the
presentation of the stimuli and the direction a person would want to move in. Therefore,
all spheres between intersections have been removed and the others, placed around the
intersections, are now only visible when the participant is close to an intersection. This
change has the added advantage of presenting all relevant stimuli at the same time, since
the spheres no longer come into view at the moment they enter the view distance. In case of
the latter, a sphere at the left or right would come into view before a sphere straight ahead,
which is now no longer the case. The alternative, more realistic, solution to this issue, re-
ducing the view distance significantly, was also considered. However, this would result in
a tiny view distance, making it difficult for participants to build any situation awareness.

Since it is no longer necessary to collect all spheres, it is also no longer necessary to visit
the entire map. Therefore, a final change was made in the automated movement, to allow
for a more equal distribution of trials. In the newer version, there is a chance of 0.5 to go
straight and an chance of 0.25 to go either to the left or to the right. This does no longer
change when revisiting an intersection.

In total, the experiment lasted for about 90 minutes. The first 30 minutes were used
to set up the EEG and explain the experiment to the participants. Then, offline data was
collected over the course of an hour, after which the experiment was finished.

4.5 Recording

To record the EEG data, 64 active scalp electrodes are placed on a Brain Products actiCAP,
according to the standard extended international 10-20 system for 128 electrodes. This is
represented by the green and yellow electrodes in figure 4.2. Although for BCI purposes,
using 32 electrodes could already be sufficient, using 64 electrodes is preferred since this
will also allow us to perform an Independent Component Analysis (ICA) if desired, which
can be used to gain additional knowledge of what is happening in the user’s brain. The
data from the electrodes is amplified using a Brain Products BrainAmp amplifier and sent
to a recording computer in another room, where it is combined with markers sent from the
virtual environment using lab streaming layer (LSL).

4.6 Virtual Environment Design

The virtual environment is written in Python, using the Panda3D gaming engine. The main
advantage of this engine is that it allows the use of the Simulation and Neuroscience Appli-
cation Platform (SNAP) framework, which has been specifically designed to simplify the
setup of an experiment with Panda3D. SNAP contains several helper functions that allow
users to quickly create very basic experiments, although most of this has limited usefulness
in the virtual environment used here. More importantly, SNAP also provides a connection
with lab streaming layer (LSL), a system for the collection and recording of data in research
experiments. LSL also allows the collection of EEG data and can return the result of the
Matlab analysis of this data to the SNAP framework. By using these frameworks, it is very
simple to integrate the EEG hardware into the virtual environment both for the measure-
4.6 Virtual Environment Design

The virtual environment itself is, in its current form, a rather basic environment. It consists of two main classes, the World and the UserAgent, and several smaller python files such as the FileReader (which reads the map from a txt file). The World class is responsible for setting up the environment and loading the 3D models. The UserAgent contains the navigation logic and provides methods for the World class to call at each frame.
The environment used in the experiment is built by reading a map from a simple txt file. Here, each road section is indicated by an ‘x’ symbol, while each building is indicated by a ‘0’ symbol. Additionally, the starting position and direction can be indicated on this map. By loading the environment in this way, the map can quickly and easily be changed by reading it from a different file. Although it is no longer required to change the map during the experiment, this method is maintained for flexibility in future experiments.
The results presented in this section are for the offline classification between two classes:

1. There is a sphere present either on the left or on the right (or both). No sphere is present straight ahead.

2. There is only a sphere present straight ahead. No spheres are present to the left or right.

A sphere on the left side is treated the same as a sphere on the right side, since we are only interested in whether or not the user wants to make a turn. Which direction to turn in can later be established using a different BCI, such as one based on SSSEP, or other input method. This will reduce the complexity of the classification from three classes to two classes.

5.1 Feature Extraction and Classification

The classification of the EEG data was initially done using the Spectrally Weighted Common Spatial Pattern (Spec-CSP) BCI paradigm, developed for the Berlin BCI [37]. This paradigm is a generalised version of the Common Spatial Pattern (CSP) paradigm [33]. The CSP approach creates a decomposition of the EEG data into a spatial pattern that maximises the variance between two classes. Compared to the CSP paradigm, Spec-CSP uses a non-homogeneous spectral filter in addition to the Spatial Patterns. After extracting features from the EEG signal, a Linear Discriminant Analysis (LDA) classifier is trained and used for offline classification.

In the initial classification approach, a time window for the Spec-CSP paradigm is determined automatically, based on the best performance. As start time for this window, either 0 or 0.5 seconds after the appearance of the spheres is used. The end time of the window can be either 1.0, 1.5 or 2.0 seconds after stimulus presentation. As a result, this can be different for each subject, allowing for a more personalised BCI paradigm. This approach does, however, make the classification less generalizable. The classification result are presented in Table 5.1. On average, an accuracy of 64.1% was achieved. Here, only one pattern pair for the Spec-CSP paradigm is used, since this resulted in, on average, a slightly better performance than using more pairs.

However, partly due to the automatic detection of the optimal time window, for some subjects classification is based on eye movement artefacts, as can be seen in figure 5.1, showing the pattern pair used for classification for subject 5. In the case where spheres are presented on the left or right, more lateral eye movement occurs than in the case where
5.2 Event Related Activity

To get a better understanding of what is happening in the subjects’ brain when navigating the environment, we can look at the presence of an Event-Related Potential (ERP). This is an electrophysical response to the occurrence of an event, which can be detected using EEG. In this case, this event is the appearance of the spheres when approaching an intersection. Once we know more about a possible ERP, we can use this to improve the classification accuracy by choosing an appropriate paradigm and setting the time window correctly to classify on this ERP. This could result in better classification and less focus on the eye-movement artefacts sometimes used now. As an added benefit of setting the time window based on an ERP, this can be the same for all subjects. Therefore, we no longer need to search for the optimal time window for each subject.

Figure 5.2 shows the average EEG at PO8 (back of the head, right side), across all subjects after stimulus presentation. Additional images can be found in Appendix C. Here,
5.2 Event Related Activity

Figure 5.2: ERP at electrode position PO8. The green line represents class 1, while the blue line represents class 2. The pink line indicates the difference between classes. This ERP is different between classes, as can be seen by the pink line. A similar effect can be found on the left side of the brain, in the area around PO7 (back of the head, left side).

Another way to investigate for an ERP, is by plotting the Event-Related Spectral Perturbation (ERSP) and the Inter-Trial Coherence (ITC) [10]. The ERSP represent the mean of the power spectrum in an electrode in relation to an event, measured in dB. The ITC is a measure of the event-related synchronisation in the frequency domain. By using these techniques we may be able to find an event-related response of the brain that would not be visible in a simple ERP, which may not be visible if an ERP is not stable across trials or if it is not completely independent of the EEG [27].

Figure 5.3 and Figure 5.4 show the ERSP and ITC in a couple electrode positions for subjects 4 and 8 respectively. Additional images can be found in Appendix C. In these images, Condition 1 represents the first condition presented at the start of this chapter, with only spheres at the left or right. Condition 2 represents the case where there is only a sphere straight ahead. The third column shows the difference between the two conditions, with the area of interest marked by a black ellipsis. From these figures we can see some event related activity, indicated by the black ellipsis, which could potentially be used for classifications. Although several other subjects show similar activity, not all of them do.

By choosing a time window from 300ms to 1000ms, based on the ERSP and ITC found, and using the regular CSP paradigm, we can achieve the classification results as shown in
Table 5.2. For the feature extraction, two pattern pairs are used for all subjects. Although the mean classification accuracy is the same compared to using an automatically determined time window, there is a large difference in accuracy for several subjects. Mainly subjects 5, 6 and 8 showed a large improvement.
5.2 Event Related Activity

Results

Figure 5.3: ITC and ERSP of subject 4. Condition 1 represents the condition where only spheres at the left or right are present. Condition 2 represents the case where there is only a sphere straight ahead. The third column shows the difference between the two conditions, with the area of interest marked by a black ellipsis.
Figure 5.4: ITC and ERSP of subject 8. Condition 1 represents the condition where only spheres at the left or right are present. Condition 2 represents the case where there is only a sphere straight ahead. The third column shows the difference between the two conditions, with the area of interest marked by a black ellipsis.
### 5.2 Event Related Activity Results

<table>
<thead>
<tr>
<th>Subject number</th>
<th>Classification Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.66</td>
</tr>
<tr>
<td>2</td>
<td>0.59</td>
</tr>
<tr>
<td>3</td>
<td>0.56</td>
</tr>
<tr>
<td>4</td>
<td>0.60</td>
</tr>
<tr>
<td>5</td>
<td>0.70</td>
</tr>
<tr>
<td>6</td>
<td>0.64</td>
</tr>
<tr>
<td>7</td>
<td>0.63</td>
</tr>
<tr>
<td>8</td>
<td>0.79</td>
</tr>
<tr>
<td>mean</td>
<td>0.646</td>
</tr>
</tbody>
</table>

Table 5.2: Offline Classification Accuracy based on the CSP paradigm with a time window from 300ms to 1000ms. For all subjects, two pattern pairs were used in the feature extraction.
Chapter 6

Discussion

6.1 Offline Classification

In the previous Chapter, the results have been presented based on two different methods. First, a time window was found automatically for each subjects. The second method used a similar time window for all subjects, based on visually inspecting the ERPs for each class, and their difference. In both cases the mean classification accuracy was around 64-65%, with the latter showing the best performance for a single subject (79%).

When looking at the patterns found by the CSP and Spec-CSP approaches, we can see that for some subjects this also includes eye movement artefacts, an example of which was shown in figure 5.1. This can be explained by looking at the experiment setup. Instead of giving the subjects a single fixation point at the centre of the screen, the stimuli (spheres), were present around intersections. Although it is more realistic to have a subject look around in his environment to some extent, this also caused more lateral eye movement in case spheres were present to the left or right of the intersection. As a result, at least for some subjects, eye movement was an important feature for the classification.

However, when investigating the ERP, ERSP and ITC in more detail for each subject, we can see activity in the area around PO8 as well as in the area around PO7 for some of them. This suggests that there is more brain activity that can be detected with EEG and used to predict a user’s intention than just eye movement artefacts.

