MAKING BRAIN-COMPUTER INTERFACES BETTER

IMPROVING USABILITY THROUGH POST-PROCESSING

DANNY PLASS-OUDE BOS

MAKING BRAIN-COMPUTER INTERFACES BETTER

IMPROVING USABILITY THROUGH POST-PROCESSING

DANNY PLASS-OUDE BOS

Ph.D. Dissertation Committee

Chair & secretary	Prof.dr. Peter Apers	University of Twente, NL
Promotor	Prof. dr. A. Nijholt	University of Twente, NL
Assistant promotor	Dr. M. Poel	University of Twente, NL
Members	Prof. dr. D.K.J. Heylen	University of Twente, NL
	Prof. dr. F. van der Velde	University of Twente, NL
	Dr. G.F. van der Hoeven	University of Twente, NL
	Prof. dr. D. Mattia	Fondazione Santa Lucia, IRCCS, IT
	Prof. dr. R.J.K. Jacob	Tufts University, US
	Dr. W. Haselager	Radboud Universiteit Nijmegen, NL
	Dr. J. van Erp	TNO, Soesterberg, NL

CTIT Ph.D. Thesis Series No. 14-312
ISSN: 1381-3617
Centre for Telematics and Information Technology
P.O. Box 217, 7500 AE Enschede, The Netherlands

SIKS Dissertation Series No. 2014-38
The research reported in this thesis was carried out under the auspices of SIKS, the Dutch Research School for Information and Knowledge Systems.

The author gratefully acknowledges the support of the BrainGain Smart Mix Programme of the Netherlands Ministry of Economic Affairs and the Netherlands Ministry of Education, Culture and Science.

The research reported in this dissertation was carried out at the Human Media Interaction group of the University of Twente.

Cover design and various graphics by Sjusjun
LaTeX template: `arsclassica`, `classicthesis`
Printed by Wöhrmann Print Service

ISBN: 978-90-365-3779-7
DOI: 10.3990/1.9789036537797

MAKING BRAIN-COMPUTER INTERFACES BETTER

IMPROVING USABILITY THROUGH POST-PROCESSING

DISSERTATION

to obtain
the degree of doctor at the University of Twente,
on the authority of the Rector Magnificus
Prof.dr. H. Brinksma
on account of the decision of the graduation committee,
to be publicly defended
on Friday, 21st of November 2014 at 16:45

by

Danny Plass-Oude Bos
born on March 17, 1983
in Almelo, The Netherlands

This thesis has been approved by:
Prof. dr. ir. A. Nijholt (promotor)
Dr. M. Poel (assistant promotor)

Voor ANNA.

Lieve schat,

Je eindeloze nieuwsgierigheid
inspireert me elke dag.
Hou het vast.

♡ MAMA.

PREFACE

Eight years ago, I took my first BCI-related course. A lot has changed since then, not only in brain-computer interaction, but also in human-computer interaction in general. Touch interfaces were barely functional back then. Now they are everywhere and it is the only mode of interaction my daughter knows, aside from the physical buttons on her toys and the remote control.

Now we have a similar situation with brain-computer interfaces making their first tentative steps in commercial applications for the general public. I hope in another eight years, they too will be common-place. Although they are still a bit clunky right know, like touch interfaces then, brain-computer interfaces have a lot to offer. Not only for patients, but for everybody.

ACKNOWLEDGEMENTS

A thesis is something that develops over a long period of time with many people contributing to it in various ways, sometimes perhaps without even knowing it. There is no way I can do justice to everyone.

All the lovely people I got to know through BrainGain: thank you for the fun collaborations and interesting talks.

Dear HMI people, you have provided a fun and supportive environment all these years. Thank you for all the advice, interesting lunch discussions, and other *gezelligheid*. Special thanks to Charlotte and Alice, Lynn, Hendri, my roommates Mark and Jorge, Maral, Andreea, Olga, Randy, Dirk, Vanessa, Egon — wait, now I'm going to end up listing everybody! Mannes and Anton, thank you for your patience and support, but also for all the freedom you have given me to find my own way. Christian, Bram, Hayrettin, Boris, and Femke, thanks for all the advice, discussions, and the good times! Boris and Femke, I'm incredibly honored that you are willing to be my paranymphs. Let's make it a great day!

Dear family and friends, you have been like a trampoline: allowing me to jump high and catching me gently when I fall. Thank you. Mom, dad, sister, thank you for teaching me the important things in life, your encouragement, and all your help. Dear Anna, during this period you've grown into such a smart, sweet, self-reliant three year-old. Thank you for your patience every time I had to work once again on my thesis. Martijn, love of my life, as you were there every step of the way, you've had the most to endure. Thank you for your never-ending support. Words fall short. ♡

Danny Plass.

CONTENTS

ACRONYMS

BCI brain-computer interface.

EEG electroencephalography.

ERP event-related potential.

FMRI functional magnetic resonance imaging.

FNIRS functional near-infrared spectroscopy.

HCI human-computer interaction.

MEG magnetoencephalography.

MI motor imagery.

PET positron emission tomography.

SCP slow cortical potential.

SSVEP steady-state visually-evoked potential.

GLOSSARY

Application A computer program that helps a user with some particular task or goal.

Asynchronous The user can provide input to the system at any time. See also *synchronous*.

BCI cycle The cycle of interaction that takes place within and between the user and the system, including the processing.

BCI pipeline The sequence of data processing steps from the user to the BCI-controlled application. This simplified BCI system view ignores the effect the BCI pipeline can have on the user, which in turn will affect the system.

Biosemi ActiveTwo A high-grade EEG system, using a cap for positioning and gel for conduction. 256 electrodes can be mounted in one cap. The sampling rate can be a maximum of 16 kHz.

Brain-computer interface A system that recognizes mental tasks or states based on the user's brain activity. This allows you to control (or otherwise affect) devices (or applications) directly with your brain.

Cerebral cortex The part of the brain closest to the scalp, of which we measure the neuronal activity with EEG. The cortex consists of four lobes which are roughly related to planning and motivation, integration of sensory information, sound and verbal memory, and sight.

Control interface The control interface translates the logical control signal to a semantic control signal: something meaningful in terms of the application.

Electroencephalography The recording of voltage changes along the scalp. These changes are the result from activity of groups of neurons in the cortex.

Emotiv EPOC A commercially available, wireless head set for measuring EEG that is easy to use. It comes with 14 electrodes (plus CMS and DRL) at a 128Hz sampling rate.

Event-related potential A brain response that occurs related to some specific event (stimulus).

Feature extraction After pre-processing, the purpose of feature extraction is to magnify those characteristics that are most distinctive for mental state detection, and to suppress or remove the rest.

Feature translation When the most distinctive features have been derived, they can be translated to some logical meaning, which expresses what mental activity has been detected. This feature translation is commonly achieved through regression or classification.

Feedback Information from the system to the user. Feedback can inform the user on changes in the application state, but also on what user input the BCI detected.

Ground truth A label indicating the actual class of a certain data sample. This allows us to either train a classifier to label such samples as that class, or to determine the performance of a classifier by comparing the classification results with this known ground truth.

Human-computer interaction The interaction between people (users) and computers. As an area for research and design, HCI involves the study, planning, design and uses of this interaction.

Input modality Category of sensors or devices which provides a pathway over which the user can provide a certain type of input to the computer.

Input task An action the user has to perform to provide certain input to the system.

Interface The space that enables the user to communicate with a computer (input) and vice versa (feedback). This includes both hardware (such as EEG-caps) and software (such the graphical user interface).

Motor activity Motor activity — or: actual movement — results in brain activations similar to motor imagery. Actual motor activity, however, is easier to detect, easier to instruct, and provides a ground truth.

