BrainBrush

A Multimodal Interactive System for Creative Expressions

Ivo Brugman

August 23, 2012

Master's Thesis
Human Media Interaction
EEMCS Faculty
University of Twente

Graduation Committee:
dr. F. Nijboer (first supervisor)
dr. M. Poel
B.L.A. van de Laar MSc

Abstract

In this Master's thesis, we combined the new developments of multimodal Brain-Computer Interfaces (BCI) and wireless EEG headsets with art by creating BrainBrush, a multimodal interactive system for creative expressions. With BrainBrush, users can paint on a virtual canvas using only their heads. They control the system by moving their heads, blinking their eyes and performing selections using a BCI.

We used BrainBrush to research how the different modalities contribute to the user experience and what the added value of BCI is to the system. Thirteen healthy persons painted with the BrainBrush system. Afterwards, a questionnaire was administered to assess the usability of the system. Furthermore, extensive interviews were conducted to evaluate the user experience of the various input modalities and the added value of BCI.

Most participants who were able to achieve good control over the modalities, were able to express themselves creatively. However, a number of participants was not able to express themselves creatively, mostly due to problems with the reliability of the modalities.

The user experience of the modalities varied. Participants were most positive about the use of head movement. They were also positive about the use of eyeblinks, but less positive about the use of the BCI because of the low reliability and higher relative cost of an error. Recommendations for improvement of the reliability of the modalities have been given.

Even though the reliability of the BCI was low, the BCI was considered to have an added value: the use of BCI was considered to be fun and interesting.

The BrainBrush system offers a good basis for further developments and research. By incorporating the suggested improvements, the user experience should improve further.

Contents

Chapter 1

Introduction

1.1 Motivation

The act of creative expression is considered by many to be a purely human ability and skill [Future BNCI Roadmap, 2012]. Creative expression allows humans to express their identity and it can take multiple forms: for instance making music, dancing, painting, writing or acting. Apart from these more traditional forms of art, new art forms have emerged, such as computer art, motion graphics and the use of virtual reality environments for art.

In recent years, a neurorevolution is taking place in which neurotechnology is becoming a hot topic. For instance, research has been done on improving task performance using neurotechnology. Some athletes now use neurofeedback training to enhance their performance by managing the stress of training and competition [Dupee and Werthner, 2011]. Furthermore, Ros et al. show that microsurgical skills of ophthalmic microsurgeons can be improved significantly with neurofeedback training [Ros et al., 2009]. Neurotechnology has even been applied to the field of economics, creating the science of neuroeconomics: combining neuroscience, economics and psychology to explain the human decision making process [Rustichini, 2009]. In neuromarketing, neurotechnology is used to research why consumers make certain choices: for instance, why consumers prefer either Pepsi or Coca Cola [McClure et al., 2004]

Another example of research in the field of neuroscience are Brain-Computer Interfaces. A Brain-Computer Interface, or BCI, provides a direct interface between a human brain and a computer, without using peripheral nerves or muscles. BCI research has mainly focused on assistive technology for people with disabilities for years. For instance, patients with the motor neuron disease Amyotropic Lateral Sclerosis (ALS) can now benefit from BCIs like the P300 speller [Farwell and Donchin, 1988], brain-controlled wheelchairs [Galán et al., 2008] or BCIs to control their environment [Hoffmann et al., 2008].

In recent years, BCI research also focuses on applications for healthy people. One example is the 'NeuroPhone' system developed by Campbell et al. which allows neural signals to drive mobile phone applications on an iPhone using a wireless EEG headset, for instance to dial a phone number using the same principles as the P300 speller [Campbell et al., 2010].

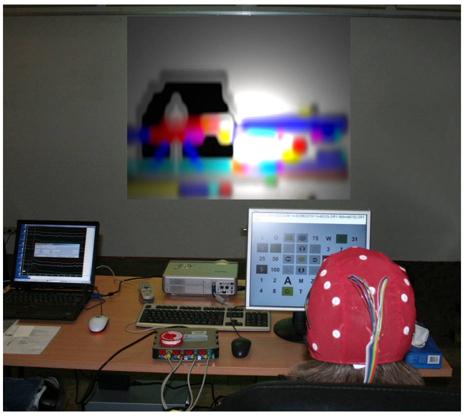


Figure 1.1: Brain Painting setup: users have to look at the screen directly in front of them to use the P300 speller grid while the painting develops on a large screen in the back

Furthermore, research is done on incorporating BCI technology in games. At the Human Media Interaction (HMI) department at the University of Twente, Plass-Oude Bos et al. developed AlphaWoW, incorporating BCI in the game 'World of Warcraft®' [Plass-Oude Bos et al., 2010]. They use alpha waves in the EEG for automatic adaptation of the avatar shape from bear to elf and vice versa. Also at the HMI department, Gürkök et al. used SSVEP for sheep herding in the game 'Mind the Sheep!' [Gürkök et al., 2011].

BCI for art

Brain-Computer Interfaces can also provide a unique link between the source of creativity, the brain, and art. The interactive installation Staalhemel[1], created by Christoph De Boeck, is a responsive environment with 80 steel segments suspended in a room, above the visitor's head. The visitors wear a portable EEG headset and as they walk through the room, tiny hammers are activated by their brainwaves, tapping rhythmical patterns on the steel segments.

Other examples of the use of Brain-Computer Interfaces for art include the 'Brain-Computer-Music Interface' which enables a disabled person to create music [Miranda et al., 2011], and the 'Câmara Neuronal', a performance where the brain signals of the performer are translated, in realtime, into audio and visual compositions[2].

The German artist Adi Hösle, in cooperation with the Institute of Medical Psychology and Behavioural Neurobiology at the University of Tübingen, designed the application 'Brain Painting', a painting application which is controlled using a BCI and enables paralyzed patients to express themselves creatively. In 'Brain Painting', all actions are performed using the P300 paradigm. The system uses two screens for the painter: one screen displays the P300 matrix while another, larger, screen shows the painting canvas, see figure 1.1. The

[1] http://www.staalhemel.com
[2] http://projects.jmartinho.net/3486412/Camara-Neuronal-Video-Teaser

standard P300 speller matrix, containing characters and numbers, was adapted to contain symbols indicating different colors, objects, grid sizes, object sizes, transparency, zoom and cursor movement. By repeatedly making selections using this P300 speller matrix, users can paint pictures on the virtual canvas. In the first evaluation of the 'Brain Painting' application, with 3 ALS-patients and 10 healthy subjects, both the ALS-patients and the healthy subjects were able to use the application with high accuracies: during a copy-painting task, the ALS patients achieved an average copy-painting accuracy of 70.18% while the healthy subjects scored an average accuracy of 80.53% [Münßinger et al., 2010]. One participant in the 'Brain Painting' study, who was severely disabled due to ALS, described her experience with the system: *"I am deeply moved to tears. I have not been able to paint for more than 5 years. Today I again had butterflies in my stomach, a feeling that I have missed for so much, so much* [sic]. *I was so sad, I was plagued by fears of loss, I was in shock because I could not paint. For me the picture I have created is so typical for me, no other paints in my style, and despite five years of absence, I am simply an artist again; I'm back to life!"*. Even though this feedback is very positive, the artistic freedom of the painter is limited due to the fact the cursor can not be moved freely; the cursor can only be moved in a predetermined grid.

In continuation of the 'Brain Painting' research, Holz et al. developed the 'Brain Drawing' application to overcome the cursor movement limitation of the 'Brain Painting' system. In the 'Brain Drawing' application, imagined movement is used to control the cursor when drawing. During the first evaluation with 1 subject, the subject performed a Copy Drawing task in which he was instructed to draw a simple object (circle, ellipse or rectangle) on a virtual canvas. Holz et al. considered 4 out of 36 copy drawings to be succesful by visual inspection and the subject found it very difficult to draw. The subject had to focus his attention for a long time period because he continuously had to imagine movement, which resulted in high workload [Holz et al., 2012].

Furthermore, Todd et al. used two different BCIs in their research on how creativity can be supported and assessed using a BCI [Todd et al., 2012]. With their first BCI, users could only control a drawing cursor in horizontal and vertical directions by looking at one of four LEDs placed on the top, bottom, left and right of the screen. The cursor would move in the direction of the LED the user looked at, and continue drawing in that direction until the user looked at another LED. For the second application, the four LEDs were mapped onto four shapes (circle, star, square and line). After choosing a shape, the shape would be drawn on the canvas. Users did not have any control over the position, size or color of the shapes. Todd et al. concluded relying completely on the efficiency of a BCI for image production is not practical as BCI technology is not yet mature enough for 100% reliability. A possible solution they suggest is to create a hybrid, or multimodal, BCI by combining a BCI with other input modalities such as an eye-tracker.

Barriers for BCI

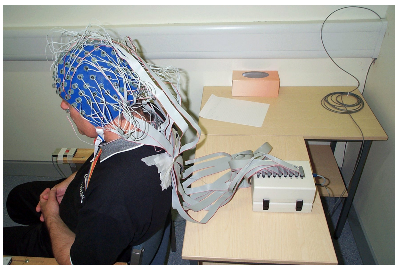
Even though more and more BCI applications exist, there are still a number of barriers BCIs need to overcome to become interesting for the large public.

First of all, the EEG sensors are far from ideal. Traditional EEG systems like the Biosemi ActiveTwo consist of a cap, a lot of wires and up to 256 EEG electrodes (figure 1.2a). Another downside is the fact that conductive gel needs to be used to achieve a good signal quality which leaves residue in the user's hair after using the system. Furthermore, the system costs thousands of euros. The g.SAHARA system produced by g.tec does not require conductive gel, but still needs a lot of wires and a cap covering the user's head (figure 1.2b). In recent years, EEG systems have been introduced which are more aimed at consumers instead of researchers. For instance, Emotiv produced the EPOC headset which gives users more freedom of movement because it is wireless, while it provides 14 EEG channels and features gyrosensors to track head movement (figure 1.2c). Furthermore, the Emotiv EPOC does not require conductive gel, is easier to put on, users do not have to wash their hair after using the headset and the headset costs only a couple of hundreds of euros. On the downside, the Emotiv EPOC provides worse signal quality than, for instance, the Biosemi ActiveTwo. Other headsets include the four channel Enobio system from Starlab (figure 1.2d) and the four channel Mindwave system from Neurosky (figure 1.2e).

Furthermore, BCI is often the only input modality in applications which have been developed for research projects. The 'Brain Drawing' application showed how this can be problematic: having to control a cursor continually by means of imagined movement results in a high workload. It would probably be better if BCI was one of multiple modalities used to control an application. Examples of such multimodal applications, or hybrid BCIs, are the previously mentioned 'AlphaWoW', where brainwaves in the alpha band are combined with keyboard and mouse inputs, and 'Mind the Sheep!', where SSVEP is combined with mouse input. Other examples include a touchless Human-Computer Interaction system created by Zander et al. which combines eye gaze for cursor control with a BCI for making selections [Zander et al., 2010].

Moreover, the focus in BCI research should shift from reliability to usability and user experience [Future BNCI Roadmap, 2012]. This shift in focus is necessary in order for BCIs to migrate out of the lab, into society. Healthy persons can choose from various alternative input modalities. So, for healthy persons to choose for BCI, the user experience and usability must be adequate. Most people have never used a BCI and the novelty of this new technology can be a reason for people to decide to use a BCI instead of alternative input modalities, even if BCI is less reliable and slower. However, if the user experience and usability are not good, people are expected to choose a different input modality which provides a better user experience and usability. Due to the fact that the focus in BCI research has mainly been on the reliability, no standardized methods to assess the user experience for BCIs exist yet. Gürkök et al., Plass-Oude Bos et al. and Van de Laar et al. addressed the need for standardized methods to assess the user experience for BCIs [Gürkök et al., 2011, Plass-Oude Bos et al., 2011, van de Laar et al., 2011a]. Van de Laar et al. proposed a questionnaire consisting of a core containing general questions and modules for the different kinds of mental tasks and ways of interacting with the BCI [van de Laar et al., 2011].

Finally, for BCIs to become succesful products for everyone, the acceptance of these projects in society is key. This acceptance is influenced by ethical and societal issues such as the safety, side-effects, privacy of mind, social stratification and communication to the media [Future BNCI Roadmap, 2012].



(a) Biosemi EEG system

(b) g.tec g.SAHARA system

(c) Emotiv EPOC system

(d) Starlab Enobio system

(e) Neurosky Mindwave system

Figure 1.2: EEG headsets

Combining multimodal BCIs, art and wireless EEG headsets

In this Master's thesis, we combined the new developments of multimodal BCIs and wireless EEG headsets with art by creating a multimodal interactive system which allows healthy persons, but possibly also patients, to express themselves creatively, called BrainBrush. We used this system to research how the different modalities contribute to the user experience. Furthermore, we researched whether BCI has an added value for this system.

We aimed to design the BrainBrush system in such a way that it would be appealing to healthy persons and would also be usable for patients who merely have control over their heads. We expected healthy persons would find it appealing to be able to operate the BrainBrush system using only their heads because it is completely different to how healthy people normally use a computer, namely using their hands to operate a keyboard and mouse. For patients who do not have control over their limbs, it is a necessity to be able to operate the system using only their heads. Therefore, the challenge was to design and develop the BrainBrush system in such a way that users would only need their heads to operate the BrainBrush system.

We chose to use head movement, the P300 speller and eyeblinks as the input modalities for the BrainBrush system. These input modalities satisfy the condition that they should not require the use of any limbs; users are only required to have control over their heads. These three input modalities were selected because they are expected not to interfere with each other. Head movement has been used successfully in the past as an input modality, for instance to control a cursor [Evans and Blenkhorn, 1999]. Furthermore, the P300 speller has proven to be a robust BCI paradigm [Guger et al., 2009] and has been used in many BCI systems. Finally, eyeblinks have been used for communication systems for ALS patients, such as the system for making selections on a computer screen proposed by Takeshita et al. ([Takeshita et al., 2003]). However, the eyeblinks are usually being detected using a camera and image processing techniques, instead of using an EEG headset. Chambayil et al. have shown promising results for their virtual keyboard BCI which uses eyeblinks to select characters [Chambayil et al., 2010]. Therefore, we included the eyeblink input modality.

The BrainBrush system was designed with the functionality for drawing, to select from various brushes and colors, to undo and redo actions in case of mistakes and the ability to start a new painting. Using these functionalities, users are expected to be able to create paintings to their liking.

Scenario's

As mentioned, BrainBrush is aimed at two kinds of users: healthy persons and disabled patients who are unable to use their limbs. Both types of users are illustrated using two fictive users: William and Jessica.

William is our first user, a 31 years old healthy male. William likes to paint in his spare time to get away from his busy 9-to-5 job. However, having painted on a real canvas using brushes and paint for quite some years, he found painting using his hands had become too common and he wanted to try something new. William had always been interested in new technologies and he wanted to combine his love for painting with his affinity with technology. William finds that, using BrainBrush, there is a direct link between his mind, where the creativity comes from, and his paintings.

Our second user is Jessica. Jessica is a 56 years old female who used to paint during most of her life. Unfortunately, Jessica suffers from ALS and is now unable to move her arms. She has therefore not been able to paint for the last couple of years. Jessica still has good control over her head movement and the blinking of her eyes. This allows her to use BrainBrush to, once again, enjoy painting and to express herself creatively.

1.2 Research questions

The research questions which can be formulated following the motivation section are:

RQ1: How can we create a multimodal interactive system which allows persons to express themselves creatively?

This thesis shows how we designed and implemented BrainBrush: a multimodal interactive system which persons can use to express themselves creatively by painting on a virtual canvas using eyeblinks, head movement and the P300 speller.

RQ2: How do the different modalities eyeblinks, head movement and the P300 speller contribute to the user experience?

For this thesis, we incorporated eyeblinks, head movement and the P300 speller in BrainBrush and researched the contribution of these input modalities to the user experience. We researched the contribution of the input modalities independently and the combination of the three input modalities.

The use of head movement as an input modality was expected to be an improvement compared to the use of only the P300 speller in the 'Brain Painting' system, and also compared to the continuous use of imagined movement in the 'Brain Drawing' system due to lower workload and head movement being a more intuitive action.

Provided that eyeblinks are detected properly, the use of eyeblinks as an input modality was expected to have a positive influence on the user experience because the action of eye blinking was expected to be easy.

The P300 speller input modality was expected to work reasonably, but not perfectly due to the fact we used the Emotiv EPOC headset. The headset comparison by Nijboer et al. showed the P300 accuracy with the Emotiv EPOC was for most users lower than with other EEG systems [Nijboer et al., 2012]. Furthermore, we expected the P300 speller required the highest level of concentration of all input modalities. Therefore, the user experience was expected to be less positive compared to the user experience for the other two input modalities.

We expected the combination of the three input modalities would provide a positive influence on the user experience because the switching between input modalities means users do not have to continually focus or concentrate on the same action. Furthermore, when incorporated in the system in such a way that the P300 speller does not have to be operated at the same time as the other two input modalities, the chosen input modalities are not expected to negatively influence each other.

RQ3: Does BCI have an added value?

The added value of BCI was expected to be the novelty effect of using ones brain to control (part of) an application. At least for healthy people, the practical value of incorporating BCI in the system was expected to be limited because they have alternatives which are likely to provide them with higher accuracy and allow them to work faster: they could use software applications without BCI, or they could even paint in real life. Whether the novelty outweighs the practical disadvantages was expected to depend on whether users would view the system as being a fun application for recreational painting, or an application which should allow them to be as productive as when using more traditional painting applications.

1.3 Approach

The research described in this thesis consists of two parts: we start with the design and development of the BrainBrush system, and then describe our research into the user experience of the resulting system as a whole and for the various input modalities.

Design and development

The BrainBrush system was developed using an iterative process where the stages of designing and implementing the software have been carried out in a cyclic manner, see the upper part of figure 1.3.

First, we defined use cases based on the system we had in mind. Then, we defined the requirements which entail the conditions which have to be met by the system. Next, we designed the system with respect to the way the input modalites were to be used and the user interface. In the next stage, we developed the system according to the design, thereby completing the first iteration.

This first version of BrainBrush was then evaluated with a pilot test during the Fourth BrainGain Consortium Meeting in Maastricht. During this meeting, we presented the BrainBrush system at a poster presentation session and offered interested people the opportunity to try the system, see figure 1.4. During this pilot test, 5 users worked with the system to create some paintings. Observations made during this pilot test and remarks made by participants were taken into account during the second iteration of the development process.

In this second iteration, all stages of the development cycle were revisited and changes were made where necessary: the use cases and requirements were adapted, the user interface design was changed, and the changes were implemented in the system. Furthermore, we noticed some small software bugs during the pilot test, which were also fixed during this second development iteration;

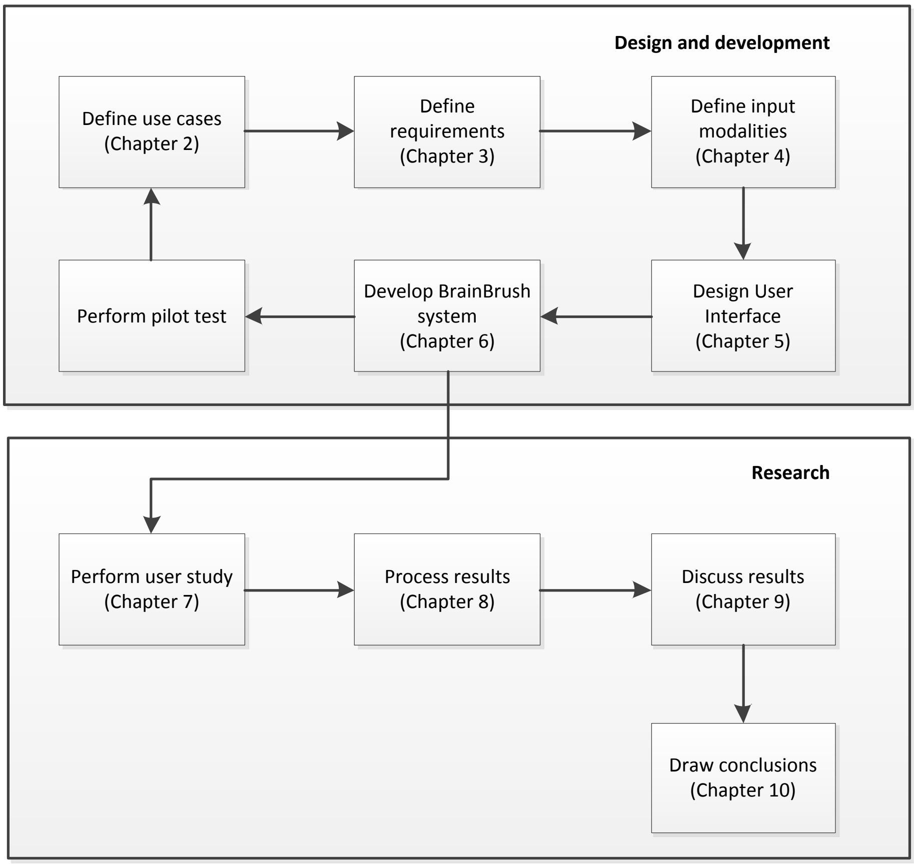


Figure 1.3: Approach

the software bugs will not be discussed in this thesis.

The use cases will be shown in chapter 2, followed by the requirements in chapter 3. Next, chapter 4 describes the input modalities and how they have been implemented in BrainBrush. Chapter 5 details the user interface design and how the user interface design changed after the first pilot test. Next, chapter 6 provides an overview of the software architecture of the BrainBrush system.

User experience research

This second and, for now, final version of the BrainBrush system was used for our user experience study. We performed a user study with 13 healthy people who performed a free painting task using BrainBrush. The research was performed qualitatively to gain an understanding of underlying reasons and motivations for either positive or negative opinions the participants formed about their user experience with the system and the various input modalities.

All participants were asked to complete a questionnaire to assess the usability of the system, and an interview was conducted to investigate why participants liked or disliked the various aspects of the system.

Figure 1.4: A user painting with BrainBrush at the Fourth BrainGain Consortium Meeting

The lower part of figure 1.3 shows the process followed for the user experience research. Chapter 7 describes the user study. The results of this user study will be shown in chapter 8, followed by a discussion of the results in chapter 9. Finally, we present our conclusions in chapter 10.

Part I

Design and development

Chapter 2

Use cases

In this section, we describe the use case scenarios for BrainBrush. Each use case includes a narrative description of the task, followed by the definition of the interaction with the system.

We define one actor for the BrainBrush system: the user. For the purpose of these use cases we will call him John. We assume John is seated in front of his computer, with the Emotiv EPOC EEG headset on his head and all necessary software started.

While painting using the BrainBrush system, John has to choose the brush he likes, choose the color he likes, he has to create brushstrokes, undo erroneous brushstrokes, redo previously undone brushstrokes and finally he has to save his painting and start with a clean canvas for his next masterpiece.

Task 1: Select brush

John wants to select a specific type of brush for painting, so he triggers the 'Change brush' option. The system shows the available brushes and lets John select the brush he likes. When John has selected a brush, the system switches back to the canvas and confirms the brush selection. John can now start using the brush.

See table 2.1 for the use case.

Table 2.1: Task 1: Select brush

User intention	System responsibility
Select a brush	
	Switch to brush selection mode
	Show available brushes
Choose desired brush	
	Select brush
	Switch back to canvas
	Confirm brush selection

Task 2: Select color

Having selected the brush he wants to use, John now wants to select a nice color. He triggers the ‘Change color’ option and the system responds by showing the available colors and lets John select the color he wants to use. When John has selected a color, the system again switches back to the canvas and confirms the color selection. John can now start painting using the selected color.

See table 2.2 for the use case.

Table 2.2: Task 2: Select color

User intention	System responsibility
Select a color	
	Switch to color selection mode Show available colors
Choose desired color	
	Select color Switch back to canvas Confirm color selection

Task 3: Create brushstroke

John now wants to create a brushstroke from point A on the canvas to point B, using the selected brush and color. He moves the cursor to point A, then signals the system that he wants to start his brushstroke at that location. The system puts the brush to the canvas and confirms to John that it has done so. John can now start his brushstroke and he moves the cursor to point B via the path he envisioned, thereby creating the brushstroke. When the cursor has reached point B, John signals the system that he wants the brushstroke to end. The system takes the brush off the canvas and confirms this action. The brushstroke is now finished.

See table 2.3 for the use case.

Table 2.3: Task 3: Create brushstroke

User intention	System responsibility
Create a brushstroke from point A to point B	
	Start drawing at point A Confirm the brush is turned on Draw along the path intended by the user Stop drawing at point B Confirm the brush has been turned off

Task 4: Undo brushstroke

John is not happy with the last brushstroke he made and wants to remove it so he can try again. Therefore, he directs the cursor to the 'Undo' option. The system removes the last brushstroke John made and confirms this action. John can now try to make a nicer brushstroke.

See table 2.4 for the use case.

Table 2.4: Task 4: Undo brushstroke

User intention	System responsibility
Undo last brushstroke	
	Remove the last brushstroke made by the user Confirm the last brushstroke has been removed

Task 5: Redo brushstroke

After having removed his last brushstroke using the 'Undo' option, John feels that particular brushstroke was not that bad after all and he wants to redo it, so he triggers the 'Redo' option. The system adds the brushstroke to the painting once again and confirms this action.

See table 2.5 for the use case.

Table 2.5: Task 5: Redo brushstroke

User intention	System responsibility
Redo previously undone brushstroke	
	Add the previously undone brushstroke to the painting Confirm the brushstroke has been added

Task 6: New painting

John is done with his painting and wants to save it so he can start a new painting without destroying the current painting. He therefore triggers the 'New painting' option. The system saves his current painting, clears the canvas and notifies John that it has done so. John can now start his new painting.

See table 2.6 for the use case.

Table 2.6: Task 6: New painting

User intention	System responsibility
Save the current painting and start a new painting with a clean canvas	
	Save the current painting for later viewing
	Confirm the painting has been saved
	Clear the canvas

Chapter 3

Requirements

In section 1.1, we outlined our system: a multimodal system with which people can express themselves creatively. The input modalities must include eyeblink detection, head movement and the P300 speller grid. Furthermore, the system has to offer specific functionalities as defined in the use cases in chapter 2.

The system will, first and foremost, be used for our research. It is therefore designed and developed in such a way that we are able to find answers to our research questions and with users in mind who want to create art in a new and innovative way. However, the system could also be interesting for people who are unable to create paintings or drawings in conventional ways (e.g. paraplegics).

Using the outline of the system, the use cases and keeping the intended users in mind, we defined a set of functional and non-functional requirements for the BrainBrush system.

3.1 Functional requirements

The functional requirements define what the BrainBrush system is supposed to *do*. In the following requirements, *system* refers to the BrainBrush system and *users* refers to the persons painting with the BrainBrush system. The requirements have been linked to related tasks from the use cases, see chapter 2.

R1	The system must enable users to control BrainBrush using only their heads.
	As mentioned in section 1.1, we expected healthy persons would find it appealing if they were able to operate BrainBrush using only their heads and no other limbs because it is something new and completely different from the standard, and most common, way to use a computer: using their hands to operate a keyboard and mouse. Furthermore, for disabled people without control over their limbs, it is a necessity to be able to use the system using only their heads. Related task(s): 1, 2, 3, 4, 5, 6

R2	The system must be multimodal.
	As defined in research question RQ1 in section 1.2, the BrainBrush system must be multimodal in the sense that it uses multiple input modalities. Furthermore, after evaluating two BCIs for creativity, Todd et al. suggested developing a multimodal BCI for creativity, instead of a system using only BCI, as BCI is not yet mature enough for 100% reliability [Todd et al., 2012]. Related task(s): 1, 2, 3, 4, 5, 6

R3	The system must use the P300 speller as an input modality.
	In section 1.2, we defined the research questions. In order to answer research question RQ2 we needed to study the influence of the three input modalities on the user experience. Since the P300 speller is one of these modalities, it had to be included in the system. Related task(s): -

R4	The system must incorporate eyeblink detection as an input modality.
	Similar to requirement R3. The use of eyeblinks also is one of the three input modalities we wanted to study, see section 1.2. Therefore, eyeblink detection had to be incorporated in the system. Related task(s): -

R5	The system must use head movement as an input modality.
	Similar to requirements R3 en R4. The use of head movement is the last of the three input modalities we wanted to study, see section 1.2. Therefore, head movement has to be incorporated in the system. Related task(s): -

R6	The system must enable users to control the cursor freely.
	The 'Brain Painting' application developed by Münßinger et al. did not provide users with the ability to control the cursor freely. The cursor could only be moved in a predetermined grid. Münßinger et al. noted that the inability to move the cursor restricted the artistic freedom of the painter [Münßinger et al., 2010]. Therefore, BrainBrush is required to enable users to control the cursor freely. Related task(s): 3

R7	The system must enable users to create brushstrokes.
	In order to paint, users have to be able to put the paintbrush onto the virtual canvas, thereby creating brushstrokes, and to take the paintbrush off the canvas to end the brushstroke. Related task(s): 3

R8	The system must offer users a variety of different brushes to choose from.
	The brush selection functionality was included to extend the possibilities for users to express themselves creatively. Furthermore, brush selection could be implemented with the P300 speller grid, fulfilling requirement R3. Related task(s): 1

R9	The system must offer users a variety of different colors to choose from.
	Similar to the brush selection in requirement R8, color selection functionality was included to extend the possibilities for users to express themselves creatively. Moreover, color selection could also be implemented with the P300 speller grid, fulfilling Requirement R3. Related task(s): 2

R10	The system must enable users to undo their last brushstroke.
	The undo functionality was included to lower the cost of an error made by either the user or the algorithms of the input modalities. In case a brushstroke does not end up the way a user intended, the undo functionality offers an easy way to correct such a mistake. The undo functionality was also included by Mandel in his list of Golden Rules of Interface Design [Mandel, 1997]. Related task(s): 3, 4

R11	The system must enable users to redo their previously undone brushstroke.
	The redo functionality was included to offer an easy way to redo a previously undone brushstroke. This functionality lowers the cost of accidentally triggering the undo function, enabling users to correct such a mistake. The redo functionality was also included by Mandel in his list of Golden Rules of Interface Design [Mandel, 1997]. Related task(s): 3, 5

R12	The system must enable users to save their current painting and start over with a clean canvas.
	The ability to save paintings was included for two reasons. First, for our research it was beneficial to be able to view what users created during the experiments: in the interviews users referred to the paintings they created and in order to exactly understand what they mean, it was good to be able to view their creations. Second, we wanted to offer users the possibility to save their masterpieces for later viewing. The ability to start over with a clean canvas was important so users were not limited to creating a single painting during the experiment. Related task(s): 6

3.2 Non-functional requirements

The non-functional requirements define how the BrainBrush system is supposed to *be*. They define qualities we feel the system should possess. In the following requirements, *system* refers, once again, to the BrainBrush system.

R13 The system must be able to run on a relatively simple computer.

We feel it is possible to design and develop the BrainBrush system in such a way that no extraordinary requirements for the computer hardware are necessary. Not demanding extraordinary processing power, enormous amounts of memory and the presence of multiple screens has multiple advantages. First, it does not restrict the group of possible users further than anyone with a computer running Microsoft Windows and an Emotiv EPOC headset. Second, the BrainBrush system could be used for nice demonstrations of the current possibilities for Brain-Computer Interfacing. Since such demonstrations often take place outside the laboratory, it would be helpful if the BrainBrush system were able to run on, for instance, a relatively simple laptop.