When basing our classification on this ERP, we can also see a large difference in performance between subjects. While the best subjects achieved a classification accuracy of 79%, several others did not even achieve anything above 60%.

One explanation for the differences between subjects could be due to the design of the experiment. Here, the participants had a passive role where they simply watched the movement through a virtual environment. Although this approach minimised the effects of EEG artefacts caused by muscle movement, it may also have caused some subjects to lose attention. Additionally, their intrinsic motivation was likely relatively low due to this.

One possible method of increasing the user’s motivation and attention, is to give him more control of the movement through the environment. After stopping at each intersection, the participant should then indicate which direction they want to go by pressing a key on the keyboard. This could introduce artefacts due to muscle movement. However, the negative effects of this could possibly be reduced by instructing participants only to press a key after stopping at an intersection. This would then occur several seconds after the stimuli have been presented and should not overlap with the classification window.

Another method to keep subjects more involved in the experiment is to use an online
6.2 Eye blinks

During the initial analysis, it was found that the best classification results are achieved on different time windows for each subject and even, to some extent, between trials of a single subject. Because of this, a method of finding an appropriate time window, based on eye-blinks, has been investigated as well. The rationale behind this, is that while someone is paying attention and waiting for certain information to present itself (in this case the spheres), the interval between blinks will be longer [4, 36]. After detection of the stimuli this blink interval decreases again. This is expected to result in an eye blinks after perception of the stimuli.

By picking a time-window relative to an eye blink, it was expected that the classification results could be improved. This was done using an automatic eye-blink detection algorithm for offline processing of the EEG data, finding the first blink after stimulus presentation. Then, based on these new time-windows, ERPs have been investigated and classification has been attempted.

Unfortunately, this did not lead to any usable results, for two different reasons. First of all, in a large number of trials there was no eye blink present within the first 2 seconds after stimulus presentation. Classifying after more than 1.5-2 seconds after stimulus presentation is not an option, since at that time the subjects can notice that he is either going to stop at an intersection or not, introducing new stimuli. Therefore, this did not only reduce the number of trials available for training and testing of the classifier, but also made it impossible to find a correct time window on which to classify for the remaining trials.

A second issue with the approach mentioned above is caused by the algorithm used to detect the eye-blinks. The automatic detection was not accurate enough to use reliably, both missing eye-blinks and identifying false positives. Although it is possible to detect these blinks from the EEG data [15], it would have been better to measure Electrooculography (EOG) into the experiment as well. With this, it would have been much easier to detect eye blinks. Another, also more reliable approach for this, would be to use a camera focussed on the subject’s eyes.

6.3 Subjects

This experiment used healthy participants, instead of envisioned, severely disabled, users. Although there are a large amount of BCI studies that use healthy participants, the number of studies with disabled subjects is a lot smaller. An even smaller number compares the performance of able-bodied subjects with that of disabled subjects, with mixed results. One of these studies comparing both types of participants was performed by Hoffmann et al. [18], using a P300-based BCI. Although they found an equal maximum classification accuracy, the maximum achieved bit-rate was higher for able bodied subjects. Another study by Hill et al. [17] compared both types of subjects for a motor-imagery (MI) based BCI without subject training. In this study, it was found that able-bodied subjects were capable of using the BCI, while the completely locked-in patients did not produce any suitable signal. Kübler et al. [22] however, did obtain positive results with disabled subjects using a MI BCI, albeit after several months of training. Therefore, although performing the experiment with able-bodied subjects is the fastest method to get the basics of the BCI working, further tests
with disabled users is recommended to determine whether or not they can achieve similar results.

Additionally, at what accuracy the performance would be perceived as acceptable for use in a real world environment is also dependant on the user. An able-bodied user, for example, will have plenty of alternatives for navigation that will provide an accuracy of almost 100%. As a result, under normal conditions these alternatives will most often be preferred. However, once a user is severely physically disabled, unreliable control methods such as a BCI may be one of the only options left. In such cases, a lower accuracy may be accepted. Part III of this thesis will investigate this in more detail.
Conclusion and Recommendations

The goal for this part of this thesis, was to answer the following research questions:

1. How can a passive BCI be used to estimate the user’s intention to change course?

2. What is the accuracy of the passive BCI’s estimation?

From the results presented in Chapter 5, we can only partially answer these questions. Two approaches have been examined. First, the time window on which to classify was determined for each subject separately, based on the best classification accuracy achieved with the offline data. This led to a mean classification accuracy of 64%. Next, a more extensive investigation of the subjects’ brain activity was performed, in an attempt to find an ERP on which to correctly classify, which led to a classification accuracy of 65%. Although not significantly better, this result was achieved by using the same time window for all subjects, making the BCI more similar between subjects and requiring less subject specific calibration. Even though the average accuracy is above chance level, it is not high enough to be used online and in a real world environment.

However, despite not being able to give a definitive answer to the first research question, the ERP that was found in some of the subjects does indicate that there is some event related activity that can be used to create a passive BCI for intention detection. This provides an interesting starting point for future research. Should it be possible to better stimulate this response and improve the feature extraction of it, it may be possible to achieve more reliable results.

Not all subjects have shown an equal accuracy, especially when using a fixed time window. On visual inspection of the ERP, there was not much activity for some subjects. The most likely explanation for this is the passive role of the participants during the experiment, possibly causing some of them to lose attention and motivation. Should further research be conducted into the detection of a user’s intention, it is recommended that the subjects are given a more active role.

Another recommendation for future research would be to attempt to further minimise the artefacts caused by eye movements, for example using another method to indicate the direction to move in. By doing so, it may be possible to better analyse the event related activity that occurs after stimulus presentation and which should be used for classification.

Finally, instead of simply attempting to improve the results obtained by the passive BCI, it should also be investigated how it can still be used without a high classification accuracy. Even if this accuracy improves, it is not expected to achieve perfect results, which is a problem inherent to all BCIs due to the low spatial accuracy of EEG. The next part of this thesis will therefore propose and test a navigation paradigm that uses the passive BCI.
introduced in this part for navigation, while attempting to reduce the limitations caused by it.
Part III

Experiment 2: Virtual Assistant for Online BCI Use
Chapter 8

Virtual Assistant Prototype

A specified in Chapter 1, Part III of this thesis will attempt to answer the third research question:

3. How can BCIs for intention detection, error detection and control selection be combined for an effective, efficient and satisfactory navigation?

The method used for this prototype is presented in Chapter 9. Then, Chapter 10 presents the results of this experiment, followed by a discussion in Chapter 11. Finally, this part is concluded in Chapter 12.

8.1 Prototype Functionality

The prototype of the virtual assistant and navigation paradigm tested in this experiment, only has a part of the functionality of the paradigm designed and described in Chapter 2. This was done in order to keep the paradigm somewhat basic for the experiment and to test the basic functionality. This left us with the basic paradigm as described by Table 1.1. If successful, further experiments could then be performed to test the full functionality of the agent.

In short, the paradigm used in this experiment has the following functionality:

- Automated movement through the environment. The user is only asked for input at an intersection when the assistant cannot determine which direction to go in.

- Accept input from the reactive BCI. The assistant combines several single trial classifications over a few seconds in order to achieve a higher classification accuracy. The amount of time used to collect the single trials will be manually optimised per subject at the beginning of the experiment.

- Simulated intention detection. Since the results from experiment 1 were not good enough to reliably use online, the intention detection used as input for the assistant is simulated instead. This input is still used by the assistant to reason determine what input options to offer the user after stopping at an intersection.

- Error detection on the intention detection BCI. An Error Detection BCI is also simulated, in order to detect errors made by the intention detection BCI. Other errors, such as those made by the reactive BCI, are ignored.

- The assistant provides visual feedback to the user, both on the classification results of the reactive BCI, as well as when an error is detected.
8.2 Technical Design

Figure 8.1: The most important classes in the design of the virtual assistant.

Figure 8.1 presents the design of the main classes of the system. The most important class of the assistant is the UserAgent. This class controls the navigation, determining when to stop and when/where to move to. Additionally, it gets its beliefs about the environment from the World class. Since this is a virtual environment, it is simple to have complete and accurate knowledge about it. However, should the assistant be implemented in a real-world environment, this knowledge would be less certain, based on which methods are used to establish beliefs about the environment (GPS, cameras, etc.). Although in some situations it can be useful to take the uncertainty of the data collection into account, the UserAgent assumes that this is accurate, since this depends significantly on the approach used to gather this information.

Apart from the environment, the assistant can also get information from the BCIs. This consists of three different classes:

- **ReactiveBCI** This BCI is used to allow the user to provide binary input using a BCI based on SSSEP. It can enable the tactile stimuli and make a classification every x ms. In the case of this experiment, x is set to 100ms. After several seconds, the assistant can then request the overall result of this classification, which is an average of multiple single trial classifications.

- **IntentionDetectionBCI** This BCI can give an estimation of the user’s intention to make a change in direction. When approaching an intersection, it can be consulted by the assistant in order to determine the desired course of action.

- **ErrorDetectionBCI** This BCI can alert the assistant if an error has been detected. The assistant can then take the appropriate action, depending on its current state and beliefs.

In the case of this experiment, only the Reactive BCI has been used online. The other two BCIs have been simulated. However, if desired and technically feasible, these could easily be converted into an online BCI without requiring any large changes to the UserAgent.

Depending on the beliefs about the environment and the input from the various BCIs, the assistant can enter several states. Which state the assistant is in determines its behaviour and response to various inputs.
Finally, the assistant is able to provide feedback to the user. In this case, this is implemented using the FeedbackGUI. This class controls the display of text and progress during the reactive BCI classification on the bottom of the screen, giving feedback on the classification results.
Virtual Assistant Evaluation

During this experiment, participants will move through a virtual environment, regularly encountering intersections. At each intersection, several spheres are shown which can be collected when moving over them. Since the goal is to collect multiple spheres, these serve as an indication of the direction to move in. Using a reactive BCI, the user can give binary input on which direction to move in. Two conditions will be compared:

1. Only the reactive BCI is used for navigation, to indicate which way to go at each intersection. Moving forwards and stopping is done automatically.