Motor imagery A mental task. The basis for its detection is that when we imagine a certain movement, this results in similar activity in the brain as *actually* executing that movement.

Mutual information The amount of information one sequence provides over another, in bits. When there is no relation between the two sequences, the mutual information is 0 bits. If one sequence completely determines the other, the mutual information is equal to the amount of information in the sequence, its entropy.

Non-stationarities Background changes in the observed brain activity recordings due to changes in the environment, in the user, or differences between users, which can interfere with the detection of mental states and mental tasks.

P300 A brain response identified by a positive voltage change around 300ms after a rarely-occurring event that is relevant to your current task. This brain response is mostly used for BCI spelling applications.

Post-classification processing Methods that aid in the translation from logical to semantical control signals, that is from classification (feature translation) results to input with meaning in the application context.

Post-processing See *post-classification processing*.

Pre-processing In the pre-processing stage, the data obtained through signal acquisition is processed to reduce artifacts and noise in the data.

Relaxation A mental state often used in commercial BCI systems which can be detected from the amount of alpha activity over the parietal lobe. This alpha activity is attenuated by mental effort.

Self-paced See *asynchronous*.

Signal acquisition The process of recording the user's brain activity (in the case of a BCI). Sensors measure the brain activity. The obtained samples are then converted to digital values to be used by the receiving device. See Emotiv EPOC and Biosemi ActiveTwo.

Slow cortical potential A category of slowly changing potentials. In BCIs, the most-used SCP is the Bereitschaftspotential, which is elicited in preparation for movement.

Steady-state visually-evoked potential A brain response. When a stimulus changes at a specific frequency, such as a flickering image that is inverted at regular intervals, we can observe this frequency and its harmonics in the brain. In the case of a visual stimulus, the brain response is called steady-state *visually*-evoked potential.

Synchronous Input is only observed during specific time slots, often when the input from the user is dependent on some stimulus from the system.

System-paced See *synchronous*.

Usability How easy and pleasant something is to use. This consists of various aspects, such as efficiency and effectiveness, learnability and memorability, error handling and user satisfaction.

User A person interacting with a computer or other device.

User experience The feelings and perceptions of a user as a result of interacting with a system.

ABSTRACT (NL)

BREIN-COMPUTER INTERFACES BETER MAKEN: VERHOOG HET GEBRUIKSGEMAK MET NA-VERWERKING

Met brein-computer interfaces (BCIs) kun je dingen rechtstreeks aansturen met je hersenen. Dit soort invoerapparaten gebaseerd op metingen van het lichaam hebben echter last van ruis, veranderingen en ambiguïteit. In het laboratorium kunnen we de systemen daar enigszins voor beschermen, maar in 'de echte wereld' kunnen BCIs wel wat extra hulp gebruiken.

Hoe belangrijk is goede besturing eigenlijk? Hoe goed kunnen gebruikers überhaupt hun controle inschatten? Veertien proefpersonen evalueerden ieder vijf weken lang drie verschillende sets van mentale taken. Het belangrijkst vonden ze dat de taken goed werden herkend door het systeem en dat ze makkelijk waren om te doen. Als mensen weten wat voor invoer ze geven, weten ze vrij goed hoeveel controle ze hebben. Zevenentachtig proefpersonen speelden een browserspelletje met verschillende mate van controle. De werkelijke mate van controle verklaarde 72% van de controle die men dacht te hebben.

Een simpele oplossing die de herkenning van hersensignalen kan verbeteren en de invoer kan vergemakkelijken is *post-processing* ('naverwerking'). Post-processing verandert hoe de herkende hersensignalen daadwerkelijk worden gebruikt in een applicatie. Post-processing is standaard bij andere invoersignalen, maar bij BCIs is dat nog niet het geval. Van de meer dan 200 BCIs waarover gepubliceerd is tot 2006 gebruikt maar 15% post-processing, volgens een eerdere literatuurstudie. Een vervolgstudie laat zien dat post-processing methodes nog steeds worden ondergewaardeerd in BCI onderzoek, hoewel de gerapporteerde verbeteringen met deze methodes erg veelbelovend zijn! Ik geef een overzicht van post-processing-methoden met richtlijnen voor toepassing, om bewust gebruik van en discussie te stimuleren. Tegelijkertijd blijft het belangrijk deze methodes te testen in de praktijk. Het doel van een experiment met achttien proefpersonen was de inspanning te verlagen met post-processing. Hoewel de tijd dat men de actieve taak moest uitvoeren significant werd verminderd, had het niet het verwachte effect op de *gevoelde* inspanning. Het afwisselen tussen de actieve en inactieve taak kostte meer moeite.

Dit werk bevestigt het belang van goede besturing voor de gebruiker en biedt onderzoekers en ontwikkelaars van BCIs een oplossing: post-processing. Een overzicht en richtlijnen worden aangegeven om bewust gebruik en discussie te stimuleren. Het onderzoek laat ook zien hoe essentieel gebruikersevaluaties zijn.

ABSTRACT (EN)

MAKING BRAIN-COMPUTER INTERFACES BETTER: IMPROVING USABILITY THROUGH POST-PROCESSING

Brain-computer interfaces (BCIs) allow you to control things directly with your mind. Unfortunately, such input devices based on observations of the body are plagued by noise, non-stationarities, and ambiguity. In the lab, we can protect systems somewhat from these influences, but in 'the real world', BCIs could use a little help.

How important is good control anyway? How well can users even assess their level of control? Fourteen participants evaluated three sets of mental tasks each for five weeks. Most important to them was good task recognition and easy task execution. When people know the input they provide, they have a good perception of their level of control. Eighty-seven participants played a browser game with varying levels of control. The actual amount of control explained 72% of the control they thought they had.

Post-processing is a simple solution to improve the recognition of brain signals and make it easier to provide. Post-processing changes the way detected brain signals are actually being used in an application. Although post-processing is standard practice with other inputs, this is not yet the case with BCIs. Of the more than 200 BCIs published about until 2006 only 15% used post-processing, according to an earlier literature study. A follow-up review shows that post-processing methods are still under-appreciated in BCI research, even though the improvements using these methods look very promising! To stimulate conscious use of and discussion about these post-processing methods, I provide a method overview with guidelines for application. At the same time, it is important to test these methods in practice. The goal of an experiment with eighteen participants was to reduce the necessary effort with post-processing. Although it did reduce the amount of active task execution time, this did not result in the expected reduction in *perceived* effort. Switching between the active and passive tasks cost more effort.

This work confirms the importance of good control to the user and offers BCI researchers and developers a solution: post-processing. An overview and guidelines are provided to stimulate deliberate use and discussion. The research also shows how essential user tests are.

EXTENDED SUMMARY

Controlling things with your thoughts is the domain of science fiction and fantasy. Brain-computer interfaces (BCIs) promise to bring this fantasy into the real world, as they can recognize mental tasks and mental states based on the user's brain activity. *Chapters 1 and 2* *Motivation*

Parts of this promise have been held up, but other parts seem to be more difficult. Being able to control things, does not necessarily mean you can control them well or easily. Many inputs based on measurements from the body suffer from similar problems related to noise, non-stationarities, and ambiguity. And these problems get worse the more we move towards real-world applications, with more noise, distractions, and multitasking.

Most research on BCIs is devoted to improving the *detection* of the mental tasks that drive these interfaces. My work focuses on improving the *control* over these systems. To overcome the problems inherent in this uncertain input modality based on observations from the body, we should take note from human-computer interaction research, where the user is at the center of design and evaluation. We can also learn from solutions used by other such uncertain input modalities.