R14 The system must incorporate a portable, wireless and comfortable BCI headset that may be used by users themselves at home and which is able to track head movement.

Because BrainBrush uses BCI, the system must incorporate a BCI headset. In section 1.1, we discussed a number of BCI headsets and their advantages and disadvantages. For BrainBrush, it is important the BCI headset is portable because users may want to use the system at multiple locations. Furthermore, the headset should be wireless to ensure users do not feel limited in their movements due to large amounts of wires connecting them to the computer. The headset should be comfortable so users can complete a painting without having to stop due to physical discomfort. Users should be able to set up the headset without an expert present since they would otherwise not be able to use the system at home. Finally, the headset must be able to track head movement in order to be able to satisfy requirement R5.

We chose to use the Emotiv EPOC headset which is a commercially available wireless headset. The headset features 16 saline EEG sensors and does not require conductive gel. From those 16 sensors, 2 sensors are reference sensors (Common Mode Sense and Driven Right Leg), at locations P3 and P4. The other 14 sensors are located at AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8 and F4 from the international 10-20 system. The Emotiv EPOC outputs 128 samples per second per sensor. Furthermore, the headset offers horizontal and vertical tilt information through the use of gyroscopes, which can be used to track head movement.

R15 The system must be usable with a minimal amount of training and calibration.

BrainBrush incorporates multiple input modalities. If each of those modalities would require an extended training session to calibrate algorithms, to train classifiers, or to train the user, overall training time would be too long. It would extend the duration of the experiments to the point that participants might not want to sign up for the experiment, or if they have signed up, chances are they will want to stop the experiment early because of discomfort due to the headset. Furthermore, if users were to use the BrainBrush system in their home environment, the training time should be as short as possible so that it does not keep them from using the system whenever they have some spare time. We therefore aim for a maximum of 15 minutes total to calibrate and train any classifiers needed and a further 5 minutes for the user to get used to the system. It might be necessary to repeat the training of classifiers once in a while, but it should not be necessary to repeat the user training.

R16 The system must be easy to set up and start.

Since most BCI applications are made for research goals, usually little effort is made to design and develop the application in such a way that it is easy to start up and run. This limits the possibilities of everyday use of such an application outside the research environment. With BrainBrush, a layman must be able to set up, start and use the system after a short introduction from an experienced BrainBrush user. The BrainBrush application can have a maximum of five configuration options. The ease of use during training is considered to be outside the scope of this thesis.

Chapter 4

Input modalities

This chapter outlines how the BrainBrush system incorporates the three input modalities defined in requirements R3 (P300 speller), R4 (eyeblink detection) and R5 (head movement) in chapter 3 in order to be able to execute the tasks defined in chapter 2.

The task of creating a brushstroke (chapter 2, task 3) requires the system to provide users with a way to signal when they want the paintbrush to be put to the canvas and to signal once again when they want to take the paintbrush off the canvas, thereby ending the brushstroke. This on/off switch for the paintbrush is discussed in section 4.1.

For users to be able to create nice brushstrokes, more is required than just an on/off switch. They also require a way to move the cursor via the path they intend (chapter 2, task 3). This cursor movement is discussed in section 4.2.

Finally, the system must allow users to choose from a variety of brushes (chapter 2, task 1) and colors (chapter 2, task 2). The design for making brush and color selections is discussed in section 4.3.

4.1 On/Off switch

To create a brushstroke in any conventional drawing application, users would need to press down a mousebutton, hold it down and release the button at the end of the brushstroke ('mouse drag'). In other words, the action (pressing a mousebutton) needs to be performed for the whole duration of the drawing-action. However, this is no requirement for the replacement input modality to be used within BrainBrush. Using this new input modality, users have to be able to signal to the BrainBrush application when a drawing-action should start, and signal to the application again when the action is supposed to end. Both the start and end could even be signalled using the same input.

Not having to continually perform some kind of action to mimic the mouse drag has another advantage: users do not need to continually focus on performing some action with their heads while also moving the mousecursor with another input modality. Therefore, this lowers the workload for the user.

Using the same input modality to denote the start and the end of a brush-

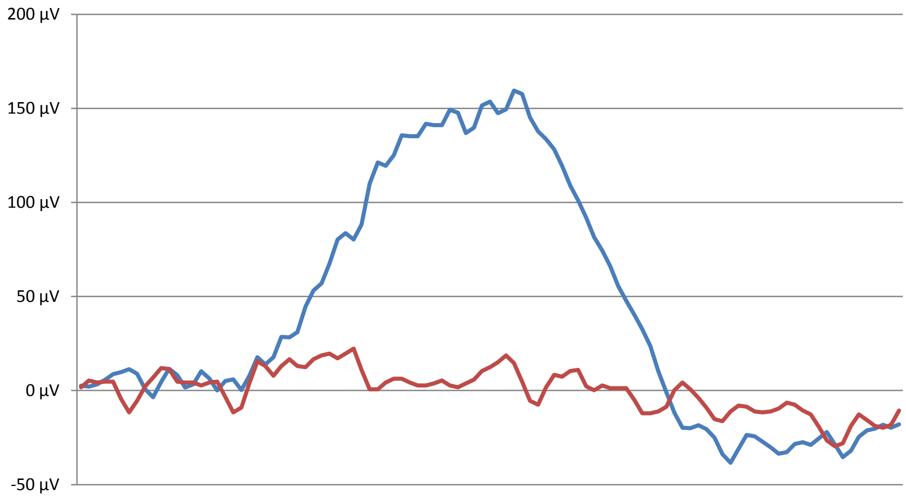


Figure 4.1: The eyeblink template and EEG without eyeblink. Offset correction was performed to show how much EEG during an eyeblink differs from normal EEG.

stroke should make it easier to remember for users and is therefore expected to increase the usability of BrainBrush.

For BrainBrush, we chose to use eyeblinks as the on/off switch to signal the start and end of a brushstroke because we expected blinking would not interfere with the other input modalities and this way, requirement R4 (*'The system must incorporate eyeblink detection as an input modality'*) is fulfilled.

In order to detect eyeblinks, we implemented an eyeblink detection algorithm based on the eyeblink detection algorithm described by Plass-Oude Bos et al. which detects eyeblinks in EEG data using a template of the general shape of the EEG data during an eyeblink [Plass-Oude Bos et al., 2010].

To construct the template, an experiment was done with 1 subject where the subject was instructed to blink whenever he noticed a visual stimulus on the screen. We used the StimulusPresentation module of the BCI2000 framework to present the same visual stimulus 40 times, with 4 second intervals. This way, 40 eyeblinks were recorded. After visual inspection of EEG data containing eyeblinks, we decided to use channels AF3 and AF4 for the eyeblink detection because eyeblinks were most clearly distinguishable in the EEG data for those channels. For the construction of the template, only the data from AF3 was used. From this EEG data, 40 epochs were extracted which contained the eyeblinks. All epochs start the same amount of samples after the stimulus, and also have the same length. These 40 epochs were baselined and averaged, which resulted in the template shown in figure 4.1. The EEG data for channels AF3 and AF4 were found to be very similar and we therefore decided to use the same template for both channels.

For the online eyeblink detection, the template is compared with the real-time EEG data. The template is compared with a piece of the live data which contains the same amount of samples, 103, as the length of the template. After this comparison, the first sample of the live data is discarded, and a new sample is added to the end. The template is then once again compared with this data,

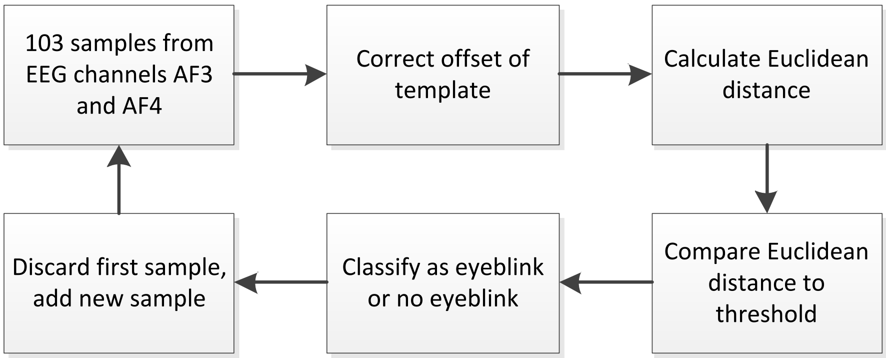


Figure 4.2: Overview of the eyeblink detection pipeline

and so on.

The comparison is done by first performing an offset correction for the template to account for drifts in the signal: for both the template and the live data, the average of the first 10 and the last 10 samples is calculated and the difference between those two averages is subtracted from the template, thereby performing the offset correction. Next, the distance between the (offset corrected) template and the live data is calculated.

This distance is measured using the Euclidean distance. When comparing the 103 EEG measurements from the template ($p = (p_1, p_2, ..., p_{103})$) with 103 EEG measurements from the live EEG data ($q = (q_1, q_2, ..., q_{103})$), the euclidean distance is given by:

$$d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + ... + (p_i - q_i)^2 + ... + (p_{103} - q_{103})^2}$$

The distance is calculated for both the AF3 and AF4 channels using the same template. Whenever the Euclidean distance for a certain segment of EEG data on any of the two channels is below a predetermined threshold, that segment of live data is classified as being an eyeblink. The threshold to be used was empirically found and is usually around 650, but can be adjusted for each user.

A schematic overview of the pipeline used for eyeblink detection can be found in figure 4.2.

Because unintentional eyeblinks are usually much shorter than intentional eyeblinks [Takeshita et al., 2003], unintentional eyeblinks should not match the template well enough to be classified as eyeblinks and should thus not trigger the BrainBrush system.

4.2 Cursor movement

For users to be able to create brushstrokes, they also need an input modality which offers a way to move the cursor freely, see requirement R6 in chapter 3. In a conventional drawing application, users would move the cursor by moving the mouse or, for instance, drawing on a graphics tablet.

For BrainBrush, we chose to use head movement to control cursor movement because head movement has proven to be suitable for cursor control

[Bates and Istance, 2003]. Furthermore, by using head movement, requirement R5 (*'The system must use head movement as an input modality'*) was fulfilled.

To convert head movement to cursor movement, we chose to use the gyrosensors of the Emotiv EPOC headset. A gyrosensor measures angular acceleration which, if checked in regular, small, intervals, can be used to calculate the amount of rotation of the object the gyrosensor is attached to. Using multiple gyrosensors, it is possible to calculate the rotation in multiple directions. With the Emotiv EPOC, we can measure how users turn their heads horizontally and vertically which is ideal for moving a cursor on the screen.

Instead of developing our own software, we opted to use the *Mouse Emulator* application provided by Emotiv as this application does exactly what is needed: it translates head movement to movement of the cursor.

4.3 Selecting brushes and colors

Besides creating brushstrokes, users also need to be able to select brushes (chapter 2, task 1) and colors (chapter 2, task 2). In a conventional drawing application, users would choose from a variety of brushes and colors by selecting an icon in a list of available brushes or colors. The selection is usually done by clicking the icon of the desired brush or color with the mouse.

In BrainBrush, we use P300 speller grids for the selection of brushes and colors because the P300 speller is suitable for making selections from a large set of options in a relatively short period of time [Aloise et al., 2011a], compared to other Brain-Computer Interfacing paradigms. In a study performed by Guger et al. with healthy participants, 88.9% of the participants was able to achieve at least 80% accuracy using the P300 speller paradigm [Guger et al., 2009], showing the robustness of the P300 signal.

The original version of the P300 speller included the letters of the alphabet and a few 1-word commands [Farwell and Donchin, 1988]. Since the development of the original P300 speller, variations to this layout have been developed. One of the reasons to develop a new interface for the P300 speller was the fact the performance of the P300 speller highly depends on the ability of the user to fixate the target with gaze [Treder and Blankertz, 2010, Brunner et al., 2010]. Therefore, alternative interfaces for the P300 Speller have been developed for patients without control over their eye movements: Hex-o-Spell (and some variations to the same principle) [Treder et al., 2011] and GeoSpell [Aloise et al., 2011b]. Both variations are slower than the standard P300 Speller interface, but for patients without the ability to move their eyes they might be very useful.

For BrainBrush, we use the original grid structure, but instead of grids with characters, grids with pictures are used where each picture depicts a certain brush or color. The *P3Speller* module of the BCI2000 framework is used for the implementation of the P300 speller grids, which is an implementation of the P300 speller grid, with various additional options, such as cascaded menus and using icons instead of characters. The stimulus duration is set to 100ms and the inter-stimulus interval to 175ms. For the selection of a brush or color, 15 sequences of flashes are shown, meaning each row and column flashes 15 times. Therefore, the target is flashed 30 times in total.

By using the P300 speller grids for the selection of brushes and colors, requirement R3 (*'The system must use the P300 speller as an input modality'*) was fulfilled.

Chapter 5

User Interface design

The user interface for BrainBrush was designed to enable users to perform the tasks described in chapter 2. Users need to be able to select brushes (requirement R8) and colors (requirement R9), create brushstrokes (requirement R7), undo (requirement R10) and redo (requirement R11) brushstrokes and to save the painting so they can start a new painting (requirement R12).

As mentioned in section 1.3, the development of BrainBrush was done using an iterative process. Because the user interface was changed significantly after the pilot test following the first iteration, we discuss both iterations separately in this chapter. Section 5.1 outlines the user interface as it was designed and implemented during the first iteration, section 5.2 outlines the changes which were made to the user interface after the pilot test and the reasons for doing so.

5.1 Iteration 1

The Brain Painting application of Münßinger et al. required the availability of two screens: one for the painting and one for the P300 speller grid [Münßinger et al., 2010]. Because BrainBrush is required to run on a single screen setup (chapter 3, requirement R13), showing the P300 speller grid constantly would reduce the available drawing space considerably. However, because in BrainBrush, unlike in Brain Painting where everything was controlled using the P300 speller grid, the cursor movement is controlled by head movement and switching the brush on and off is done with eyeblinks, it is unnecessary to show the P300 speller grid constantly. It is sufficient to only show the P300 speller grid when selecting a brush or color for drawing. The user interface for BrainBrush can therefore be divided in two separate modes: a painting mode and a selection mode.

Section 5.1.1 discusses how switching between the painting and selection modes is performed, section 5.1.2 discusses the painting mode and in section 5.1.3, the selection mode is discussed.

5.1.1 User Interface mode switching

For BrainBrush, the painting mode as shown in figure 5.1 is the main mode. In this mode, users can draw on the virtual canvas, and they are expected to spend most time in this mode. According to the golden rules of user interface design as defined by Mandel, when introducing different modes in an application it is important to let users choose when they want to go into a particular mode and to provide users with immediate visual feedback whenever they change modes [Mandel, 1997]. We designed the user interface so users can switch to the selection mode from the painting mode by selecting the 'Change Brush' or 'Change Color' menu options displayed as rectangles in the top left and top right corners of the frame of the painting.

Users can select the rectangles by hovering the cursor over one of the menu options. Because users are not required to confirm the selection of such a menu option in any way, this could lead to unintended selections by accidentally pointing the cursor at one of the menu options. This problem of unintended selections is an example of the Midas-Touch-Problem [Ihme and Zander, 2011], named after King Midas who had the gift of turning everything he touched into gold, even his food. Because of this, he almost starved to death.

This risk of unintended selections was reduced by placing the menu options in places where users are unlikely to accidentally move the cursor: the corners of the screen.

Another solution would have been to require users to confirm their selections using eyeblinks, thereby eliminating the Midas-Touch-Problem. However, this way, the ability to select a menu option would depend on the eyeblink detection accuracy and it would also require more effort from the users. We therefore chose not to require users to confirm the selection of a menu option.

5.1.2 Painting mode

When in painting mode, users are able to paint using the selected brush and color by moving their heads and switching the brush on and off by blinking their eyes (chapter 2, task 3). Furthermore, as discussed in section 5.1.1, users are able to switch to the selection mode whenever they want to switch to another brush or color (chapter 2, tasks 1 and 2).

To give BrainBrush a nice look, a picture frame[1] is displayed at the edge of the screen. The painting canvas is shown on the inside of the picture frame, thereby occupying the largest part of the screen. The selected brush and color are displayed in text at the bottom of the picture frame. See figure 5.1 for a picture of the user interface in painting mode.

The brush selection and the color selection were separated in two P300 grids, see section 5.1.3 for more information on this. Since we chose not to use eyeblinks to confirm the selection of a brush or color menu option, it was important to minimize the possibility of accidentally selecting a menu option, as mentioned in section 5.1.1. Therefore, the 'Change brush' and 'Change color' menu options have been placed on opposite sides of the screen: in the top left and top right

[1] Picture created by Max Stanworth, provided under the *CC BY 2.0* license: http://creativecommons.org/licenses/by/2.0/deed.en

Figure 5.1: User Interface in Painting mode with the canvas in the center, the 'Change Brush' and 'Change Color' menu options in the top left and right corners and the selected brush and color on the bottom - Iteration 1

corners of the picture frame, see figure 5.1.

5.1.3 Brush and color selection modes

For the selection mode, BrainBrush uses the *P3Speller* module from the BCI2000 framework, see section 6.2. The *P3Speller* module largely defines the layout of the selection mode: a fullscreen window with a black background and a grid showing rows and columns filled with symbols or images.

The *P3Speller* program offers the possibility to use cascaded menu's: selecting an item in the main P300 grid would display a new P300 grid with new options. For the user interface it would therefore be possible to use a single menu option to display a main P300 grid with two options: 'Select brush' and 'Select color'. After selecting either of the two options, the user would be presented with the appropriate P300 grid to select the desired brush or color. However, this would require users to perform two selections to select a single brush or color. The chance of users being able to select the brush or color they want would be lower because both selections would need to be correct to achieve the desired result. Furthermore, the process of selecting a brush or color would take more time during which users have to remain concentrated.

BrainBrush therefore uses two separate P300 grids, one for brush selection and one for color selection, both directly accessible from the painting mode. The only difference in the user interface for both selection modes is the symbols used for the specific selection mode (i.e. brush symbols or color images).

For the brush selection, a 4 x 4 grid filled with pictures corresponding to the available options is used, see figure 5.2. The options consist of 13 different

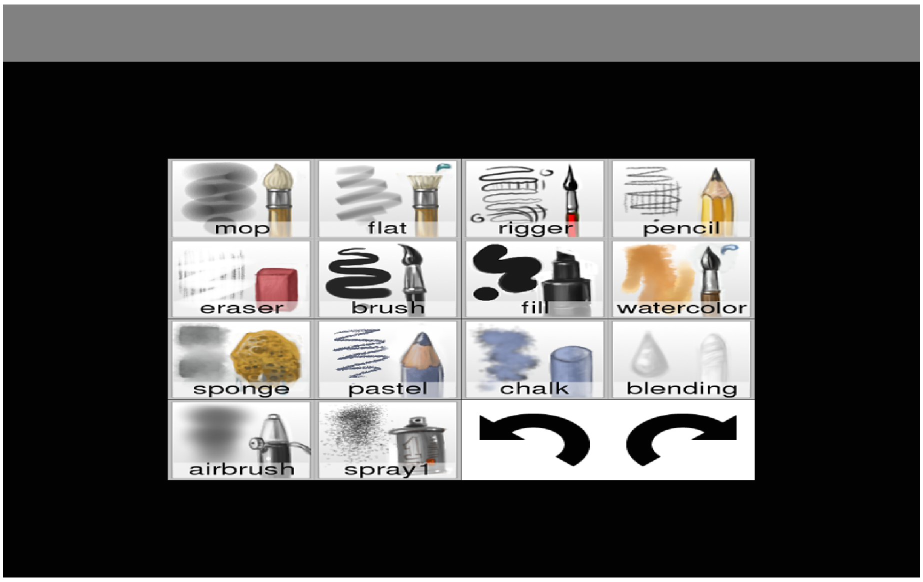


Figure 5.2: User interface during brush selection showing the 13 available brushes, 1 eraser and the undo and redo options - Iteration 1

Figure 5.3: User interface during color selection showing the 12 available colors - Iteration 1

brushstyles, 1 eraser, the undo function and the redo function. The pictures for the symbols have been taken from MyPaint, the drawing application used for BrainBrush, see section 6.3.

The P300 speller grid for the color selection consists of a 3 x 4 grid with images showing the available colors: red, green, blue, yellow, cyan, purple,

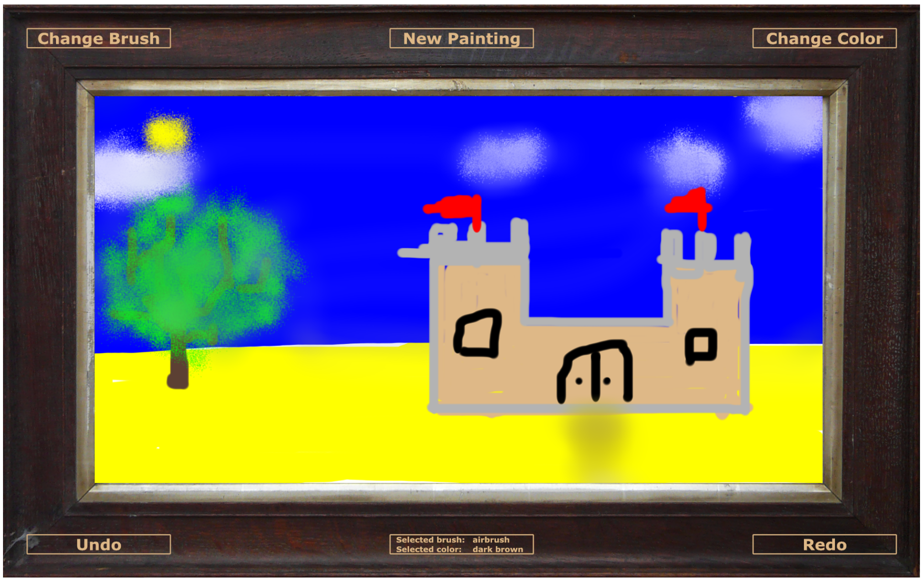


Figure 5.4: User interface in Painting mode with the 'New Painting' menu option at the top and the 'Undo' and 'Redo' menu options at the bottom - Iteration 2

black, burlywood, dark brown, grey, orange and white. The images for the colors consist of rectangles in the specific colors, see figure 5.3.

5.2 Iteration 2

During the pilot test of the BrainBrush system, several remarks were made by participants regarding the user interface. Furthermore, we observed the interaction with the system. With these remarks and observations in mind, we adapted the user interface of BrainBrush during the second iteration of the development process.

First, we removed the undo and redo functions from the brush selection grid because we observed that participants were not using the undo and redo options, even when they made small mistakes during painting. We expect this might be due to the fact that in order to undo a brushstroke, participants had to use a P300 speller session which lasts 20 seconds. Furthermore, the undo and redo functions were located in the brush selection grid, even though both undo and redo are not brushes. Users therefore might not have noticed the undo and redo functions existed. We then placed the undo and redo functions as menu options in the bottom left and right corners of the picture frame in painting mode, see figure 5.4. By placing them in the bottom left and right corners, we ensured they are not located near other menu options, thereby minimizing the risk of accidental selections.

By removing the undo and redo options from the brush selection grid, 14 items remained in the P300 speller grid. Because 14 items can not be divided in a good number of columns and rows, we removed 2 additional items from the grid: the *pastel* and *blending* brushes. These brushes were chosen because the

Figure 5.5: User interface during brush selection showing the 11 available brushes and the eraser - Iteration 2

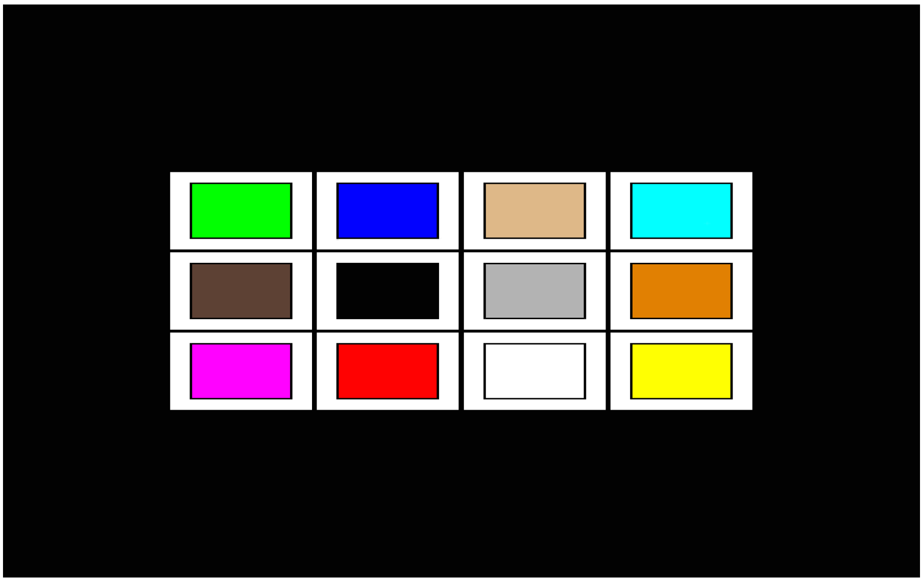


Figure 5.6: User interface during color selection showing the 12 available colors with white borders - Iteration 2

pastel brush is very similar to the *chalk* brush, and the *blending* brush is very similar to the *watercolor* brush. An additional advantage of having one row less in the brush selection grid is the fact that the speed of the brush selection is improved by 12.5%. See figure 5.5 for the resulting brush selection grid.

Finally, remarks by participants indicated that for some colors it was not easily visible whether they were flashing. Obviously, this design was not likely to elicit a strong P300 response. Therefore, the images in the P300 speller grid were changed, see figure 5.6. In this updated version, the colored rectangles have a thick white border. The flashing of this border is easily visible and therefore all colors should be able to elicit a stronger P300 response.

During this second iteration of the development process, two additional changes were made to the user interface which were not directly related to remarks and observations during the pilot test. First of all, the 'New Painting' functionality was added (see chapter 2, task 6), thereby fulfilling requirement R12 (*'The system must enable users to save their current painting and start over with a clean canvas'*). In order for users to be able to start making a new painting, a 'New Painting' menu option was added to the picture frame in Painting mode, see figure 5.4. The option was added in the middle of the top of the picture frame thereby ensuring it is not placed near other menu options to minimize the risk of accidental selections. The final change we made was to remove the grey bars at the top of the P300 speller grids. When using the P300 speller grid to spell words, this grey bar is used to display the text a user is typing, but it has no use in BrainBrush. During the second development iteration, we found a way to remove the grey bars. The result can be seen in figure 5.5 and figure 5.6.

Chapter 6

Software architecture

This chapter shows the architecture for the BrainBrush software. The BrainBrush software allows users to perform the tasks defined in chapter 2. The BrainBrush software is designed and implemented according to the requirements defined in chapter 3. Furthermore, the software incorporates the input modalities defined in chapter 4 and provides the graphical user interface defined in chapter 5.

6.1 System overview

Using the design described in the previous chapters, the following components are defined for the BrainBrush system: the P300 speller containing images of brushes and colors, an eyeblink detection component, a head tracking component, and a paint component providing the virtual canvas and the ability to paint using various brushes and colors.

For the P300 speller and the eyeblink detection components, we used the Brain-Computer Interfacing platform BCI2000. BCI2000 provides an implementation of the P300 speller and furthermore provides the functionality to add our eyeblink detection algorithm. See section 6.2.

For the paint component, we used the open-source MyPaint application. In order to be able to interface with MyPaint, we adapted the application. Section 6.3 describes MyPaint and how we adapted MyPaint for BrainBrush.

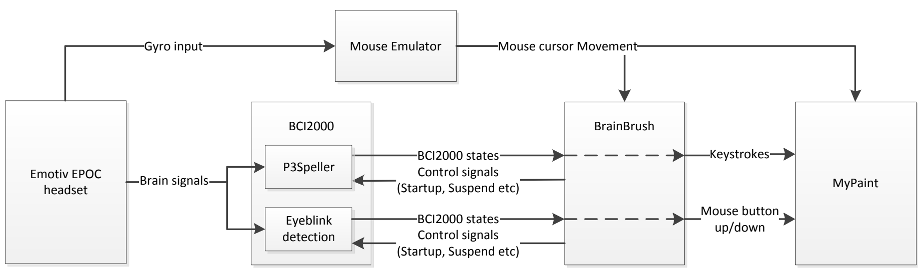


Figure 6.1: BrainBrush system overview

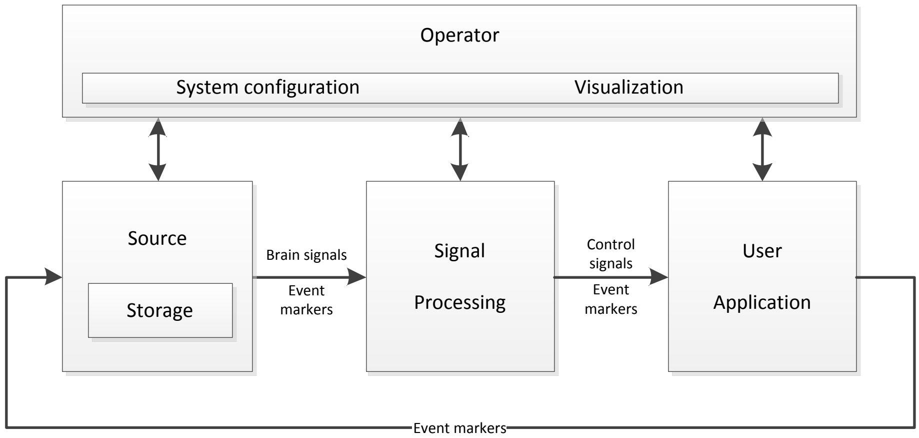


Figure 6.2: BCI2000 platform

In section 6.4, we describe Emotiv's Mouse Emulator program which was used to track the user's head.

In order to integrate the software components into one system that handles all user inputs and switches between painting and brush or color selection modes, we developed the BrainBrush application. The BrainBrush application component implements all logic necessary for the system to operate as designed. We describe the BrainBrush application in section 6.5.

Figure 6.1 shows how all components are connected. Details of the connections between the various components are discussed in the following sections.

6.2 BCI2000

BCI2000 is a software platform for BCI research [Schalk et al., 2004]. It aims to facilitate research and the development of applications in all areas that depend on real-time acquisition, processing, and feedback of biosignals. BCI2000 is mainly developed by the Brain-Computer Interface R&D Program at the Wadsworth Center of the New York State Department of Health in Albany, New York, USA with substantial contributions by the Institute of Medical Psychology and Behavioral Neurobiology at the University of Tübingen, Germany. The information in this section is largely based on [Schalk and Mellinger, 2010].

6.2.1 BCI2000 modules

BCI2000 consists of four separate modules: the SignalSource, SignalProcessing, Application and Operator modules (see figure 6.2). This modular design ensures that individual modules can easily be replaced without the need to redesign the whole application. So, for instance, other headsets can be connected without the need to make major changes to the software.