2. Apart from the reactive BCI’s input, the assistant now uses information from two passive BCIs that will support in the navigation. These BCIs will attempt to estimate the user’s intention and detect errors in the intention detection.

The comparison of these two conditions should answer research question 3, how BCIs for intention detection, error detection and control selection can be combined for an effective, efficient and satisfactory navigation. The following results are expected, when comparing the first and second condition:

- In the second condition, a higher performance in the execution of the navigation task should be present, making it more effective and efficient.

- In the second condition, a higher user satisfaction will be found.

9.1 Participants

Although the eventual BCI system created is aimed at people suffering from severe physical disabilities, due to time limitations this study used two able-bodied participants. Although using subjects with a severe physical disability does have its advantages, an initial experiment with able-bodied users can be performed significantly faster and should not have too much of an impact on the results, as explained in Section 6.3.

9.2 Stimuli

Participants will have a tactor attached to the palm of each hand. This tactor is made from a loudspeaker and vibrations are produced by playing an audio file on it. These tactors will be activated when user input is required, asking the user to focus on one of the vibrations. This can then be classified using the reactive BCI.
Apart from the tactile stimuli used for user input, the most important stimuli are visual. Each participant will watch a 27 inch Iiyama ProLite B2776HDS computer monitor running the virtual environment at a resolution of 1440 x 900. In this environment, yellow spheres will appear that need to be collected by the participant.

9.3 Experimental Task and Procedure

The experimental task of this research consists of the user moving through the virtual environment as described in Section 4 and shown in figure 9.1. A large part of the movement is done automatically by the system, while the user is only able to use a reactive BCI to give binary input. Whenever the stimuli for the SSSEP BCI are presented, the grey/red bar at the bottom of figure 9.1 will appear. At first, it will be filled halfway, up till the centre. After each single trial classification, the red bar will slightly move towards the left or right. If, after a predetermined amount of time, the red bar is beyond one of the black indicators, on average one of the classes was chosen significantly more, which will be returned to the assistant. If the bar stays about halfway filled, presentation of the stimuli and classification will start over after a short break.

At each intersection, one or more spheres can be collected by the participant, by moving over them. The goal for the participant is to collect 12 spheres. Two different conditions have been compared: one with minimal support for navigation and one with a virtual assistant and passive BCIs aiding in the navigation. Participants used both conditions, half of them started with condition one, while the other half started with condition two. This allows a comparison between the basic condition, without any support, and the designed system, which provides partly automated support for navigation in several ways.

9.3.1 Limited Support (Condition 1)

For condition one, a large amount of movement through the virtual environment is automated. At each intersection, a stop will be made. At this point, the user can indicate where to go using the reactive BCI. This is done in two steps:

1. Indicate whether to go straight at this intersection or make a turn.
2. If the user wants to make a turn, then indicate whether to go left or right.

This process is indicated in the flowchart in Figure 9.2.

9.3.2 Full Support (Condition 2)

For condition two, the paradigm is introduced, which assists in the navigation, thereby automating more of the movement compared to condition one. The assistant can give feedback to the user when input is required or when some of the automatic decisions have been made. Automatic decisions are based both on environmental information, as well as on classification results of two passive BCIs. Since the online implementation of the latter is beyond the scope of this project, this is simulated. However, the participant is not told that this is actually simulated.

The first passive BCI used by the assistant is based on the first part of this research, where a user’s intention to either make a change in direction or continue on the current course is detected. This will be done with an assumed classification accuracy of 65%, which is the same as the mean accuracy found in Part II. If an intention to make a change in direction at the next intersection has been found, then this information is fed back to the
9.3 Experimental Task and Procedure

Virtual Assistant Evaluation

Figure 9.1: A typical scene of the virtual environment, where the subject is standing still at an intersection. The sphere to be collected can be seen on the right, indicating the desired direction. The bottom of the screen shows the feedback from the reactive BCI classification, with the red/grey bar indicating the progress to either the left or to the right.

The assistant, which will then stop at the intersection and ask the user whether to go left or right. If no intention is detected, the assistant will continue to go straight.

The second passive BCI used by the assistant for error detection is based on Error Potentials (ErrP). If the intention to either make a turn or continue on the current course is misclassified, then the user can see error this by either maintaining speed or slowing down, respectively. In this case, an ErrP will occur which can be classified. A classification accuracy of 84% is simulated for this, as found in [40]. If this situation occurs, we know that either the user’s intention was misclassified, or that the ErrP was misclassified. In this case, the assistant will notify the user and ask what to do, similar to Condition 1.

This entire process of the tested system is presented in the flowchart in Figure 9.3. For a more detailed explanation of the complete designed system, see the use cases in Section B.1.

Appendix A explains the expected chances of making correct turns in more detail, comparing this condition with the limited support condition. As seen here, the chance of making a correct decision for the user is higher when providing full support. If we want to go straight, the chance of doing this increases from 65% to 81%. If we want to make a turn, the chance of making the right turn increases from 42% to 53%.

Unfortunately, the increased chances of making a correct decision are still far from perfect. The main cause of this is the low accuracy of the reactive BCI. Should we implement the extended support as described in table 2.1, then we can greatly reduce the amount of errors caused here. Each time an error has been detected on the classification of the reactive BCI, we can stop and ask the user again what to do. This can be repeated until no error is detected anymore. The only time this can then still go wrong, is when both the reactive BCI and the error detection fail, a chance of about 5%. However, repeated errors can result in a long time required to make a decision.
Figure 9.2: A flowchart describing each step in the decision process used in Condition One. A square indicates an action taken by the assistant, while a diamond represents the use of a BCI.
Additionally, as can be seen in table A.1, increasing the accuracy of the reactive BCI and intention detection BCI will also greatly increase the chances of making a correct decision. However, both improvements mentioned above are beyond the scope of this experiment.

9.4 Variables

9.4.1 Independent Variables

The independent variable in this experiment is the full support provided by the assistant, as described above. This is absent in the first condition, but present in the second.

9.4.2 Dependent Variables

The dependent variables that have been measured are as follows:

- **Performance** The performance has been measured automatically, by storing data while the experiment is running. It consists of four different variables:

  - **Navigation Errors Made** At each intersection, only one sphere will appear, which can only be collected by going in the right direction. Therefore, this dependent variable is defined as the ratio of spheres picked up and the total number of intersections encountered.

  - **Decision Speed** The decision speed is defined as the average time it takes to make a decision at an intersection.

  - **Navigation Speed** The navigation speed is the total time it takes to collect the 12 spheres required in the trial.

  - **Reactive BCI Error Rate** The Reactive BCI Error Rate is defined as the number of errors made on classifying the reactive BCI.

- **Perceived Performance** Even though the performance can be higher in one condition than in the other, the user may not experience it that way. The perceived performance will be measured with several open questions at an interview at the end of the experiment. Additionally, this is one of the rating scales in the NASA TLX questionnaire.

- **Task Load** The task load relates to the mental effort required to navigate using the BCI. Although Condition one requires more input from the user than Condition two, the actions required from the user are the same at every turn. The latter is not the case for Condition two. The task load was measured using the NASA TLX questionnaire.

- **Satisfaction** To measure satisfaction, the user was asked which condition they prefer to use, and why in the interview at the end of the experiment.

- **Trust** In order for the automated system of Condition two to be used, it is important that the user trusts the system enough. This has been measured with several open questions in the interview at the end of the experiment.
Figure 9.3: A flowchart describing each step in the decision process used in the full support condition. A square indicates an action taken by the assistant, while a diamond represents the use of a BCI.
9.5 Hypotheses

From the claims in Section B.3, we can establish the following hypotheses to test in this experiment:

1. The number of wrong turns taken is decreased when using passive BCIs for intention and error detection. (Claims C002 and C005)

2. The average time it takes to make a decision at an intersection is decreased when using passive BCIs for intention and error detection. (Claim C004)

3. The total time it takes to collect 12 spheres is lower when using passive BCIs for intention and error detection. (Claim C001)

4. The perceived performance is increased when using passive BCIs for intention and error detection. (Claims C001, C002, C004 and C005)

5. The task load is decreased when using passive BCIs for intention and error detection. (Claim C007)

6. The overall satisfaction of the system is increased when using passive BCIs for intention and error detection.

9.6 Measurement

During the experiment, the performance is measured in several different ways. The number of wrong turns made is being recorded, along with the Decision Speed and Navigation Speed. Additionally, the error rate of the SSSEP classification is being measured. This should allow the testing of hypotheses 1, 2 and 3.

Although theoretically, we know that condition two should provide a more efficient method of navigation, as seen in Appendix A, the user will have less direct control over the actions taken. Additionally, an ErrP can be wrongfully detected when the correct decision has been made. Such a false positive will cause the assistant to ask the user what to do, while it was already doing the right thing or when the user would not expect it. To investigate the effect of providing additional support and automation on the user experience, a questionnaire is performed afterwards. Since fatigue when using the system can be a significant issue for a severely disabled users, as well as frustration due to errors made, the NASA TLX questionnaire is used to compare the task load required for navigation with the basic system of condition one to the system with added support of condition two. This should allow the confirmation or rejection of hypothesis 4.

At the end of the experiment, after the participant has used both Condition one and Condition two, a structured interview is conducted to get more qualitative information on the user’s trust and satisfaction, as well as on perceived performance and overall usability. These two final methods should provide more information on hypotheses 5 and 6, as well as give information on the user’s trust in the system.

In this case, only qualitative research into the UX is was conducted, using 2 participants. This should be sufficient to identify most serious flaws in the system. Should certain trends emerge from this, further (quantitative) research could then be conducted into those trends in a further study with disabled users.