The main research goals were first to examine the importance of control through user tests, and secondly to explore a possible solution with a literature review and a concluding experiment to test it in practice. *Research goals*

As a first step, I investigated what users prefer in their mental tasks for BCI control. For five weeks, fourteen people played a role-playing game using three different novel mental tasks to change their avatar from human to animal and back. The results were very consistent: What users want is first of all that the mental tasks are well recognized by the system, and secondly that these tasks are easy to do. Another important observation was that the perceived task recognition significantly impacted other user experience measurements. *Chapter 3*

Having said that, how well can users even assess how good the input recognition really is? Eighty-seven people played a browser game with a varying amounts of control. The actual level of task recognition explained 72% of the participants' perception of control. When people know what input they are providing, they appear to be competent at estimating their amount of actual control over the system. Uncertainty over provided brain-computer interface input will decrease with training, making the actual level of control more and more important to the user's sense of control. *Chapter 4*

Good control based on inputs that are easy to provide — that seems to be the opposite of what BCIs currently have to offer, especially in the case of consumer-grade hardware used in real-world situations. Fortunately, it is *Chapter 5* not that difficult to significantly improve the performance and reduce the effort it takes for the user to provide ‘thought input’, simply by adjusting the way detections are used through post-classification processing methods. Such methods are already commonly applied in all other input modalities. However, a 2007 survey of over 200 BCIs showed that only 15% of these systems used some form of post-processing to improve performance. After a follow-up literature study, I have to conclude that this undervaluation of these post-processing methods in BCI research still persists, despite the fact that most reported performance gains from adding post-processing methods are very promising.

Chapter 6

Only when the application of post-processing methods is done deliberately, and informed through discussion and structural evaluation, can we fully benefit. I have created an overview of post-processing methods, combined with guidelines for their application, to support this.

Chapter 7

To investigate how well this theory translates into practice, I conclude with a final experiment with eighteen participants which evaluates the influence three post-processing methods had on the perception of control and effort. Although the post-processing did result in a significant reduction of the amount of active task execution time, the *perceived* effort did not decrease accordingly. Apparently, switching between the active and passive tasks took more effort. This points to the importance of evaluating systems with users.

Chapter 8

Contribution

This work confirms the importance of task recognition accuracy in brain-computer interfaces from the users’ point of view, and offers a solution to the lack of accuracy inherent in this input device. It brings post-processing methods and their benefits to the attention of researchers and developers of brain-computer interfaces, and encourages their deliberate use with an overview and guidelines. This research also points to the significance of considering the user in the loop.

1 INTRODUCTION

> The design of the interface is a design of human experience and, as such, the interface becomes a locus of power.
>
> *Teena Carnegie*
> *Interface as Exordium [1]*

Brain-computer interface (BCI) is a somewhat futuristic term for somewhat futuristic technology. Controlling things with your thoughts has long been the domain of science fiction and fantasy. Yet I say '*somewhat* futuristic', because these interfaces are already available to consumers, right now.

A popular formal definition of BCI: a communication system in which messages or commands that an individual sends to the external world do not pass through the brain's normal output pathways of peripheral nerves and muscles [2].

A BCI is an input device, not that different from a keyboard which sends the keys you press to a computer. Instead of detecting key presses, brain-computer interfaces detect specific brain activations. As such, a BCI allows you to control devices directly with your brain. The term 'control' here should be interpreted loosely, as: "providing input for other devices so they in turn can respond to it in some way". Similarly, a 'device' can be anything that can respond to the output signal of a BCI, whether it is a wheelchair [3], a video game [4], or the international space station [5].

Brain-computer interfaces provide private, hands-free interaction. And as they are based on brain activity, they could come closer to assessing intent than any other interface [6]. Most BCI applications are aimed at health (assistive technology, therapy, wellness), finance (neuro-economics and neuro-marketing), and entertainment (mainly gaming) [7]. To give some concrete examples of products that are currently on the market: NeuroInsight, a market research company, analyzes how the brain responds to advertisements [8]. No Lie MRI offers a brain-based lie detector [9]. The Muse headband, created by InteraXon, comes with an app which helps you to manage stress and stay focused [10]. In short, BCIs can be pretty useful.

But there is one big complication: BCIs do not provide perfect recognition of what the user attempts to convey. Like other input modalities based on observations of the body, BCIs suffer from a number of basic problems that are difficult to combat [11, 12]. The sensors are highly sensitive to noise [13, 14]. It is very difficult to distinguish between activity intended for control and other activity — particularly similar activity that is triggered naturally: the so-called Midas Touch problem [15]. Moreover, there is the challenge of robustness to changes in the environment, changes in the user, and of differences between users [16]. It has also been posed that the high variability in BCI performance is exactly due to the fact that we look at the brain *directly*, when it is our cerebellum and our spinal motoneurons that make our interactions with the outside world smooth, adaptive, and accurate[1] [17].

The cerebellum, our 'little brain', is beneath the cortex at the back of the head.

All in all, it can be said that BCIs suffer from an inherent uncertainty in their detections. This is reflected in the way most BCI experiments are set up: the user is preferably put in a shielded room; is instructed not to move or blink, and stay relaxed; and is doing a very simple task.

In controlled settings, accuracies range from 61% up to 100% [18]. For this technology to become an accepted part of our everyday lives, however, it needs to be able to function in real-world situations, where the user is allowed to behave naturally. Potential users will also demand minimal training times (preferably none at all: plug-'n-play), and for the system to be as cheap as possible. Besides, they will probably be multitasking. Very few current applications have the input itself at the centre. Generally, the purpose for providing input is to meet some other user goal, which will require attention as well. Each of these needs will mean a reduction in recognition accuracy. Additionally, most of these needs are not only important for healthy users, but also for people with physical disabilities [19].

In this thesis I use 'I' when I speak for myself or describe something I did (mostly) by myself. I've also been part of a lot of team work, so when I use 'we' I'm discussing a collaborative effort. I also use 'we' in a more general sense, such as 'us BCI researchers and developers'.

Current research for improving brain-computer interfaces focuses mostly on increasing recognition through comparing various methods for feature extraction and mental state detection. With this approach, we leave out many aspects that also have large effects on how a brain-computer interface is experienced, such as the user, the mental tasks, and the mapping from the mental task input to application controls. The studies in this thesis are built around exactly these three aspects.

In the first half of the book, I investigate the need for a solution for this inherent uncertainty in BCIs by looking at the user and the mental tasks. The second half of the book explores the current state and promise of one specific solution: post-processing.

In Chapter 3 I look into the user preference and experience for a number of mental tasks, using user-centred methods from the field of human-computer interaction. It confirms that how well the task is recognized by the system is very important to the users, which takes us to Chapter 4, which looks at how well users can assess this system recognition aspect of control.

Then we move on to a solution to deal with the uncertainty inherent in mental task recognition, through the mapping of the mental input tasks to the application controls. This post-classification processing can significantly increase detection performance and the ease with which users con-

1 As a solution Wolpaw proposes to use goal selection instead of process control, by which he is trying to reduce the need for this smooth, adaptive, and accurate control.

THIS THESIS

trol the application. Post-processing to improve usability is already common practice for all other input modalities. The literature study in Chapter 5 shows the benefits and current state of post-processing in brain-computer interface research. Chapter 6 provides an overview of post-processing methods, combined with guidelines for their application. In Chapter 7 I investigate the effects of some of these methods in practice. This experiment also reveals the gap between the theory and practice when applying post-processing methods, thereby pointing the way to future research.

But before all that, I will quickly explain some of the basics of BCIs in the next chapter. If you have no previous experience with brain-computer interfaces, this information will help you understand the chapters that follow. That chapter also provides the motivation for BCI-related decisions that are common across the studies in this thesis.

KEY POINTS

Every chapter ends with key points, the most important statements of that chapter according to the author. It provides a quick reference and allows you to skip certain chapters while still getting the information that is essential for the chapters that follow.