The modules are separate programs which communicate through a TCP/IP-based protocol and the programs could therefore be run on different computers.

SignalSource module

The SignalSource module acquires brain signals from the BCI headset and passes the signals on, in blocks containing a fixed number of samples, to the SignalProcessing module. The SignalSource module consists of a data acquisition component and a data storage component, which stores the brain signals and the event markers.

SignalProcessing module

The SignalProcessing module extracts signal features and translates those features into control signals and sends them to the Application module.

The process consists of two stages: first, features are extracted from the brain signals using a series of filters, next, the signal features are translated into control signals using another set of filters.

Application module

The Application module receives the control signals from the SignalProcessing module and uses them to control an application, for instance the P300 speller grid. The resulting event markers are sent back to the SignalSource module where they and the raw signals are stored to disk.

Operator module

The Operator module enables the user to start, stop, suspend or resume the application. Furthermore, it provides the functionality to load parameter files with settings for the various modules and it can provide the user with a graphical display of the current system parameters and real-time analysis results.

We chose not to use the Operator executable from the BCI2000 distribution, but to create an own implementation of the Operator module in C# instead. By using our own Operator module, we can provide seemless integration of BCI2000 into the BrainBrush system, which would otherwise not be possible.

6.2.2 P300 speller component

BrainBrush uses a P300 speller for brush and color selection, as discussed in section 4.3. The BCI2000 framework offers a complete P300 speller implementation: the *P3Speller*. The *P3Speller* application module can be customized; we used the *P3Speller* to show images of brushes and colors instead of characters, as discussed in section 5.1.3.

In order to be able to use the Emotiv EPOC headset, we used the *Emotiv* SignalSource module which is not part of the BCI2000 core distribution, but is a module provided by Griffin Milsap from the Rensselaer Polytechnic Institute.

The *P3SignalProcessing* module from BCI2000's core distribution was used for signal feature extraction, P300 classification and for the translation into control signals.

6.2.3 Eyeblink detection component

When painting with BrainBrush, users can put the brush to the canvas by blinking their eyes. BrainBrush uses an eyeblink detection algorithm to detect the eyeblinks in the EEG signals, as discussed in section 4.1. The eyeblink detection algorithm had to be implemented from scratch. For this, we created a new SignalProcessing module called *EyeblinkSignalProcessing*. First, we created a new project *EyeblinkSignalProcessing* and added this to the existing BCI2000 source using the *NewBCI2000Module.exe* tool which is available with the BCI2000 package. This tool added the *EyeblinkSignalProcessing* module to the BCI2000 source tree and created a number of sourcefiles within this new project, the most important being *PipeDefinition.cpp*. This sourcefile allowed us to define which filters have to be used to process the EEG, and in which order the filters should be used. Since BCI2000 does not come with a filter for eyeblink detection, we added our own filter, *EyeblinkFilter*, to the *EyeblinkSignalProcessing* module using BCI2000's *NewBCI2000Filter.exe* tool. This created the sourcefile *EyeblinkFilter.cpp*, which contained a framework in which we implemented our algorithm. For details about the implementation of the eyeblink detection algorithm, see appendix A. After implementing the algorithm, we added the *EyeblinkFilter* to the *PipeDefinition.cpp* file. Compiling the sourcecode resulted in the *EyeblinkSignalProcessing* executable we used as our SignalProcessing module for eyeblink detection.

The eyeblink detection component of the BrainBrush system merely needs to be able to detect eyeblinks, and continually output the classification results via state information. It does not need a graphical interface. Therefore, we used the *DummyApplication.exe* application module from the BCI2000 core distribution, which exists for this purpose.

Just as with the P300 speller, we used the *Emotiv* module as the SignalSource module for eyeblink detection.

6.2.4 Interfacing with BCI2000

As mentioned in the system overview (section 6.1), the BrainBrush application is responsible for the operating logic of the whole BrainBrush system. Part of this logic is to start and set up BCI2000, to read and process BCI2000's state information while BCI2000 is running and finally to stop BCI2000 when necessary.

We implemented our own BCI2000 Operator module within the BrainBrush application, as mentioned in section 6.2.1. When either the P300 speller component or the eyeblink detection component has to be started, the BrainBrush application first starts our operator module. This module runs in a separate thread within the BrainBrush application. Next, the *Emotiv* SignalSource module is started, followed by the appropriate SignalProcessing and Application modules. All modules connect with the Operator module via TCP/IP connections. When all modules are started, the Operator module loads the appropriate settings for the various modules by processing parameter-files, supplied by the BrainBrush application. When all parameters have been set, the BCI2000 system becomes operational.

BCI2000 keeps record of it's internal state using state variables correspond-

Figure 6.3: MyPaint Graphical User Interface

ing to, for instance, event markers and classification results. These state variables are transmitted to the BrainBrush application 16 times per second via a UDP connection. This way, the BrainBrush application is able to monitor whether the P300 speller session is finished, what the P300 classification result is, or, for the eyeblink detection, whether an eyeblink has been detected or not.

When the BrainBrush system switches from painting mode to selection mode or vice versa, BCI2000 must be stopped before it can be restarted with the correct modules and parameters for the other mode. To stop BCI2000, the BrainBrush application sends a 'stop' command to BCI2000. This stop command is sent by modifying the *Running* state of BCI2000 via a UDP connection: by sending the string *'Running 0\r'* to BCI2000, the *Running*-state is set to *false*. BCI2000 detects the state-change and shuts down all modules. A new BCI2000 instance can then be started.

6.3 MyPaint

A key component of the BrainBrush system is the paint component which provides users with a virtual canvas to paint on. We chose to use the MyPaint application for this component. MyPaint is 'a fast and easy open-source graphics application for digital painters'[1] which was very suitable to be used in BrainBrush for several reasons. First, it is open source which means the source code is available and extendable; MyPaint is largely written in Python, parts are written in C++. Second, the MyPaint graphical user interface can be reduced to only the virtual canvas, showing no titlebars, menubars or toolbars. This enabled us to run MyPaint in true fullscreen mode which made it easier to integrate MyPaint in the BrainBrush system. Finally, the use of keyboard shortcuts is very common in MyPaint. Because the source code already contained the appropriate structure to handle keyboard shortcuts, new shortcuts were easily

[1] http://mypaint.intilinux.com/

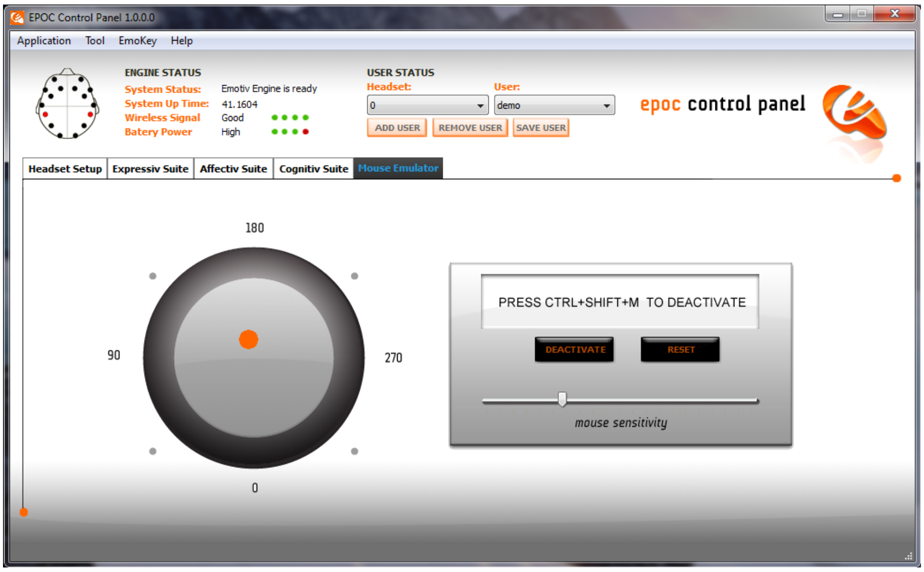


Figure 6.4: Emotiv Mouse Emulator application

added. The keyboard shortcuts for selection of specific brushes and colors provide an easy way for the BrainBrush application to interface with MyPaint for the selection of brushes and colors.

See figure 6.3 for a screenshot[2] of MyPaint. Appendix B lists the changes which were made to the MyPaint program.

6.4 Emotiv Mouse Emulator

The BrainBrush system uses Emotiv's Mouse Emulator program, see figure 6.4, to track the user's head movement as discussed in section 4.2. The Mouse Emulator translates acceleration information provided by the gyroscopes in the Emotiv EPOC headset to cursor movement so users can move the brush and create brushstrokes on the canvas by moving their heads.

The Mouse Emulator program provides one configuration option: the sensitivity can be changed using a slider. The sensitivity can be adjusted to fit the needs or preferences of specific users or environments.

Before starting the BrainBrush system, the Mouse Emulator program has to be started by the user, but the user should not activate the Mouse Emulator. Once the BrainBrush application is started, it activates the Mouse Emulator by programmatically sending a keystroke, CTRL+SHIFT+M, to the Mouse Emulator window. This keystroke activates the Mouse Emulator and users can then move the cursor by moving their heads.

[2]Provided by 'sfepa' under the *CC BY* license: http://creativecommons.org/licenses/by/3.0/

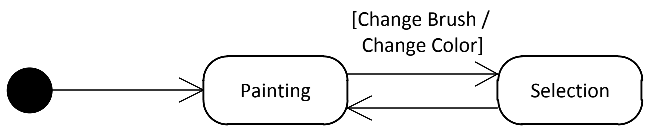


Figure 6.5: State diagram of the BrainBrush application

6.5 BrainBrush application

The BrainBrush application is a C# application which integrates BCI2000 and MyPaint, and activates the Emotiv Mouse Emulator. By integrating the separate applications as much as possible, the BrainBrush system gets the look and feel of a single application instead of a number of separate components. In chapter 3, we defined requirement R16: *'The system must be easy to set up and start'*. More specific: a layman should be able to set up, start and use the BrainBrush system. We feel that integrating the components as much as possible contributes to the fulfillment of this requirement.

As mentioned in section 6.1, the BrainBrush application implements all logic necessary for the system to operate as designed. When running, the BrainBrush application can be in one of two states: the 'Painting' state, or the 'Selection' state, see figure 6.5. In the 'Painting' state, the canvas is shown and users can paint. When the application is in the 'Selection' state, the P300 speller is shown and users can select a brush or color. BrainBrush starts in the 'Painting' state. When a user selects 'Change brush' or 'Change color', the BrainBrush application switches to the 'Selection' state. After selecting a brush or color, the application returns to the 'Painting' state.

Most of the graphical user interface in the 'Painting' state consists of the MyPaint canvas. The BrainBrush application adds a picture frame at the borders of the screen which functions as a menubar, see the design in chapter 5. This picture frame is always visible when the BrainBrush application is in the 'Painting' state and provides a number of 'active regions' which can trigger specific actions, such as starting the P300 speller, as discussed in section 5.1.1.

In the 'Selecting' state, the graphical user interface consists of the P300 speller grid, which is shown fullscreen. Whenever a P300 speller grid is shown, no mouse cursor should be visible because a moving mouse cursor might distract the user. When using a conventional mouse, it would be sufficient to move the cursor to the edge of the screen. This would ensure the cursor does not interfere with the P300 spelling session. However, with the BrainBrush system, the mouse cursor movement is controlled by movement of the user's head. Because users will most likely move their heads during a P300 spelling session, this would cause the cursor to move into view and could adversely affect the classification. Therefore, we chose to hide the cursor whenever a P300 speller grid is shown. However, it is not possible to alter the mousecursor from an external application. During the selection of brushes or colors, the *P3Speller* program is the active application and the BrainBrush application is thus unable to hide the cursor directly. Therefore, the BrainBrush application adds an invisible window on top of the P300 speller grid, and hides the mouse-cursor on this screen. This way, no mouse cursor is visible during P300 spelling sessions.

Depending on the active state of the BrainBrush application, the BrainBrush application ensures the correct BCI2000 module is running: the eyeblink detection module for the 'Painting' state, or the *P3Speller* for the 'Selection' state. As shown in figure 6.1, the Emotiv EPOC headset sends brain signals to the running BCI2000 module, which processes the realtime EEG. BCI2000 continually sends state information concerning current operating status and classification results to the BrainBrush application. The BrainBrush application filters the BCI2000 state information for specific states, such as whether an eyeblink has been detected or whether the *P3Speller* has finished classifying. Depending on the state information BCI2000 provides, the BrainBrush application can send control signals back to the active BCI2000 component. For instance, when the *P3Speller* has finished classifying a target, the BrainBrush application receives state information denoting the classification result and signals BCI2000 to stop running.

Besides integrating the components of the BrainBrush system, the BrainBrush application also determines the flow of the system. For example, the BrainBrush application determines at what point external applications have to be launched or stopped, which commands need to be executed, when those commands have to be executed, what the graphical user interface should show at any particular moment and which input modalities should be active. Figure 6.6 depicts the flow of the BrainBrush application.

When users start the BrainBrush system, the BrainBrush application first displays the graphical user interface, launches the MyPaint application and activates the Mouse Emulator program. After startup, the BrainBrush application is in the 'Painting' state.

Next, the BrainBrush applications starts tracking the cursor movement and launches the BCI2000 eyeblink detection component. Whenever the BrainBrush application receives the information that an eyeblink has been detected by the eyeblink detection module, it emulates a mouse-down or mouse-up event in Windows, which in turn triggers MyPaint to start or stop drawing brushstrokes, as discussed in section 4.1.

Emotiv's Mouse Emulator program converts acceleration information provided by the gyrosensors in the Emotiv EPOC headset to cursor movement. When in the 'Painting' state, the BrainBrush application continually tracks the position of the cursor. Whenever the cursor hovers over one of the menu options, the appropriate action is triggered. For instance, when the 'Change Brush' or 'Change color' menu option is triggered, the BrainBrush application stops the eyeblink detection module and starts the *P3Speller* module with the appropriate P300 speller grid. In case the 'Undo' or 'Redo' active regions were triggered, the BrainBrush application sends a keystroke to MyPaint to signal MyPaint to undo or redo the last painting action. Finally, when the 'New Painting' active region is triggered, a screenshot is saved to the hard drive and the canvas is cleared by sending a keystroke to MyPaint.

Detailed information about the implementation of the BrainBrush application can be found in appendix C.

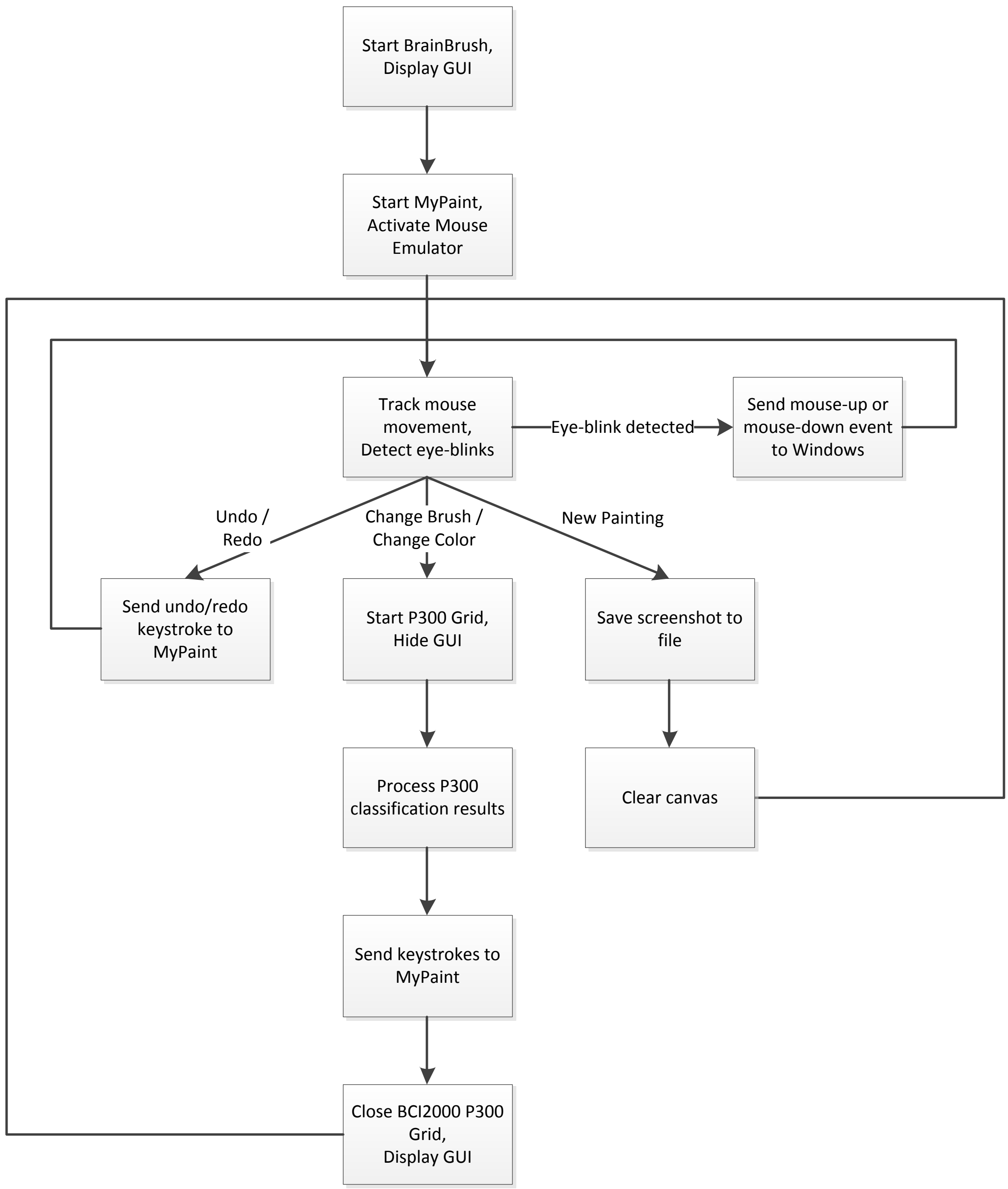


Figure 6.6: Flow diagram for the BrainBrush application

Part II

User experience research

Chapter 7

User study

We developed the BrainBrush system during two development iterations. Next, we evaluated the user experience for the BrainBrush system with a qualitative user study. We want to gain an understanding of underlying reasons and motivations for both the positive and negative opinions the participants formed about the user experience with the system and the various input modalities.

First, we discuss how we recruited the participants for the user study in section 7.1. The procedure for the experiments is outlined in section 7.2. Next, we discuss how we acquired our data during the experiments in section 7.3. This is followed by the materials used in the experiments in section 7.4. Finally, we discuss how we analyzed the data in section 7.5.

7.1 Recruitment of participants

The goal was to recruit ten people to participate in the user study. We were looking for participants who were healthy, 18 years or older, who did not suffer from epilepsy and possessed reasonable English literacy. For this purpose, a call for participants was issued in the form of a webpage listing the requirements for the participants, as well as contact information and a link to another webpage where they could choose a timeslot for the actual experiment. Participants were offered a compensation of 6 euros, unless the participant was an employee of the University of Twente. See appendix D for the call for participants.

Thirteen people signed up for the user study. We decided not to reject any participants and performed the user study with thirteen participants instead of ten. The participants received a reminder one day before their experiment. In this email, additional information was provided and they were asked to make sure to be rested and sober at the time of the experiment. Furthermore, they were asked not to use too much hairgel or hairspray before coming in for the experiment.

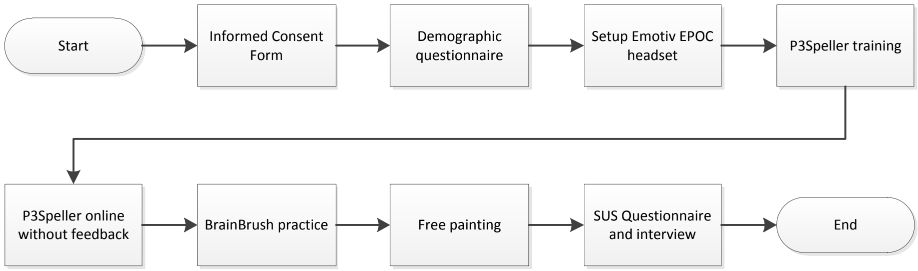


Figure 7.1: Experiment protocol overview

7.2 Experiment procedure

In order to ensure that all experiments were carried out under similar circumstances as much as possible, we defined a protocol for the experiments. By adhering to this protocol we hope to have limited accidental biases to individual results and furthermore limit the chance of the experimenter forgetting essential actions during the experiment. A schematic overview of the experiment protocol can be seen in picture 7.1.

Each participant was first asked to read and sign the Informed Consent Form (appendix E) which explained the goal, procedures, risks, discomforts, rights and the confidentiality of the recorded data. All participants agreed with and signed the Informed Consent Form. Participant 005 added a comment to the Informed Consent Form before signing, stating he did not give permission for publication of the audio of the interview.

Next, we asked the participants to complete a demographic questionnaire (appendix F) to provide us with information about their age, gender, education, their medical history, prior experience with BCIs and their creative background.

We then asked the participants to place the Emotiv headset on their heads, using an image on the Emotiv box as an example. The placement of the headset was then checked by the experimenter and corrected if necessary to ensure proper signals. The signal quality was checked using the Emotiv TestBench, a program which can be used to assess the signal quality. A crude estimate of the signal quality can be made using the status reported by the Emotiv TestBench for each sensor. Possible statusses are *black* (no signal), *red* (very poor signal), *orange* (poor signal), *yellow* (fair signal) and *green* (good signal). Following the experiment protocol, we required all channels to have a *green* status. Next, the EEG signals were visually checked for unusual artifacts like drifts or other issues, also using the Emotiv TestBench.

The participants were then asked to read the instructions for the *P3Speller* (appendix G) which describes the *P3Speller* interface, how the columns and rows flash and how a certain character can be chosen. After reading the instructions, participants were given the chance to ask any questions they might have had. To ensure the participants fully understood how the *P3Speller* interface works, a spelling session was started and with the *P3Speller* running they were explained what they were expected to do. Once the *P3Speller* was finished flashing 15 sequences (one sequence is one flash for every row and column, so 12 flashes in total) and thus started with the second character, the spelling session was stopped and the actual training session was started. In this training session, participants had to spell the word 'BRAINPOWER', using 15 sequences of flashes with a stimulus duration of 100ms, an inter-stimulus interval of 175ms and a 8s pause between characters. During this P300 speller session, the word 'BRAINPOWER' was shown at the top of the screen. Furthermore, the target letter was always shown in parentheses. Participants were not provided with feedback of the results.

During the training session, the EEG data was recorded. After the session, we used stepwise linear discriminant analysis (SWLDA) for classification of the P300 event-related potential. SWLDA identifies the suitable discriminant function by adding spatiotemporal features (i.e., the amplitude value at a particular channel location and time sample) to a linear equation based on features that demonstrate the greatest unique variance [Nijboer et al., 2008]. Details of SWLDA can be found in [Krusienski et al., 2006]. We used the *P300Classifier* tool from the BCI2000 framework to train the linear classifier. The *P300Classifier* tool was configured to use a maximum of 60 spatiotemporal features. Furthermore, a Common Average Reference filter was applied and all 14 EEG channels of the Emotiv EPOC headset were used.

After training the classifier, the classifier was used in an online spelling session where participants had to spell the word 'PAINT'. For this online spelling session, the same parameters and settings were used as for the training session. The results from this session were used to compare with the results users obtained using the *P3Speller* in the BrainBrush system. The results of this online spelling session where not shown to the participants to avoid biases in the opinions of the participants regarding the use of the *P3Speller*.

Next, the participants were given a document with instructions for BrainBrush, see appendix H. The document shows what the user interface looks like during drawing, how to move the cursor, how to start and stop drawing, how to choose a brush or color, how an action can be undone or redone and how to save a painting and start with a clean canvas. The participants were given the chance to ask questions.

Meanwhile, the experimenter started the Mouse Emulator program and set the mouse sensitivity to the optimal setting as determined during the pilot test. Figure 7.2 shows the mouse sensitivity setting: the slider was placed underneath the first 'A' of 'DEACTIVATE'. BrainBrush was then started, a new userprofile was created for the participant and the P300 classifier was loaded.

The participants were given 5 minutes to familiarize themselves with the the BrainBrush program. During this time, the threshold for the eyeblink detection was adjusted if necessary. The threshold defaults to 650. In case a participant experienced too many false positives, the threshold was lowered in steps of 50, or raised in steps of 50 in case of too many false negatives.

After 5 minutes, BrainBrush was stopped and the participants were given a document containing the description of their task for the experiment (free painting) and how they could signal the experimenter when they wanted to stop painting. See appendix I for the task description.

When the participant was ready, BrainBrush was started and the experimenter left the room to ensure the participant would feel comfortable while painting and would not get distracted by the presence of the experimenter.

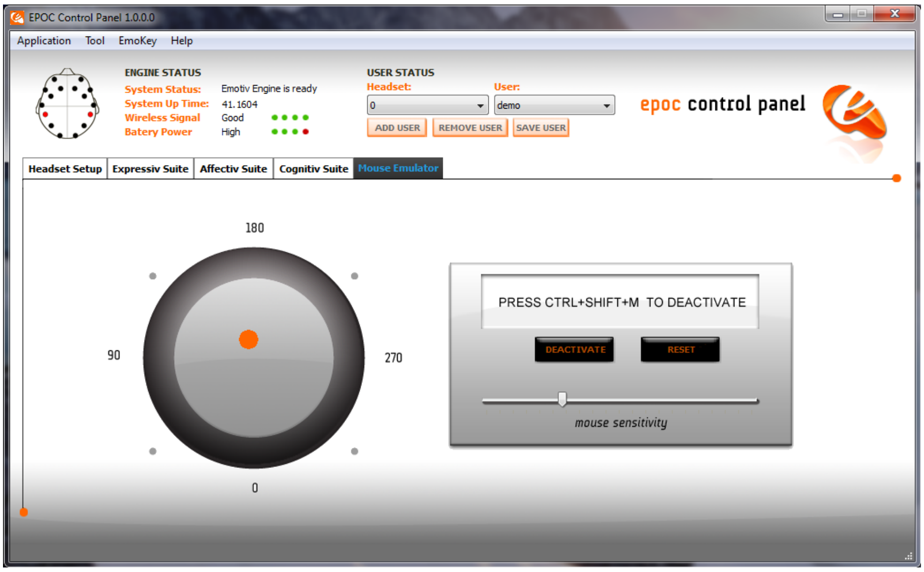


Figure 7.2: Emotiv Mouse Emulator

The participants could stop drawing at any time they wanted by ringing a bell, or if they had not stopped after 30 minutes of drawing, the experimenter would go back into the room and ask them to stop drawing and save their paintings.

The Emotiv Control Panel was then used by the experimenter to check whether the status of all EEG channels were still *green*. Participants were then instructed to take off the headset.

Next, the participants were asked to complete the System Usability Scale questionnaire. Afterwards, interviews were conducted with the participants. The audio of all interviews was recorded. The System Usability Scale questionnaire and the interviews will be discussed in section 7.3.

Finally, the participants were thanked for their participation in the user study and asked to fill out a declaration form for the monetary compensation.

7.3 Data acquisition

The dependent variables we acquired during the user study consists of the P300 copy spelling accuracy scores, the System Usability Scale scores and the interview answers.

P300 copy spelling accuracy

The participants performed two P300 copy spelling sessions during the experiment (see section 7.2): the first to record EEG to train the P300 classifier, the second to assess the accuracy participants were able to achieve using the P300 speller.

For this second session, the trained P300 classifier was used to perform online classification. The classification results were, however, not shown to the participants but written to a logfile instead.

The participants had to copy spell the word 'PAINT', which contains 5 characters. We calculated the P300 copy spelling accuracy as the percentage of correctly classified characters. So a participant who spelled 'XAINT' would have achieved 80% accuracy, a participant who spelled 'R5IZT' only 40% accuracy.

System Usability Scale questionnaire

The System Usability Scale questionnaire (SUS) was developed by John Brooke as part of the usability engineering programme in integrated office systems development at Digital Equipment Co Ltd., Reading, United Kingdom. SUS is a ten item, five point Likert scale which provides a global view of subjective assessments of usablity by using 10 statements covering different aspects of system usability such as the need for support, training and complexity [Brooke, 1996].

The ten statements consist of five positive and five negative statements. The positive and negative statements alternate to make sure each statement has to be read and an effort has to be made to give an answer. The System Usability Scale must be issued before the participant has had time to debrief or discuss the system. The SUS questionnaire, as used in the user study, can be found in appendix J.

Completing the SUS questionnaire results in a score from 0 - 100. To calculate the SUS score, 1 is subtracted from the answers to the odd items, and for the even items, the answer is subtracted from 5. The resulting scores are summed up and multiplied by 2.5.

The resulting score is, for instance, useful when comparing different versions of a software product. Is is important to note that the scores for individual items are not meaningful.

Interview

Each participant was asked a set of 13 main questions, followed by subquestions, during the interview. In some cases additional questions were asked to help clarify remarks made by the participants. The interviews were conducted in Dutch, see appendix K for the questions.

First, a couple of general questions were asked concerning the BrainBrush program and the user study. We asked what expectations the participants had and whether the system matched those expectations. This provides insight into whether participants had different expectations which could have influenced their opinions about the system. Furthermore, we asked whether the participants were able to transfer their creative ideas to the virtual canvas, if they had a specific picture in mind and if they were able to draw that picture. The answers to these questions indicate whether they felt they were able to express themselves creatively and helped answer research question RQ1 (section 1.2). We also asked why the participants stopped drawing at some point (in case they did) or if they would have continued drawing (in case they were stopped by the experimenter after the maximum allowed time for the free painting task).

The last of the general questions was whether they would buy the system and for what price. This provided insight into how much the participants liked the system and whether it would be marketable.

Next, we asked questions about the three input modalities. For each input modality, the questions consisted of (a subset of): whether the system was easy to control using that specific input modality, if the input modality was accurate, whether the use of that input modality distracted them from the other actions they had to perform and whether it was tiring to use that input modality. The answers to these questions were used to answer research question RQ2 (section 1.2).

The participants then were asked about their opinions concerning the use of BCI technology to transfer their creative ideas to the virtual canvas. Furthermore, they were asked to arrange the input modalities in the order of most pleasant to the least pleasant. Next, their opinion was asked about the way the three input modalities were combined within the BrainBrush program. These questions provided us with the data to answer research question RQ3 (section 1.2).

Finally, they were asked whether they liked the user study and/or the program.

7.4 Materials

The materials we used during the user study are the Emotiv EPOC headset, several software packages, a laptop and a experiment room.

Emotiv EPOC

The EEG headset used in the user study was the Emotiv EPOC, as this headset satisfied the conditions outlined in requirement R14 (*'The system must incorporate a portable, wireless and comfortable BCI headset that may be used by users themselves at home and which is able to track head movement'*). The Emotiv EPOC features 16 saline EEG sensors, two of which are used as reference channels, leaving 14 usable EEG channels. Furthermore, the Emotiv EPOC headset offers positional information through the use of gyrosensors.