In total, the experiment takes about 2.5-3 hours. The first 20 minutes are used to set up the EEG, after which training data for the reactive BCI can be collected. The time this data collection takes depends on the subjects, but around 30 minutes can be expected. After
a classifier has been trained, the parameters for the reactive BCI, such as bias or stimulus presentation time, can be tweaked, which will again take about 30 minutes. Next, the experiment can be conducted, taking just under an hour. Finally, an interview is conducted and the NASA TLX questionnaire is filled in in roughly 20 minutes.

9.7 Recording

To record the EEG data, the same hardware and frameworks are being used as in the previous experiment, as explained in Section 4.5. However, contrary to the previous experiment, only 32 electrodes are used instead of 64, since this would be sufficient for the reactive BCI. Additionally, the EEG data is not only used for classification of the reactive BCI, but is also recorded for offline analysis of the user's intention.
Chapter 10

Results

In total, the experiment has been performed with five participants. However, for the first three experiments, the reactive BCI did not work online. As a result, the full experiment could only be performed with the last two participants. Both of them were male Human Factors students, aged 26 and 27, with some basic knowledge about BCIs. The results of this are described in this chapter.

10.1 Performance Measures

Tables 10.1 through 10.4 present the performance data that has been collected per subject. Table 10.1 shows a decrease in the navigation errors made of about a third for both subjects. The speed with which decisions were made, along with the total navigation speed, as shown in tables 10.2 and 10.3 respectively, is also lower in Condition two compared to Condition one. Additionally, both these values are also lower for subject two than for subject one. This is partly caused by the lower stimulus presentation time used for subject two, which was 3 seconds compared to the 3.8 seconds used for subject one.

Finally, Table 10.4 shows the classification results of the reactive BCI. For participant 1, four sets of training data were collected (consisting of 40 samples each), and ultimately, the second set was used to train the model for online classification. For participant 2, six sets were collected, from which the fifth set was used online. In the online use, an error in the data collection for participant 1 caused the online error rate to be unavailable. This error was corrected when the experiment was performed with the second subject.

<table>
<thead>
<tr>
<th></th>
<th>Condition 1 (no support)</th>
<th>Condition 2 (support)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject 1</td>
<td>0.4</td>
<td>0.59</td>
</tr>
<tr>
<td>Subject 2</td>
<td>0.42</td>
<td>0.63</td>
</tr>
<tr>
<td>Mean</td>
<td>0.41</td>
<td>0.61</td>
</tr>
</tbody>
</table>

Table 10.1: Navigation Errors Made. Spheres collected (12) / intersections encountered.

10.2 Task Load

To measure the task load, the NASA TLX questionnaire has been filled in by each participant [16]. Table 10.5 presents the results of the questionnaire. The sub scales are defined as follows:
Table 10.2: Decision Speed. The average time it takes to make a decision at an intersection in seconds.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Condition 1 (no support)</th>
<th>Condition 2 (support)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.87s</td>
<td>4.55s</td>
</tr>
<tr>
<td>2</td>
<td>3.05s</td>
<td>2.37s</td>
</tr>
</tbody>
</table>

Table 10.3: Navigation Speed. To total time in seconds it takes to collect all 12 spheres.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Offline</th>
<th>Condition 1 (no support)</th>
<th>Condition 2 (support)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.35</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>2</td>
<td>0.40</td>
<td>0.49</td>
<td>0.41</td>
</tr>
</tbody>
</table>

Table 10.4: Reactive BCI Error Rate. The online and offline error rates of the reactive BCI. Online data is only available for subject 2.

- **Mental Demand** How much mental and perceptual activity was required (e.g., thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?

- **Physical Demand** How much physical activity was required (e.g., pushing, pulling, turning, controlling, activating, etc.)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?

- **Temporal Demand** How much time pressure did you feel due to the rate or pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?

- **Performance** How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?

- **Frustration level** How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

- **Effort** How hard did you have to work (mentally and physically) to accomplish your level of performance?

Since some subscales, such as physical or temporal demand, are less relevant to this task than others, only the weighted scores are reported. In total, these subscales add up to an overall score between 0 and 100. The range of the individual values depend on the subject specific weights.

Table 10.5 shows that while the added support of the virtual assistant decreases the average effort experienced by the users, the mental demand and frustration stays roughly the same. Both the physical and the temporal demand are almost zero after weighting them, as they did not play a large role in the experiment. Finally, the performance is rated slightly lower in condition two. Overall, the NASA TLX score decreased by about a quarter.
Due to the low sample size, these figures are not statistically significant. Combined with the answers from the interview, they do however provide some valuable insight in the users’ experience, as discussed in Chapter 11.

<table>
<thead>
<tr>
<th></th>
<th>Condition 1</th>
<th></th>
<th>Condition 2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Subject 1</td>
<td>Subject 2</td>
<td>Mean</td>
<td>Subject 1</td>
</tr>
<tr>
<td>Mental Demand</td>
<td>26.67</td>
<td>11.8</td>
<td>19.24</td>
<td>28.33</td>
</tr>
<tr>
<td>Physical Demand</td>
<td>1.2</td>
<td>0</td>
<td>0.6</td>
<td>0.87</td>
</tr>
<tr>
<td>Temporal Demand</td>
<td>0</td>
<td>0.07</td>
<td>0.04</td>
<td>0</td>
</tr>
<tr>
<td>Performance</td>
<td>11.07</td>
<td>24.67</td>
<td>17.87</td>
<td>9.87</td>
</tr>
<tr>
<td>Effort</td>
<td>14.6</td>
<td>16.8</td>
<td>15.7</td>
<td>6</td>
</tr>
<tr>
<td>Frustration</td>
<td>15.73</td>
<td>17</td>
<td>16.37</td>
<td>14.13</td>
</tr>
<tr>
<td>NASA TLX Score</td>
<td>69.27</td>
<td>70.34</td>
<td>69.81</td>
<td>59.2</td>
</tr>
</tbody>
</table>

Table 10.5: The NASA TLX scores for each participant across the two conditions.

10.3 Interview Results

Both participants mentioned that it seemed easy for them to focus on one of the vibrations when collecting training data for the reactive BCI, but found it a lot more difficult to use it online. The main cause of this is the low classification accuracy when using the BCI online. Additionally, the long time the training took was also a cause of this, mainly for the second participant.

10.3.1 Satisfaction

Both participant preferred Condition two, with the virtual assistant, over Condition one, without the assistant. The main reason they gave for this was the improved performance.

Additionally, both participants found Condition two at least somewhat pleasant to use, mainly because it more often works like it is supposed to (making the right turns), less input is required and the overall accuracy is higher. Alternatively, when asked if the Condition one is pleasant to use, only one of the participant gave a positive answer. This was mostly because he liked the idea of controlling the navigation with a BCI. Both participants where highly motivated, but reported frustration because the reactive BCI has a low reliability, making it difficult to control the movement and increasing the amount of errors made.

When asked if they were satisfied with the working of the two different conditions as a control mechanism for a navigation task, neither condition was rated satisfactory. However, both participants noted that they were more satisfied with Condition two than with Condition one, albeit still not satisfied enough.

10.3.2 Trust

On the subject of trust, participant one reported that he trusted the system with the virtual assistant less than the system without the support, even though it made less errors. This is due to the fact that in Condition one, the user could give all commands. In Condition two however, it was sometimes not possible to choose the right direction, because the virtual assistant made an error in the classification of the passive BCIs and a decision was made without asking the user the right question.
Participant two mentioned that in order to trust either system, he would want to see statistics or other proof that it works well enough. Additionally, in the case of the virtual assistant, he would want to see more errors detected and automated decisions only being made when there is a really high certainty that this is the correct decision.

In both cases, the main point of improvement required was the reliability of the system. Finally, subject two mentioned that for either system it has to be failure safe, so no bad things could happen on a misclassification, in order to trust it enough to use it.

### 10.3.3 Perceived Performance

The performance of the reactive BCI was judged to be similar for both conditions, which is to be expected since the same model and settings are used for classification. As a whole though, the number of navigation errors made in Condition two were judged to be lower, since the automated system often made the correct decision for you. Additionally, this was also judged to be faster, since automatic decisions were made without requiring user input.

### 10.3.4 Additional remarks

One of the participants mentioned that it was harder to go left than right when two questions are asked, due to the nature of the binary input of the reactive BCI. In the case of a right turn, the participant could first focus on his right hand to indicate that he wants to make a turn, and then focus on his right hand again to indicate to go right. When going left however, his focus has to shift from his right hand, to indicate a turn, to his left hand, to indicate the turn direction.

Additionally, one subject mentioned some confusion when an error was detected by the virtual assistant in the case he wanted to make a turn. Then, instead of directly indicating the direction to go in, he would first need to answer the additional question on whether he wants to make a turn or not. This required a larger amount of focus on reading the questions presented to the user, since these were not always the same in every situation.
Chapter 11

Discussion

During the experiment, it was found that in its current form, the training and setup of the reactive BCI is not as streamlined as it should be for easy online use. This is not only caused because it suffers from similar issues as other BCIs, but also by issues specific to this type of BCI. First of all, it is difficult for participants to focus on one of the two vibrations and as a result reasonable results were only achieved after a longer training session. Secondly, the time window for the model on which the best classification results are achieved, is not the same for each subject nor for each training session for one subject. As a result, finding the right time window can be a time consuming process. Finally, the calibration of the BCI settings parameters, such as the stimulus presentation time, added some additional time to the experiment duration. These factors combined resulted in a longer setup time of the BCI than anticipated in the experimental design, possibly causing some additional fatigue and frustration for the subject as well.

Although online use of the reactive BCI does introduce some issues, it does have a different effect on the user’s perception of it. When comparing online use to simulating the BCI, the user has less or no control over its performance in the latter. The same effect is likely present in the simulation of the two passive BCIs, for both intention and error detection. Since these were simulated in the experiment, this could have had an effect on the user’s perception of the automation provided by the assistant.