- Brain-computer interfaces allow you to control (or affect) devices (or applications) directly with your brain. This private, hands-free input modality based on mental states can be used for a large variety of applications.
- The detection of mental tasks is imperfect, largely due to problems inherent in the type of input based on bodily measurements: problems such as noise, non-stationarities, and ambiguity. As a result, it is problematic to use BCIs in real-world situations.
- In this thesis, I propose to use post-classification processing methods to address a big part of this problem.
- To move towards a more holistic view in BCI development, I recommend the use of methods from human-computer interaction to observe these systems as a whole, including the user and the application.

REFERENCES

[1] T. A. M. Carnegie. “Interface as exordium: The rhetoric of interactivity.” In: *Computers and Composition* 26.3 (2009), pp. 164–173 (cit. on p. 3).

[2] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan. “Brain-computer interfaces for communication and control.” In: *Clinical neurophysiology* 113.6 (2002), pp. 767–791 (cit. on p. 3).

[3] A. R. Satti, D. Coyle, and G. Prasad. “Self-paced brain-controlled wheelchair methodology with shared and automated assistive control.” In: *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2011 IEEE Symposium on*. IEEE. 2011, pp. 1–8 (cit. on p. 3).

[4] A. Lécuyer, F. Lotte, R. B. Reilly, R. Leeb, M. Hirose, and M. Slater. “Brain-computer interfaces, virtual reality, and videogames.” In: *Computer* 41.10 (2008), pp. 66–72 (cit. on p. 3).

[5] L. Rossini, D. Izzo, and L. Summerer. “Brain-machine interfaces for space applications.” In: *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*. IEEE. 2009, pp. 520–523 (cit. on p. 3).

[6] D. Plass-Oude Bos, B. Reuderink, B. L. A. van de Laar, H. Gürkök, C. Mühl, M. Poel, A. Nijholt, and D. K. J. Heylen. “Brain-Computer Interfacing and Games.” In: *Brain-Computer Interfaces: Applying our Minds to Human-Computer Interaction*. Ed. by D. S. Tan and A. Nijholt. Springer, 2010. Chap. 10 (cit. on p. 3).

[7] F. Nijboer, B. Z. Allison, S. Dunne, D. Plass-Oude Bos, A. Nijholt, and P. Haselager. “A Preliminary Survey on the Perception of Marketability of Brain-Computer Interfaces and Initial Development of a Repository of BCI Companies.” In: (2011). Ed. by G.R. Mueller-Putz, R. Sherer, M. Billinger, A. Kreilinger, V. Kaiser, and C. Neuper, pp. 344–347 (cit. on p. 3).

[8] NeuroInsight. *NeuroInsight*. `http://www.neuro-insight.com/`. Last accessed: August 4, 2014. (cit. on p. 3).

[9] No Lie MRI. *No Lie MRI*. `http://www.noliemri.com/`. Last accessed: August 4, 2014. (cit. on p. 3).

[10] InteraXon. *Muse*. `http://www.choosemuse.com/`. Last accessed: August 4, 2014. (cit. on p. 3).

[11] L. Deng and X. Huang. “Challenges in adopting speech recognition.” In: *Communications of the ACM* 47.1 (2004), pp. 69–75 (cit. on p. 3).

[12] R. J. K. Jacob and K. S. Karn. “Eye tracking in human-computer interaction and usability research: Ready to deliver the promises.” In: *Mind* 2.3 (2003), p. 4 (cit. on p. 3).

[13] E. B. J. Coffey, A.-M. Brouwer, E. S. Wilschut, and J. B. F. van Erp. “Brain-machine interfaces in space: using spontaneous rather than intentionally generated brain signals.” In: *Acta Astronautica* 67.1 (2010), pp. 1–11 (cit. on p. 3).

[14] R. R. Wehbe and L. Nacke. “An Introduction to EEG Analysis Techniques and Brain-Computer Interfaces for Games User Researchers.” In: *DiGRA 2013: DeFragging Game Studies*. Digital Games Research Association DiGRA. 2013 (cit. on p. 3).

[15] M. M. Moore. “Real-world applications for brain-computer interface technology.” In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 11.2 (2003), pp. 162–165 (cit. on p. 4).

[16] B. Reuderink, J. Farquhar, M. Poel, and A. Nijholt. “A subject-independent brain-computer interface based on smoothed, second-order baselining.” In: *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*. IEEE. 2011, pp. 4600–4604 (cit. on p. 4).

[17] J. R. Wolpaw. “Brain-computer interfaces as new brain output pathways.” In: *The Journal of Physiology* 579.3 (2007), pp. 613–619 (cit. on p. 4).

[18] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi. “A review of classification algorithms for EEG-based brain-computer interfaces.” In: *Journal of neural engineering* 4 (2007) (cit. on p. 4).

[19] F. Nijboer, D. Plass-Oude Bos, Y. Blokland, R. van Wijk, and J. Farquhar. “Design requirements and potential target users for brain-computer interfaces – recommendations from rehabilitation professionals.” In: *Brain-Computer Interfaces* 1.1 (2014), pp. 50–61 (cit. on p. 4).

2 BCI BASICS

This chapter gives a short introduction into brain-computer interfaces. It explains how brain-computer interfaces work and introduces some of the terminology common in this field, with special focus on the context of the research in this thesis.

2.1 BCI AND GAMES

A brain-computer interface allows you to provide brain-based input to a computer. The result is private, hands-free interaction. And as the input is based on your brain activity, BCIs could come closer to assessing your intent than any other interface [1]. As already mentioned in the introduction, most BCI applications are aimed at health (assistive technology, therapy, wellness), finance (neuro-economics and neuro-marketing), and entertainment (mainly gaming) [2].

In this thesis, the focus is on games. A large part of the population plays games, and it is known that gamers are often among the first to adopt new technology [3]. Learning a new skill like this could be part of the challenge of the game [4]. It comes as no surprise then that many of the current BCI applications are game-oriented.

Games are a compelling target for brain-computer interfaces, but brain-computer interfaces also have a lot to offer to games. Immersion may be increased through such intuitive input or by having the player's mental state reflected in the game [5]. Through neurofeedback mechanisms, BCI games can also train players to be more relaxed or concentrated, or may even help with ADHD and anxiety [6]. For further reading, there are many interesting overview papers on the use of BCI in games, such as Lécuyer et al. [7], Nijholt [8], and Marshall et al. [9]. There are also sources of inspiration aimed at game developers specifically, such as the 'brain-enhanced gaming concepts' published by Neurosky, a neuro-headset manufacturer [10].

From a scientific point of view, games are also interesting. Games provide a safe virtual test ground (as opposed to, for example, navigating a mind-controlled wheelchair through actual traffic). Besides, games can help experiment participants to stay motivated and focused for longer periods [11].

2.2 THE COMPONENTS OF A BCI

The interaction of the user with a computer is often visualized as a cycle. A good example is the model by Chapanis, see Figure 1 [12]. The user (human) provides information to the computer (machine), and vice versa. The interface is in between the user and the computer. Note the distinction between how the information is provided, and how it is perceived. Physical actions by the user are being perceived as control inputs in the machine. In our case, the brain-computer interface determines which brain responses are listened for (the equivalent of the 'motor responses'), and which subsequent information is sent to the machine ('controls').