Software

All software as described in section 6 (BrainBrush, BCI2000, MyPaint and the Emotiv Control Panel and Mouse Emulator) was used during the user study: the BrainBrush program, BCI2000 for the *P3Speller* and the eyeblink detection, MyPaint provided the paint functionality, the Emotiv Control Panel was used to assess the signal quality, and the Mouse Emulator was used to translate head movement to cursor movement.

Laptop

A Lenovo ThinkPad W700 laptop was used to run all software. This laptop has a 17 inch screen with a resolution of 1920x1200 which provided the participants with a good size canvas. The laptop contains an Intel® Core™ 2 Extreme Q9300 2.53GHz processor and 4 GB of memory. Furthermore, the laptop features a built-in color calibrator which automatically adjusts the display's color for true Pantone color. The laptop was running the 64-bit version of Windows 7 Enterprise with Service Pack 1 installed.

Experiment room

All experiments were conducted in room A128 of the Zilverling building at the University of Twente. The experiment room does not contain any windows. The lights in the room were on during the experiments. The participants were seated at a table with the laptop directly in front of them. A bell was placed next to the laptop so the participants could signal the experimenter whenever they wanted to stop the experiment. During the *P3Speller* training sessions, the practice sessions and the interviews, the experimenter sat next to the participant. During the painting task, the experimenter was outside the room.

7.5 Data analysis

The data gathered in the user study (the P300 copy spelling accuracies, SUS scores and answers to our interview questions) was analyzed to answer our research questions.

First, we used the P300 copy spelling accuracy for a rough within-subjects comparison with the P300 accuracy achieved by participants when selecting brushes and colors using the *P3Speller* during the free painting task. For participants who experienced bad P300 classification accuracy during free painting, we used this comparison to determine whether the bad P300 classification accuracy was due to the use of images instead of the traditional P300 speller interface with characters, or whether the P300 classification accuracy was also low for that participant during copy spelling.

Next, we used insight into the meaning of the SUS score, provided by Jeff Sauro[1]: after analyzing data from over 5000 users in 500 different evaluations, he concluded a SUS score above 68 would be considered above average and anything below 68 is below average. We used the SUS questionnaire to assess whether the BrainBrush system scores above or below average concerning the usability. Other than that, we only report the score so future comparable systems, perhaps even a new and improved version of BrainBrush, can be compared to the current system regarding the usability.

The final and most important data analysis was the analysis of the interview data. The audio of the interviews was recorded, as mentioned in section 7.2, and transcribed for further analysis. To be able to qualitatively analyze the opinions and remarks pertaining to a specific topic, we summarized and grouped

[1] http://www.measuringusability.com/sus.php

all opinions and remarks of all participants by topic. During the interviews we encouraged participants to elaborate on their answers to our questions. This resulted in participants sometimes not strictly answering the question, but also uttering remarks about one or more other topics of interest. During the analysis, we attributed such remarks to the correct topic and not strictly to the topic of the questions they were asked. Next, comparable remarks and opinions of participants were grouped and finally we rated the opinions and remarks from positive to negative.

Chapter 8

Results

We outlined the design of the user study and the methods used in the analysis of the data in chapter 7. In this chapter, we present the results. Information about the participants of the user study can be found in section 8.1. In section 8.2, we present the results of the P300 copy spelling session and the P300 speller accuracy during the free painting task, as well as a comparison of both accuracies. Section 8.3 contains the results of the System Usability Score questionnaire. In section 8.4, we present the results of the interviews. Then we discuss additional observations we made during the experiments which do not belong to one of the result categories in section 8.5. Finally, section 8.6 contains suggestions of participants concerning BrainBrush.

8.1 Participants

The group of participants for the user study consists of 8 males (61.5%) and 5 females (38.5%). All participants are aged 20 to 29, the average age is 24.8 (standard deviation: 2.9). 11 participants (84.6%) are right-handed, 2 left-handed (15.4%). All participants are students: 4 Bachelor-students, 7 Master-students and 2 PhD students. All participants have the Dutch nationality.

None of the participants reported relevant medical conditions. Out of the 13 participants, 2 participants (15.4%) indicated they had previously participated in BCI research. 4 participants (30.8%) indicated they regularly exercise some form of creative expression: participant 001 paints, draws and designs daily for study-related purposes; participant 012 paints once a week as a form of recreation; participant 008 regularly uses Photoshop and participant 009 sometimes plays the drawing-game 'Draw Someting' on the iPad.

8.2 P300 results: copy spelling vs BrainBrush

As described in section 7.2, an online spelling session without feedback was performed using the *P3Speller*. Participants had to copy spell the word 'PAINT'. Furthermore, during the interviews after the free painting session, the participants were asked what accuracy they thought they had been able to achieve

Table 8.1: P300 classification accuracy during copy spelling and free painting

Participant	Copy spelling Result	Accuracy	BrainBrush Accuracy
001	R6INW	40%	Regularly incorrect
002	XAINT	80%	70%
003	PACNZ	60%	80%
004	RHINT	60%	Regularly incorrect
005	R5IZT	40%	Regularly incorrect
006		-	< 50%
007	LAINT	80%	At the beginning: 50% At the end: 100%
008	RAIPT	60%	70%
009	RAINT	80%	21 out of 24 sessions correct (88%)
010	RAINT	80%	70%
011		-	Brush selection: 50-60% Color selection: 90%
012	L5BNN	20%	Choice decided beforehand: 100% Otherwise: lower
013	RGUNC	20%	25-33%
Average		56.4%	

with the brush and color selection during the free painting session. Both results are shown in table 8.1.

During the experiments of participants 006 and 011, the classifier was accidentally not loaded during the online spelling session. Therefore, no copy spelling results are available for both participants. For the other 11 participants, the average accuracy during copy spelling was 56.4% (standard deviation: 23.4%, minimum 20%, maximum 80%).

For the accuracy during free painting, not all participants were able to name a percentage of correct classifications. Average accuracies can therefore not be given. For three participants, additional explanations of their reported accuracies are needed. First, participant 007 explained she achieved only 50% accuracy at the beginning of the free painting session, but once she had learned how to focus, she achieved 100% accuracy. Second, participant 011 reported different accuracy levels for brush selection (50-60%) and for color selection (90%). Finally, participant 012 reported she was not always able to decide which selection she wanted to make before the flashes started. Whenever she had decided which option she wanted to focus on beforehand, the accuracy was 100%. If she was still deciding what option to focus on when te flashing started, the accuracy was lower. But she reported the second session was always correctly classified in the case the first classification was wrong.

The results can be used for a rough within-subjects comparison of the accuracies for the condition using the *P3Speller* with characters (copy spelling) and the condition using pictures (free painting). It must be noted that the accuracies for free painting are estimates made by the participants and are therefore not objective and may not be completely accurate. However, we can conclude that

Table 8.2: System Usability Scale questionnaire results

	Participant 001	002	003	004	005	006	007	008	009	010	011	012	013
Item 1	2	2	4	2	1	2	1	4	4	1	4	1	1
Item 2	4	5	1	2	2	4	1	2	1	2	3	1	1
Item 3	2	4	5	4	2	2	5	4	5	4	1	4	1
Item 4	1	1	1	2	1	3	1	1	1	1	4	2	1
Item 5	3	5	3	4	2	4	5	2	4	4	4	4	5
Item 6	2	2	1	1	3	2	1	2	1	2	2	2	5
Item 7	5	5	4	4	3	2	5	4	4	4	4	5	5
Item 8	3	2	2	4	4	3	2	3	2	2	5	3	1
Item 9	2	4	4	2	2	2	5	2	4	4	2	3	5
Item 10	1	2	2	3	1	2	1	1	2	1	2	2	1
Score	57.5	70	82.5	60	47.5	45	87.5	67.5	85	72.5	47.5	67.5	70

the accuracies for all but one participant are similar in both conditions: participant 012 only achieved 20% accuracy during copy spelling, but for the free painting task, she reported a significantly higher accuracy. As mentioned before, participant 012 found it difficult to decide which option to focus on before the flashes started. In case she had decided which option to focus on beforehand, she achieved 100% accuracy. We believe the low accuracy for participant 012 during copy spelling can be explained by the fact that the connection of most EEG sensors failed during the *P3Speller* copy spelling task. The bad connections were corrected as much as possible before the free painting task, so this could explain the large improvement in accuracy for participant 012.

8.3 System Usability Scale questionnaire results

The System Usability Scale score is calculated as described in section 7.3. The resulting scores are shown in table 8.2. To reiterate: for the odd statements, a higher score is better, but for the even statements a lower score is better. The average SUS score was 66.2 (standard deviation: 14.2, minimum: 45, maximum: 87.5). As mentioned in section 7.5, a score of 68 would be average, so the BrainBrush system scored a little below average. Scores for individual statements are not meaningful on their own [Brooke, 1996]. Looking at individual scores provided by the participants, 6 participants scored the system above average, 7 below average.

8.4 Interview results

We analyzed the answers to the interview questions as discussed in section 7.5: first, we summarized and grouped the answers by topic; next, we grouped comparable remarks and opinions of participants. The results of this analysis are listed in sections 8.4.1 through 8.4.12. Finally, we rated the opinions and remarks from positive to negative, see section 8.4.13.

8.4.1 Expectations

First, we asked the participants what expectations they had before the experiment and whether BrainBrush matched those expectations.

The expectations reported by the participants varied. Some mainly had general expectations like *'being able to draw by using brainsignals'* (004), *'creating a painting with my head'* (009) or *'a Paint-like program with which you can draw by doing something with your head'* (008).

Seven participants (001, 002, 006, 008, 010, 012, 013) indicated they had not expected to be able to use the movement of their heads to control the cursor. Participant 002 thought it was a nice change and an addition to not need your hands for drawing. On the other hand, she thought it would be cool to actually draw by using your brainwaves. Participants 006 and 012 indicated that moving the cursor was easier than expected, due to the fact that it could be controlled by moving their heads. Participant 006 did expect it to be easier to draw using the system. Participant 013 expected having to move the cursor by hand.

Participant 001 thought everything worked better than expected; she expected everything would work less smoothly.

Participant 005 thought the program was quite comprehensive. For instance, he had not expected as much different brushes to choose from. Other than that, the functionality was as he expected. He would, however, rather see the *P3Speller* not being used.

Participant 010 had also expected less options; he had expected to be able to draw and maybe select some colors, but nothing else.

Participant 011 had expected to be able to choose a color by *'simply aiming his head at it and the color would be chosen'*. He had not expected having to count the flashes. He therefore expected controlling the system to be much easier; he found controlling the system to be hard.

8.4.2 Transfer creative ideas to the virtual canvas

Next, we asked the participants whether they felt they were able to transfer their creative ideas to the virtual canvas.

Four participants (003, 007, 009, 010) indicated they were able to transfer their creative ideas to the virtual canvas. Participant 010 noted the result was a bit cruder compared to when using a mouse or a pen.

Three participants (002, 006, 008) somewhat managed to transfer their creative ideas. Participant 002 felt like her head movements were not neatly enough and that she would do better after having had more practise. Participant 008 also noted some problems controlling the cursor: drawing smooth lines was hard.

Six participants (001, 004, 005, 011, 012, 013) were unable to transfer their creative ideas to the virtual canvas. Participants 012 and 013 indicated, as participants 002 and 008 did above, that problems with controlling the cursor were the main cause for the inability to transfer their creative ideas. For participants 001 and 004, problems with the *P3Speller* were the main cause. Participant 005 indicated he was used to be able to directly do what he thinks of, and he was unable to do that using this program.

8.4.3 Ability to draw the picture in the participant's mind

We then asked the participants whether they had a specific picture in mind while drawing and if they were able to draw that picture. If not, we asked them whether they adapted the picture they had in mind and how they adapted it.

Eleven out of the thirteen participants had a specific picture in mind, the other two participants (001 and 007) indicated they did not always have a specific picture in mind: sometimes they did, sometimes they did not. Participant 007 noted she found that the drawings she started without a picture in mind and where she made something up along the way worked out better than the pictures where she started with a specific picture in mind: *'there was a blot and I thought: that looks like such and such a shape and I drew the rest around it'*.

Three participants (002, 003, 009) indicated they were able to draw the picture they had in their minds. Participant 003 noted that maybe not all pictures were equally beautiful, but at least it was clear what they depicted.

Two participants (001, 013) were unable to draw the picture they had in mind. Participant 001 indicated her drawings ended up being abstract. Participant 013 adjusted what he wanted to draw according to the color he was able to select instead of the color he wanted to select (*'I was unable to select yellow, so the sun ended up being grey and then it ended up being clouds'*) and he also adjusted to the movements he was able to make with his head (*'the first clouds ended up being birds'*).

Participant 006 noted she was not completely able to draw what she had in mind. For her also, not being able to select the right color had an influence: *'I wanted red, it became blue, so I decided to start painting the sea'*.

Participants 008 and 010 indicated they thought it was difficult to draw smooth lines or circles. Participant 008 just kept on trying, but participant 010 adjusted what he wanted to draw: *'the circle ended up being a rounded square, so then I thought, I can make a head out of that'*.

Participant 004 was partly able to draw the picture he had in mind: he simplified the idea (*'trees exist in all shapes and sizes'*).

Both participants 011 and 012 where unable to draw what they had in mind for the first couple of drawings, but ended up being able to make a couple of pictures they found to be reasonably succesful. Participant 011 said he did not adjust his ideas, but instead *'tried to make do with the possibilities he had at that time'*.

Participant 005 indicated he was unable to do certain things (*'drawing a house was not possible'*); he adjusted by drawing other types of pictures (*'very simple childlike things'*, like clouds and suns).

8.4.4 Time spent on free painting

We measured how much time each participant spent performing the free painting task, see table 8.3. Eleven out of the 13 participants stopped painting before the maximum allowed painting time of 30 minutes passed. Two participants (009, 010) were stopped by the experimenter after the 30 minute limit passed. The average amount of time participants spent free painting was 21:58 minutes (standard deviation: 5:12 minutes, minimum: 14:00 minutes, maximum: 30:00

Table 8.3: Amount of time spent free painting

Participant	001	002	003	004	005	006	007
Time	16:30	24:00	24:30	17:30	24:30	18:30	24:00

Participant	008	009	010	011	012	013
Time	15:30	30:00	30:00	24:30	22:00	14:00

minutes). We asked the eleven participants who stopped painting before the 30 minute limit passed why they stopped. The two participants who had to be stopped by the experimenter were asked whether they would have wanted to continue painting for much longer.

Participant 001 indicated that at some point she understood how the program worked and then decided to stop. Participant 002 felt like she had been drawing for a very long time and she thought she should not continue drawing for too long (*'I thought, oh, I have been drawing for a long time, I will just quickly do this, I will do this, oh, I have been drawing for a really long time now, I will just do this, and then I stopped'*).

Three participants (003, 005, 007) stopped due to, among other things, physical discomfort due to the Emotiv EPOC headset.

For 6 participants (003, 004, 005, 008, 012, 013) annoyances and/or frustrations played a role in the decision to stop drawing. The annoyances and frustrations had a couple of different causes: a loss of focus resulting in bad *P3Speller* accuracy (003), *'simply was unable to do it'* (005), *'the fact that it reacts a bit with difficulty, not completely the way you want'* (008), problems with the Emotiv EPOC headset which resulted in the EEG-modalities not working correctly (012, 013).

Four participants (006, 007, 008, 012) indicated they did not have any inspiration to draw more paintings.

Furthermore, 3 participants (011, 012, 013) experienced problems with the Emotiv EPOC headset which contributed to the decision to stop drawing. Participant 011 indicated he would have liked to finish the last drawing he was working on, but that was impossible due the problems with the headset (bad signals and the *P3Speller* grid flashing at varying speeds, possibly due to a low battery level). Participant 013 would have liked to finish the experiment by writing that he finally had managed to select red, using the color red, because he had been unable to select the color red earlier in the experiment.

8.4.5 Purchasing the system

Next, we asked the participants whether they would want to buy the BrainBrush system if it was on the market for 300 euros, including the headset, and whether they would want to keep on drawing with the system.

All participants indicated they would not buy the system for 300 euros. Three participants (003, 006, 009) added painting in general is not their thing, participant 013 noted he is better able to draw using MS Paint. Furthermore, participant 001 indicated the *P3Speller* made the program too difficult to control and that she therefore would not consider buying it.

When asked whether they would consider buying the system for a lower price, only two participants were able to name a price: participant 002 would buy it for 100 euros (*'because it is cool'* and *'to impress friends'*), participant 004 thinks of the system more like a gadget and would then pay 40 to 50 euros for it. Furthermore, two participants (001, 003) indicated it should be *'a bit cheaper'* than 300 euros, but they were unable to name a price. Participant 003 noted he would probably want to experiment with the system. The other participants would not want to buy the system.

Participant 005 was only interested in buying the Emotiv EPOC headset.

Participants 011 and 013 indicated they failed to see the use of a drawing-application which uses BCI. On the other hand, participant 013 thought the system was a nice thing to play with and he thought it might be nice to practise using it in order to be able to operate it properly.

Participant 007 would like to have the system when there are more possibilities for it, instead of only a drawing application. For instance, a Harry Potter game in which you *'can think certain things and he goes up-up-and-away'*.

For participant 008 the system was *'not enjoyable enough to overcome the annoyances it brings'*.

Furthermore, two participants (005, 012) noted it might be better suitable for people with a physical disability.

8.4.6 Eyeblinks

We then asked the participants how they felt about the use of eyeblinks to control the BrainBrush system, whether they felt the system was easy to operate using eyeblinks, whether their eyeblinks were detected correctly, whether the use of eyeblinks distracted them from the other actions they had to perform to control the system and if the use of eyeblinks was tiring.

Eight participants (001, 002, 004, 007, 008, 009, 010, 013) were generally positive about the use of eyeblinks as an input modality. We first mention the results of this group of participants.

Participant 001 thought it worked well, handy and easy, and she thought she had quite good control over it. Participant 002 noted she thought the use of eyeblinks worked very smooth, but she felt she had to be careful because she apparently blinks a lot with her eyes. Participant 007 mentioned she thought blinking her eyes was a very easy and obvious action. Participant 008 also noted he thought the action was easy. Participant 009 noted the eyeblinks required some effort at times, but he did not feel it was very difficult. Participant 010 indicated he had to get used to it, but in general he thought it worked well. For participant 013, the use of eyeblinks was easy at the beginning of the experiment, but later on it stopped working due to problems with the Emotiv EPOC headset. He thought it was not much more difficult than clicking a mousebutton.

For most of the participants from the group of eight participants who were generally positive about the use of eyeblinks, the eyeblinks were detected reasonably well. The eyeblink detection at times failed for everyone: most of those eight participants experienced both false positives and false negatives. Participants 001, 004 and 013 mentioned that whenever there was a false positive, or if they blinked accidentally, the error was easily corrected. Participant 008 mainly encountered false positives, not really false negatives. For participant 010, however, there was a low amount of false positives, but more false negatives. Participant 013 encountered quite a lot of false positives and he noted this depended on what he was doing at the time: when moving his head, he got more false positives. This may be due to the observed problems with the Emotiv EPOC headset during his session.

Seven participants (001, 002, 004, 007, 008, 010, 013) out of this group of eight participants felt the use of eyeblinks did not distract from the other actions they had to perform to control the system. Participant 002 even mentioned it felt 'very natural to do so, as if you always have been using your eyeblinks'. Participant 008 mentioned he feels like every action you have to perform, other than moving a mouse because you are already very used to that, is a bit distracting because you have to think about it momentarily. Participant 009 did feel the use of eyeblinks was a bit distracting because he had some problems seeing the on-screen feedback whether the brush turns on or off. This is related to the placement of the text 'brush on' or 'brush off' on the screen.

From this group of eight participants, one participant (008) thought the use of eyeblinks was tiring: he felt having to continually be cautious about not blinking his eyes accidentally and thinking about blinking when he did need to blink was mentally tiring. Participant 007 also mentioned it was sometimes a bit difficult to have to think about not blinking accidentally, but she thought it was not too bad. The other participants (001, 002, 004, 009, 010, 013) did not feel it was tiring. Participant 002 even indicated she felt like she could do it all the time.

The other five participants (003, 005, 006, 011, 012) were not positive about the use of eyeblinks. Participant 003 noted he did not feel in control; he even felt there was no correlation with him blinking his eyes and thought the brush turned on and off randomly. Whenever he was unable to turn the brush off he then had to move the brush via the picture frame to avoid ruining the drawing. Because of this, 003 found the use of eyeblinks to be distracting. Participant 005 indicated he felt it was difficult when drawing: he experienced so many false positives that he felt it was troublesome. He felt he needed more practise to be able to use his eyeblinks properly. Participant 006 also noted she had a hard time using her eyeblinks because she regularly experienced false positives. Participant 011 felt it was not functioning the way it is supposed to: out of every 5 deliberate eyeblinks, only 1 was recognized as such by the system. Furthermore, he experienced a lot of false positives. For participant 012 it worked reasonably well, but at times she experienced quite a lot of false positives, which she felt was annoying. Participant 012 also felt it was troublesome that she could not blink her eyes at any time she wanted; after a while she experienced dry eyes and she felt that was distracting and tiring. Participants 003, 005, 006 and 011 did not feel using their eyeblinks was tiring.

8.4.7 Head movement

Next, we asked the participants how they felt about the use of head movement to control the BrainBrush system, whether they felt the system was easy to operate using head movement, whether the use of head movement distracted them from the other actions they had to perform to control the system and if the use of head movement was tiring.

Nine participants (001, 003, 006, 007, 008, 009, 010, 012, 013) indicated they thought moving their heads to control the cursor was easy. Participant 006 noted she thought it felt very *'natural'*, participant 008 thought it worked *'natural'* and *'cool'*. Participant 012 mentioned she thought it worked intuitively and that she did not have to think about it. According to participant 007, although it is easy to understand, it was not very easy to develop a technique for using it.

The issues most mentioned by the participants were the fact that the cursor needed to be realigned regularly and issues with the precision.

The problem of needing to realign the cursor was mentioned by six participants (003, 006, 009, 011, 012, 013). Participant 003 felt the use of head movement was extra tiring because of this. Participant 006 thought it was distracting. Participant 012 mentioned realigning the cursor was a bit difficult at times because of the menu options at the top and bottom: one could accidentaly hit one of the menu options. Participant 013 found realigning the cursor was frustrating because it had to be done quite often.

Remarkably, participant 010 mentioned he used the possibility of realigning the cursor to his advantage: when drawing at the top of the screen he realigned the cursor to be a bit higher. He felt that this way he did not get tired as quickly as he otherwise would.

Seven participants (002, 003, 005, 007, 008, 010, 013) mentioned the level of precision they were able to achieve using the movement of their heads. Five out of these seven participants (003, 005, 007, 008, 010) mentioned diagonal lines, smooth lines and circles being difficult to draw. Participant 002 indicated she tought the precision would get better once she would have used it more often. Participant 013 noted controlling the cursor using your head is not as accurate as when using your hands.

Most participants did not feel the use of their heads to move the cursor was tiring. However, 3 participants (003, 004, 005) did. Participant 003 felt this was mainly due to the constant need to realign the cursor and the fact that he regularly had to move the cursor via the picture frame to avoid accidentally drawing lines because of bad functioning eyeblink detection. Participants 004 and 005 reported some physical discomfort: they experienced tense neck muscles.

Six participants (002, 004, 005, 006, 008, 010) indicated they felt having to move their heads in order to control the cursor was a bit distracting at times. As mentioned before, participant 006 felt having to realign the cursor was distracting. Participant 008 felt it was distracting because he was able to draw straight lines, but not able to draw smooth lines. Participant 002 noted she lost sight of the complete picture whenever she needed to turn her head. Participant 004 mentioned the need to constantly be aware of the fact you are using your head as a pointer and that you can not look away for a moment. Participant 005 thought he accidentally blinked because of moving his head. Participant 010 thought the combination of the head movement with the eyeblinks was difficult at times, specifically when his head was turned to an extreme position and he then was wondering whether the eyeblink was detected or not.

This final remark of participant 010 is mainly due to a problem which was also mentioned by participant 005, and earlier on by participant 009: the 'brush on' or 'brush off' feedback is always being displayed in the same position, which is a bit above the middle of the canvas. This visual feedback is therefore difficult to see when drawing at the edges of the canvas.

8.4.8 P3Speller

We then asked the participants how they felt about the use of the *P3Speller* to control the BrainBrush system, whether they felt the system was easy to operate using the *P3Speller*, whether the system detected their desired choices correctly and if they felt the use of the *P3Speller* was tiring.

Six participants (002, 003, 005, 009, 011, 012) indicated they thought using the *P3Speller* to select brushes and colors was easy. Participant 002 estimated approximately 30% of the classifications was wrong, which was more than she expected. She expected this was due to the fact that she found it more difficult to concentrate after a while; she found it mentally tiring. Participant 003 estimated about 20% of the classifications was wrong. He thought this was a good result and did not really find it tiring. Participant 005 indicated he thought using the *P3Speller* was easy because you *'do not have to do anything, you just have to look and count'*. However, a lot of the classifications for participant 005 were wrong. Instead of trying again, he then settled for the brush or color he was given and drew something else than he had planned to.

At some point during the experiment of participant 009, the classification was wrong three times in a row. The other 21 classifications were correct though. Participant 009 noted he felt like it worked effortlessly. Participant 011 also indicated he felt the *P3Speller* was easy to use. However, during his experiment, 50-60% of the classifications for the brushes was correct compared to 90% for the colors. The reason why participant 012 found the *P3Speller* easy to use was because it did not last very long and you only *'have to stare at the thing and wait, that is fine'*. For her, the classification was often correct; if the classification was not correct the first time, it was correct at the second try.

Participant 012 did give a possible explanation for this: after choosing 'Change brush' or 'Change color' the flashes start almost instantly. She found it annoying that she was unable to choose which option she would want to select before the flashing started. Participant 012 suspected this to be the reason why sometimes the first try did not yield the desired result. Participants 003 and 008 reported the same issue.

Seven participants (001, 004, 006, 007, 008, 010, 013) indicated they thought the use of the *P3Speller* was not easy. For participants 001, 004, 006 and 013 the classification was often wrong. Participants 001 and 013 indicated they thought it was frustrating when the classification was constantly wrong even though they felt like they did try to focus and count correctly. Participants

001 and 007 observed that often when the classification was wrong, the chosen option was adjacent to the correct option. This was also observed by participant 003. Participant 004 felt the *P3Speller* process took too long: *'apparently quite often the wrong option is chosen and then you have to watch those pictures for another 30 seconds before you can continue'*. Participant 008 mentioned he found it troublesome the color-options flashed by using negative colors: for instance, red flashes green and vice versa. He found this was distracting and increased the difficulty.

Seven participants (001, 002, 006, 007, 008, 009, 010) indicated they thought the use of the *P3Speller* was tiring. For participants 001 and 009 this was mostly a physical discomfort: they experienced problems with their eyes. The other five participants especially found it mentally tiring. Participants 006 and 007 only found it slightly mentally tiring; participants 002, 008 and 010 indicated they found it to be more mentally tiring. The cause was reported to be the need to continually concentrate well in order to get the desired result. Participant 010 noted that whenever you make a mistake during drawing, you can select undo. If you make a mistake while using the *P3Speller*, however, you have to do a complete series of flashes again.

Participants 005 and 007 indicated they found the learning curve to be steep. Participant 005 noted he, for instance, did not know whether staring was good or bad and what to pay attention to. Participant 007 also experienced a learning curve during the experiment. In the beginning approximately 50% of the classifications was wrong, but at the end, when she learned that *'it goes wrong when you are not concentrated enough and not staring at 1 point'*, she thought it was easy to use. From then on, the classification was correct each time.

8.4.9 BCI technology for creative expression

Next, we asked the participants how they felt about the use of BCI technology to transfer their creative ideas to the virtual canvas.

Nine participants (001, 002, 004, 005, 006, 007, 009, 011, 013) indicated they thought the use of BCI technology for creative expressions was *'fun'* or *'cool'*. Participant 002 said *'you are kind of watching what you are doing from a distance, that provides a different kind of input, i think'*. Participant 009 mentioned he thought it was *'very nice that you can just think stuff and it then ends up on the screen'*. Participant 007 thought it was mainly fun as a form of recreation rather than to actually transfer an idea from your head to the paper.

Participant 001 noted she felt it was not very easy to use. Participants 004 and 005 indicated they felt using the BrainBrush system was a lot slower / more cumbersome compared to, for instance, pen and paper. Participant 011 noted he felt it would take too much time when having to be productive.

Out of the remaining four participants (003, 008, 010, 012), who did not indicate they thought the use of BCI for creative expressions is fun or cool, participants 003 and 008 also indicated they would rather use a mouse or pencil for these kind of activities. Participant 012 just wants to draw in real life.

Participants 004, 005, 006, 010 and 013 thought it was interesting and/or fun because it is something new.

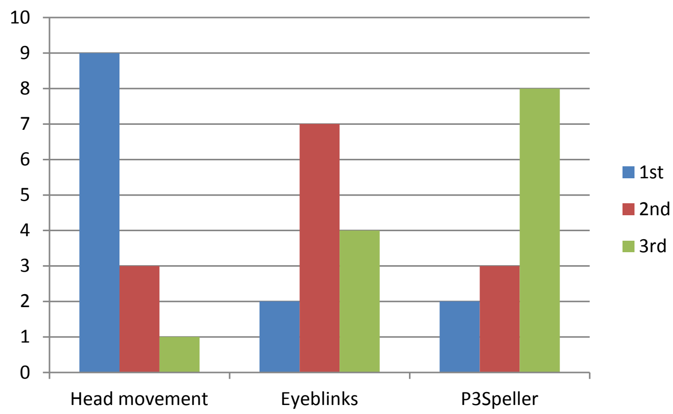


Figure 8.1: Head movement was most often ranked most pleasant, eyeblinks most often second and the P3Speller most often last

8.4.10 Arrangement of input modalities

We asked the participants to arrange the input modalities from most pleasant to least pleasant. Figure 8.1 shows how often each input modality was placed on the first, second or third place by participants.

Seven out of 13 participants (001, 004, 006, 007, 008, 009, 010) chose the same arrangement for the input modalities. From most pleasant to least pleasant: head movement, eyeblinks, *P3Speller*.

Two other participants (003 and 012) also found head movement to be the most pleasant input modality, but they found the *P3Speller* to be more pleasant than eyeblinks.

Participant 002 chose the arrangement: eyeblinks, *P3Speller*, head movement. Participant 013 also thought the eyeblinks were the most pleasant input modality, but he preferred head movement over the *P3Speller*.

Two participants (005, 011) found the *P3Speller* to be the most pleasant input modality, followed by head movement and finally eyeblinks.

8.4.11 Combination of the modalities

Next, we asked the participants how they felt about the combination of the three input modalities (eyeblinks, head movement and *P3Speller*) to control the application.

Nine participants (001, 002, 003, 006, 007, 009, 010, 011, 012) were positive (*'fun'*, *'worked well'*, *'nice that it is not all the same'*, *'logical choice'*, *'well balanced'*) about the combination of the different modalities within the BrainBrush program.

Participant 001 noted she thought it would be good to have some time in between returning to the canvas after using the *P3Speller* and the ability to start the *P3Speller* again. This might avoid accidentally selecting 'Change brush' or 'Change color' immediately upon returning to the canvas.