After the BCI setup, the main experiment could be conducted in order to find an answer to the six hypotheses:

1. The number of wrong turns taken is decreased when using passive BCIs for intention and error detection.

As expected from the calculations presented in Appendix A, there does indeed seem to be a trend towards a lower number of wrong turns when assistance is provided by the virtual assistant. Although the intention detection on its own would only present a minor increase in the BCI performance, when this is combined with the classification of ErrPs, which has a much higher classification accuracy, the difference between the two conditions is increased. This suggests that the estimation of the user’s intention, combined with error correction, does indeed provide a higher performance than navigation using just the reactive BCI.

2. The average time it takes to make a decision at an intersection is decreased when using passive BCIs for intention and error detection.

As expected from the fact that a lower amount of questions is asked from the user, there does indeed seem to be a trend towards a lower average time required to make a decision when automating parts of the movements using a virtual assistant based on passive BCIs.
Discussion

3. The total time it takes to collect 12 spheres is lower when using passive BCIs for intention and error detection.

The results described in Section 10 show a trend towards a lower total time required to collect the 12 spheres when the proposed system supports the user in his navigation compared to the basic condition using only the reactive BCI to control the movement. This result is not surprising, considering hypotheses 1 and 2 seem to be accepted as well. A lower time required to make a decision, combined with a lower error rate, should indeed speed up the overall navigation.

4. The perceived performance is increased when using passive BCIs for intention and error detection.

When looking at the perceived performance of the reactive BCI, no change was reported by the participants. However, both participants did mention that they noticed the increase in correct turns made after the introduction of the passive BCIs.

When looking at the subjects’ own performance rating in the NASA TLX questionnaire however, a decrease in the performance rating can be observed. A possible cause for this is the lesser control the subjects had, reducing their influence on the performance. As such, they rated their own performance lower, even though the overall performance increased.

5. The task load is decreased when using passive BCIs for intention and error detection.

From the result of the NASA TLX questionnaire, it would seem that the overall task load is perceived lower in Condition two compared to Condition one. In this case, this is mainly caused by the lower effort experienced by the users.

Unlike the expectations, the mental demand does not seem to decrease when the virtual assistant is introduced. In both conditions, this is one of the larger contributions to the TLX score. A likely explanation for this high value is the mental demand required by the reactive BCI. Although less input is required when the virtual assistant is added, some new factors may be introduced that have an effect on the mental effort. One of these is the error detection, which caused different questions to be asked of the user. In the control task using the reactive BCI, the same two questions were always asked, unless this would not make sense based on the environment (when encountering a T-junction for example). However, when the virtual assistant is added, the question wether to make a turn or go straight ahead is often omitted and only asked if an error has been detected. As mentioned by one of the participants, this required him to pay more attention to what question he is supposed to answer when user input is required. Therefore, making the error detection more obvious to the user, with for example auditive feedback, could improve the assistant and lower the mental demand.

The level of frustration is rated the same in both conditions, which can also be explained by looking at the interview results. As mentioned there, a main cause for frustration is the low accuracy of the reactive BCI and the long training required to use it. Although this is partly solved by introducing the virtual assistant, the passive BCIs also introduce some new issues. It was mentioned that it is frustrating when a wrong decision is made and there is no chance for the user to correct this. So although the better performance does not seem to reduce the frustration experienced by the user due to some added complexity, it does not seem to increase either.

Finally, the extremely low ratings for the physical demand seems promising when the system would be applied for usage by a severely disabled user, who is unable to perform any physically demanding tasks.
6. **The overall satisfaction of the system is increased when using passive BCIs for intention and error detection.**

The overall satisfaction with the system was only measured from the interview conducted at the end of the experiment. Although both participants preferred the inclusion of the virtual assistant based on passive BCIs over the control using only the reactive BCI, neither of these was rated satisfactory. The main reason for this is the low accuracy of the reactive BCI, which was not considered reliable enough.

This could potentially be improved by using the BCI for multiple sessions, after which participants may be better able to focus on one of the vibrations in such a way that this can be classified. Another possible improvement could be the implementation of the entire navigation support system specified in Section B. By doing this, error detection can also be applied on the classification of the reactive BCI. A wrong classification can then be corrected either automatically by the system, or by asking the relevant question again. This could in turn have an additional negative effect when a correct decision is reverted.

The interview also showed a low trust in the use of the reactive BCI as a control mechanism in this task. This trust was still low when the virtual assistant was introduced. One participant mentioned the loss of control in case a wrong decision is made for the user as a reason for this. Due to this, he was not able to correct some of the errors. Another participant mentioned his uncertainty about the performance of the reactive BCI as a reason for the low trust.
Chapter 12

Conclusion and Recommendations

The goal of the second experiment presented in this thesis, was to answer the third research question:

3. How can BCIs for intention detection, error detection and control selection be combined for an effective, efficient and satisfactory navigation?

To answer this question, we have to look at the results presented in Chapter 11 in more detail. Although the users preferred the system with the added support from passive BCIs and the virtual assistant over the basic system using only the reactive BCI, there are some shortcomings to the current paradigm that should be improved upon. Users are still not satisfied enough with the system, nor do they trust it enough to use it outside a lab setting. This is mainly caused by the low accuracy of the BCIs used and the mistakes made because of this while navigating. This is a problem inherent to the use of a BCI in a task such as this, as well as in many other tasks. Although this is partly solved by the virtual assistant and error detection BCI, it is not yet sufficient for use in a real world environment.

Additionally, although the proposed system did not decrease the mental demand required, it did seem to reduce the effort required by users to navigate effectively. Overall, there seems to be a trend towards a lower task load.

From this we can conclude that the navigation paradigm appears to provide some improvement in usability when using a BCI for a navigation task. However, further improvement is required before it could be used in a real world environment.

Despite providing some improvements over a basic control method using only a BCI, there is room for improvement for the paradigm. One possible way to improve the usability of the system is by implementation of the extended paradigm presented in Appendix B. This would include, among other things, error correction for the reactive BCI, reducing the errors made here. Together with the other functions described there, a much more complete system for navigation should be available. However, this could also introduce new problems, which should be investigated in a future experiment.

Some changes in the experiment setup are also recommended, should this system be investigated further. First of all, the subjects that used the BCI in his experiment were all healthy, with no need for any assistive technologies. As a result, they are used to input methods with a much higher accuracy than a BCI. Therefore, it would be valuable to repeat the experiment with subjects with a severe physical disability. It is expected that mainly the satisfaction rating of the system would be affected by this, but it could have an effect on other aspects of the usability as well.

Secondly, if a new experiment would be conducted with a larger sample size, the trends that this research show could be investigated further and validated.
Thirdly, if it is possible to use the passive BCI for intention detection online, another change in the experiment setup could be made. Then, instead of tasking the user with picking up a sphere at each intersection he encounters, he could instead be tasked to travel from one place to another. This setup would be closer to navigation in the real world, making the user care more about a wrong turn and thereby also making the experiment results more reliable. However, in this case it would no longer be possible to simulate the two passive BCIs. Since multiple routes can lead to the same goal, we can no longer determine what the user wants at each intersection. As a result, these need to be used online.

Finally, streamlining the training and parameter adjustment for the reactive BCI could speed up the first phase of the experiment, allowing more time to investigate the usage of the virtual assistant and causing less initial fatigue and frustration for participants.

In conclusion, this research has shown how a virtual assistant, in combination with passive BCIs, could improve the effectiveness, efficiency and satisfaction of a BCI system when used for a navigation task, although there is definitely need for further improvement.
Part IV

Conclusion and Recommendations
Chapter 13

Recommendations to Enhance BCI Based Navigation

Of the many assistive technologies available, BCIs can be one of the most difficult ones to use properly. This is caused by several factors, including their low accuracy, low bitrate and high required mental effort. However, for some people they may be among the last available methods of controlling devices, making the improvement of their functionality and performance an important area for research. Therefore, during this Master’s Thesis, the goal was to answer the following research questions:

1. **How can a passive BCI be used to estimate the user’s intention to change course?**
2. **What is the accuracy of the passive BCI’s estimation?**
3. **How can BCIs for intention detection, error detection and control selection be combined for an effective, efficient and satisfactory navigation?**

These questions have been answered in two steps. First, the investigation of the passive BCI for intention detection when navigating, presented in Part II. Here it was found that this could be detected with a low classification accuracy, partly based on eye movement artefacts. However, an ERP has also been found in several subjects, which should be investigated for further improvements.

Next, Part III used the passive BCI for the detection of intention in combination with another passive BCI for error detection and a reactive BCI, based on SSSEP, for binary input, along with a virtual assistant to improve the navigation through a virtual environment. It was shown here that although some improvements were made when compared to navigation using only a reactive BCI, further improvement is required before it is feasible to be used online in a real world environment.

Therefore, we can conclude that it is, to some extent, possible to enhance BCI based navigation in a virtual environment using the proposed paradigm. This can be done both by estimating a user’s intention to make a change in direction, as well as by providing additional support using a virtual assistant.

However, further investigation is recommended. Not only should the two experiments described here be further examined in order to verify the trends visible here and improve on their results, but improving the reactive BCI used is also of great importance. In its current state, the SSSEP BCI used is not the most accurate in providing a binary input, partly due to restrictions in its design. Improving the BCIs performance would decrease the errors that need to be resolved using techniques such as described in this thesis, making these techniques more reliable in the process.
Additionally, the proposed navigation paradigm should also be tested in a real world environment. Here, the user might also be occupied by other tasks, which could influence the performance of the passive BCI for intention detection. Although other types of passive BCI have been shown to work in more realistic environments, as described in Section 2, the effect on the passive BCI investigated here is unknown. However, since this research was aimed at users with a severe physical disability, simply performing a single task can already be difficult. The amount of time spent doing other tasks is therefore also expected to be limited, but its effects can however not be ignored.