Human-computer interaction

Interface

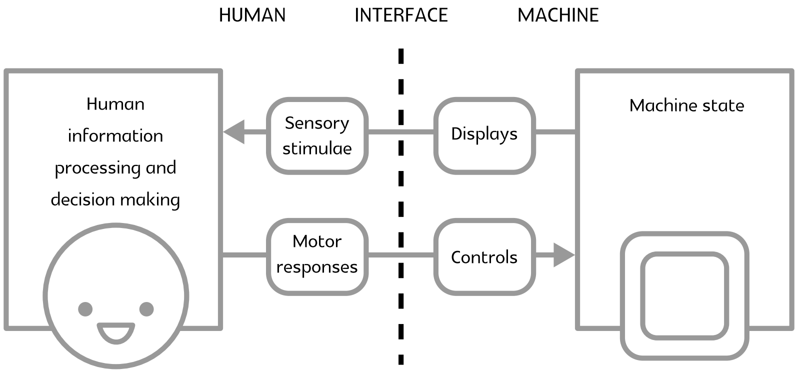


FIGURE 1: **Human-computer interaction model**, based on Chapanis, 1965 [12].

Mason and Birch propose a more elaborate cycle for brain-computer interfaces specifically, which distinguishes various key steps in the process from measuring brain activity to using the interpretations for control, see Figure 2 [13]. This interaction cycle is often referred to as the *BCI cycle* [14]. The word 'cycle' emphasizes that there is a feedback loop. Adjusting one analysis step does not only affect the steps that follow, but may therefore also influence the steps before. This is an important reason to test BCI systems as a whole, with the user in the loop. The sequence of processing steps is sometimes also referred to as the *BCI pipeline*, which infers a simplified linear point of view. With this kind of thinking, it is possible to test different pipelines on one pre-recorded dataset of brain signals. It is important to remember, however, that this is a simplification that may not hold in practice, as a different pipeline may cause the user to provide different input.

BCI cycle

BCI pipeline

The BCI cycle consists of the following steps: the user, signal acquisition, pre-processing, feature extraction, feature translation, the control interface, application, and feedback[1].

1 The main difference between the Mason and Birch cycle and most other models is the presence of the *control interface*. This element is crucial to the research presented in this thesis. The

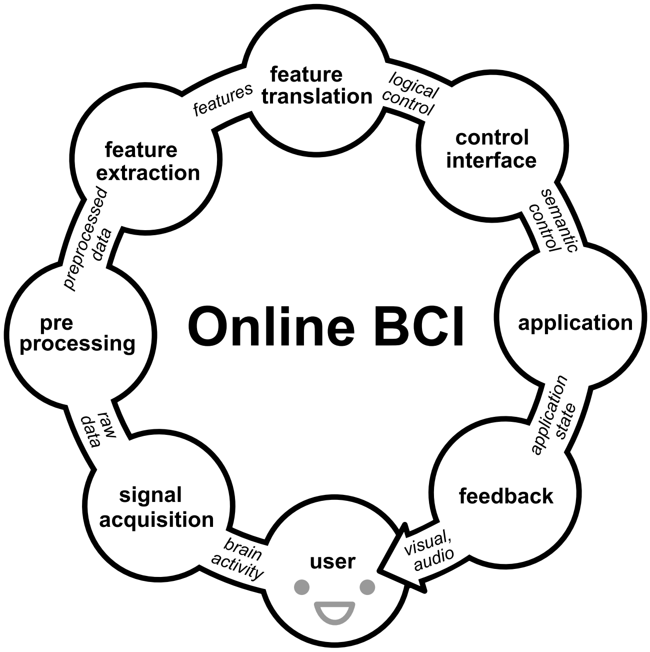


FIGURE 2: **Model of the online BCI cycle**, which indicates the various processing steps, and the data streams in between.

Signal acquisition *Preprocessing*

The BCI hardware records the user's brain activity, so it handles signal acquisition. Often it also applies some pre-processing (pre feature extraction) to reduce artefacts and noise in the data.

Feature extraction

Feature translation

The subsequent processing steps of the brain-computer interface can theoretically be part of the hardware, but are generally implemented in the software of the device receiving the hardware input. This software part can do additional preprocessing, followed by feature extraction and translation. During feature extraction, the preprocessed brain signals are transformed to magnify those characteristics that are most distinctive for what we are trying to detect. The rest is suppressed or even thrown out completely. These transformed features are then translated so they provide some logical meaning. For example, high activity over the left sensorimotor cortex (central-left on the head), could be translated to a high probability of right hand movement (yes, it is on the opposite side). This logical con-

main difference between my model and that of Mason and Birch is the substitution of the *device* and *device controller* by *application*. This is because my model is focused on software, while Mason and Birch focused on controllers for hardware such as wheelchairs. With hardware as well as software, it is recommended to dedicate as little screen estate or physical space to the controller as possible. The controller is simply a means to an end, and it is the end that the user should be able to focus on. A controller device can even physically block the user from participating in social settings or get in the way of using the device they are trying to control in the first place, for example when you cannot see where the wheelchair is going [15].

trol signal expresses what mental activity has been detected by the brain-computer interface.

This detection then needs to be translated into something meaningful in terms of the application that is being controlled. This can either be handled by the BCI, the application, or a separate software module altogether: the control interface. A big part of this thesis is about this translation from logical to semantic control. Two concrete examples from this thesis: If the BCI indicates low relaxation, then turn the player avatar into a bear (Chapter 3). And: If the BCI observes hand movement, select current option (Chapter 7).

Control interface

Looking at the way input device software is currently implemented, the control interface is simply the final step in the device driver[2]. The description on Wikipedia for *Device driver* clearly shows the relationship between *driver* and *control interface*:

Device driver

> "Device drivers [act] as translator between a hardware device and the applications or operating systems that use it." [17]

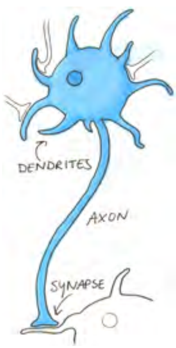
Both pieces of software are described as a translator between the device output and the application. The article further explains the benefits of this:

> "Programmers can write the higher-level application code independently of whatever specific hardware the end-user is using."

As a programmer, you should not have to know how a keyboard works exactly, or what specific keyboard the user has. The application can simply catch the key presses that occur. Similarly, an application programmer should not need to know how a BCI works, only what input it can provide. The application should simply be able to listen for specific events, such as changes in the user's level of relaxation. Drivers often offer the user some customization options. Changes in the driver will affect the input received by all applications.

If the control interface is the driver, then examples of in-application post-processing are application-specific key-bindings or to context-dependent responses in the application. This means that any application should be able to listen for the user's level of relaxation as detected by the BCI, but different applications may have different processing of that same input. It is those additional in-application translation steps that create the stage for optimal usability.

In-application post-processing



A neuron receives input via dendrites. This can trigger the neuron to send an electrochemical signal over its axon.

2.3 SIGNAL ACQUISITION

So, how can we observe brain activity? The brain is an enormous network of neurons. Neurons are cells which communicate with one another by sending electric currents. This electrical activity, or the resulting magnetic fields,

2 Allison describes BCI drivers as a crucial part of BCI software integration [16].

can be measured, with EEG or MEG. Another indication of brain activity is the change in blood flow as active cells require more oxygen carried by red blood cells. This principle is used by methods such as PET, fMRI, and fNIRS.

One of the most-used methods for recording brain activity is EEG, electroencephalography. Electrodes on the outside of the head measure voltage differences that are the result of the activity of large groups of neurons. It is so popular, because it does not require surgery, is relatively cheap, portable, and responds quickly to changes in brain activity. EEG has also been the method of choice for the BCI experiments described in this thesis.

An electrode and the cortex.

This brain imaging method also comes with some drawbacks, however. The electrodes measure only superficially, so we can observe the cortex, but none of the deeper brain structures. As the electrodes are on the outside of the head, the measurements are highly attenuated and spread out by the fluids, bone, and skin in between the neurons and the sensors. Besides, EEG has a poor spatial resolution: we cannot fit that many sensors onto a certain area on the head, making it less precise in terms of location[3]. Additionally, the electrodes are highly sensitive to artifacts, both from the body and from the environment, while the voltage differences to be measured are weak. This results in a low signal-to-noise ratio.