Participant 002 indicated she thought the program contained a lot of different functionalities and she felt this clearly adds value to the program. Furthermore, she noted the combination of head movement with eyeblinks definitely works well.

Participants 003 and 005 indicated they would have liked an alternative to the use of eyeblinks, such as keeping your eyes closed for a certain period of time or to clench your teeth.

Five participants (004, 005, 008, 009, 013) noted the *P3Speller* could be replaced by using head movement and confirming a choice with eyeblinks. These are the same participants who were not predominantly positive about the combination of the input modalities.

Participant 006 thought selecting an option, such as 'Change brush' or 'Undo', by hovering the cursor over the option was obvious. Participant 008, however, felt this was not handy and he would like to see the need to confirm the selection of an option by with an eyeblink. He felt that is a more 'traditional' way of control and would prevent accidental selections when moving the cursor.

Participant 010 indicated he felt the combination of modalities was well balanced. According to him, the action one has to perform the most, drawing, should be the easiest to do. He felt this was the case, followed by turning the brush on/off and finally switching tools.

Participant 012 felt the combination of modalities was very obvious and she felt it was good to have the ability to turn the brush on and off independently from everything else, and not, for instance, by *'having to look somewhere'*.

8.4.12 Fun

Finally, we asked the participants whether they enjoyed participating in the user study.

All 13 participants enjoyed participating in the user study. Participant 003 said he got childhood memories from when he was making his first drawings by hand. Participant 006 felt it was funny to be able to do something like this without using your hands. For participant 011, it was one of the nicest things he had done lately, even though it did not work very well for him. Participant 004 however, noted the fun was fading, probably because of the problems he experienced trying to control the program. But as an experience he thought it was very interesting and fun. Participant 005 indicated he always likes to try new things and therefore also liked this experience. Participant 012 said she thought it was fun to do once, but she did not feel like doing it again some time. She did not enjoy the use of BrainBrush for painting that much because she prefers to paint in real life.

8.4.13 Summary of interview results

To give a clearer overview of the user experience of the participants, the results from the interviews have been summarized.

For each of the topics in the previous sections, the remarks made by the participants were categorized as being positive (+), neutral (+/-) or negative

(-). The results are shown in table 8.5. The meanings of a positive, neutral or negative result per topic are described in table 8.4. If the meaning of a neutral result is not explicitly specified, it is in between the positive and negative results.

For the eyeblinks, head movement and *P3Speller*, a +, +/- or - was awarded according to the general remarks of the participant being positive, neutral or negative. Depending on the remarks of the participant concerning the accuracy and whether the modality was distracting or tiring, the result was modified to be more positive or negative where necessary, but never more than one step. In other words, a positive general opinion never changed to a negative result, or vice versa.

The topic 'Arrangement of input modalities' was not included in this summary because the remarks of the participants could not be categorized as being positive, neutral or negative.

8.5 Additional observations

During some of the experiments, we made additional observations which could have influenced the performance of the system and the user.

First, for participant 001, the status of one of the EEG sensors changed to *orange* during the experiment, indicating the connection was poor. This was corrected before starting the main task by adding contact lens solution.

Second, while checking the EEG signals of participant 003 during the setup stage of the experiment, the experimenter noticed the amplitude of the EEG signals during eyeblinks were almost triple the size of the amplitude for previous users. Because the current eyeblinkdetection algorithm uses a fixed template with a much smaller amplitude, accurate eyeblinkdetection was not achieved for participant 003.

Third, a couple of hours after the experiment, participant 004 indicated he noticed he felt the experiment had been more tiring than he previously thought: he felt like he had been 'driving for four hours'. Furthermore, he indicated he might have trained the *P3Speller* incorrectly. He experienced a sort of 'tunnel-effect' due to continuous staring at the target character; he tried to prevent this from happening by not looking at the target directly, but next to it. He indicated he 'more or less saw the target flashing in the corner of his eyes'. This might not be sufficient for a good P300 response.

Fourth, during the experiment of participant 007, the status of two EEG sensors changed to *orange* (poor signal): once after the online *P3Speller* session and once after the practise session, just before the main task. Both times the status of the EEG sensors could be corrected to *green* (good signal) by adding contact lens solution. After performing the main task, the sensors appeared not to be having a very good connection: whenever the participant smiled, the statuses of all sensors changed to *orange* and back to *green* when she stopped smiling.

Fifth, the experimenter noticed participant 011 had issues with his eyes during the *P3Speller* training session. Furthermore, when the participant stopped the main task, he showed the experimenter the *P3Speller* was flashing at a variable speed. It seemed likely the battery of the Emotiv EPOC was low, resulting

Table 8.4: Criterion for attribution of a positive, neutral or negative value to user's experiences

Topic		Criterion
Expectations	+	System worked as expected or better
	+/-	
	-	Expected more of the system
Transfer creative ideas to the virtual canvas	+	Able to transfer ideas to the virtual canvas
	+/-	Somewhat able to transfer ideas to the virtual canvas
	-	Not able to transfer ideas to the virtual canvas
Ability to draw the picture in the participant's mind	+	Able to draw the picture in the participant's mind
	+/-	Able to draw some pictures, but not all, or not completely able to draw the pictures in the participant's mind
	-	Not able to draw the picture in the participant's mind
Time spent on free painting	+	Spent the full 30 minutes painting and would have wanted to continue painting or stopped painting but not due to bad experiences
	+/-	
	-	Had negative reasons to stop painting, including physical discomfort or problems with the headset
Purchasing the system	+	Buy the system for 300 euros
	+/-	Buy the system for a lower price
	-	Not buy the system as it is
Eyeblinks, head movement and P3Speller	+	Positive experience
	+/-	
	-	Negative experience
BCI technology for creative expression	+	Positive remarks
	+/-	
	-	Negative remarks
Combination of modalities	+	Positive remarks
	+/-	
	-	Negative remarks
Fun	+	Was a fun experience
	+/-	Was fun for one time, but would not want to do it again
	-	Did not like the experience

Table 8.5: Summary of interview results

	001	002	003	004	005	006	007	008	009	010	011	012	013	+	+/-	-	Total
Expectations	+/-	+	+	+	+/-	+/-	+	+/-	+	+/-	+/-	+/-	+	6	7	0	13
Transfer creative ideas	-	+/-	+	-	-	+/-	+	+/-	+	+	-	-	-	4	3	6	13
Draw picture in mind	-	+	+	+/-	+/-	-	+/-	+/-	+	+/-	+/-	+/-	-	3	7	3	13
Time spent painting	+/-	+	-	-	-	+	-	-	+	+	+/-	-	-	4	2	7	13
Purchasing the system	+/-	+/-	+/-	+/-	-	-	-	-	-	-	-	-	-	0	4	9	13
Eyeblinks	+	+	-	+	-	-	+	+/-	+	+	-	+/-	+	7	2	4	13
Head movement	+	-	+/-	+/-	+/-	+	+/-	+	+	+	-	+	+	7	4	2	13
P3Speller	-	+	+	-	+/-	-	+/-	-	+	-	+/-	+	-	4	3	6	13
BCI for creative expression	+	+	+/-	+	+/-	+	+	-	+	+	+	-	+	9	2	2	13
Combination of modalities	+	+	+	-	-	+	+	-	+	+	+	+	+/-	9	1	3	13
Fun	+	+	+	+	+	+	+	+	+	+	+	+/-	+	12	1	0	13
+	5	8	6	4	1	5	6	2	10	7	3	3	5				
+/-	3	2	3	3	5	2	3	4	0	2	4	4	1				
-	3	1	2	4	5	4	2	5	1	2	4	4	5				
Total	11	11	11	11	11	11	11	11	11	11	11	11	11				

in such behaviour.

Sixth, with participant 012, it was very difficult to put the headset on correctly; it took more than 15 minutes before the status of all EEG sensors was *green*. After the online *P3Speller* session, all EEG sensors, with exception of the reference-sensors, reported a *red* (very poor signal) or *orange* status. This was corrected as much as possible, but multiple sensors remained problematic. After the main painting task, barely any sensors reported any form of connection at all: some sensors reported a red status, others had a *black* status (no signal). The participant reported she had issues controlling the system at the end of the main task, which likely is due to the bad connection status.

Finally, during the last experiment, with participant 013, the status of two sensors changed to *yellow* (fair signal) after the *P3Speller* online session. The status of one additional sensor was constantly changing between *yellow* and *green*. More contact lens solution was added, but this did not really improve the connection. When the participant stopped performing the main task, because he felt the eyeblink detection was not working anymore, the status of all EEG sensors was *black*.

The fact that issues with the Emotiv EPOC headset and the signal quality mainly occurred during the last three experiments is striking. It is possible the headset suffered from some kind of defect which influenced the ability of the participants to control the system.

8.6 Recommendations by participants

Several participants gave recommendations for further development of the BrainBrush system.

First, suggestions were made to change the eyeblink detection because of the amount of false postives experienced by the participants. Participant 006 suggested to change the eyeblinks to a longer blink of about 1 second to avoid involuntary blinks from being detected. Participant 005 thought it might be good to use a series of blinks, instead of a single eyeblink, again to avoid the brush being turned on or off when it is not supposed to. Participant 005 also suggested replacing the eyeblinks, but by clenching your teeth. He figured that, as one would not accidentally clench his teeth, this would never go wrong. Participant 008 also suggested replacing the eyeblinks to avoid false positives, he suggested squeezing your fists to turn the brush on or off.

Second, the *P3Speller*'s rows and columns start flashing directly after the grid is fully visible. Participants 003, 008 and 012 suggested changing the *P3Speller* to allow users to check which option they would want to focus on, before the flashing starts. Five participants felt the *P3Speller* was unnecessary: participants 004, 005, 008, 009 and 013 suggested to replace the *P3Speller* by a similar grid with options, but to select by using head movement to point the cursor at the desired option and confirm by using eyeblinks.

Third, some participants reported accidental selections of menu options, such as starting the *P3Speller*. For participant 001 the accidental selections occurred after just having selected a new brush or color using the *P3Speller*, returning to the canvas and then accidentally having the cursor pointed at either ‘Change brush' or 'Change color', resulting in the *P3Speller* being started again and having to select a brush or color. Her suggestion was to introduce a timeout after returning to the canvas. This way, users would be able to move the cursor to avoid accidentally starting the *P3Speller* or selecting one of the other menu options. Participant 008 suggested changing the way menu options are selected. Instead of selecting an option by hovering the cursor over it, he thought having to confirm the selection by blinking your eyes would be better.

Fourth, not being able to turn the system off for a moment was an issue for participant 005. He suggested creating a place on the screen where the cursor could be 'parked' and nothing would happen. This way, users would be able to relax every now and then while painting.

Finally, four participants (005, 008, 009, 010) noted the 'brush on'/'brush off' feedback was not always clearly visible. Participants 005, 009 and 010 indicated the feedback was not visible when, for instance, one was drawing near the border of the canvas. They suggested the feedback should always be presented near the location of the cursor. Furthermore, participant 008 noted the visibility of the feedback depended on what was drawn at the location where the feedback was presented. When the colors of the painting are similar to the color of the feedback, it is not clearly visible.

Chapter 9

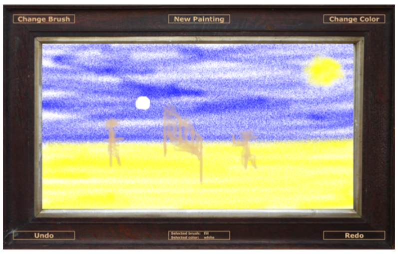
Discussion

In this chapter we discuss the results of the user study. First, we discuss the results with respect to the three research questions in sections 9.1 through 9.6. In these sections, several issues with the BrainBrush system emerge which will be discussed in section 9.7. Finally, we will provide technical recommendations for improvements to the BrainBrush system in section 9.8.

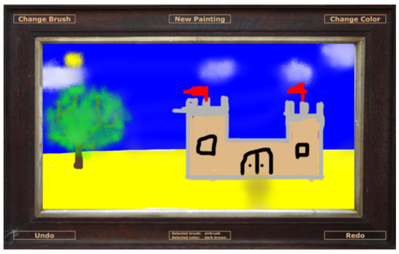
9.1 Ability for creative expression

In this thesis, we set out to create a multimodal interactive system which persons can use to express themselves creatively. Our results show that the participants who did not suffer from low reliability of the input modalities (mostly problems controlling the cursor and low P300 speller accuracy) too badly were able to express themselves creatively. When comparing some of the drawings made by participants to drawings made using the 'Brain Painting' system of Münßinger et al [Münßinger et al., 2010], we feel participants were able to create more elaborate and complicated drawings with BrainBrush. Compare the paintings in figures 9.1a, 9.1b and 9.1c with the example drawing from the Brain Painting system in figure 9.1d. The ability to move the cursor freely and the fact BrainBrush offers a larger variety of brushes and colors seem to contribute to this.

However, results from the interviews show that almost half the participants were not able to transfer their creative ideas to the virtual canvas and a further quarter of the participants were somewhat able to. Participants mostly attributed their inability to transfer their creative ideas to a combination of problems controlling the cursor, they were unable to draw smooth lines for instance, and low P300 speller accuracy, which made it difficult to select the desired brushes and colors. These issues are discussed further in sections 9.7.1 and 9.7.2, respectively.

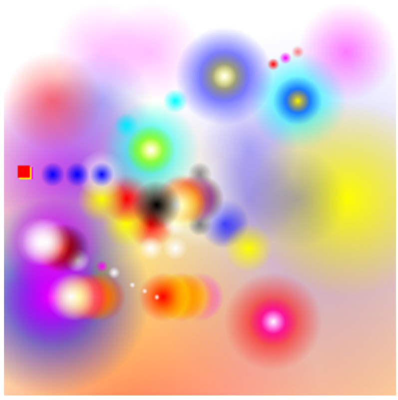


(a) BrainBrush example 1: a game of beach volleyball, by participant 007



(b) BrainBrush example 2: a castle, by participant 009

(c) BrainBrush example 3: a sailing boat, by participant 013



(d) Brain Painting example, from http://www.brain-painting.com

Figure 9.1: Drawings made with the BrainBrush and Brain Painting systems

9.2 Contribution of eyeblink detection to the user experience

Our second aim was to elucidate how the input modalities contribute to the user experience. Most participants were positive about the use of eyeblinks as an input modality even though the eyeblink detection was faultless for none of the participants. Participants felt the errors made by the eyeblink detection were mostly easily corrected, e.g. by blinking again. In other words, the cost of an error was low. Furthermore, they felt blinking their eyes was an easy action to perform. One participant even noted he felt it was not much more difficult than clicking a mousebutton. Another noted using eyeblinks felt very natural, as if she had always been using her eyeblinks.

For most participants, using this input modality was not distracting. However, for some participants the accuracy of the eyeblink detection algorithm was low, at which point the eyeblinks became distracting from the other tasks they had to perform to use the system. Participants did not feel using eyeblinks as an input modality was tiring: one participant even felt she could do it all the time.

Negative user experiences concerning the use of eyeblinks as an input modality mainly concerned bad accuracy. Primarily, this was caused by a large amount of false positives. When users experienced a large amount of false positives, they often attributed this to themselves blinking accidentally. As a result of this, they did not dare to blink at all, which in turn resulted in physical discomfort for some participants. The issue of bad eyeblink detection accuracy is discussed further in section 9.7.3.

For most BCIs, eyeblinks are considered noise in the EEG signal, masking important information or other control signals. Most existing eyeblink detection algorithms aim to detect eyeblinks so they can be ignored or removed from the signal ([Shoker et al., 2005], [Schlögl et al., 2007], [Erfanian and Mahmoudi, 2005]), not to use eyeblinks as a control signal.

However, Chambayil et al. developed a virtual keyboard which uses eyeblinks as a control signal to select characters [Chambayil et al., 2010]. They reported perfect eyeblink detection for their system, allowing them to achieve a correct spelling rate of 1.00 character/min. They did however not report on the user experience of the use of eyeblinks as a control signal.

To conclude, our results show that the use of eyeblinks as a control signal is promising with respect to the user experience. As long as the eyeblink detection accuracy is reasonably high, and the cost of an error is low (e.g. the effects of a false positive are not too big), the user experience is positive.

9.3 Contribution of head movement to the user experience

Furthermore, we researched how the use of head movement as an input modality contributes to the user experience. Most participants were positive about the use of head movement to control the cursor while painting. They thought it was cool to be able to control the cursor by moving their heads. Users also thought it felt intuitive, which seems to be confirmed by Bates et al., who observed the use of head movement to control a cursor was extremely quick to learn [Bates and Istance, 2003]. Users mentioned the use of head movement felt natural, which was also found by Evans et al. [Evans and Blenkhorn, 1999].

We can only speculate how these results would transfer to users with disabilities such as multiple sclerosis and spinal cord injuries. LoPresti et al. found that patients with such disabilities had limited range of neck motion [LoPresti et al., 2000]. This could complicate the use of head movement for cursor control. For patients with limited range of neck motion to be able to use BrainBrush, it might be necessary to increase the sensitivity setting in the Mouse Emulator program, which effectively increases the speed of cursor movement. This way, they would still be able to reach the borders of the canvas. However, the downside would be reduced accuracy since low speed movements would be harder to perform.

The user experience was negatively influenced by issues with the cursor becoming misaligned with the head of the user. Users were able to realign the cursor, but felt this was distracting, or even frustrating. We suspect the misalignment is caused by issues with the Emotiv EPOC headset, or with the Emotiv Mouse Emulator program, which is used to convert head movement to cursor movement.

Furthermore, a large number of participants noted that it was difficult to draw a smooth line, circle or diagonal line. Some participants thought the cause was them not being able to move their heads smoothly. However, we suspect this is caused by issues with the Emotiv Mouse Emulator program.

The issues of misaligned cursors and the lack of smoothness are discussed further in section 9.7.1.

In conclusion, the contribution of head movement to the user experience can be positive: users thought it was cool, intuitive, and felt natural. However, issues with misalignment of the cursor and the smoothness of lines should be solved to further improve the user experience.

9.4 Contribution of the P300 speller to the user experience

Next, we researched the influence of the P300 speller to the user experience. Unlike with the other two input modalities, the user experience with the P300 speller was mainly negative. Some participants experienced physical discomfort due to staring at the P300 speller grid. Participants also noted that whenever the classification yielded a wrong result, they had to concentrate for a complete new series of flashes. For some participants the classification was often wrong and they then got frustrated.

Furthermore, participants felt having to continually concentrate when using the P300 speller was mentally tiring. This seems to contradict the findings of Zickler et al. when researching the usability of the 'Brain Painting' system, which uses the P300 speller as the only input modality. Zickler et al. reported subjective workload was low to moderate [Zickler et al., 2012]. It could be the case Zickler et al. used less sequences of flashes, thereby reducing the time users have to concentrate and possibly lower the workload and increasing the user experience. However, the amount of flashes used by Zickler et al. was not reported. Another factor could be that using the Emotiv EPOC headset, P300 accuracy is lower compared to systems like the Biosemi ActiveTwo [Nijboer et al., 2012]. Lower accuracy could lead to users trying to concentrate harder to get the correct classification result, which would have an adverse effect on the user experience.

Participants noted they did not know how to focus on the flashing images, for instance whether staring was good or bad. One participant made a remark which illustrates the importance of knowing how to focus: at the beginning of her experiment, half the classifications made by the P300 speller were wrong; at the end, when she had found a way to focus, all classifications were correct. This particular participant learned how to focus during the experiment, but not all participants managed to figure that out.

Almost a quarter of the participants mentioned they felt the flashes started too soon after the P300 speller grid was displayed. The flashes started when the participants were still deciding which brush or color they would want to select. Therefore, they were unable to focus on the desired option from the beginning and this is expected to have resulted in lower classification accuracy.

The issues of the P300 speller are discussed further in section 9.7.2.

To conclude, the contribution of the P300 speller to the user experience was mainly negative. Reasons for the negative user experience were: physical discomfort, it was frustrating and mentally tiring, participants did not know how to focus and the flashes started too soon. If these issues can be resolved or at least reduced, the user experience could improve.

9.5 Contribution of the combination of the input modalites to the user experience

We also researched the contribution of the combination of the input modalities to the user experience. We asked the participants to rank the input modalities by how pleasant they were. Furthermore, we asked their opinion about they way the three input modalities were combined within BrainBrush. We found that most participants were positive about the combination of the three modalities in the BrainBrush system. They felt the input modalities were combined in a logical way, and found the modalities to be well balanced: one participant commented he felt the actions one has to perform most often should be the easiest. He felt this was the case with drawing being the easiest, using head movement, followed by turning the brush on or off, using eyeblinks, and finally the hardest part, switching brushes or colors using the P300 speller.

Five participants however argued the P300 speller was obsolete and it could be replaced by a combination of head movement and eyeblinks. The brushes and colors could be presented in a grid, just as with the P300 speller, but then the cursor should be pointed at a brush or color and the selection should then be confirmed by blinking the eyes. This suggestion is discussed further in section 9.7.2, together with the P300 speller issues reported above.

9.6 Added value of BCI

Finally, we researched the added value of BCI for a multimodal interactive system for creative expression. We found that, even though all participants reported problems with the BCI technology incorporated in BrainBrush, most participants thought the use of BCI was fun, cool and interesting, mainly because it was something new. One participant described the experience as: *'you are watching what you are doing from a distance, that provides a different kind of input'*, another felt it was *'very nice that you can just think stuff and it then ends up on the screen'*. However, participants noted it was much slower and more cumbersome than, for instance, using pen and paper. This issue is discussed further in section 9.7.6.

The fact that most participants were positive about the use of BCI shows the current potential: even though the technology is not yet perfect, people are interested in using BCIs. As pointed out in the Future BNCI Roadmap: a gamer can choose a BCI due to the novelty, increased challenge, and richer user experience, although the reliability and information transfer rate (technical issue) are much lower than for a traditional input device [Future BNCI Roadmap, 2012].

Van de Laar et al. researched the influence of unreliable input on fun when playing a simple game in which a hamster has to be guided to the exit of a maze while the amount of control the user has over the hamster is varied [van de Laar et al., 2011b]. They concluded a perfect BCI is not necessarily a prerequisite for games and the shortcomings of a BCI could be turned into a challenge for the user to create a fun game. Therefore, it is not necessary to wait for the technology to become perfect and to work just as well as traditional input devices in terms of reliability and information transfer rate before putting it on the market; BCIs can be introduced on the market while meanwhile work on improvements continues. On the other hand, if a BCI performs too bad, this could result in people not using the BCI and forming a negative opinion about BCIs in general. Then it might be questionable if people would give BCI technology a second chance once new developments have improved the technology.

9.7 Discussion of issues

Several issues with the BrainBrush system emerged in the previous parts of the discussion: issues with the cursor control, the P300 speller, the eyeblink detection, the selection of menu options, the 'Brush on/off' feedback, the fact participants felt they could not be productive with the BrainBrush system and finally issues with the Emotiv EPOC headset. In this section, we discuss these issues further.

9.7.1 Cursor control

In BrainBrush, users control the cursor by moving their heads. The movement of their heads is detected by gyrosensors in the Emotiv EPOC headset and then translated to movement of the cursor by Emotiv's Mouse Emulator program. During the user study, two issues concerning the cursor control were reported by the users: the lack of smoothness of the cursor movement and the fact the cursor is sometimes misaligned with the head of the user.

Lack of smoothness

The users reported the system seemed to have a preference for straight lines: circles ended up being rounded squares and diagonal lines ended up being shaped like stairs. The cause of this issue seems to be the Mouse Emulator program. It would seem this program is intended to be used to move the cursor from a certain point A on the screen to point B on the screen, for instance to select a button. In this case the exact route the cursor travels from point A to B is not important; getting to the destination in a controlled manner is more important. It would seem the software therefore uses some threshold to filter out small movements in direction X when moving the cursor in direction Y and vice versa. This way, the cursor moves from point A to point B in nice, straight lines. This behaviour may be appropriate in such circumstances, but when the route travelled by the cursor is as important as the destination, like in BrainBrush, such behaviour is unwanted.

We found the lack of smoothness influenced the ability for creative expression negatively (section 9.1). Therefore, we recommend writing new software to replace the Mouse Emulator program and interpret the data from the gyrosensors without filtering small movements, or to reduce the filter.

Misaligned cursor

The second issue with the cursor control reported by the users was the fact the cursor sometimes becomes misaligned with the location they were looking at: for instance, the user is looking at the center of the screen, but the cursor is located at the right side of the screen. Users can correct this by continuing to move their heads in the direction of the cursor. At some point, the cursor will hit the side of the screen and stop moving. Users can then realign their heads with the cursor.

The gyrosensors in the headset measure the movement of the headset and report this movement to the software in the form of acceleration information. The software reads these acceleration values a certain number of times per second and uses this to calculate how far the cursor should be moved during that time interval. Apparently, this process of translating head movement to cursor movement introduces errors.

A possible explanation of the perceived error could be the software not reading the acceleration information often enough. This could result in the software reading a lower acceleration value than the real acceleration level achieved during that interval. This would then result in misalignment.

Another explanation could be the fact the gyrosensors have a built-in limit of amount of acceleration which can be measured. If the real acceleration were to be higher than the measured acceleration, the software would not get accurate acceleration information and thus not be able to calculate the correct amount of cursor movement. The cursor would then be misaligned.

Finally, the problem could be caused by the issue we reported with respect to the lack of smoothness: small movements in direction X are filtered out when moving in direction Y. When filtering out such movements, the user's head automatically becomes misaligned with the cursor because the head moves in direction X but the cursor does not follow.

We feel this last explanation is the most likely cause of the issue of the misaligned cursors. Since replacing the Emotiv Mouse Emulator program with new software is already necessary to solve the lack of smoothness, we recommend trying whether the new software also solves the misalignment issue.

9.7.2 P300 speller

The classification accuracy for the P300 speller during free painting was low for almost half the participants, as shown in section 8.2. As mentioned in section 9.4, low classification accuracy resulted in frustrations and made the use of the P300 speller mentally tiring because participants had to concentrate for a complete new series of flashes to correct the mistake. Furthermore, staring at the P300 speller grids caused physical discomfort. It is therefore important to achieve higher classification accuracy. With higher classification accuracy, users less often have to repeat a P300 speller session to correct erroneous classifications. If users have to perform less P300 speller sessions, the use of the P300 speller is expected to be less mentally tiring and expected to cause less physical discomfort due to staring. The low P300 speller accuracy may have been caused by a number of factors.

First, as discussed in section 9.4, a number of participants noted they felt it was difficult to learn how to focus. This contradicts the common belief that the P300 is an involuntary evoked response. We hypothesize that users may indeed benefit from learning. By performing more P300 training sessions, we expect users will learn how to focus on the flashing images more easily. This is supported by Aloise et al. who demonstrated that using a P300 based BCI can train user's attention and positively affect the P300 event related potential [Aloise et al., 2011a].

Second, the colors of the images in the P300 speller grids are inverted when they are being flashed. For the color selection grid, the red square flashes green and the green square flashes red, for instance. As discussed in section 9.4, at least one participant felt this was distracting and therefore increased the difficulty. Other methods to flash the images, for instance hiding the flashed images resulting in black rows and columns, should therefore be considered.

Third, the Emotiv EPOC may not be the best EEG headset for the detection of the P300 event related potential [Nijboer et al., 2012]. If an other EEG headset would become available which satisfies the requirements defined in requirement R14 (*'The system must incorporate a portable, wireless and comfortable BCI headset that may be used by users themselves at home and which is able to track head movement'*), we recommend using that headset.

Finally, the flashes start almost immediately after the P300 speller grids are displayed. At that time, users may not have decided which option they want to select yet. Therefore, they are then unable to focus on the desired option from the start of the first flashes. By adding a delay of 5 seconds after the grid is displayed, users should be able to decide which brush or color they would like to select.

We suspected the use of images in the P300 speller grids instead of the more traditional layout with characters could have had an adverse effect on the classification accuracy. However, we concluded there were no significant differences between the P300 classification accuracy during copy spelling and during free painting, see section 8.2. Furthermore, research by Hoffmann et al. also used P300 grids with images and they were able to achieve high classification accuracy [Hoffmann et al., 2008]. Therefore, we do not expect the use of images instead of characters in the P300 speller grids caused the low classification accuracy.

To summarize, first, we recommend to consider other flash methods for the images. Second, we recommend to look for other EEG headsets which meet the requirements and are better for P300 detection. Third, we recommend to add a delay before the columns and rows start flashing. Finally, we recommend users are allowed more time to train how to use the P300 speller.

9.7.3 Eyeblink detection

To paint with BrainBrush, users have to turn the brush on and off by blinking their eyes. Their eyeblinks are detected by the eyeblink detection algorithm. However, during the experiments, many users experienced bad eyeblink detection accuracy due to false positives.

The eyeblink detection algorithm works by continually comparing two EEG channels to a predetermined template. Whenever the difference between the template and one of the EEG channels is below a certain threshold, that section of EEG is classified as being an intentional eyeblink. The predetermined template is based on the intentional eyeblinks of one person. During the experiments we observed that some participants had significantly different EEG signals during intentional eyeblinks. For one participant, the maximum peek during the eyeblink was approximately three times higher compared to the maximum peeks for other participants. The duration of an intentional eyeblink also varies for participants. Because of these differences in EEG signals during intentional eyeblinks, the threshold for the eyeblink detection algorithm can not be set very low since otherwise intentional eyeblinks would also not be classified as such. However, the consequence of a higher threshold is the fact that random EEG noise or unintentional eyeblinks can also be classified as an intentional eyeblink.

During the practice sessions of the experiments, we observed the EEG signals during intentional eyeblinks had the same characteristic shape for all participants: a steep peek which then slowly levels off. An improvement for the eyeblink detection could be to take into account this characteristic shape of the EEG signals during an intentional eyeblink. The presumption is that unintentional eyeblinks are shorter than intentional eyeblinks. By also including a minimum duration for an eyeblink, unintentional eyeblinks could then be filtered out. This way, the eyeblink detection algorithm might produce less false positives because random noise and unintentional eyeblinks would less often be classified as intentional eyeblinks. Because the algorithm would also be less vulnerable to differences in the maximum of the peek in the signal and differences in the length of an intentional eyeblink, it would also produce less false negatives.

Because of the influence of bad eyeblink detection on the user experience, distraction and physical discomfort, we recommend studying the eyeblinks of more subjects to search for the similarities between subjects in EEG signals during intentional eyeblinks and for the differences between intentional and unintentional eyeblinks. This can then be used to develop an improved version of the eyeblink detection algorithm.

9.7.4 Menu option selection

The menu options in BrainBrush, located on the picture frame, can be selected by hovering the cursor over the menu option. The selection of the menu option does not have to be confirmed in any way. In section 5.1.1, we discussed this design choice and the looming Midas-Touch-Problem: the risk of unintentional selections [Ihme and Zander, 2011]. We argued that, because the menu options in BrainBrush are located at the borders of the screen with space in between the various menu options, the risk of unintended selections would be limited under normal circumstances.