Finally, at what accuracy the overall performance would be perceived as acceptable for use in a real world environment is also dependant on the user and should be investigated further. An able-bodied user, for example, will have plenty of alternatives for navigation that will provide an accuracy of almost 100%. As a result, under normal conditions these alternatives will most often be preferred. However, once a user is severely physically disabled, unreliable control methods such as a BCI may be one of the only options left. In such cases, a lower accuracy may be accepted. This should be determined by repeating a modified version of the experiment presented in Part III, with severely disabled users. In this experiment, the only independent variable should be the simulated classification accuracy of the BCIs. Different accuracies can then be compared, in order to find the minimum required accuracies for use of the navigation paradigm in a real world environment.
Appendix A

Expected Navigation Accuracy

In order to decide on the best method of support the assistant provides, it is important to know its effect on the effectiveness of the system. To some extent, this can be calculated without testing the system. This appendix compares the expected probabilities of taking certain paths for the two conditions described in Section 9.3. However, this does not take into account the effect of the user experience on the performance of the BCIs, which could have an impact on the performance.

The performance of the system is highly dependant on the accuracy of the BCIs used. In the examples presented here, the following classification accuracies are used:

- SSSEP: 65%
- Intention Detection: 65%
- Error Detection: 85%

Here, the accuracies for both the intention detection and the error detection are the same as used in the experiment described in Chapter 9. For the SSSEP BCI, the single trial offline accuracy as described in [43] is used.

Two different cases are compared. First, Figures A.1 and A.2 describe the expected outcomes when the user wants to go straight. Secondly, Figures A.3 and A.4 describe the expected outcomes when the user wants to make a left turn. In both cases, it is assumed that we are approaching a regular intersection, not a T-junction. Only in Figures A.2 and A.4 is the virtual assistant active. The green boxes indicate the desired action, while the red boxes indicate the wrong states the user could end up in. The probabilities presented next to the arrows indicate the chance to make a certain decision, while the probabilities inside the blocks indicate the chance to encounter that state when starting from the beginning of the flowchart.

As seen in Figures A.1 through A.4, the chance of making a correct decision for the user is higher when providing additional support using the virtual assistant and passive BCIs. If we want to go straight, the chance of doing this increases from 65% to 81%. If we want to make a turn, the chance of making the right turn increases from 42% to 53%.

It could be argued, that instead of asking the user what to do when an error has been detected, to simply take the opposite action as was detected by the intention detection BCI. Such a system is described in Figure A.5. Apart from the effects this has on the user experience, this also has some effect on the expected outcome. Whether this is better or worse, however, depends mostly on the classification accuracies used. Table A.1 presents the effect of choosing various different accuracies for the BCIs. Here we can see, that when
the accuracy of the reactive BCI and intention detection BCI increases, the overall chance of
making the right decision increases faster when we do ask the user what to do after an ErrP
has been detected. Although in the settings used in the experiment it would be better not to
ask the user, when the accuracies increase to only 70% the difference is already negligible.
For higher accuracies, the system tested in this research is superior.

In the examples presented in this appendix, it would be better to use the system as
presented in Figure A.5 instead of the support tested in the experiment. However, the ac-
curacies used for the reactive BCI and intention detection are quite low. Both of these are
expected to potentially increase. For the reactive BCI, this is attempted in the experiment
by not using the single trial classification accuracy, but instead use the average of multiple
classifications taken over a period of several seconds. Additionally, when the classification
is uncertain, the user can be asked to provide input again. Further more, several other pos-
sible improvements are presented in [43]. Improvements to the intention detection BCI may
be possible as suggested in Chapter 6. Because of this, the tested system can be superior to
the system presented in Figure A.5.

Should we implement the extended support as described in table 2.1, then we can
greatly reduce the amount of errors caused by the reactive BCI. Each time an error has
been detected on the classification of the reactive BCI, we can stop and ask the user again
what to do. This can be repeated until no error is detected anymore. The only time this can
then still go wrong, is when both the reactive BCI and the error detection fail, a chance of
about 5%. However, repeated errors can result in a long time required to make a decision.
This situation is illustrated by the bottom row of table A.1.
Figure A.1: The expected outcomes when the user wants to go straight at the upcoming intersection. The support provided is based on the limited support condition of the experiment described in Part III. In this case, navigation is done without the help of the virtual assistant. A square indicates an action taken by the assistant, while a diamond represents the use of a BCI.
Expected Navigation Accuracy

Figure A.2: The expected outcomes when the user wants to go straight at the upcoming intersection. The support provided is based on full support condition of the experiment described in Part III. In this case, navigation is done with the help of the virtual assistant. A square indicates an action taken by the assistant, while a diamond represent the use of a BCI.
Expected Navigation Accuracy

Figure A.3: The expected outcomes when the user wants to make a left turn at the upcoming intersection. The support provided is based on limited support condition of the experiment described in Part III. In this case, navigation is done without the help of the virtual assistant. A square indicates an action taken by the assistant, while a diamond represent the use of a BCI.
Expected Navigation Accuracy

Figure A.4: The expected outcomes when the user wants to make a left turn at the upcoming intersection. The support provided is based on full support condition of the experiment described in Part III. In this case, navigation is done with the help of the virtual assistant. A square indicates an action taken by the assistant, while a diamond represent the use of a BCI.
Figure A.5: An alternative method of support for the virtual assistant. Instead of asking the user what to do after an ErrP has been detected, the opposite action of the result from the intention detection BCI is executed. A square indicates an action taken by the assistant, while a diamond represent the use of a BCI.
<table>
<thead>
<tr>
<th>BCI Accuracies</th>
<th>Reactive</th>
<th>Intention</th>
<th>Error</th>
<th>User wants to go straight</th>
<th>Reactive</th>
<th>No Reactive</th>
<th>BCI on error</th>
<th>No BCI on error</th>
<th>User wants to make a turn</th>
<th>Reactive</th>
<th>No Reactive</th>
<th>BCI on error</th>
<th>No BCI on error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>60%</td>
<td>60%</td>
<td>85%</td>
<td></td>
<td>77%</td>
<td>85%</td>
<td></td>
<td>46%</td>
<td>51%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>65%</td>
<td>65%</td>
<td>85%</td>
<td></td>
<td>81%</td>
<td>85%</td>
<td></td>
<td>53%</td>
<td>55%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>70%</td>
<td>70%</td>
<td>85%</td>
<td></td>
<td>85%</td>
<td>85%</td>
<td></td>
<td>59%</td>
<td>60%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>75%</td>
<td>75%</td>
<td>85%</td>
<td></td>
<td>88%</td>
<td>85%</td>
<td></td>
<td>66%</td>
<td>64%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>80%</td>
<td>80%</td>
<td>85%</td>
<td></td>
<td>91%</td>
<td>85%</td>
<td></td>
<td>73%</td>
<td>68%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>95%</td>
<td>65%</td>
<td>85%</td>
<td></td>
<td>93%</td>
<td>85%</td>
<td></td>
<td>88%</td>
<td>81%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table A.1: The percentage of correct turns taken for different BCI accuracies. Here, two cases are compared. In one case, the reactive BCI is used to ask the user what to do after an error has occurred. In the other case, no reactive BCI is used to ask the user what to do after an error.
Appendix B

Specification

After establishing a foundation, the requirements baseline can be specified. This consist of three parts, presented here: Use cases, requirements and claims.

B.1 Use Cases

UC001: Intention to change course detected
Assist the user in navigating by stopping at the next intersection and to ask whether he wants to turn left or right.

Pre Conditions
• Approaching an intersection.

Post Conditions
• Turned left or right at the intersection.

Action Sequences
Action sequence:
1. The user approaches an intersection and wants to make a left turn there.
2. The Passive BCI detects this intention and sends this information to the virtual assistant.
3. The virtual assistant detects that it is possible to go either left or right at this next intersection.
4. The virtual assistant stops at the intersection.
5. The assistant enables the stimuli for the SSSEP BCI to determine whether the user wants to go left or right.
6. A signal to go left is classified and this information is sent to the assistant.
7. The assistant starts moving again, making a left turn.
Action sequence:

1. The user approaches an intersection, with only a road straight and to the left, and wants to make a left turn there.
2. The Passive BCI detects this intention and sends this information to the virtual assistant.
3. The virtual assistant detects that it is only possible to go left or straight.
4. The virtual assistant turns left at the intersection without stopping.

Requirements
R001, R002, R012
UC002: Intention to stay on course detected

Assist the user in navigating by moving straight ahead at the next intersection, without requiring any input from the user.

Pre Conditions

• Approaching an intersection

Post Conditions

• Went straight ahead, crossing the intersection

Action Sequences

Action sequence:

1. The user approaches an intersection and wants to go straight there.

2. The Passive BCI detects this intention and sends this information to the virtual assistant.

3. The virtual assistant detects that it is possible to go straight.

4. The virtual assistant continues moving forward, without any action required by the user.

Requirements

R001, R002, R012
UC003: T-junction ahead
Assist the user by stopping at the T-junction ahead and activating the SSSEP BCI to ask him whether to go left or right.

Pre Conditions
- Approaching a T-junction.

Post Conditions
- Left or right turn made at the T-junction.

Action Sequences
Action sequence:
1. The virtual assistant detects a t-junction ahead, where it is possible to go either left or right.
2. The virtual assistant stops at the junction.
3. The assistant enables the stimuli for the SSSEP BCI to determine whether the user wants to go left or right.
4. A signal to go left is classified and this information is sent to the assistant.
5. The assistant starts moving again, making a left turn.

Requirements
R012, R013
UC004: Error in SSSEP classification

Pre Conditions
- A stop has been made at an intersection, after which the user can indicate to go either left or right using the SSSEP BCI.

Post Conditions
- A turn in the desired direction has been made.

Action Sequences
Action sequence:
1. The user approaches an intersection and wants to make a left turn there.
2. The Passive BCI detects this intention and sends this information to the virtual assistant.
3. The virtual assistant detects that it is possible to go either left or right at this next intersection.
4. The virtual assistant stops at the intersection.
5. The assistant enables the stimuli for the SSSEP BCI to determine whether the user wants to go left or right.
6. A signal to go right is classified and this information is sent to the assistant.
7. The assistant starts moving again, making a right turn.
8. An ErrP is generated automatically by the user and this is correctly classified.
9. The virtual assistant notifies the user of the detected error and asks the user again what to do.

Requirements
R004, R006
UC005: Error in Intention Detection - False Positive Intention

Pre Conditions
- When approaching an intersection, an intention to make a turn has been wrongfully detected. An ErrP is generated and correctly classified.