One of the most entertaining explanations of why it can be so difficult to interpret EEG is the metaphor by Dr David Lewis, cited in a book by John Naish: "Using the [EEG] machine is like standing outside a football ground, trying to interpret the action in the game by listening to the roars of the crowd" [20].

EEG systems range from high quality medical systems to much cheaper consumer-grade EEG headsets. Higher grade EEG systems generally result in better measurements, from more electrodes, which are more precisely positioned. These systems are also 100 times more expensive, require conductive gel, and a trained person to mount the electrodes on your head. Consumer EEG sets, on the other hand, are more easy to use. You can put them on yourself, they are generally wireless, and work either 'dry' or with a little contact lens fluid. And, not to be underestimated, they are designed to look good too. In my earlier research, I used a high grade BioSemi ActiveTwo system (Chapter 3). Later on, I switched to the consumer-grade Emotiv EPOC, see Figure 3 (more on this headset in Chapter 7).


The Biosemi headset.

3 Putting things in perspective with some numbers: the average adult male human brain has 86 billion neurons. The cerebral cortex contains only 16 billion of those neurons [18]. And when we then use an EEG headset with 16 electrodes, we are trying to observe about one billion neurons with just one electrode. Secondly, the amplitude of an action potential, which is the electrochemical impulse neurons send over their axon to communicate, is about 100mV [19]. What we measure with EEG on the outside of the brain is in the microvolt range.

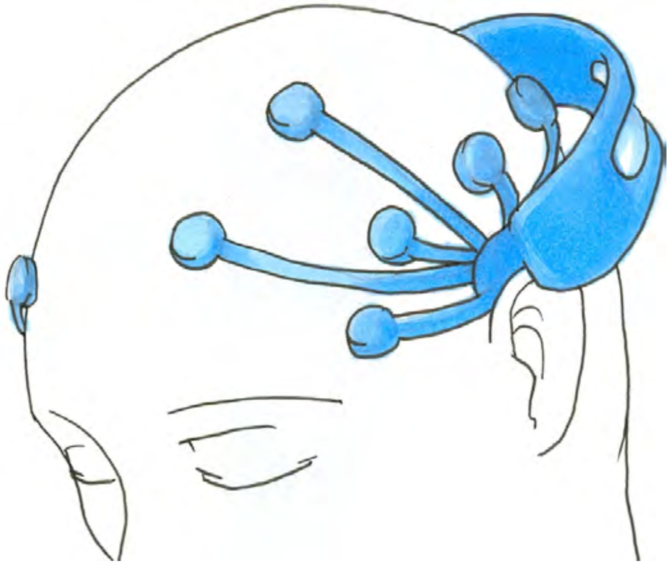
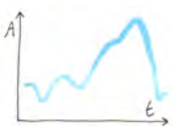


FIGURE 3: **The Emotiv EPOC.** A commercially available, wireless head set that is easy to use. It comes with 14 electrodes (plus CMS and DRL) at a 128Hz sampling rate.

2.4 MENTAL INPUT

Which mental inputs could we use to control a brain-computer interface? The four most-used inputs are: P300, SSVEP, SCP, and MI [21, 22]. They have been thoroughly researched in neuroscience, and are still popular research topics today. They have proven to be relatively easy to detect, and are from that point-of-view quite suitable to control things with[4]. I will quickly describe six mental inputs: the classic four mentioned above, followed by relaxation and motor activity, which are two other inputs you will encounter in this thesis. The accompanying brain responses are specified in terms of area in the brain where they are most dominant, and whether it is a potential (a wave), or a rhythm (repeating waves).



A P300 potential.

P300 When you see (or hear, or feel) something that occurs rarely, and that is relevant to what you are currently doing, we can observe a specific wave in your brain activity [21]. As it is related to some specific event, this wave is called an Event Related Potential (ERP). This positive wave oc-

4 Other points of view could include how intuitive the user task can be matched to a system response. Imagining to move your right hand to move an object to the right is more logical than having to imagine moving your tongue to move the object. Another important aspect is how much effort it takes the user to provide this kind of input. See Chapter 3 for an experiment on this topic. But currently, the main criterion for selecting brain-based inputs is how well they can be detected.

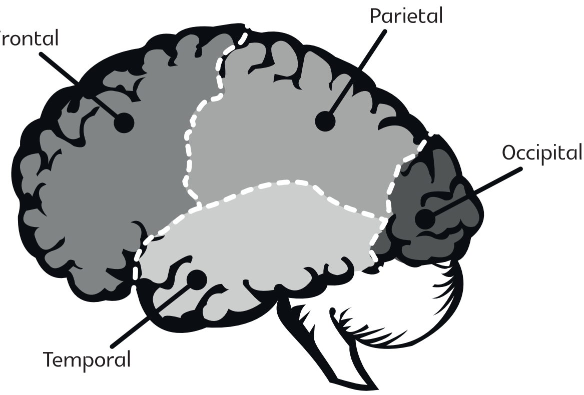


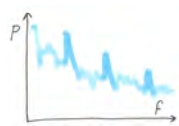
FIGURE 4: **A side view of the brain, showing the four main lobes of the cerebral cortex:** frontal, parietal, temporal, and occipital. These lobes are roughly related to planning and motivation, integration of sensory information, sound and verbal memory, and sight, respectively.

curs with a delay of around 300 milliseconds, hence the name 'P300', and is strongest in the parietal lobe (center back on the head, see Figure 4). Currently, this input is mostly used in brain-based spelling applications, where you can select characters from a matrix by concentrating on the one you want to type, and mentally counting each time this character is highlighted (making it task-relevant) [23]. See Chapter 5 for a more elaborate description and example of such a P300 speller.

A	G	M	S	Y	5
B	H	N	T	Z	6
C	I	O	U	1	7
D	J	P	V	2	8
E	K	Q	W	3	9
F	L	R	X	4	_

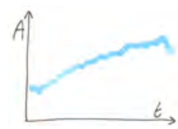
A P300 speller matrix.

SSEP Steady-State Evoked Potentials (SSEP) happen when you focus on a flickering image, listen to modulated sound (that repeatedly toggles on and off), or feel some vibration. When you observe a change, your brain responds. When this change occurs at a specific frequency, we can observe this frequency and its harmonics in your brain [24]. Depending on the sense in question, the frequency can be observed in different areas of the brain. Steady-state *visually*-evoked potentials (SSVEPs) appear in the occipital lobe (see Figure 4). Most-often this is used to select an option on screen. Each option flickers at its own frequency, and you concentrate on the one you want to select.



SSEP base frequency peak and harmonics.

SCP Slow Cortical Potentials (SCPs) is the name for a group of slow moving potentials mostly observed in the frontal and central parts of the cortex [25]. In BCIs, often the user task is "preparation of movement", to trigger the so-called Bereitschaftspotential [26]. One can, for example, imagine preparing to shoot an arrow with a bow in order to cause a negative brain



A positive slow cortical potential.

signal shift [27]. As opposed to P300 and SSVEP, this input requires extensive user training.

The mu rhythm.

MI MI stands for Motor Imagery. It is the only one of these mental tasks that is not named in terms of the brain response, but for the task the user has to execute. Motor imagery can actually be detected both from potentials (the aforementioned Bereitschaftspotential), and from specific rhythms (the mu rhythm in the alpha range, and beta rhythms) in the sensorimotor cortex (see Figure 5) [28]. The basis for this mental task is that when we imagine a certain movement, this results in similar activity in the brain as actually executing the movement. Popular body parts for this task are hands, feet, and tongue, as they are represented by relatively large parts of the brain, making them more easy to detect.