Due to the problems with the cursor misalignment discussed earlier, users had to move the cursor to the border of the screen more often, and for other reasons than to select a menu option: they had to carefully navigate the cursor around the menu options to the border of the screen in order to correct the alignment. One user noted he experienced unintentional selections because he accidentally moved the cursor over the menu options. He would therefore like to see the need to confirm the selection of a menu option with an eyeblink.

Ihme et al. describe a similar issue for eye-gaze systems. They mention that when using dwell-times for making selections, unintentional selections can be made due to spontaneous dwellings at random objects. Zander et al. solved this issue by using an active BCI where selections had to be confirmed using imagined hand movement [Zander et al., 2010], but because of the extra cost for the user, Ihme et al. researched the possibility for using a passive BCI to confirm selections [Ihme and Zander, 2011].

The difference between the situation described by Ihme et al. and the situation in BrainBrush is the fact that the issue in BrainBrush is caused by a faulty input modality and not by uncontrollable actions of the user. If the software used to translate head movement to cursor movement would not result in misalignments of the cursor, or if the cursor could be realigned in a different way, the Midas-Touch-Problem would not necessarily be an issue in BrainBrush.

Therefore, we do not recommend adding a mechanism for confirmation of the selection of a menu option.

9.7.5 Brush on/off feedback

When an eyeblink is detected while painting, the BrainBrush program displays a 'Brush on' or 'Brush off' message on the screen. This message informs the user about the fact that an eyeblink has been detected. This message is important for the user, because the eyeblink detection algorithm is not faultless and therefore the user needs confirmation that his eyeblink has been detected, and whenever an eyeblink has been detected, but the user did not blink, the user has to be informed that the brush has been turned on or off to minimize confusion.

The feedback is shown just above the center of the canvas, as shown in picture 9.2. Participants indicated they found the feedback to be very useful when they were drawing near the feedback. However, when they were not drawing near the location of the feedback but, for instance, on the left of the canvas, they were unable to see the feedback properly. Furthermore, the visibility of the feedback depends on what the user has drawn at the location where the feedback is shown. If the user has painted at the location of the feedback, using similar colors as the color of the feedback, the feedback is also not visible properly.

By always showing the feedback near the location of the cursor and adding a border with a different color to the text of the feedback, we can ensure the 'Brush on' and 'Brush off' messages are properly visible. Because the changes needed for this improvement are minimal, we recommend implementing them.

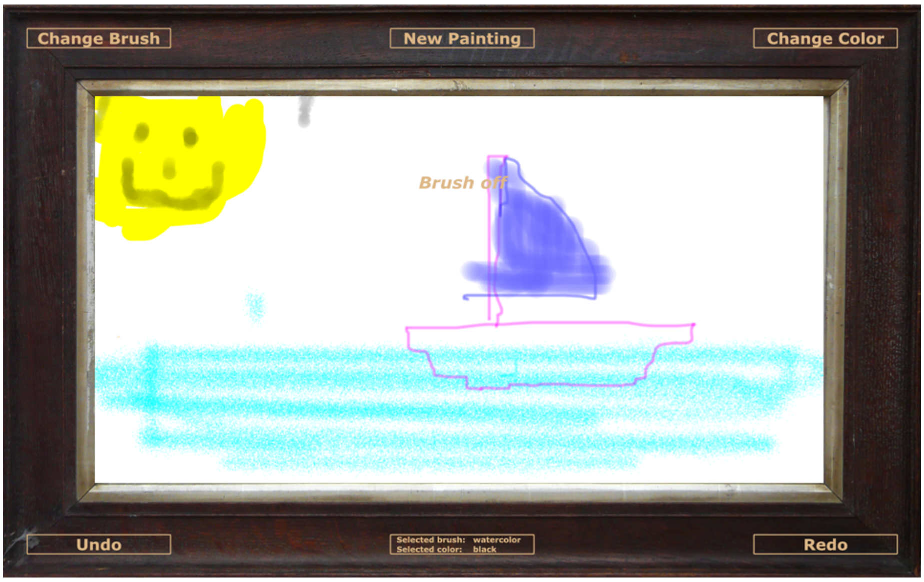


Figure 9.2: Screenshot showing the 'Brush off' feedback

9.7.6 Productivity

During the experiments, participants felt their progress was slow while painting. They felt this was due to the use of the BCI technology. Furthermore, most participants failed to see the use of a multimodal painting program which uses BCI. Remarks made by the participants during the interviews indicated they saw the system as something with which you have to be productive. Because they were unable to be productive using the BrainBrush system, most participants felt the application was of no use to them. Because it was of no use to them, most participants would not buy the system, not even for low prices.

Several participants were asked whether they liked the system as just being a funny application, not an application with which one has to be productive. The responses were more positive then. Therefore, it is important the BrainBrush system is used in a fitting situation: preferably not in situations where people have to be productive.

Instead of changing how BrainBrush is used, the concept of the BrainBrush system could be changed. If the system were to provide a game-like environment, users are not expected to feel the need to be productive and are therefore expected to be more positive. Another advantage of a game-like environment would be that a perfect BCI is not necessarily a prerequisite for games [van de Laar et al., 2011b]. Therefore, unreliability issues with, for instance, the P300 speller may have a less negative impact on the user experience.

9.7.7 Emotiv EPOC headset

Nearly a quarter of the participants in the user study stopped painting due, in part, to physical discomfort caused by the Emotiv EPOC headset. Nijboer et al. reported the same issue with the Emotiv EPOC headset in their headset comparison study [Nijboer et al., 2012].

Another almost quarter of the participants stopped painting due to technical issues with the Emotiv EPOC headset: low battery level even though the battery status indicator indicated there should be enough power left, and bad EEG electrode connections.

In case users experience physical discomfort due to the headset or other hardware issues, this has a negative impact on the user experience and could result in people not using the system. Therefore, it is important for companies such as Emotiv to keep improving the quality of their EEG headsets.

9.8 Technical recommendations

In section 9.7 we recommended a number of future changes to be implemented for the BrainBrush system in order to improve the usability and user experience.

First of all, there are three minor changes which are expected to have a direct and positive effect on the user experience. The visibility of the 'Brush on' and 'Brush off' feedback has to be improved by always showing the feedback near the location of the cursor and by adding a border to the text. Furthermore, when users select 'Change brush' or 'Change color', the P300 speller grids must start flashing after a delay of 5 seconds to allow users to determine which option they would like to select.

Next, research has to be done to determine the influence of other flash methods for the P300 grids. The outcome of this research could result in changes to the BrainBrush system if better flash methods were found.

Furthermore, to solve the lack of smoothness in controlling the brushes, new software has to be written to replace the Mouse Emulator program. Hopefully, this will also solve the issue of the misaligned cursors. Otherwise, this issue has to be investigated further.

Finally, it remains to be investigated how intentional eyeblinks can be properly detected and distinguished from unintentional eyeblinks. Using the outcome of this research, an improved version of the eyeblink detection module has to be implemented.

After implementing these changes, the SUS questionnaire could be issued again and the results be used to compare the usability of the improved version with the current version of BrainBrush.

Chapter 10

Conclusions

In this thesis, we set out to develop a multimodal interactive system which allows persons to express themselves creatively and included Brain-Computer Interfacing technology. Using this multimodal interactive system, we wanted to research how the different modalities contribute to the user experience and whether BCI has an added value.

We developed the BrainBrush system, which lets users paint on a virtual canvas using their head movement for brush control, eyeblinks to turn the brush on and off, and the P300 speller to select different brushes and colors. We defined a number of requirements before developing the system, and we feel the final BrainBrush system meets those requirements.

A user study with thirteen participants showed that the BrainBrush system does not enable all users to express themselves creatively. The group of users who were able to achieve good control over all three input modalities were able to expres themselves creatively and made nice paintings. However, for all users to be able to achieve this, the reliability of the input modalities must be improved.

The head movement modality was considered to be the most pleasant. However, misalignment of the cursor and a lack of smoothness negatively influenced the user experience. By replacing the Emotiv Mouse Emulator program with new software to translate head movement to cursor movement, the user experience is expected to improve.

After the head movement modality, the use of eyeblinks to turn the brush on or off was considered to be the most pleasant. In general, the user experience for this input modality was positive. However, the user experience can be further enhanced by performing more research into the topic of intentional eyeblink detection and improving the detection.

The P300 speller was considered to be the least pleasant input modality. Unlike with the other two input modalities, the user experience was not positive: using the P300 speller was considered to be mentally tiring and it caused physical discomfort and frustration. Improvements to the P300 training setup and to the BrainBrush system have been suggested which are expected to improve the user experience.

We can conclude the multimodal aspect of the system was good for the user experience. The combination of the three input modalities within the

BrainBrush system was considered to be positive: it was a logical combination and the input modalities were well balanced.

Finally, concerning the value of BCI for a multimodal interactive system for creative expression, we can conclude that BCI does have an added value: even though there were some issues due to the BCI modality, the use of BCI was considered to be fun, cool and interesting.

We feel the BrainBrush system in its current state offers a good basis, and with the suggested improvements, the user experience should improve further.

10.1 Future research

We tested our BrainBrush system with healthy participants. However, patients with disabilities could also be an interesting target audience for BrainBrush. The 'Brain Painting' system was tested with patients and the results of this research were positive, but available options were limited and the cursor could not be moved freely, thereby limiting the artistic freedom of the painter [Münßinger et al., 2010]. Our approach differs from the approach used in the 'Brain Painting' system. Where 'Brain Painting' uses a P300 speller as the only input modality, BrainBrush combines three input modalities to offer the painter the advantage of free cursor movement, and to make the painter less dependent on the P300 speller input modality, which most participants in our study found to be the least pleasant input modality. It would be interesting to research whether patients are able to use the BrainBrush system for creative expression. Patients should be able to use the P300 speller with good accuracy, as the use of a P300 speller for patients was already shown in the 'Brain Painting' research. As for the use of eyeblinks: even severe ALS patients retain muscle functions around the eyes [Takeshita et al., 2003]. Whether the EEG during an eyeblink of patients has the same characteristics as for healthy people would have to be researched. The third input modality, head movement, could pose a challenge for patients with limited range of neck motion. Changes to the BrainBrush system in terms of increased sensitivity for head movement may be necessary which would limit the accuracy of the cursor movement. Furthermore, it remains to be seen whether the Emotiv EPOC headset can be mounted on the heads of patients. If a patient is bedridden, for instance, the Emotiv EPOC might be difficult to put on.

For healthy users, it would be interesting to build an application which is purely meant for entertainment. One possibility would be to implement a game like the immensely popular smartphone game 'Draw Something' where one player has to draw something, and another player has to guess what it is. If users were able to use a BCI to play such a game, they are less likely to think about the program in terms of productivity, and more in terms of enjoyment. Since most participants in our study felt using BCI technology in the BrainBrush system was fun and cool, and since the game Draw Something is very popular, the combination of the two could be promising.

Finally, how cool would it be if users could picture shapes in their minds and a BCI would recognize the shapes and put them on the painting canvas? Esfahani et al. researched the possibility to capture a mental representation of a shape using a BCI [Esfahani and Sundararajan, 2011]. More concrete, they investigated the feasibility of using a BCI to distinguish between five primitive shapes (cubes, cylinders, spheres, cones and pyramids) imagined by the user. They developed a classification method to distinguish these primitive objects using the EEG of the subjects. During their experiment, subjects had to imagine one of the five primitive objects. Their classification method was able to achieve 44.6% accuracy, which is significantly better than chance accuracy (20%), but not yet well enough to be used in a painting application. So, for now, this is merely a harbinger of things to come, but it is interesting to think about the idea of using such a method in a painting application as it could be very useful for patients and it certainly would be very cool for healthy users.

Bibliography

[Aloise et al., 2011a] Aloise, F., Aricò, P., Schettini, F., Lucano, E., Salinari, S., Babiloni, F., Mattia, D., and Cincotti, F. (2011a). Can the p300-based bcitraining affect the erps? *International Journal of Bioelectromagnetism*, 13(3):148–149.

[Aloise et al., 2011b] Aloise, F., Aricò, P., Schettini, F., Riccio, A., Risetti, M., Salinari, S., Mattia, D., Babiloni, F., and Cincotti, F. (2011b). A new p300 no eye-gaze based interface: Geospell. In Babiloni, F., Fred, A. L. N., Filipe, J., and Gamboa, H., editors, *BIOSIGNALS*, pages 227–232. SciTePress.

[Bates and Istance, 2003] Bates, R. and Istance, H. (2003). Why are eye mice unpopular? a detailed comparison of head and eye controlled assistive technology pointing devices. *Universal Access in the Information Society*, 2:280–290. 10.1007/s10209-003-0053-y.

[Brooke, 1996] Brooke, J. (1996). Sus - a quick and dirty usability scale. In Jordan, P. W., Thomas, B., McClelland, I. L., and Weerdmeester, B., editors, *Usability Evaluation In Industry*, pages 189–194. CRC Press.

[Brunner et al., 2010] Brunner, P., Joshi, S., Briskin, S., Wolpaw, J. R., Bischof, H., and Schalk, G. (2010). Does the 'p300' speller depend on eye gaze? *Journal of Neural Engineering*, 7(5):056013.

[Campbell et al., 2010] Campbell, A., Choudhury, T., Hu, S., Lu, H., Mukerjee, M. K., Rabbi, M., and Raizada, R. D. (2010). Neurophone: brain-mobile phone interface using a wireless eeg headset. In *Proceedings of the second ACM SIGCOMM workshop on Networking, systems, and applications on mobile handhelds*, MobiHeld '10, pages 3–8, New York, NY, USA. ACM.

[Chambayil et al., 2010] Chambayil, B., Singla, R., and Jha, R. (2010). Virtual keyboard bci using eye blinks in eeg. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2010 IEEE 6th International Conference on*, pages 466 –470.

[Dupee and Werthner, 2011] Dupee, M. and Werthner, P. (2011). Managing the stress response: The use of biofeedback and neurofeedback with olympic athletes. *Biofeedback: Fall 2011*, 39(3):92–94.

[Erfanian and Mahmoudi, 2005] Erfanian, A. and Mahmoudi, B. (2005). Real-time ocular artifact suppression using recurrent neural network for electro-encephalogram based brain-computer interface. *Medical and Biological Engineering and Computing*, 43:296–305. 10.1007/BF02345969.

[Esfahani and Sundararajan, 2011] Esfahani, E. T. and Sundararajan, V. (2011). Classification of primitive shapes using braincomputer interfaces. *Computer-Aided Design*, (0):–.

[Evans and Blenkhorn, 1999] Evans, G. and Blenkhorn, P. (1999). A head operated joystick – experience with use. In *Proceedings of the CSUN conference on technology and persons with disabilities*, California State University, Northridge.

[Farwell and Donchin, 1988] Farwell, L. and Donchin, E. (1988). Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*, 70(6):510 – 523.

[Future BNCI Roadmap, 2012] Future BNCI Roadmap (2012). Future bnci: A roadmap for future directions in brain / neuronal computer interaction research.

[Galán et al., 2008] Galán, F., Nuttin, M., Lew, E., Ferrez, P. W., Vanacker, G., Philips, J., and Millán, J. d. R. (2008). A Brain-Actuated Wheelchair: Asynchronous and Non-Invasive Brain-Computer Interfaces for Continuous Control of Robots. *Clinical Neurophysiology*, 119(9):2159–2169.

[Guger et al., 2009] Guger, C., Daban, S., Sellers, E., Holzner, C., Krausz, G., Carabalona, R., Gramatica, F., and Edlinger, G. (2009). How many people are able to control a p300-based braincomputer interface (bci)? *Neuroscience Letters*, 462(1):94 – 98.

[Gürkök et al., 2011] Gürkök, H., Hakvoort, G., and Poel, M. (2011). Evaluating user experience with respect to user expectations in brain-computer interface games. In Müller-Putz, G. R., Scherer, R., Billinger, M., Kreilinger, A., Kaiser, V., and Neuper, C., editors, *Proceedings of the 5th International Brain-Computer Interface Conference, BCI 2011, Graz, Austria*, pages 348–351, Graz, Austria. Verlag der Technischen Universität Graz.

[Gürkök et al., 2011] Gürkök, H., Plass-Oude Bos, D., van de Laar, B. L. A., Nijboer, F., and Nijholt, A. (2011). User experience evaluation in bci: Filling the gap. *International Journal of Bioelectromagnetism*, 13(1):54–55.

[Hoffmann et al., 2008] Hoffmann, U., Vesin, J. M., Ebrahimi, T., and Diserens, K. (2008). An efficient p300-based braincomputer interface for disabled subjects. *Journal of Neuroscience Methods*, 167(1):115 – 125.

[Holz et al., 2012] Holz, E. M., Kaufmann, T., Franz, D., Hösle, A., and Kübler, A. (2012). Brain drawing: First evaluation results. In *Proceedings of the 3rd TOBI Workshop*, pages 109–110, Würzburg, Germany.

[Ihme and Zander, 2011] Ihme, K. and Zander, T. (2011). What you expect is what you get? potential use of contingent negative variation for passive bci systems in gaze-based hci. In D'Mello, S., Graesser, A., Schuller, B., and Martin, J. C., editors, *Affective Computing and Intelligent Interaction*, volume 6975 of *Lecture Notes in Computer Science*, pages 447–456. Springer Berlin / Heidelberg.

[Krusienski et al., 2006] Krusienski, D. J., Sellers, E. W., Cabestaing, F., Bayoudh, S., McFarland, D. J., Vaughan, T. M., and Wolpaw, J. R. (2006). A comparison of classification techniques for the p300 speller. *Journal of Neural Engineering*, 3(4):299.

[LoPresti et al., 2000] LoPresti, E., Brienza, D. M., Angelo, J., Gilbertson, L., and Sakai, J. (2000). Neck range of motion and use of computer head controls. In *Proceedings of the fourth international ACM conference on Assistive technologies*, Assets '00, pages 121–128, New York, NY, USA. ACM.

[Mandel, 1997] Mandel, T. (1997). *The elements of user interface design.* John Wiley & Sons, Inc., New York, NY, USA.

[McClure et al., 2004] McClure, S. M., Li, J., Tomlin, D., Cypert, K. S., Montague, L. M., and Montague, P. (2004). Neural correlates of behavioral preference for culturally familiar drinks. *Neuron*, 44(2):379 – 387.

[Miranda et al., 2011] Miranda, E. R., Magee, W. L., Wilson, J. J., Eaton, J., and Palaniappan, R. (2011). Brain-computer music interfacing (bcmi) from basic research to the real world of special needs. *Music and Medicine*, 3(3):134–140.

[Münßinger et al., 2010] Münßinger, J., Halder, S., Kleih, S., Furdea, A., Raco, V., Hösle, A., and Kübler, A. (2010). Brain painting: First evaluation of a new braincomputer interface application with als-patients and healthy volunteers. *Front Neurosci*, 4:182. doi: 10.3389/fnins.2010.00182.

[Nijboer et al., 2008] Nijboer, F., Sellers, E., Mellinger, J., Jordan, M., Matuz, T., Furdea, A., Halder, S., Mochty, U., Krusienski, D., Vaughan, T., Wolpaw, J., Birbaumer, N., and Kübler, A. (2008). A p300-based braincomputer interface for people with amyotrophic lateral sclerosis. *Clinical Neurophysiology*, 119(8):1909 – 1916.

[Nijboer et al., 2012] Nijboer, F., van de Laar, B. L. A., Brugman, I. H. G., Gerritsen, S., and Poel, M. (2012). Reliability and user experience of different eeg recording systems for brain-computer interfaces - a within-subject comparison. In preparation.

[Plass-Oude Bos et al., 2010] Plass-Oude Bos, D., Duvinage, M., Oktay, O., Delgado Saa, J., Guruler, H., Istanbullu, A., van Vliet, M., van de Laar, B. L. A., Poel, M., Roijendijk, L., Tonin, L., Bahramisharif, A., and Reuderink, B. (2010). Looking around with your brain in a virtual world. In *Proceedings of the 6th International Summer Workshop on Multimodal Interfaces, eNTERFACE'10*, pages 12–23. University of Amsterdam.

[Plass-Oude Bos et al., 2011] Plass-Oude Bos, D., Gürkök, H., van de Laar, B. L. A., Nijboer, F., and Nijholt, A. (2011). User experience evaluation in bci: Mind the gap! *International Journal of Bioelectromagnetism*, 13(1):48–49.

[Plass-Oude Bos et al., 2010] Plass-Oude Bos, D., Reuderink, B., Laar, B., Gürkök, H., Mühl, C., Poel, M., Nijholt, A., and Heylen, D. (2010). Brain-computer interfacing and games. *Brain-Computer Interfaces*, pages 149–178.

[Ros et al., 2009] Ros, T., Moseley, M., Bloom, P., Benjamin, L., Parkinson, L., and Gruzelier, J. (2009). Optimizing microsurgical skills with eeg neurofeedback. *BMC Neuroscience*, 10(1):87.

[Rustichini, 2009] Rustichini, A. (2009). Neuroeconomics: what have we found, and what should we search for. *Current Opinion in Neurobiology*, 19(6):672 – 677.

[Schalk et al., 2004] Schalk, G., McFarland, D., Hinterberger, T., Birbaumer, N., and Wolpaw, J. (2004). BCI2000: a general-purpose brain-computer interface (bci) system. *Biomedical Engineering, IEEE Transactions on*, 51(6):1034 –1043.

[Schalk and Mellinger, 2010] Schalk, G. and Mellinger, J. (2010). *A Practical Guide to Brain-Computer Interfacing with BCI2000*. Springer, first edition.

[Schlögl et al., 2007] Schlögl, A., Keinrath, C., Zimmermann, D., Scherer, R., Leeb, R., and Pfurtscheller, G. (2007). A fully automated correction method of eog artifacts in eeg recordings. *Clinical Neurophysiology*, 118(1):98 – 104.

[Shoker et al., 2005] Shoker, L., Sanei, S., Wang, W., and Chambers, J. (2005). Removal of eye blinking artifact from the electro-encephalogram, incorporating a new constrained blind source separation algorithm. *Medical and Biological Engineering and Computing*, 43:290–295. 10.1007/BF02345968.

[Takeshita et al., 2003] Takeshita, K., Uchibori, A., Mizukami, Y., Satoh, T., Tanaka, K., and Uchikado, S. (2003). A communication system for als patients using eye blink. *International Journal of Applied Electromagnetics and Mechanics*, 18(1-3):3–10.

[Todd et al., 2012] Todd, D. A., McCullagh, P. J., Mulvenna, M. D., and Lightbody, G. (2012). Investigating the use of brain-computer interaction to facilitate creativity. In *Proceedings of the 3rd Augmented Human International Conference*, AH '12, pages 19:1–19:8, New York, NY, USA. ACM.

[Treder and Blankertz, 2010] Treder, M. and Blankertz, B. (2010). (c)overt attention and visual speller design in an erp-based brain-computer interface. *Behavioral and Brain Functions*, 6(1):28.

[Treder et al., 2011] Treder, M. S., Schmidt, N. M., and Blankertz, B. (2011). Gaze-independent braincomputer interfaces based on covert attention and feature attention. *Journal of Neural Engineering*, 8(6):066003.

[van de Laar et al., 2011] van de Laar, B., Gürkök, H., Plass-Oude Bos, D., Nijboer, F., and Nijholt, A. (2011). Perspectives on user experience evaluation of brain-computer interfaces. In Stephanidis, C., editor, *Universal Access in Human-Computer Interaction. Users Diversity*, volume 6766 of *Lecture Notes in Computer Science*, pages 600–609. Springer Berlin / Heidelberg. 10.1007/978-3-642-21663-3_65.

[van de Laar et al., 2011a] van de Laar, B., Nijboer, F., Gürkök, H., Plass-Oude Bos, D., and Nijholt, A. (2011a). User experience evaluation in bci: Bridge the gap. *International Journal of Bioelectromagnetism*, 13(3):157–158.

[van de Laar et al., 2011b] van de Laar, B., Plass-Oude Bos, D., Reuderink, B., Poel, M., and Nijholt, A. (2011b). How much control is enough? Optimizing fun with unreliable input.

[Zander et al., 2010] Zander, T. O., Gaertner, M., Kothe, C., and Vilimek, R. (2010). Combining eye gaze input with a braincomputer interface for touchless humancomputer interaction. *International Journal of Human-Computer Interaction*, 27(1):38–51.

[Zickler et al., 2012] Zickler, C., Hösle, A., Halder, S., Kleih, S., Herbert, C., and Kübler, A. (2012). Brain painting: Usability testing in end users with severe disabilities. In *Proceedings of the 3rd TOBI Workshop*, pages 41–42, Würzburg, Germany.

Part III

Appendices

Appendix A

Eyeblinkdetection algorithm

In chapter 4 we described the eyeblink detection algorithm used in BrainBrush. In chapter 6 we described how the eyeblink detection algorithm was included in our BCI2000 SignalProcessing module, *EyeblinkSignalProcessing*. This appendix shows the implementation of the algorithm.

A.1 Variable declarations

The member variables used for the eyeblink detection algorithm are declared in the *EyeblinkFilter.h* header file:

```
// holds the SampleBlocks
std::deque<double> mSignalQueue[2];
// holds the eyeblink template
std::vector<double> mBlinkTemplate[2];
// the averages of the first 10 and last 10 samples of the eyeblink templates (AF3 and AF4)
double mTemplateAvg[2];
// the eyeblink threshold for both AF3 and AF4 channels
double mThreshold[2];
```

A.2 Initialize member variables

When the eyeblink detection is started, the *Initialize* function in the *EyeblinkFilter.cpp* file is called. This function processes the supplied BCI2000 parameters (such as the eyeblink template and the threshold) and initializes the member variables.

```
void
EyeblinkFilter::Initialize( const SignalProperties& Input, const SignalProperties& Output )
```

```
{
  // prepare everything for each of the channels
  for(unsigned int c = 0; c < 2; c++)
  {
    std::stringstream sstm;
    sstm << "EyeblinkTemplate" << c+1;
    string templateParamName = sstm.str();
    std::stringstream sstm2;
    sstm2 << "EyeblinkThreshold" << c+1;
    string thresholdParamName = sstm2.str();

    // clear some queues
    mSignalQueue[c].clear();
    mBlinkTemplate[c].clear();

    // set template
    for( int i = 0; i < Parameter(templateParamName)->NumValues(); i++)
    {
      double value = static_cast<double>(Parameter(templateParamName)(i));
      mBlinkTemplate[c].push_back(value);

    }

    // set threshold
    mThreshold[c] = Parameter(thresholdParamName);

    // calculate averages of template for offset correction
    mTemplateAvg[c] = 0.0;
    for(unsigned int d = 0; d < 10; d++)
    {
      mTemplateAvg[c] += mBlinkTemplate[c][d];
    }
    for(unsigned int d = mBlinkTemplate[c].size() - 10; d < mBlinkTemplate[c].size(); d++)
    {
      mTemplateAvg[c] += mBlinkTemplate[c][d];
    }
    mTemplateAvg[c] = mTemplateAvg[c] / 20;
  }
}
```

A.3 Process sample blocks

While the eyeblink detection component is running, the *Process* function in the *EyeblinkFilter.cpp* file is called for each block of EEG samples, which is 16 times per second. This function performs the eyeblink detection by comparing the EEG data with the eyeblink template, as discussed in chapter 4.

```
void
EyeblinkFilter::Process( const GenericSignal& Input, GenericSignal& Output )
{
  Output = Input; // Pass the signal through unmodified.

  // add the signaldata from the current SampleBlock to the queue
  // only checking 2 channels: the first and last one
  // channel indexes start with 0, not with 1!
  int channels[2] = {0, Input.Channels()-1};

  for(int n = 0; n < 2; n++)
  {
    for( int el = 0; el < Input.Elements(); el++)
    {
      double d = Input.Value(channels[n], el);
      mSignalQueue[n].push_back(d);
    }
  }

  bool isEyeblink = false;

  while(mSignalQueue[0].size() >= mBlinkTemplate[0].size())
  {
    double dst[2];
    for(unsigned int c = 0; c < 2; c++)
    {
      double sum = 0.0;

      // calculate average of first 10 and last 10 items of the sample for offset correction
      double sampleAvg = 0.0;
      for(unsigned int d = 0; d < 10; d++)
      {
        sampleAvg += mSignalQueue[c][d];
      }
      for(unsigned int d = mBlinkTemplate[0].size() - 10; d < mBlinkTemplate[0].size(); d++)
      {
        sampleAvg += mSignalQueue[c][d];
      }
      sampleAvg = sampleAvg / 20;

      // calculate difference between the averages
      double diff = mTemplateAvg[c] - sampleAvg;
```

```

      // calculate euclidian distance
      for(unsigned int n = 0; n < mBlinkTemplate[c].size
          (); n++)
      {
        sum += pow(mBlinkTemplate[c][n] - diff -
            mSignalQueue[c][n], 2);
      }
      dst[c] = sqrt(sum);

      // remove first element
      mSignalQueue[c].pop_front();
    }

    if(dst[0] < mThreshold[0] || dst[1] < mThreshold[1])
    {
      isEyeblink = true;
    }
  }

  if(isEyeblink)
  {
    State("EyeBlinkDetected") = 1;
  }
  else
  {
    State("EyeBlinkDetected") = 0;
  }
}
```

Appendix B

MyPaint

In chapter 6, we described how we incorporated the MyPaint application in the BrainBrush system and how the BrainBrush application interfaces with MyPaint: the BrainBrush application sends keyboard shortcuts to the MyPaint window to select brushes and colors. Here, we list the modifications we made to the MyPaint application in order to add those keyboard shortcuts to the sources of MyPaint version 0.9.1. We modified the *src\gui\document.py* file: we defined the keyboard shortcuts and added two callback functions.