Post Conditions
- The user will continue to go straight ahead.

Action Sequences
Action sequence:
1. The user approaches an intersection and wants to go straight ahead there.
2. The Passive BCI wrongfully detects an intention to make a turn and sends this information to the virtual assistant.
3. The virtual assistant detects that it is possible to go either left or right at this next intersection.
4. The virtual assistant stops at the intersection.
5. While stopping, the user notices the mistake and an ErrP is classified.
6. The assistant notifies the user of the detected error, stops, and asks whether or not the user wants to go straight.
7. Using the SSSEP BCI, the user indicates “yes” by focussing on his right hand.
8. This is correctly classified and this information is fed back to the assistant.
9. The virtual assistant now chooses to continue straight ahead instead, in the direction the user wanted to go.

Requirements
R004, R005
UC006: Error in Intention Detection - False Negative Intention

Pre Conditions

- When approaching an intersection, no intention to make a turn has been classified, while the user does have one. An ErrP is generated and correctly classified.

Post Conditions

- A turn in the desired direction has been made.

Action Sequences

Action sequence:

1. The user approaches an intersection and wants to make a left turn there.
2. The Passive BCI wrongfully misses this intention and sends this information to the virtual assistant.
3. The virtual assistant will continue to go straight, crossing the intersection.
4. The user detects that the assistant is not slowing down at the intersection.
5. An ErrP is detected.
6. The assistant can still stop at the intersection and does so.
7. The assistant notifies the user of the detected error, stops, and asks whether or not the user wants to go straight.
8. Using the SSSEP BCI, the user indicates “no” by focusing on his left hand.
9. This is correctly classified and this information is fed back to the assistant.
10. The assistant asks the user if he wants to go left or right and enables the stimuli for the SSSEP BCI again to determine this.
11. A signal to go left is classified and this information is sent to the assistant.
12. The assistant starts moving again, making a left turn.

Action sequence:

1. The user approaches an intersection and wants to make a left turn there.
2. The Passive BCI wrongfully misses this intention and sends this information to the virtual assistant.
3. The virtual assistant will continue to go straight, crossing the intersection.
4. The user detects that the assistant is not slowing down at the intersection.
5. An error potential is detected.
6. The assistant cannot stop at the intersection.
7. After crossing the intersection, the assistant turns around and notifies the user of the detected mistake.
8. The assistant enables the stimuli for the SSSEP BCI to determine whether the user wants to go left or right.

9. A signal to go right is classified and this information is sent to the assistant.

10. The assistant starts moving again, making a right turn.

Requirements
R004, R005
UC007: Intention ErrP Misclassified - False Positive Error

An error in the intention detection classification is falsely detected. This use case describes how that issue will be handled.

Action Sequences

Action sequence:
1. The user approaches an intersection and wants to go straight ahead there.
2. The Passive BCI correctly detects an intention to make go straight sends this information to the virtual assistant.
3. The virtual assistant continues to go straight at the intersection.
4. While stopping, the an ErrP is wrongfully classified.
5. The assistant notifies the user of the detected error, stops, and asks whether or not the user wants to go straight.
6. Using the SSSEP BCI, the user indicates “yes” by focusing on his right hand.
7. This is correctly classified and this information is fed back to the assistant.
8. The virtual assistant now chooses to continue straight ahead instead, in the direction the user wanted to go.

Action sequence:
1. The user approaches an intersection and wants to make a left turn there.
2. The Passive BCI correctly detects this intention and sends this information to the virtual assistant.
3. The virtual assistant detects that it is possible to go either left or right at this next intersection.
4. The virtual assistant stops at the intersection.
5. An ErrP is wrongfully detected.
6. The assistant can still stop at the intersection and does so.
7. The assistant notifies the user of the detected error, stops, and asks whether or not the user wants to go straight.
8. Using the SSSEP BCI, the user indicates “no” by focusing on his left hand.
9. This is correctly classified and this information is fed back to the assistant.
10. The assistant asks the user if he wants to go left or right and enables the stimuli for the SSSEP BCI again to determine this
11. A signal to go left is classified and this information is sent to the assistant.
12. The assistant starts moving again, making a left turn.

Requirements

R004, R005
UC008: SSSEP ErrP Misclassified - False Positive

An error in the SSSEP classification is falsely detected. This use case describes how that issue will be handled.

Pre Conditions

- User has intention to go left
- SSSEP is correctly classified

Post Conditions

- A turn either to the left or to the right has been made.

Action Sequences

Action sequence:

1. After stopping at an intersection, the assistant asks the user if he wants to go left or right.
2. The user indicates a left turn by focussing on the vibration on his left hand.
3. This SSSEP is correctly classified.
4. The assistant starts moving again, making a left turn.
5. An ErrP is incorrectly classified by the assistant.
6. Instead of making the turn, the assistant stops and asks the user whether he wants to go left or right.
7. The user indicates a left turn by focussing on the vibration on his left hand.
8. This SSSEP is correctly classified.
9. The assistant starts moving again, making a left turn.

Action sequence:

1. After stopping at an intersection, the assistant asks the user if he wants to go left or right.
2. The user indicates a left turn by focussing on the vibration on his left hand.
3. This SSSEP is correctly classified.
4. The assistant starts moving again, making a left turn.
5. An ErrP is incorrectly classified by the assistant.
6. Instead of making the turn, the assistant stops and asks the user whether he wants to go left or right.
7. The user indicates a left turn by focussing on the vibration on his left hand.
8. This SSSEP is classified erroneously.
9. The assistant starts moving again, making a right turn instead.
Requirements
R006
B.1 Use Cases Specification

UC009: “Emergency” stop used
An additional (Active) BCI based on MI can be used to indicate that the system has made an error. Alternatively, if the user is able to use it, this active BCI can be replaced by eye-blink detection.

Pre Conditions
- The system is making an error in navigation.

Post Conditions
- The error has been corrected by asking the user several question, answered with the SSSEP BCI.

Action Sequences
Action sequence:
1. When navigating, the system makes an error that is not detected by the error detection BCI.
2. The user sees this mistake and in order to correct itimagines moving his right arm upwards.
3. This is quickly detected by the assistant, who then stops.
4. The assistant now knows that his current course is wrong and asks the user what to do using the SSSEP BCI.
5. The user answers the questions, which are classified correctly.
6. The assistant can now change his behaviour and go into another direction.

Action sequence:
1. When navigating, the system makes an error that is not detected by the error detection BCI.
2. The user sees this mistake and blinks in order to correct it.
3. This is quickly detected by the assistant, who then stops.
4. The assistant now knows that his current course is wrong and asks the user what to do using the SSSEP BCI.
5. The user answers the questions, which are classified correctly.
6. The assistant can now change his behaviour and go into another direction.
UC010: Stimulus presentation

Optimize the number of times the stimuli have to be presented. This is a trade-off between classification accuracy and speed.

Pre Conditions

- The BCI only has a generalised idea on how often to present the stimuli to this particular user.

Post Conditions

- A trade-off between speed and classification accuracy has been found.

Action Sequences

Action sequence:

1. After using the SSSEP BCI for a while, it becomes more difficult for the user to focus on one specific stimulus.

2. More mistakes are being made by the classifier, which is detected by the virtual assistant.

3. After a significant amount of mistakes has been made, the virtual assistant decides to increase the number of times the stimuli are presented before classification is performed.

4. Feedback on this decision is given to the user through both text and speech.

5. Making a turn now takes longer, but is less prone to errors.

Action sequence:

1. After using the SSSEP BCI for a while, the virtual assistant detects that hardly any classification errors are being made.

2. To speed up the navigation, the virtual assistant decides to decrease the number of times stimuli are presented before classification is performed.

3. Feedback on this decision is given to the user through both text and speech.

4. Making a turn now is now faster, but more likely to result in an error.

5. After several new turns, the assistant detects a decrease in classification accuracy.

6. The stimulus presentation is brought back to its old level.

7. Feedback on this decision is given to the user through both text and speech.

Action sequence:

1. After using the SSSEP BCI for a while, the virtual assistant detects that hardly any classification errors are being made.

2. To speed up the navigation, the virtual assistant decides to decrease the number of times stimuli are presented before classification is performed.
3. Feedback on this decision is given to the user through both text and speech.
4. Making a turn now is now faster, but more likely to result in an error.
5. After several new turns, the assistant detects that the classification accuracy remains high.
6. Steps 2-4 are repeated until the classification accuracy starts to decline.

Requirements
R008, R010, R009, R011, R015

UC011: No errors detected, even though they are present
When a classification error is present, but has not been detected, there is no way for the virtual assistant to know this without any additional input methods.

B.2 Requirements

R001: Intention detection using a passive BCI
Parent Requirement
None
Specification
A passive BCI estimates the user’s intention to deviate from his current course. This information is used by the virtual assistant.

Use Cases
UC001, UC002

Claims
C001, C003, C004, C007

R002: Intention to deviate form current course
Parent Requirement
R001
Specification
Intention to deviate from the current course will cause the virtual assistant to stop and enable the SSSEP BCI.
R003: Intention to maintain current course

Parent Requirement
R001

Specification
Intention to maintain current course will cause the virtual assistant to cross the next intersection without any user intervention.

Use Cases
UC002

Claims
C001, C003, C004, C007

R004: Error detection using a passive BCI

Parent Requirement
None

Specification
Classification of error potentials with a passive BCI reduces the amount of mistakes made in navigation.

Use Cases
UC004, UC005, UC006, UC007

Claims
C001, C002, C005
R005: Detect errors in intention detection

**Parent Requirement**
R004

**Specification**
An ErrP can be classified after the user notices a wrong intention is detected. If an ErrP has been found, then the cart will stop and user will be asked if he want to go straight or make a turn.

**Use Cases**
UC005, UC006, UC007

**Claims**
C002, C006

---

R006: Detect errors in SSSEP

**Parent Requirement**
R004

**Specification**
An ErrP can be classified after the user notices a wrong turn is going to be made. If an ErrP is found, the user can be asked which way to turn repeatedly, until no ErrP occurs.