Mental input tasks can be subdivided according to various characteristics, for example whether it requires *user training* (do you need to learn how to execute the task, or does it come naturally or automatically?), *system training* (does the system need to learn to recognize you specifically, or is the response very similar for everyone?), whether it requires conscious, *active* input or can be used *passively*, and whether it requires a *stimulus* to be presented to the user (such as the flickering in SSVEP) or can be *self-induced* (such as imagining tapping your hands). Related to this distinction between externally-evoked and self-induced tasks is *system-paced* versus *self-paced* input[5], which results in *non-stop* and *intermittent* input, respectively. In the system-paced case, the system will only listen for user input during specific moments. When the input depends on some external stimulus, the moment of stimulus presentation will happen right before the system listens for input. On a side note, the system using externally-evoked input can potentially provide the stimulation continuously, so the power of input initiation is put back into the user's 'hands'. Input can also be *continuous* or *discrete*, so either a value along some axis (such as a concentration level of 0.8), or one of a set of predefined class labels (such as concentration 'high').

When applying these characteristics to the classic four, we see that P300 and SSEP are stimulus-evoked. MI and SCP are self-induced. MI requires some user training, as most people will have had no practice with it, but can extrapolate from their experience with actual movement. SCP requires extensive user training. It is common to use system training, as generally this will result in better task recognition, but there is a lot of research into the development of subject-independent BCIs. All of these inputs generally require active, conscious action from the user, and are normally used in a discrete way, the output being either 'on' or 'off', or a label indicating a specific selection[6]

5 System-paced and self-paced is also known as synchronous and asynchronous.

6 Admittedly, this statement is a generalization. For example, for P300-detection to work well, the user actively counts occurrences of the target. But there is also research into P300-based

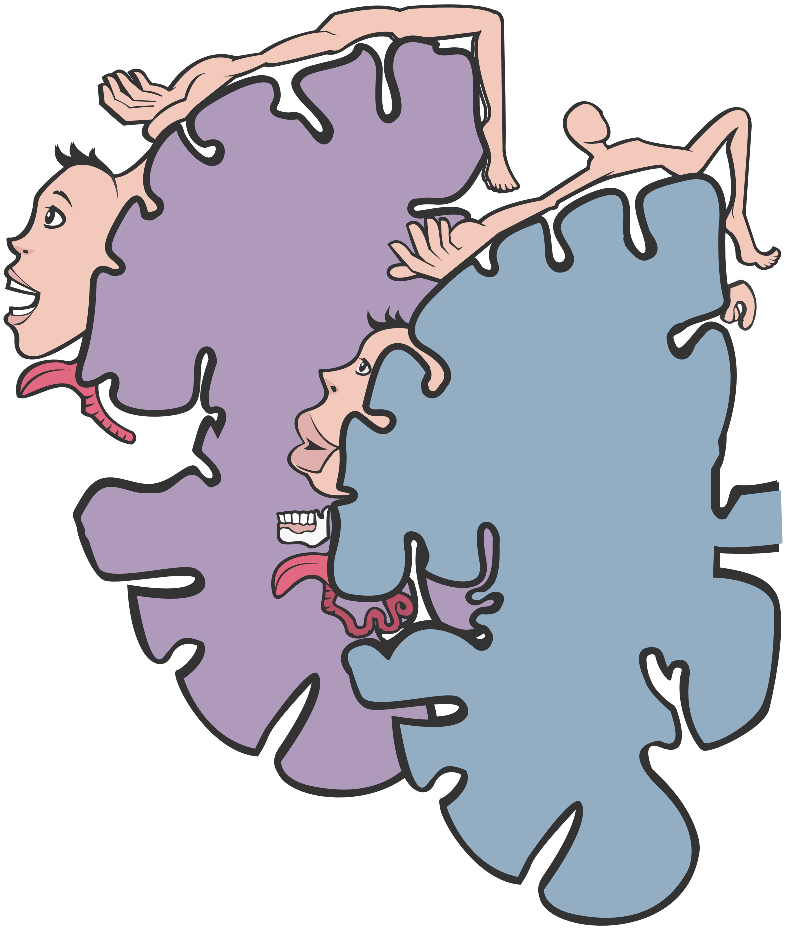
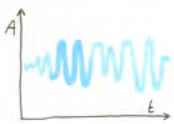


FIGURE 5: **The sensorimotor-cortex homunculi.** The central sulcus (fold) separates the frontal and parietal lobes (see Figure 4). The ridge on the side of the frontal lobe is the primary motor cortex (for motor control), and the ridge on the parietal side is the somatosensory cortex (for tactile sensations). Together, they are referred to as the sensorimotor cortex. Both contain a representation of the body, called a homunculus. The primary motor homunculus is shown here at the back; the sensory homunculus at the front. The larger the related area, the easier it will be to detect related brain activity. The hands are therefore a popular target for BCIs based on motor imagery.

This thesis features some other mental inputs. In Chapter 3 a whole new set of mental tasks is designed for a specific set of in-game actions. Below I will explain one of them, 'relaxation', in a bit more detail, as we have also used this particular mental task in other experiments and in many demonstrations. In Chapter 7 you will find the task of actual motor activity. Relaxation and motor activity are both self-induced, so the task can be initiated by the user. They are also both related to familiar concepts, so they should come more naturally, and require minimal user training.



Alpha waves.

Relaxation Like the previously described 'classic four', relaxation has been thoroughly researched in neuroscience. For the AlphaWoW prototype, further described in the example at the end of this chapter, I chose the alpha activity over parietal lobe (see Figure 4) as an indicator of relaxation, as it is often described as a correlate for a state of relaxed wakefulness [30]. Alpha activity is attenuated by attention and mental effort [31, 32]. What makes this mental input particularly interesting is that it can be used actively and passively by the user. In practice we observe that the way it is used often changes within a session.

Motor activity As already mentioned, actual movement results in brain activations that are similar to imaginary movement (see also Figure 5). The main differences are that actual motor activity is easier to detect [33, 28], easier to instruct, and one can observe what the user is doing. Such a ground truth is often missing with BCI input.

2.5 EXAMPLE: ALPHAWOW

To give a concrete example of the steps and characteristics mentioned above, I will describe one of our BCI prototypes: AlphaWoW. It has been used in research [34, 35], although no experiment details have been included in this thesis (it would have detracted from the main theme). It has been used in many demonstrations as well, with as highlight a demo talk at TEDxAmsterdam in 2009. This prototype is also closely related to the system used in

---

lie detectors. Obviously a lie detector would not be very useful it would require such a degree of voluntary participation from the user. Along similar lines, SSEPs can be used as an indicator of the amount of concentration on a target on-screen. This amount of concentration is then likely to be passed on as a continuous value, instead of discrete, as we did in our Bacteria Hunt game [29] for example. However, the statement still holds for most systems. Many of these characteristics are not inherent in the mental input task per se, but also depend on the way the input is used in the rest of the system. That being said, certain inputs will be more suitable for certain types of control. And certain inputs have been used for certain types of control for so many years, an unconscious, trained prejudice may have been created towards certain combinations.

Chapter 3, featuring the same control interface, application, and application add-on.

AlphaWoW shows what it would be like to have an intuitive, mental-state-based control in a role-playing game. In the popular game World of Warcraft® (developed by Blizzard Entertainment, Inc®), you can play a druid. Druids can shape-shift into animal forms. In this BCI version of the game, your shape depends on your level of relaxation, see Figures 6 and 7. When you are relaxed, you are in your normal human form[7], but when you get agitated, you automatically change into a bear.

FIGURE 6: **BCI control in World of Warcraft®:** When the user is relaxed (high parietal alpha activity), the avatar is humanoid. When the user gets agitated (low parietal alpha activity), the avatar becomes a bear.