B.1 Define keyboard shortcuts

We modified the *init_actions* method in the *src\gui\document.py* file to define the keyboard shortcuts for the selection of brushes and colors. In the sourcecode listed below, the added keyboard shortcuts can be found in lines 50 through 86. For each action, a keyboard shortcut and a callback function were defined.

```
1  def init_actions(self):
2    # name, stock id, label, accelerator, tooltip, callback
3    actions = [
4      ('Undo', gtk.STOCK_UNDO, _('Undo'), 'Z', None,
          self.undo_cb),
5      ('Redo', gtk.STOCK_REDO, _('Redo'), 'Y', None,
          self.redo_cb),
6
7      ('Brighter', None, _('Brighter'), None, None,
          self.brighter_cb),
8      ('Smaller', None, _('Smaller'), 'd', None,
          self.brush_smaller_cb),
9      ('MoreOpaque', None, _('More Opaque'), 's', None,
          self.more_opaque_cb),
10     ('LessOpaque', None, _('Less Opaque'), 'a', None,
          self.less_opaque_cb),
11     ('Eraser', None, _('Toggle Eraser Mode'), 'e', None,
          self.eraser_cb),
```

```
        ('PickContext', None, _('Pick Context (layer, brush and color)'), 'w', None, self.pick_context_cb),

        ('Darker', None, _('Darker'), None, None, self.darker_cb),
        ('Warmer', None, _('Warmer'), None, None, self.warmer_cb),
        ('Cooler', None, _('Cooler'), None, None, self.cooler_cb),
        ('Purer', None, _('Purer'), None, None, self.purer_cb),
        ('Grayer', None, _('Grayer'), None, None, self.grayer_cb),
        ('Bigger', None, _('Bigger'), 'f', None, self.brush_bigger_cb),

        # Context actions are also added in init_context_actions
        ('ContextStore', None, _('Save to Most Recently Restored'), 'q', None, self.context_cb),

        ('ClearLayer', gtk.STOCK_CLEAR, _('Clear'), 'Delete', None, self.clear_layer_cb),
        ('CopyLayer', gtk.STOCK_COPY, _('Copy to Clipboard'), '<control>C', None, self.copy_cb),
        ('PasteLayer', gtk.STOCK_PASTE, _('Paste Clipboard (Replace Layer)'), '<control>V', None, self.paste_cb),
        ('PickLayer', gtk.STOCK_JUMP_TO, _('Select Layer at Cursor'), 'h', None, self.pick_layer_cb),
        ('LayerFG', gtk.STOCK_GO_UP, _('Next (above current)'), 'Page_Up', None, self.layer_fg_cb),
        ('LayerBG', gtk.STOCK_GO_DOWN, _('Next (below current)'), 'Page_Down', None, self.layer_bg_cb),
        ('NewLayerFG', gtk.STOCK_ADD, _('New (above current)'), '<control>Page_Up', None, self.new_layer_cb),
        ('NewLayerBG', None, _('New (below current)'), '<control>Page_Down', None, self.new_layer_cb),
        ('MergeLayer', gtk.STOCK_DND_MULTIPLE, _('Merge Down'), '<control>Delete', None, self.merge_layer_cb),
        ('RemoveLayer', gtk.STOCK_DELETE, _('Remove'), '<shift>Delete', None, self.remove_layer_cb),
        ('IncreaseLayerOpacity', None, _('Increase Layer Opacity'), 'p', None, self.layer_increase_opacity),
        ('DecreaseLayerOpacity', None, _('Decrease Layer Opacity'), 'o', None, self.layer_decrease_opacity),
```

```

    ('ShortcutsMenu', None, _('Shortcuts')),

    ('ResetView', gtk.STOCK_ZOOM_100, _('Reset (Zoom, Rotation, Mirror)'), 'F12', None, self.reset_view_cb),
    ('ZoomIn', gtk.STOCK_ZOOM_IN, _('Zoom In (at cursor)'), 'period', None, self.zoom_cb),
    ('ZoomOut', gtk.STOCK_ZOOM_OUT, _('Zoom Out'), 'comma', None, self.zoom_cb),
    ('RotateLeft', None, _('Rotate Counterclockwise'), '<control>Left', None, self.rotate_cb),
    ('RotateRight', None, _('Rotate Clockwise'), '<control>Right', None, self.rotate_cb),
    ('MirrorHorizontal', None, _('Mirror Horizontal'), 'i', None, self.mirror_horizontal_cb),
    ('MirrorVertical', None, _('Mirror Vertical'), 'u', None, self.mirror_vertical_cb),
    ('SoloLayer', None, _('Layer Solo'), 'Home', None, self.solo_layer_cb),
    ('ToggleAbove', None, _('Hide Layers Above Current'), 'End', None, self.toggle_layers_above_cb),

    # Items added for BrainBrush
    ('BPBrushMop', None, _('Mop'), '<control><alt>a', None, self.bp_brush_select_cb),
    ('BPBrushFlat', None, _('Flat'), '<control><alt>b', None, self.bp_brush_select_cb),
    ('BPBrushBright', None, _('Bright'), '<control><alt>c', None, self.bp_brush_select_cb),
    ('BPBrushRigger', None, _('Rigger'), '<control><alt>d', None, self.bp_brush_select_cb),
    ('BPBrushPencil', None, _('Pencil'), '<control><alt>e', None, self.bp_brush_select_cb),
    ('BPBrushEraser', None, _('Eraser'), '<control><alt>f', None, self.bp_brush_select_cb),
    ('BPBrushBallpen', None, _('Ballpen'), '<control><alt>g', None, self.bp_brush_select_cb),
    ('BPBrushG-pen', None, _('G-pen'), '<control><alt>h', None, self.bp_brush_select_cb),
    ('BPBrushBrush', None, _('Brush'), '<control><alt>i', None, self.bp_brush_select_cb),
    ('BPBrushPen', None, _('Pen'), '<control><alt>j', None, self.bp_brush_select_cb),
    ('BPBrushMarker', None, _('Marker'), '<control><alt>k', None, self.bp_brush_select_cb),
    ('BPBrushFill', None, _('Fill'), '<control><alt>l', None, self.bp_brush_select_cb),
```

```
        ('BPBrushWatercolor', None, _('Watercolor'),
            '<control><alt>m', None, self.bp_brush_select_cb),
        ('BPBrushWatercolor1', None, _('Watercolor1'),
            '<control><alt>n', None, self.bp_brush_select_cb),
        ('BPBrushSponge', None, _('Sponge'),
            '<control><alt>o', None, self.bp_brush_select_cb),
        ('BPBrushPastel', None, _('Pastel'),
            '<control><alt>p', None, self.bp_brush_select_cb),
        ('BPBrushCharcoal', None, _('Charcoal'),
            '<control><alt>q', None, self.bp_brush_select_cb),
        ('BPBrushChalk', None, _('Chalk'),
            '<control><alt>r', None, self.bp_brush_select_cb),
        ('BPBrushBlending', None, _('Blending'),
            '<control><alt>s', None, self.bp_brush_select_cb),
        ('BPBrushBlending1', None, _('Blending1'),
            '<control><alt>t', None, self.bp_brush_select_cb),
        ('BPBrushBlending2', None, _('Blending2'),
            '<control><alt>u', None, self.bp_brush_select_cb),
        ('BPBrushAirbrush', None, _('Airbrush'),
            '<control><alt>v', None, self.bp_brush_select_cb),
        ('BPBrushSpray', None, _('Spray'),
            '<control><alt>w', None, self.bp_brush_select_cb),
        ('BPBrushSpray1', None, _('Spray1'),
            '<control><alt>x', None, self.bp_brush_select_cb),
        ('BPColorRed', None, _('Red'), '<control><alt>y',
            None, self.bp_color_select_cb),
        ('BPColorGreen', None, _('Green'),
            '<control><alt>z', None, self.bp_color_select_cb),
        ('BPColorBlue', None, _('Blue'), '<control><alt>1',
            None, self.bp_color_select_cb),
        ('BPColorYellow', None, _('Yellow'),
            '<control><alt>2', None, self.bp_color_select_cb),
        ('BPColorCyan', None, _('Cyan'), '<control><alt>3',
            None, self.bp_color_select_cb),
        ('BPColorPurple', None, _('Purple'),
            '<control><alt>4', None, self.bp_color_select_cb),
        ('BPColorBlack', None, _('Black'),
            '<control><alt>5', None, self.bp_color_select_cb),
        ('BPColorBurlywood', None, _('Burlywood'),
            '<control><alt>6', None, self.bp_color_select_cb),
        ('BPColorDarkBrown', None, _('Dark brown'),
            '<control><alt>7', None, self.bp_color_select_cb),
        ('BPColorGrey', None, _('Grey'), '<control><alt>8',
            None, self.bp_color_select_cb),
        ('BPColorOrange', None, _('Orange'),
            '<control><alt>9', None, self.bp_color_select_cb),
        ('BPColorWhite', None, _('White'),
            '<control><alt>0', None, self.bp_color_select_cb),
    ]

```

```
  ag = self.action_group = gtk.ActionGroup('DocumentActions')
  ag.add_actions(actions)

  toggle_actions = [
    # name, stock id, label, accelerator, tooltip, callback, default toggle status
    ('PrintInputs', None, _('Print Brush Input Values to stdout'), None, None, self.print_inputs_cb),
    ('VisualizeRendering', None, _('Visualize Rendering'), None, None, self.visualize_rendering_cb),
    ('NoDoubleBuffereing', None, _('Disable GTK Double Buffering'), None, None, self.no_double_buffering_cb),
    ]
  ag.add_toggle_actions(toggle_actions)
```

B.2 Handle brush shortcuts

All keyboard shortcuts for the selection of brushes list the *bp_brush_select_cb* function in *src\gui\document.py* as the callback function to be used. This function checks which action (i.e. keyboard shortcut) triggered the callback function and then selects the appropriate brush.

```
def bp_brush_select_cb(self, command):
  brushname = command.get_name()[7:].lower()
  if brushname == 'g-pen':
    brushname = 'G-pen'
  b = self.app.brushmanager
      .get_brush_by_name('deevad/'+brushname)
  # save color before setting brush
  (rgb) = self.app.brush.get_color_rgb()
  self.app.brushmanager.select_brush(b)
  # restore color
  self.app.brush.set_color_rgb(rgb)
```

B.3 Handle color shortcuts

The keyboard shortcuts for the selection of colors use the *bp_color_select_cb* callback function in *src\gui\document.py*. This function checks which action (i.e. keyboard shortcut) triggered the callback function and then selects the appropriate color.

```
def bp_color_select_cb(self, command):
  colorname = command.get_name()[7:].lower()

  if colorname == 'blue':
```

```
      rgb = (0,0,255)
    elif colorname == 'green':
      rgb = (0,255,0)
    elif colorname == 'red':
      rgb = (255,0,0)
    elif colorname == 'yellow':
      rgb = (255,255,0)
    elif colorname == 'cyan':
      rgb = (0,255,255)
    elif colorname == 'purple':
      rgb = (255,0,255)
    elif colorname == 'black':
      rgb = (0,0,0)
    elif colorname == 'burlywood':
      rgb = (222,184,135)
    elif colorname == 'darkbrown':
      rgb = (92,64,51)
    elif colorname == 'grey':
      rgb = (179,179,179)
    elif colorname == 'orange':
      rgb = (255,127,0)
    elif colorname == 'white':
      rgb = (255,255,255)

    # rgb values are expected in the range 0.0 - 1.0, so divide by 255
    r,g,b = rgb
    r = r / 255.0
    g = g / 255.0
    b = b / 255.0
    rgb = (r,g,b)

    self.app.brush.set_color_rgb(rgb)
```

Appendix C

BrainBrush application implementation

This appendix provides details of the implementation of the various tasks and processes in the BrainBrush application. First, we show the various threads running in the BrainBrush application in section C.1. Next, we show the flow of the BrainBrush application during various tasks in section C.2.

C.1 Threads

BrainBrush runs on a number of separate threads, all performing a specific task at specific moments and not all threads are running at the same time:

MainThread runs from startup of BrainBrush until the application quits. The MainThread controls the user interface, starts the various other threads when needed, processes the outputs from these other threads and decides, based on those outputs, what state (i.e. 'Painting' or 'Selection') the application should be in.

CursorTracker is started by MainThread whenever BrainBrush enters the 'Painting' state. This thread is used to detect when the mousecursor enters one of the trigger areas (i.e. 'Change Brush', 'Change color', 'New painting', 'Undo' or 'Redo'). When the CursorTracker notices the mousecursor is in one of the trigger areas, the thread informs the MainThread which trigger area has been hit after which the thread ends.

EyeblinkDetection is also started by MainThread whenever BrainBrush enters the 'Painting' state, right after the CursorTracker thread is started. It continually processes BCI2000 state information and whenever an eyeblink is detected, the EyeblinkDetection thread reports this to the MainThread. The EyeblinkDetection thread only exits when the MainThread requests the EyeblinkDetection thread to cancel it's processing, which is whenever the CursorTracker has detected a trigger (i.e. 'Change brush', 'Change color', 'New painting', 'Undo' or 'Redo').

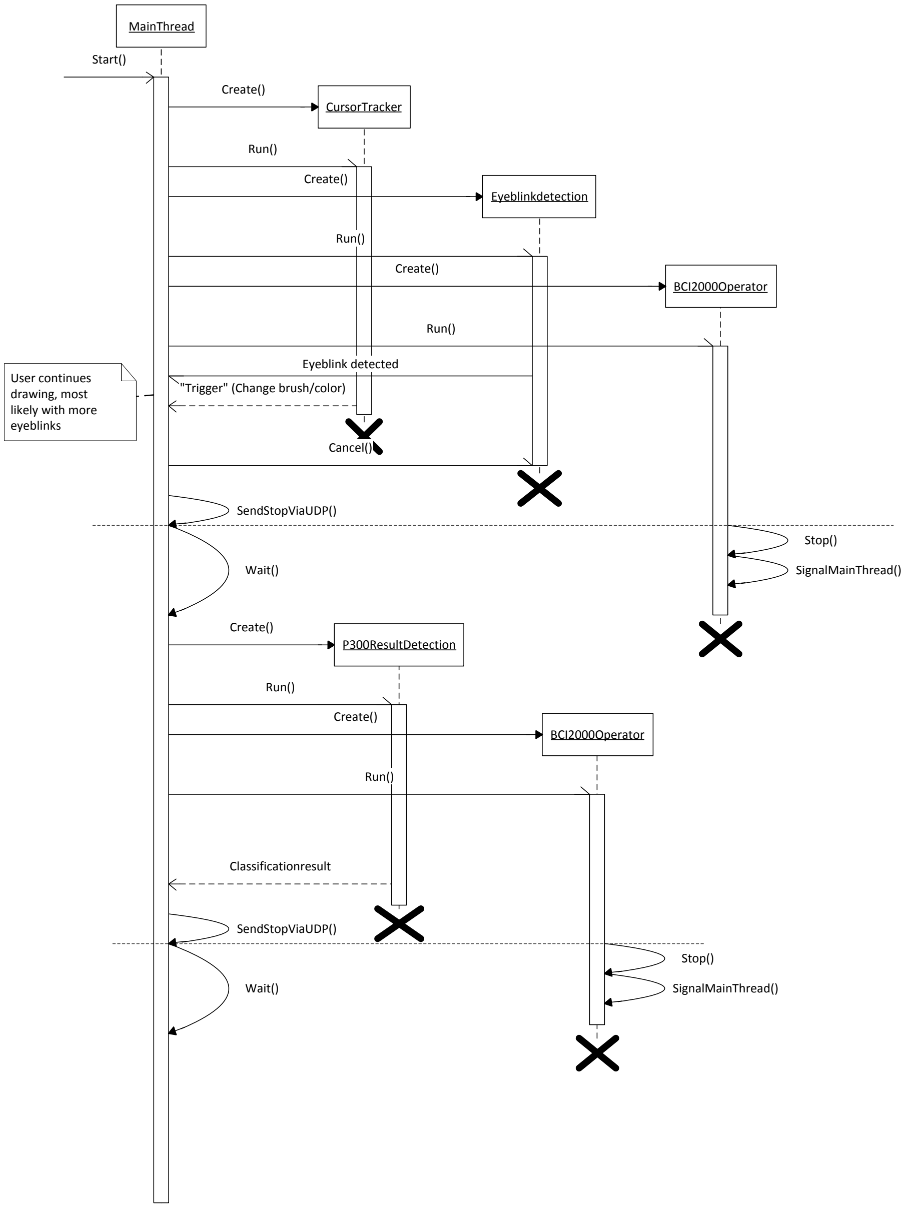


Figure C.1: Sequence diagram showing the start and exit of the various threads within BrainBrush

P300ResultDetection is started by MainThread whenever BrainBrush enters the 'Selection' state. It, like the EyeblinkDetection thread, continually processes BCI2000 state information and whenever a P300 classification result is detected, reports to the MainThread. The P300ResultDetection thread exits directly after reporting this classification result to the MainThread.

BCI2000Operator is started by MainThread whenever BrainBrush switches from the 'Painting' state to the 'Selection' state, or vice versa. The BCI2000Operator thread connects the Source, SignalProcessing and Application modules (see section 6.2.1). Furthermore, it loads the appropriate parameter files. Whenever BCI2000 receives the 'Running 0\r' signal (see section 6.2.4), the BCI2000Operator thread signals the waiting MainThread that it is finished and exits afterwards.

Figure C.1 shows an example of how a sequence of actions could trigger the start and exit of the various threads.

C.2 Flow of the BrainBrush application

C.2.1 Startup

During the startup of the BrainBrush application, a black window is displayed fullscreen and on top of all other windows. Meanwhile, MyPaint is started in fullscreen mode in the background. Due to the black window on top, this process is not visible to the user. Furthermore, the Emotiv Mouse Emulator is enabled by sending the appropriate keystroke (CTRL+SHIFT+M) to the Mouse Emulator window. The BrainBrush application then waits for 4 seconds to allow for MyPaint to fully start. Still in the background, the picture frame is loaded and drawn. This is done on top of the, now fully operational MyPaint application. Finally, the black fullscreen window is hidden, revealing the complete BrainBrush graphical user interface to the user.

Next, the CursorTracker thread is started in order to detect when the mousecursor enters an active region, thereby triggering BrainBrush to, for instance, start a P300 grid. However, at this point in the startup process, triggers are not being processed, they are discarded instead.

Next, the eyeblink detection is started. This process is described in section C.2.2.

Finally, the processing of mousecursor triggers is started. The reason why triggers are only processed after the eyeblink detection has started is that otherwise a situation could occur where the BCI2000 eyeblink detection component is not yet fully started and the cursor triggers the 'Change brush' active region. In such a case, BrainBrush would send a stop-signal to the, not yet fully started, BCI2000 instance which is not yet able to process this stop-signal. BCI2000 would then not be able to close, but BrainBrush would try to start a new instance of BCI2000 for the P300 grid. This would then fail because the BCI2000 eyeblink detection component is still running. Even though this situation is unlikely to occur it would be fatal for the BrainBrush program and therefore

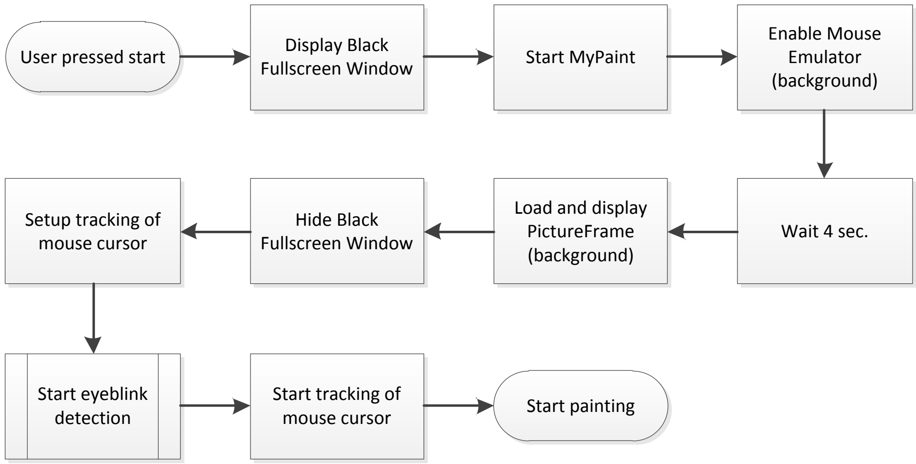


Figure C.2: Flowchart of the startup process of BrainBrush

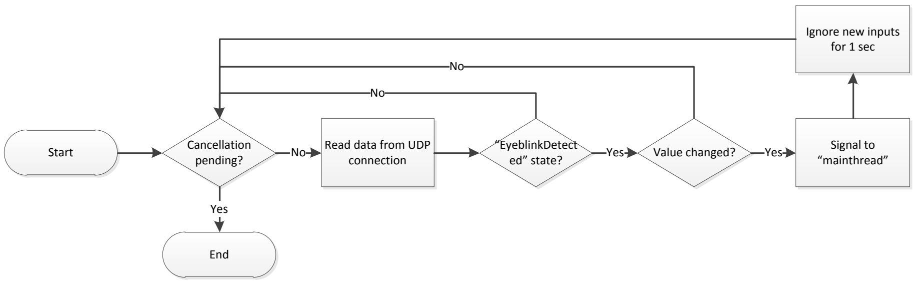


Figure C.3: Flowchart of the process executed by the EyeblinkDetection thread

the processing of mousecursor triggers is delayed. Since the delay in processing mousecursor triggers is less then a second, it does not introduce problems for the responsiveness of the application.

BrainBrush is now completely started and the user can start painting. Figure C.2 shows the startup process of BrainBrush in a flowchart.

C.2.2 Start eyeblink detection

In order to start the eyeblink detection, first the Eyeblinkdetection thread is started. This thread runs in a loop and continually processes BCI2000 state information from the BCI2000 eyeblink detection component. First, it checks whether it has been requested to stop running. Next, it reads BCI2000 state information from the UDP connection, searching for the *EyeblinkDetected* state. If this state has changed, the MainThread is signalled. The Eyeblinkdetection thread then ignores new inputs via the UDP connection for one second to avoid immediate, usually false, repeated triggers of the eyeblink detection. This process is shown in the flowchart in figure C.3.

The BCI2000 eyeblink detection component is started by first starting the BCI2000Operator thread, then loading the *Emotiv* Source module, the *Eye-*

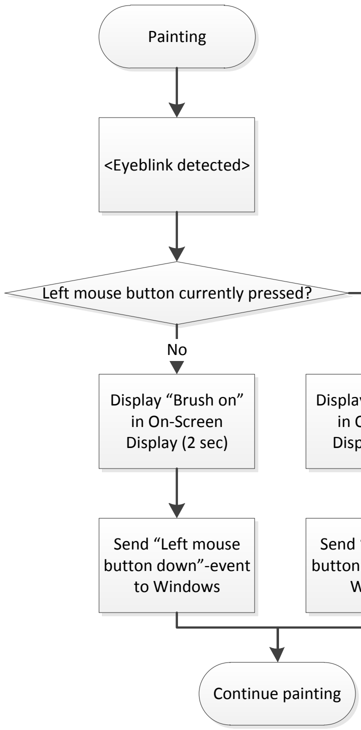


Figure C.4: Flowchart showing how the MainThread handles detected eyeblinks

blinkSiganlProcessing module, and the *DummyApplication* module. Once all modules are connected, parameterfiles are loaded, the config is set and the run is started.

When an eyeblink has been detected, the Eyeblinkdetection thread signals the MainThread. The MainThread then checks whether Windows reports the left mouse button has been pressed. The mouse button is not pressed physically, but BrainBrush sends mouse-down and mouse-up events to Windows, emulating the mouse button is pressed. If the left mouse button is currently pressed, the BrainBrush application displays the 'Brush off' feedback to the user and sends the mouse-up event to Windows. If the left mouse button is not currently pressed, the BrainBrush application displays the 'Brush on' feedback and sends the left mouse-down event to Windows. This process is shown in the flowchart in figure C.4.

C.2.3 Cursor trigger

In order to detect when the cursor enters an active region (e.g. 'Change Brush'), the position of the cursor has to be polled constantly. For this purpose, the MainThread starts the CursorTracker thread which continually executes a loop: first, it checks whether the MainThread has requested the process to be cancelled, then it checks whether the cursor is currently in one of the active regions, finally it waits 50 ms before starting over.

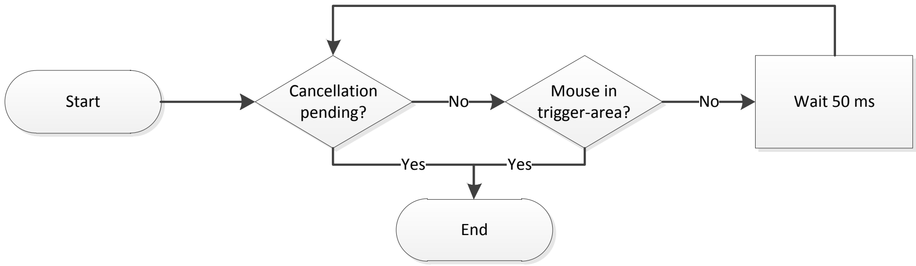


Figure C.5: Flowchart of the process executed by the CursorTracker thread

When the cursor is in one of the active regions, the CursorTracker thread ends and informs the MainThread about which active region was hit.

Figure C.5 shows the process executed by the CursorTracker thread in a flowchart.

C.2.4 Brush selection

When the BrainBrush application is in the 'Painting' state, and the user hovers the cursor over the 'Change brush' menu option, the CursorTracker detects a trigger.

First, the EyeblinkDetection thread, the thread processing BCI2000 state information from the BCI2000 eyeblink detection component, is requested to stop. If the user was creating a brushstroke at the time the trigger occurred, the left-mousebutton-up command is sent to Windows to finish the brushstroke.

Next, the running eyeblink detection instance of BCI2000 is requested to stop. The BrainBrush application has to wait for it to completely stop. Otherwise, the BCI2000 instance for the P300 paradigm can not be started.

Since the EyeblinkDetection thread is not guaranteed to stop instantly after it has been requested to stop, it is possible that in the meantime another eyeblink has been detected and the left-mousebutton-down command has once again been issued. Therefore, the status of the left-mousebutton is checked once again and, if needed, the left-mousebutton-up command is issued again.

Next, a black window is displayed fullscreen in order to be able to prepare the user interface without this process being visible to the user. The window showing the picture frame is then hidden.

Because it is not desirable to have a mousecursor visible on the P300 grid, the cursor has to be hidden. Since the P300 grid window is created by BCI2000 it is not possible to control the visibility of the mousecursor directly from BrainBrush, as described in section 6.5. Therefore, an invisible (transparent) fullscreen window is created on top of all other windows, and the mousecursor is hidden on this window. When the user sees the P300 grid, there is actually another window on top of it, but this window is invisible.

The P300 speller session is then started, see C.2.6.

After BCI2000 is completely configured and ready to start, the black fullscreen window is hidden, revealing the P300 grid to the user. The P300 speller session is then started automatically.

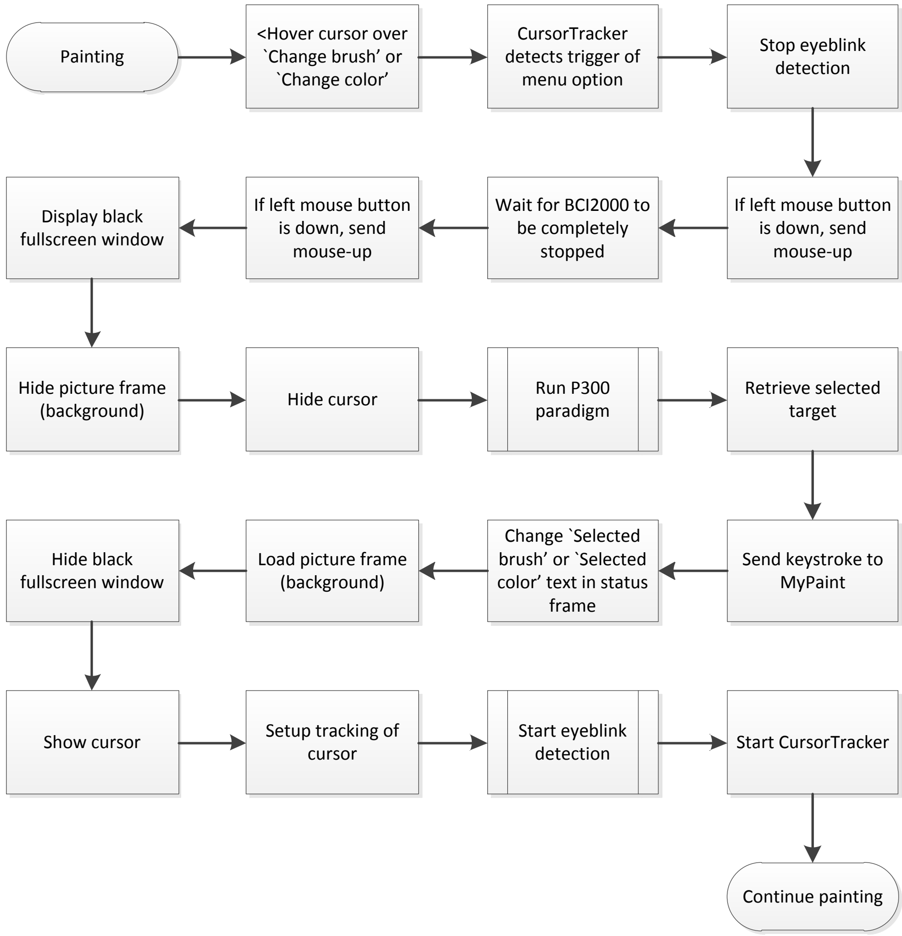


Figure C.6: Flowchart of the brush or color selection process

After the first target has been classified by the P300 speller, BrainBrush once again shows a black fullscreen window to hide the changes to the user interface. BCI2000 is stopped and BrainBrush retrieves the classification result and sends the appropriate keystroke to the MyPaint window, thereby selecting the desired brush. Furthermore, the 'Selected brush' text in the status frame is updated. Next, still in the background, the picture frame is loaded and displayed. The black fullscreen window is hidden, revealing the canvas and picture frame to the user once again. Finally, the cursor is made visible again and the tracking of the cursor and the eyeblink detection are started.

The user can now start painting with the selected brush. This process is depicted in the flowchart in figure C.6.

C.2.5 Color selection

The color selection process is started when the user hovers the cursor over the 'Change color' text in the picture frame. This selection process is similar to the

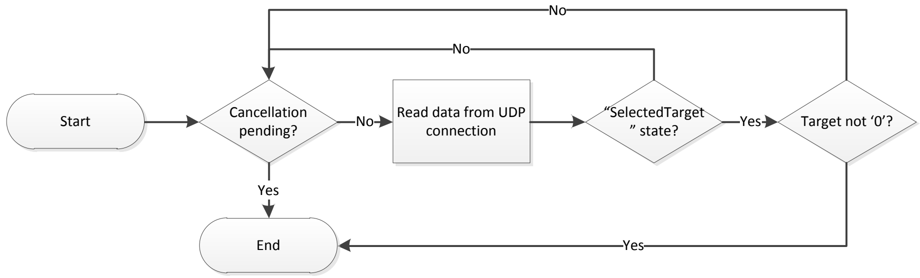


Figure C.7: Flowchart of the process executed by the P300ResultDetection thread

brush selection process described in section C.2.4.

C.2.6 P300 speller session

When a P300 speller session is started, the BrainBrush application first starts the P300ResultDetection thread. This thread consists of a loop: first, the thread checks whether it has been requested by the MainThread to cancel the processing of BCI2000 state information. Next, it reads data from the UDP connection and checks whether this data contains information about the *SelectedTarget* state. If so, it checks whether the target has been classified (*SelectedTarget* is '0' if no target has been classified yet). When the selected target has been found, the thread ends and reports the selected target to MainThread. This process is shown in the flowchart in figure C.7.

After the P300ResultDetection thread has been started, the BCI2000Operator thread is started, followed by the *Emotiv* Source module, the *P3SignalProcessing* module and the *P3Speller* application module. Once all modules are connected, parameterfiles are loaded and the config is set. This triggers the BrainBrush application to hide the black fullscreen window which was displayed before starting BCI2000, see section C.2.4.

Next, the P300 speller run is started. When the first target has been classified, the MainThread displays a black fullscreen window to hide changes to the user interface, sends the 'Running 0\r' signal to BCI2000, and waits for BCI2000 to close. The BCI2000Operator thread detects the 'Running 0\r' signal, shuts down all modules, notifies the MainThread that it has finished shutting down and exits. This process is shown in figure C.8.