**Use Cases**
UC004, UC008

**Claims**
C002, C005, C006

---

R007: Ask what to do on detection of ErrP

**Parent Requirement**
R004

**Specification**
After an ErrP has been detected, the virtual assistant will ask the user what to do using the reactive BCI.
R008: Behaviour of the virtual assistant is adjusted according to BCI performance

Parent Requirement

None

Specification

Depending on the performance of the reactive BCI, the virtual assistant can adjust its behaviour by changing the level of support provided.

Use Cases

UC010

R009: Increase stimulus presentation

Parent Requirement

R008

Specification

On reduced performance, the number of times tactile stimuli are presented is increased.

Use Cases

UC010

Claims

C005

R010: Decrease stimulus presentation

Parent Requirement

R008

Specification

On increased performance, the number of times tactile stimuli are presented is reduced.
B.2 Requirements

Use Cases
UC010

Claims
C001, C003, C004, C005

R011: Personalize Stimulus Presentation
Parent Requirement
R008

Specification
The optimal number of times to present tactile stimuli for a specific user is remembered to be used as a starting point for each session.

Use Cases
UC010

Claims
C001

R012: Virtual assistant uses basic environmental information to assist in navigation
Parent Requirement
None

Specification
The virtual assistant collects knowledge about the environment and uses this to assist in the navigation.

Use Cases
UC001, UC002, UC003

Claims
C004, C007
R013: T-junction ahead

Parent Requirement

R012

Specification

When approaching a T-junction, this is detected by the assistant and only one relevant question is asked or a turn is made automatically.

Background

If there are only two directions in which the user can go, we can eliminate some options for navigation. For example, if the user cannot go straight, we know he has an intention to make a turn. If a user can only go straight or to the right and an intention to make a turn is detected, it is no longer needed to ask which direction he wants to turn in.

Use Cases

UC003

R014: Virtual Assistant Appearance

Parent Requirement

None

Specification

The assistant is not embodied. It can only provide feedback using text or speech.

Background

There are several forms a virtual assistant can take. If it is embodied, then a social character is required for people to prefer it over a text interface. Additionally, it is important how this embodied assistant looks, to prevent effects such as the uncanny valley. Therefore, in this case the assistant will not be embodied and merely present text and voice feedback.

R015: Virtual Assistant Feedback Style

Parent Requirement

None

Specification

The assistant will provide feedback in both text and speech.
B.3 Claims

Background
Although auditive feedback can be slow, reading text can be difficult for some users.

Use Cases
UC0101

B.3 Claims

C001: Navigation Speed
The duration in which the user moves from his starting point to his destination decreases.

Positive
• Improves navigation speed

Negative
• Reduces direct control for the user

Requirements
R010, R004, R001, R002, R003, R011

C002: Navigation Errors - ErrP
The number of errors made during navigation is reduced by asking the user what to do when an error has been detected.

Positive
• Reduces amount of errors made

Negative
• Reduces direct control
• False positive ErrP can be annoying
• Changes in behaviour can be confusing

Requirements
R007, R005, R006, R004
C003: Duration of Use
By automatically making the right decision for the user and requiring less input, he can use the system for a longer time without getting tired or having difficulty focussing for the SSSEP BCI

Positive
- Improves time for which the system can be used effectively
- Less mental effort required

Negative
- Loss of control

Requirements
R010, R001, R002, R003

C004: Decision Speed
The time it takes to make a decision, with a certain chance of it being correct, is decreased.

Positive
- Faster decision speed

Negative
- Loss of control

Requirements
R010, R001, R002, R003, R012

C005: Navigation Errors - Stimulus Presentation
The number of errors made during navigation is reduced by combining multiple classification results.

Positive
- Reduces the amount of errors made

Negative
- Longer stimulus presentation
- Repeatedly ask the user the same question
**C006: Trust**
Correcting errors made by the BCIs lead to a higher level of trust in the system for the user.

**Positive**
- Higher trust
- User is more inclined to use the system

**Negative**
- Wrongfully correcting errors can lead to a decrease in trust

**Requirements**
- R007, R010, R006, R004, R009

**C007: Task Load**
Less input required leads to a lower task load for the user.

**Positive**
- Less mental effort required

**Negative**
- Automatic decisions may cause a loss of control

**Requirements**
- R001, R002, R003, R012
Appendix C

Results Experiment 1

In this appendix, additional images of the ERP, ERSP and ITC are presented, as discussed in Chapter 5.
Results Experiment 1

Figure C.1: ERP at several electrode positions. The green line represents class 1, while the blue line represents class 2. The pink line indicates the difference between classes.
Results Experiment 1

Figure C.1: ERP at several electrode positions. The green line represents class 1, while the blue line represents class 2. The pink line indicates the difference between classes.
Figure C.2: ITC and ERSP of subject 4. Condition 1 is the first condition presented at the start of this chapter, while Condition 2 represents the second. The third column shows the difference between the two conditions, with the area of interest marked by a black ellipsis.
Figure C.2: ITC and ERSP of subject 4. Condition 1 the first condition presented at the start of this chapter, while Condition 2 represents the second. The third column shows the difference between the two condition, with the area of interest marked by a black ellipsis.
Figure C.2: ITC and ERSP of subject 4. Condition 1 the first condition presented at the start of this chapter, while Condition 2 represents the second. The third column shows the difference between the two condition, with the area of interest marked by a black ellipsis.
Results Experiment 1

Figure C.3: ITC and ERSP of subject 8. Condition 1 the first condition presented at the start of this chapter, while Condition 2 represents the second. The third column shows the difference between the two condition, with the area of interest marked by a black ellipsis.

(a) Subject 8 - Oz

(b) Subject 8 - POz
Figure C.3: ITC and ERSP of subject 8. Condition 1 the first condition presented at the start of this chapter, while Condition 2 represents the second. The third column shows the difference between the two condition, with the area of interest marked by a black ellipsis.
Appendix D

Experimental Procedure Experiment 1

This appendix describes the step-by-step procedure of Part II of the experiment.

1. Welcome the participant.
2. Setup the EEG system in the observation room.
3. Move to the experiment room and hook up/test the EEG system.
4. Attach the tactors to the palm of the participant’s hands.
5. Explain what the participant should do to minimise artifacts in the EEG signal (do not move, etc.).
6. Explain the experiment to the participant.
   a) The goal is to collect 25 spheres per run, 10 runs in total.
   b) All spheres are placed at the start, but can only be seen when getting close to an intersection.
   c) The tactile stimuli are intended to simulate the situation of a later experiment, but are not relevant now.
   d) The subject cannot control the movement himself.
   e) Explain the logic behind the movement (move straight unless other intention detected, etc.).
7. Explain to the participant that they can leave or take a break whenever they want.
8. Close the blinds in the room and switch off the light.
9. Start the first run of the experiment.
10. Run the map (manhattan2.txt) 10 times in total.
11. Thank the participant for his/her help and provide shampoo etc.
Appendix E

Experimental Procedure Experiment 2

This appendix describes the step-by-step procedure of Part III of the experiment.

1. Welcome the participant.

2. Setup the EEG system.

3. Attach the tactors the the palm of the participant’s hands.

4. Explain what the participant should do to minimise artifacts in the EEG signal (do not move, etc.).

5. Explain the offline data collection
   a) A L or R will appear on the screen, indicating which hand to focus on, followed by a fixation cross.
   b) When the fixation cross appears, the factors will start to vibrate.
   c) After 3 seconds, a small ‘twitch’ occurs in the pattern, which will continue for another second afterwards.
   d) Focus mainly on the vibrations after the twitch.

6. Run the offline data collection twice, then train classifier on best set. If needed, more data will be collected.

7. Start with the online use of the SSSEP BCI, in order to personalise its control variables to the user (sensitivity, bias, stimulus duration).

8. Show the online environment and explain the next part of the experiment.
   a) The goal is to collect 20 spheres per run, 2-3 runs per condition.
   b) All spheres can only be seen when getting close to an intersection.
   c) When asked, a command can be given to the system by focussing on one of the hands.
   d) In case of condition 2, the user’s intention is also estimated. However, this is not perfect, so errors can occur. Some errors are detected, after which user input is required.
9. Run the first condition for a shorter time, in order to adjust the SSSEP control variables. This first condition can be either Condition one or Condition two, alternating between subjects.

10. When the variables are set, run the first condition 2-3 times.

11. After the all runs with this condition, ask the participants to fill in NASA TLX

12. Repeat the previous two steps with the alternate condition.

13. Perform the interview described in Appendix F

14. Thank the participant for his/her help and provide shampoo etc.
Appendix F

Interview Experiment 2

This appendix describes the open questions used to guide the interview conducted at the end of the second experiment, after the user has used both Conditions of the system. The interview consists of two parts: general questions about the user and questions specific to this

F.1 General Questions

- Name
- Age
- Gender
- Employed/Student/other
- Education level
  - In case of a higher education, in which field?
- How much do you know about BCI?

F.2 Experiment related questions

1. Did you find it hard to use the SSSEP BCI? If so, what made it difficult?
2. Satisfaction
   a) Which method for BCI navigation do you prefer and why?
   b) Was condition one pleasant to use?
   c) Was condition two pleasant to use?
   d) Are you satisfied with the way condition one worked?
   e) Are you satisfied with the way condition two worked?
3. Trust
   a) Would you trust Condition one enough to use it? Why, or why not?
   b) Was Condition one reliable?
c) If not, what would need to change for you to trust it?
d) Would you trust Condition two enough to use it? Why, or why not?
e) Was Condition two reliable?
f) If not, what would need to change for you to trust it?

4. Perceived Performance
   a) Do you think the number of errors made in Condition one is higher or lower than Condition two, and why?
   b) Do you think the decision speed in Condition one is higher or lower than Condition two, and why?

5. Virtual Assistant
   a) What is your general opinion of the added support?
   b) Did the added support make it harder or easier to control the navigation?
   c) Are there any functionalities you missed in the support?
Bibliography