C.2.7 New Painting

When the user hovers the cursor over the 'New painting' menu option, this is detected by the CursorTracker thread which then signals the MainThread and exits. The MainThread first stops the eyeblink detection to ensure the user is not creating brushstrokes during the save process. Next, the MainThread creates a screenshot and saves it to the hard disk. It then sends a keyboard shortcut to MyPaint to clear the canvas. Next, it provides the user with a 'Painting saved' notification. Finally, it restarts the CursorTracker thread and the eyeblink detection.

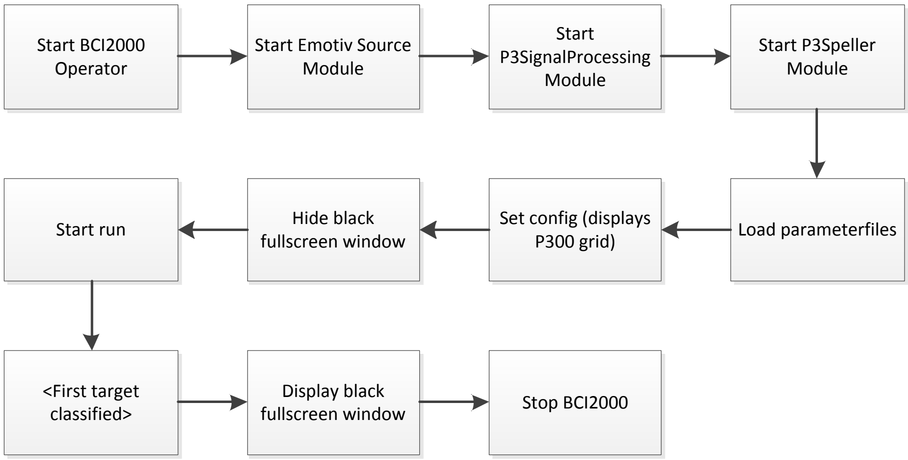


Figure C.8: Flowchart of the P300 paradigm

C.2.8 Undo/Redo

When the user hovers the cursor over either the 'Undo' or the 'Redo' menu option, this is detected by the CursorTracker thread which then signals the MainThread and exits. The MainThread provides the user with an 'Undo' or 'Redo' notification. Next, it stops the eyeblink detection to ensure the user is not creating brushstrokes. Next, the appropriate keystroke for either undo or redo is sent to MyPaint. Finally, MainThread restarts the CursorTracker thread and the eyeblink detection.

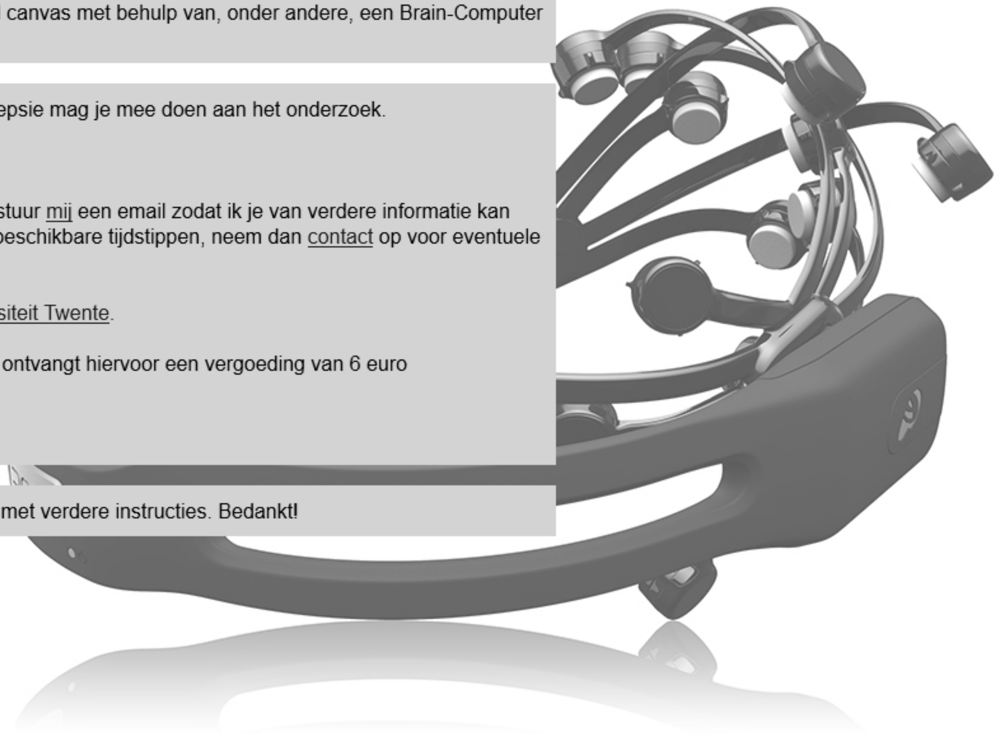
Appendix D

Call for Participants

Altijd al eens willen schilderen door enkel je hoofd te gebruiken? Doe mee aan mijn afstudeeronderzoek en je zult na een korte training de kans krijgen zelf te schilderen op een virtueel canvas met behulp van, onder andere, een Brain-Computer Interface!

- Als je gezond bent, 18 jaar of ouder en niet lijdt aan epilepsie mag je mee doen aan het onderzoek.
- Redelijke Engelse leesvaardigheid is een vereiste.
- Voldoe je hieraan, kies dan hier een geschikt tijdstip en stuur mij een email zodat ik je van verdere informatie kan voorzien. Wil je graag meedoen maar kun je niet op de beschikbare tijdstippen, neem dan contact op voor eventuele andere tijdstippen.
- Het onderzoek vindt plaats in De Zilverling op de Universiteit Twente.
- Deelname aan het onderzoek kost ongeveer 1 uur en je ontvangt hiervoor een vergoeding van 6 euro (UT-medewerkers ontvangen geen vergoeding).
- Als je nog vragen hebt, neem dan gerust contact op.

De dag voor het experiment krijg je een herinnering per email met verdere instructies. Bedankt!



Appendix E

Informed Consent Form

UNIVERSITY OF TWENTE.

Human Media Interaction

INFORMED CONSENT FORM (ICF)

Voordat u gaat deelnemen aan dit onderzoek is het belangrijk dat u deze uitleg over dit onderzoek leest. Dit document beschrijft het doel, de procedures, risico's, ongemakken en voorzorgsmaatregelen voor het onderzoek. Indien u een term of uitdrukking tegenkomt welke u niet begrijpt, vraag dan de experimentleider om uitleg. Voordat u akkoord gaat met deelname aan dit onderzoek mag u anderen hierover raadplegen. U ontvangt een kopie van deze Informed Consent Form.

Titel van het onderzoek: Gebruikerservaringen met een multimodaal interactief schilderprogramma
Experimentleider: Ivo Brugman

Doel van het onderzoek
Het doel van dit onderzoek is de evaluatie van de gebruikerservaringen bij het schilderen met een multimodaal interactief schilderprogramma.

Uitleg van de procedures
Voor het begin van het experiment wordt u gevraagd een vragenlijst in te vullen over uw demografische gegevens. Daarna mag u de draadloze brain-computer interface opzetten. Vervolgens krijgt u een document met uitleg over de zogenaamde P3Speller die u gaat gebruiken. Hierna zullen twee training-sessies volgen om de P3Speller af te kunnen stemmen op uw hersensignalen. U krijgt daarna een document te lezen met uitleg over het schilderprogramma. Nadat u deze gelezen heeft mag u 5 minuten kennis maken met het programma en het een en ander uitproberen. Tijdens deze 5 minuten worden indien nodig nog kleine aanpassingen gedaan aan de configuratie van het programma om deze beter af te stemmen op uw hersensignalen. Na deze oefensessie krijgt u de taakomschrijving en mag u beginnen met schilderen. Aan het einde van het experiment wordt u gevraagd een enquête in te vullen en zal de experimentleider een interview met u afnemen.

Duur van het onderzoek
De hele procedure zoals hierboven omschreven zal ongeveer een uur in beslag nemen.

Risico's
Dit onderzoek is gebaseerd op de huidige kennis op het gebied van brain-computer interfaces en is veilig voor de deelnemers. Alleen als u epilepsie heeft kan het experiment wel bijwerkingen hebben en mag u niet aan het experiment deelnemen. Door deel te nemen aan dit onderzoek loopt u verder geen specifieke risico's en er zijn geen bijwerkingen bekend.

Terugtrekking
Deelname aan dit onderzoek is vrijwillig; als u besluit niet mee te werken heeft dat geen gevolgen. U bent op elk moment, ook tijdens het experiment, vrij uw toestemming in te trekken en te stoppen met uw deelname aan dit onderzoek.

Vertrouwelijkheid
Uw hersenactiviteit zal tijdens het onderzoek opgenomen worden met behulp van de Emotiv EPOC headset. Het afsluitende interview zal opgenomen worden met behulp van een voicerecorder. Alle opgenomen data, uw antwoorden op de vragenlijsten en de resultaten kunnen publiekelijk beschikbaar gemaakt worden en gebruikt worden voor onderzoeksdoeleinden. Uw data zal altijd anoniem gepubliceerd worden zodat uw identiteit niet bekend wordt.

Kosten en/of vergoeding voor deelname aan dit onderzoek
Er zijn geen kosten verbonden aan deelname aan dit onderzoek. U ontvangt een vergoeding van €6,- voor uw deelname aan dit onderzoek tenzij u een wetenschappelijk medewerker van de Universiteit Twente bent.

Vragen
Vragen met betrekking tot dit onderzoek kunt u stellen aan de experimentleider, te bereiken via i.h.g.brugman@student.utwente.nl.

Ik heb bovenstaande informatie gelezen of het is aan mij voorgelezen. Ik heb de mogelijkheid gehad vragen te stellen hierover en de vragen die ik gesteld heb zijn voldoende beantwoord. Ik geef vrijwillig toestemming tot deelname aan dit onderzoek.

Naam proefpersoon: ______________________

Handtekening proefpersoon: ______________________

Plaats, datum: Enschede, .../.../2012

Ik bevestig dat de proefpersoon de kans heeft gehad vragen te stellen over dit onderzoek en dat alle vragen die gesteld zijn door de proefpersoon juist en naar mijn beste kunnen beantwoord zijn. Ik bevestig dat de proefpersoon niet is gedwongen tot het geven van toestemming tot deelname en dat de toestemming tot deelname vrijwillig is gegeven.

Een kopie van deze Informed Consent Form is aan de proefpersoon gegeven.

Naam experimentleider: ______________________

Handtekening experimentleider: ______________________

Plaats, datum: Enschede, .../.../2012

Appendix F

Demographic questionnaire

Beste deelnemer,

Hartelijk bedankt dat u mee wilt doen aan ons onderzoek. Wij willen u verzekeren dat uw informatie **vertrouwelijk behandeld** wordt. U krijgt een code van ons zodat uw naam niet gebruikt zal worden. Die code bestaat uit cijfers. Hieronder zult u zien, dat de experimentleider de code al heeft ingevuld.

Proefpersooncode:

1)	2)	3)

Datum: ______________________

Tijd: ______________________

► AUB verder bladeren

Informatie over uzelf

Demografische gegevens

Leeftijd: __________ Jaar

Geslacht: mannelijk ❒ vrouwelijk ❒

Handvoorkeur (kies 1 van de opties):

het meeste met links ❒

het meeste met rechts ❒

zowel links als rechts ❒

Hoogst behaalde diploma: ________________________

Beroep/studie: ________________________

Moedertaal: ________________________

Medische voorgeschiedenis, ontwikkeling en huidige situatie

1. **Heeft u wel eens een hele harde klap op uw hoofd gehad?**

Ja ❒ Nee ❒

Zo ja, wanneer was dat? ______________________________________

Heeft u daarbij uw bewustzijn verloren? Ja ❒ Nee ❒

Zo ja, voor hoe lang?__________________________________

Had u naderhand geheugenverlies? Ja ❒ Nee ❒

Zo ja, voor hoe lang? _________________________________

Moest u voor dit voorval naar het ziekenhuis? Ja ❒ Nee ❒

Zo ja, voor hoe lang? _________________________________

► AUB verder bladeren

2. **Heeft u wel eens een neurologische stoornis of probleem gehad?** (bijvoorbeeld: beroerte, alcohol-gerelateerde problemen, depressies)

Ja ❒ Nee ❒

Zo ja, kunt u beschrijven welk probleem?

____

____

3. **Heeft u wel eens een epileptische aanval gehad?**

Ja ❒ Nee ❒

4. **Heeft u een chronische ziekte?** (bijvoorbeeld: diabetes, allergieën)

Ja ❒ Nee ❒

Zo ja, welke? ________________________________

____

5. **Rookt u?**

Ja ❒ Nee ❒

6. **Drinkt u? (AUB ook ja aankruizen als u gewoon af en toe een glaasje drinkt)**

Ja ❒ Nee ❒

7. **Gebruikt u wel eens drugs?**

Ja ❒ Nee ❒

► AUB verder bladeren

8. **Neemt u regelmatig medicijnen?**

Ja ❐ Nee ❐

Zo ja, welke medicijnen neemt u regelmatig?

___

9. **Is er verder nog iets m.b.t. uw gezondheid waarover u denkt ons te moeten informeren?**

___

___

___

Voorgeschiedenis met Brain-Computer Interfacing

1. **Heeft u eerder deelgenomen aan onderzoek met brain-computer interfaces?**

Ja ❐ Nee ❐

Creatieve achtergrond

1. **Houdt u zich regelmatig bezig met creatieve uitingen zoals schilderen?**

Ja ❐ Nee ❐

Zo ja, welke en hoe vaak?

___

Heel erg bedankt voor uw deelname!

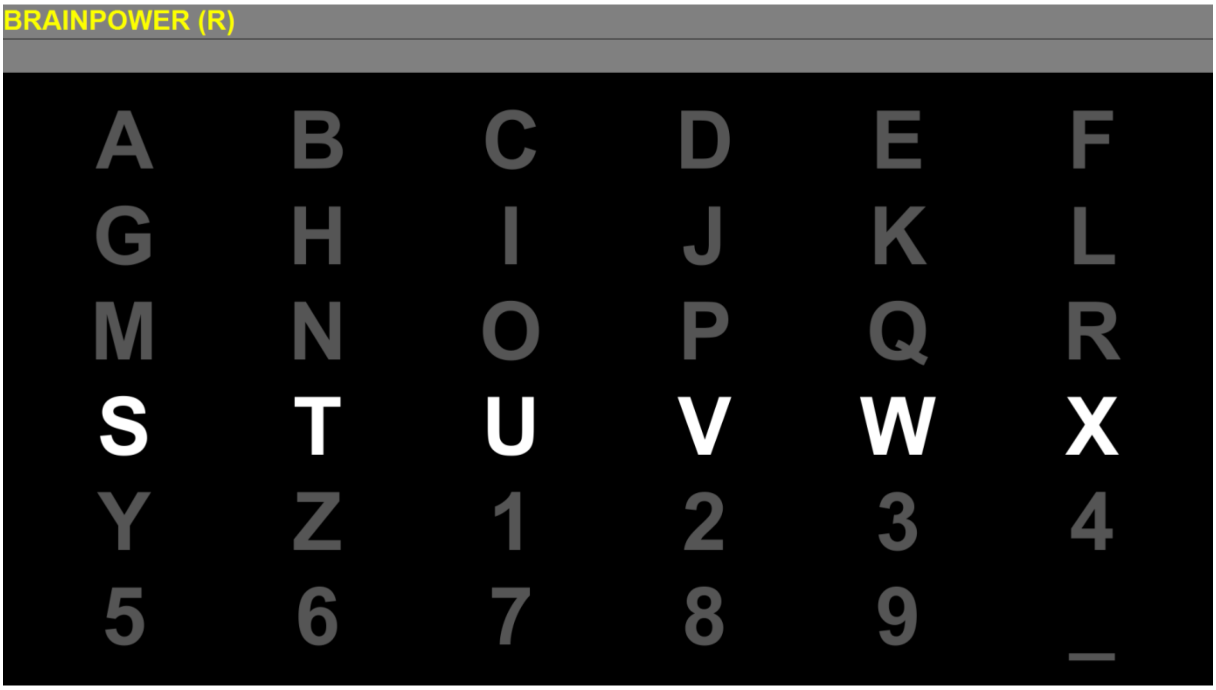
Appendix G

P3Speller instruction document

P300 training instructies

Tijdens de nu volgende sessie van 15 minuten zullen we EEG data opnemen om hiermee een algoritme te kunnen trainen zodat deze afgestemd kan worden op uw hersensignalen.

U krijgt dadelijk de zogenaamde P3Speller te zien: een matrix waarin het alfabet, de cijfers 1 tot en met 9 en de spatie verdeeld zijn over 6 rijen en 6 kolommen:



Tijdens deze trainingssessie wordt u gevraagd het woord ‘BRAINPOWER’ te spellen met behulp van deze P3Speller. De P3Speller werkt met behulp van het knipperen van de rijen en kolommen in een willekeurige volgorde.

Bovenaan het scherm wordt het woord ‘BRAINPOWER’ weergegeven als geheugensteun. Daarachter wordt tussen haakjes weergegeven wat de letter is welke u op dat moment moet kiezen. Om de letter te kiezen moet u zich concentreren op de betreffende letter door in gedachten mee te tellen hoe vaak deze letter knippert. Nadat de letter 30 keer geknipperd heeft is er 8 seconden pauze waarna de volgende letter zal beginnen. In deze 8 seconden kunt u dus kort relaxen, de volgende letter op het scherm opzoeken en hier op focussen.

Dezelfde procedure zal hierna herhaald worden met het woord ‘PAINT’.

Als u een fout heeft gemaakt zoals het concentreren op de verkeerde letter, laat dit dan weten aan de experimentleider. De trainingssessie zal dan opnieuw gestart worden.

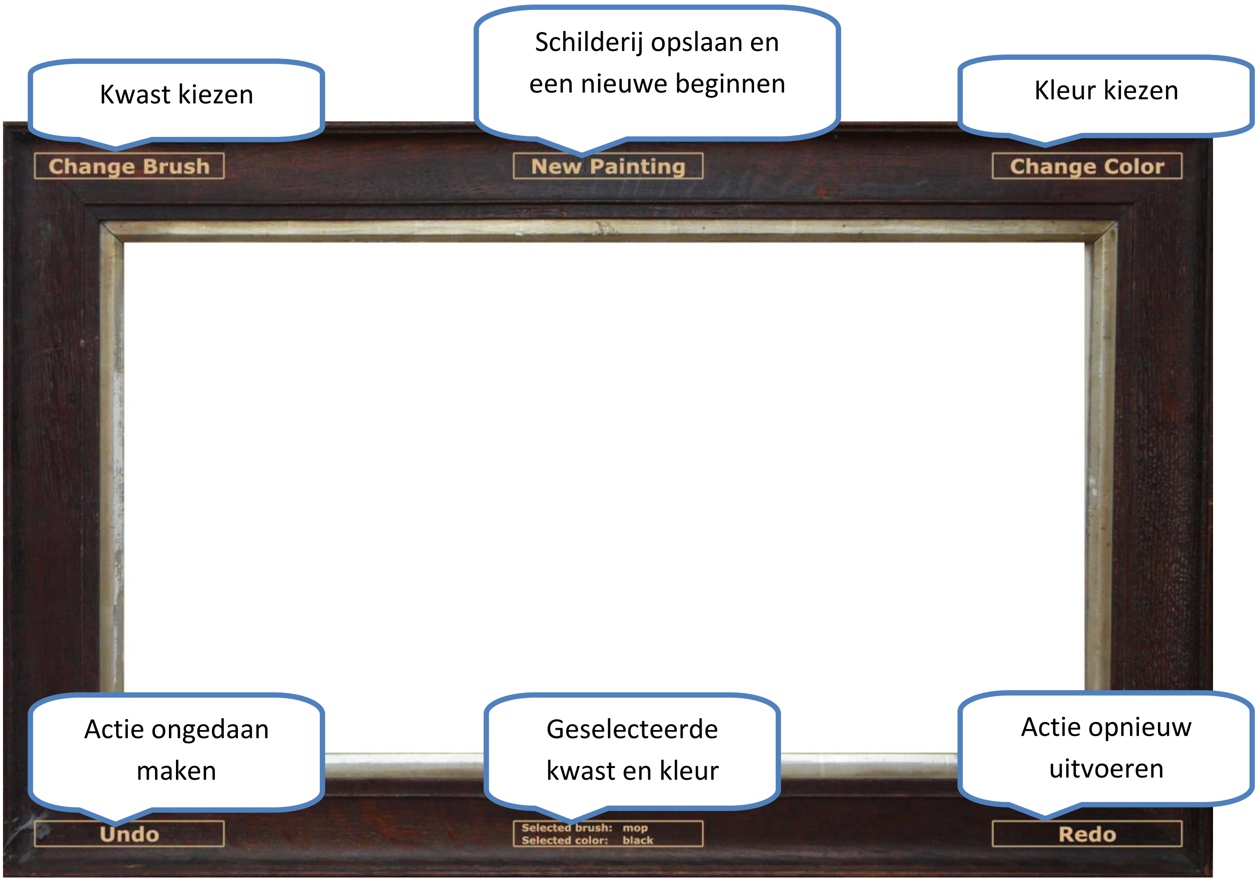
Appendix H

BrainBrush instruction document

Schilderen met je hoofd

U gaat dadelijk schilderen op een virtueel canvas door enkel uw hoofd te gebruiken. Hiervoor zult u gebruik maken van een speciaal hiervoor ontwikkeld programma. Om u enigszins wegwijs te maken in de mogelijkheden van dit programma krijgt u hier alvast enkele instructies. Lees deze goed door en laat de experimentleider weten wanneer u klaar bent. Er zal dan een korte oefensessie gestart worden waarin u het een en ander uit kunt proberen en waar nodig zullen enkele instellingen op u aangepast worden.

Na het starten van het programma ziet u het onderstaande scherm:



In het midden ziet u het virtuele canvas waar u op kunt schilderen. Op de lijst van het schilderij zijn een aantal keuzeopties weergegeven. Deze en andere functies worden hieronder uitgelegd.

Cursor bewegen

U kunt de cursor bewegen door uw hoofd te bewegen in de richting waarin u de cursor wilt verplaatsen.

Beginnen / stoppen met schilderen

Om de kwast op het canvas te zetten knippert u één keer met uw ogen. Om te stoppen knippert u opnieuw met uw ogen. Varieer de lengte van de oogknipper om te proberen hoe deze het beste herkend wordt.

Kwast kiezen

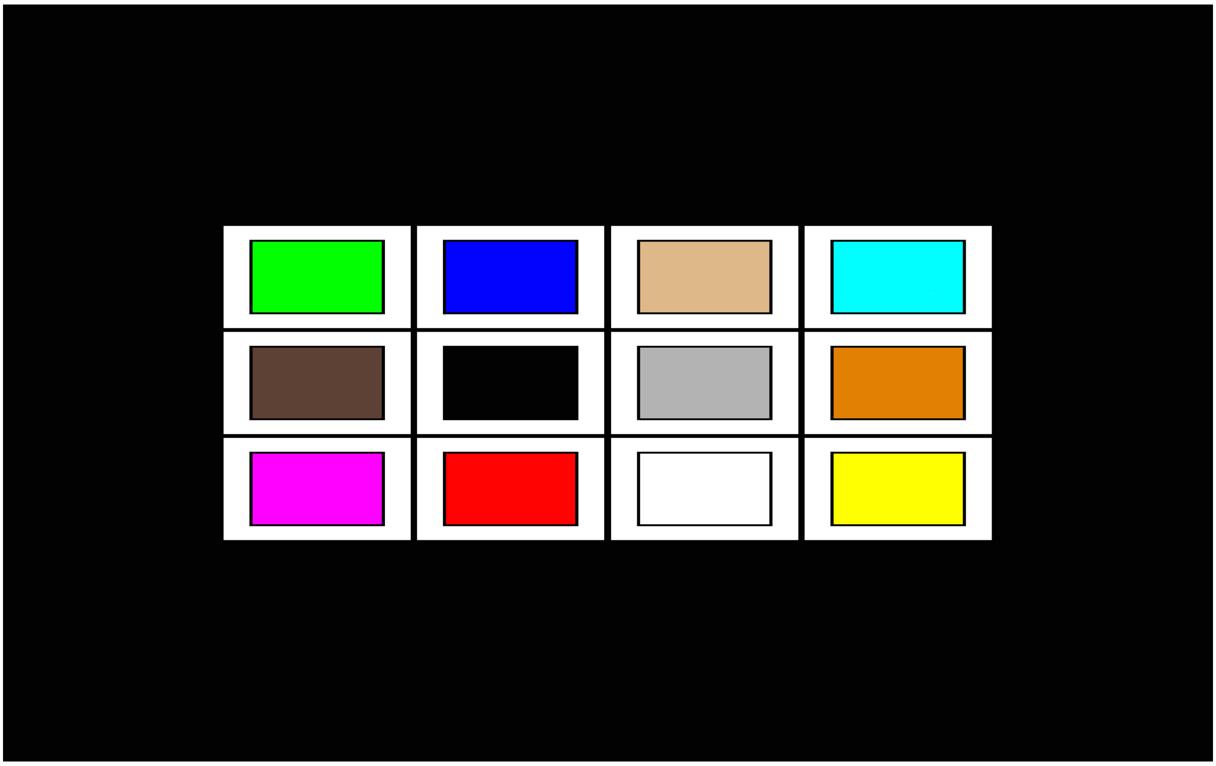
Om een andere kwast te kiezen beweegt u de cursor naar het vakje ‘Change Brush’. Hierna zal een matrix getoond worden, net als in de trainingssessie. Deze keer bevat de matrix echter plaatjes van verschillende kwasten. Het principe is hetzelfde: u moet zich concentreren op de kwast welke u wilt kiezen en het aantal keren dat deze knippert in gedachten tellen. Het programma zal daarna terugkeren naar het canvas en onderin het scherm laten zien welke kwast u volgens het programma wilde selecteren (bij ‘Selected brush’).

U kunt kiezen uit 12 verschillende soorten kwasten en andere tekenstijlen:

Kleur kiezen

Om een andere kleur te kiezen beweegt u de cursor naar het vakje ‘Change Color’. Hierna zal, net als bij het kiezen van een kwast, een matrix getoond worden. Deze matrix bevat nu de kleuren waar u uit kunt kiezen. Wederom moet u zich concentreren op de kleur welke u wilt kiezen en het aantal keren dat deze knippert in gedachten tellen. Het programma zal daarna terugkeren naar het canvas en onderin het scherm laten zien welke kleur u volgens het programma wilde selecteren (bij ‘Selected color’).

U kunt kiezen uit 12 kleuren:



Ongedaan maken / Opnieuw

Om de laatste bewerking aan het schilderij ongedaan te maken beweegt u de cursor naar het vakje 'Undo'. Om de ongedaan gemaakte bewerking opnieuw uit te voeren beweegt u de cursor naar het vakje 'Redo'.

Schilderij opslaan en een nieuwe beginnen

Als u uw schilderij wilt opslaan en een nieuw schilderij wilt beginnen beweegt u de cursor naar het vakje 'New Painting'. Het schilderij wordt dan opgeslagen en u krijgt een schoon canvas te zien. U kunt de gemaakte schilderijen naderhand toegestuurd krijgen, mocht u dat willen.

Appendix I

Task description document

Experiment

Uw taak voor dit experiment is om te schilderen op het virtuele canvas. U mag zelf bepalen wat u gaat schilderen, hoeveel schilderijen u maakt en hoe lang u hier mee door wilt gaan. Als u klaar bent slaat u uw laatste schilderij op door nog één maal ‘New painting’ te kiezen. Daarna mag u aan de experimentleider aangeven dat u wilt stoppen door de bel te rinkelen. Wacht alstublieft met het verwijderen van de EEG headset tot de experimentleider dit zegt.

Na afloop van het experiment zal de experimentleider u nog enkele vragen stellen.

Appendix J

System Usability Scale questionnaire

Proefpersooncode: ______

Please cross within one box:

	X			

And not like this:

	X (on the line between boxes 2 and 3)			

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

Appendix K

Interview questions

Algemene vragen over BrainBrush:

1. Wat waren je verwachtingen vooraf?
2. Heeft BrainBrush aan deze verwachtingen voldaan?
3. Heb je het gevoel dat je je creatieve ideeën op het virtuele canvas hebt kunnen overbrengen?
4. Had je een specifiek plaatje voor ogen en heb je dit kunnen tekenen?
    a. Zo niet: heb je het plaatje wat je in je hoofd had aangepast, en hoe heb je het aangepast?
5. Afhankelijk van of de proefpersoon zelf gestopt is of de maximale tijd vol heeft gemaakt:
    a. Zelf gestopt: waarom ben je gestopt?
    b. Tijd volgemaakt: had je nog veel langer door willen gaan?
6. Als dit programma incl de headset op de markt was voor 300 euro, zou je het dan willen hebben en er vaker mee willen tekenen?
    a. Zo niet: voor welk bedrag zou je het wel willen kopen?

Specifieke vragen over de verschillende modaliteiten:

7. Wat vond je van het gebruik van oogknippers?
    a. Is het systeem makkelijk te bedienen met oogknippers?
        i. Zo niet: waarom niet, wat maakt het lastig?
        ii. Zo ja: wat vond je het makkelijk maken?
    b. Merkt het systeem jouw oogknippers correct op?
        i. Zo niet: hoe vaak niet?
    c. Leidt het gebruik van oogknippers af van de andere handelingen die je moet uitvoeren om het systeem te bedienen?
        i. Zo ja: waar leidt het van af?
    d. Is het vermoeiend?
        i. Zo ja: waarom? Is het fysiek vermoeiend of mentaal?
8. Wat vond je van het gebruik van hoofdbewegingen voor de beweging van de cursor?
    e. Is het systeem makkelijk te bedienen met hoofdbewegingen?
        i. Zo niet: waarom niet, wat maakt het lastig?
        ii. Zo ja: wat vond je het makkelijk maken?
    f. Leidt het gebruik van hoofdbewegingen af van de andere handelingen die je moet uitvoeren om het systeem te bedienen?
        i. Zo ja: waar leidt het van af?
    g. Is het vermoeiend?
        i. Zo ja: waarom? Is het fysiek vermoeiend of mentaal?
9. Wat vond je van het gebruik van de P300 speller?
    h. Is het makkelijk te gebruiken?
        i. Zo niet: waarom niet, wat maakt het lastig?
        ii. Zo ja: wat vond je het makkelijk maken?
    i. Nam het systeem jouw keuzes correct over?
        i. Zo niet: hoe vaak niet?
    j. Is het vermoeiend?
        i. Zo ja: waarom? Is het fysiek vermoeiend of mentaal?

Vragen over de manier hoe de modaliteiten verwerkt zijn in BrainBrush.

10. Wat vond je van het gebruik van BCI technologie om je creatieve ideeën op het papier over te brengen?
11. Je hebt oogknippers, hoofdbewegingen en de P300 speller gebruikt om het programma te bedienen. Kun je deze 3 methoden rangschikken van meest aangenaam naar minst aangenaam?
    k. Waarom kies je voor deze rangschikking?
12. Wat vond je van de combinatie van deze 3 bedieningsmethoden voor de aansturing van het programma?

Afsluitende vraag:

13. Vond je het leuk?