Following the processing steps of the online BCI cycle model explained in the beginning of this chapter (Figure: 2):

User The task for the user here is to either try to stay relaxed, or to get agitated. Instead of trying to consciously control this, the user can also consider the shape to be simply feedback on their current mental state, and play the game as best as they can with whatever this state turns out to be at that moment. So the input can be active or passive, depending on user preference. In the case of active control, the input is self-induced. When used passively, one could say that the input is stimulus evoked, as the level of relaxation will fluctuate depending on what happens in the game. Whatever the case, the system listens for input non-stop.

Signal acquisition In the early days of my research, I used the Biosemi ActiveTwo headset, which is a high-grade EEG set, using cap and gel. Later *See Section 2.3.*

7 The human form is a night elf, to be exact.

FIGURE 7: **Participant playing AlphaWoW:** with the Biosemi headset. Mouse and keyboard are still used for movement, selection, and camera control.

on the system was adjusted to work with the consumer-grade Emotiv EPOC, which was a lot easier to use and demonstrate.

Pre-processing, feature extraction, translation The amount of parietal activity (in the back of the head) in the alpha frequency band (8–13 Hz) is used as an indicator of relaxation (see the description of Relaxation input in the previous section). For this, the BCI first selects the parietal electrode channels, and computes the absolute alpha-band power for each[8]. To ensure a normal value distribution, the log of the bandpower values is computed. The initial indicator value for relaxation is then obtained by taking the sum of these log bandpower values.

The indicator value we have obtained thus far may still vary widely across users. Adaptive z-score normalization (subtract the mean, divide by the standard deviation, with the mean and standard deviation based on recent observations) forces this indicator in the same range for every user: 95% of the values should now occur between -2 and +2. This automatically adjusts the system to the user, and prevents the user from getting stuck in high or low relaxation for the entire session. For easy interpretation, this value is

8 The relative power would indicate a percentage of the power over all frequency bands combined. Although this can be a convenient way to somewhat normalize the values obtained in this process, this relative power can fluctuate highly based on the activity in the other bands. To avoid this, the AlphaWoW system looks solely at the absolute alpha-band power.

scaled to be in the range from 0 to 1 (from the original -2 to +2 standard deviations). Anything lower or higher is cut off. We have now arrived at a value with a user-independent logical meaning: the amount of relaxation.

Control interface To make this relaxation value less sensitive to outliers, a weighted moving average is applied of [0.2, 0.3, 0.5], with the most recent observation contributing the most. The reduced sensitivity also makes the system respond slower to intended changes. Such trade-offs are a common theme in brain-computer interfaces. See Section 6.2 for more examples.

Then this value is passed on to a separate application which translates the level of relaxation into key presses which are used to communicate with the proprietary World of Warcraft, as this game is closed off for any other means of receiving user input. Conditions for certain keys are defined in terms of both value and duration thresholds (dwell times). Chapters 5 and 6 will discuss post-processing methods such as the moving average and dwell times in more detail.

Application World of Warcraft was extended with a small add-on which would provide the user with some basic feedback on the currently-observed level of relaxation. This feedback bar animates towards each newly observed relaxation level, instead of simply jumping to it. The point of this feature is to make the interface match user expectations better.

Aside from relaxation-level feedback, there is also feedback at command-level when the avatar is about to shape-shift. When the user crosses the low-relaxation threshold and is about to change to bear, the screen flashes red. When the dwell time is exceeded, the shape-shifting actually occurs. Similarly, when crossing the high-relaxation threshold and the user is about to change to human form, the screen flashes blue. The shape-shifting itself, and the game, are all part of the basic World of Warcraft application.

KEY POINTS

- BCIs and games are a good combination. BCIs can increase the sense of immersion, while games provide a motivating environment. Besides, games offer a large target audience, with eager early adopters.
- Human-computer interaction occurs in a loop. Changes in any of the interface processing steps are likely to affect the input from the user.
- Post-processing translates the initial interpretation of the user input (by a classifier, for example) into semantic control commands that make sense in the context of the application.
- Brain-computer interfaces try to do the equivalent of determining what happens on the field in a soccer game by standing outside the stadium listening to the cheers of the crowd. It is not mind-reading.

REFERENCES

[1] D. Plass-Oude Bos, B. Reuderink, B. L. A. van de Laar, H. Gürkök, C. Mühl, M. Poel, A. Nijholt, and D. K. J. Heylen. "Brain-Computer Interfacing and Games." In: *Brain-Computer Interfaces: Applying our Minds to Human-Computer Interaction*. Ed. by D. S. Tan and A. Nijholt. Springer, 2010. Chap. 10 (cit. on p. 9).

[2] F. Nijboer, B. Z. Allison, S. Dunne, D. Plass-Oude Bos, A. Nijholt, and P. Haselager. "A Preliminary Survey on the Perception of Marketability of Brain-Computer Interfaces and Initial Development of a Repository of BCI Companies." In: (2011). Ed. by G.R. Mueller-Putz, R. Sherer, M. Billinger, A. Kreilinger, V. Kaiser, and C. Neuper, pp. 344–347 (cit. on p. 9).

[3] A. Nijholt and D. S. Tan. "Playing with your brain: brain-computer interfaces and games." In: *Proceedings of the international conference on Advances in Computer Entertainment Technology*. ACM. 2007, pp. 305–306 (cit. on p. 9).

[4] A. Nijholt, D. Plass-Oude Bos, and B. Reuderink. "Turning shortcomings into challenges: Brain–computer interfaces for games." In: *Entertainment Computing* 1.2 (2009), pp. 85–94 (cit. on p. 9).

[5] G. Hakvoort, H. Gürkök, D. Plass-Oude Bos, M. Obbink, and M. Poel. "Measuring Immersion and Affect in a Brain-Computer Interface Game." In: *Human-Computer Interaction - INTERACT 2011*. Berlin/Heidelberg, Germany: Springer-Verlag, 2011, pp. 115–128 (cit. on p. 9).

[6] Q. Wang, O. Sourina, and M. K. Nguyen. "EEG-based 'serious' games design for medical applications." In: *Cyberworlds (CW) 2010, International Conference on*. IEEE. 2010, pp. 270–276 (cit. on p. 9).

[7] A. Lécuyer, F. Lotte, R. B. Reilly, R. Leeb, M. Hirose, and M. Slater. "Brain-computer interfaces, virtual reality, and videogames." In: *Computer* 41.10 (2008), pp. 66–72 (cit. on p. 9).

[8] A. Nijholt. "BCI for games: A 'state of the art' survey." In: *Entertainment Computing-ICEC 2008*. Springer, 2009, pp. 225–228 (cit. on p. 9).

[9] D. Marshall, D. Coyle, S. Wilson, and M. Callaghan. "Games, gameplay, and BCI: The state of the art." In: *Computational Intelligence and AI in Games, IEEE Transactions on* 5.2 (2013), pp. 82–99 (cit. on p. 9).

[10] Neurosky. *Brain Enhanced Gaming Concepts*. `http://download.neurosky.com/gameconcepts.pdf`. Last accessed: September 20, 2014. 2012 (cit. on p. 9).

[11] B. Graimann, B. Z. Allison, and A. Gräser. "New Applications for Non-invasive Brain-Computer Interfaces and the Need for Engaging Training Environments." In: *BRAINPLAY 07 Brain-Computer Interfaces and Games Workshop at ACE (Advances in Computer Entertainment)*. 2007, pp. 25–28 (cit. on p. 9).

[12] A. Chapanis. "Man-machine engineering." In: (1965) (cit. on p. 10).

[13] S.G. Mason and G.E. Birch. "A general framework for brain-computer interface design." In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 11.1 (2003), pp. 70–85 (cit. on p. 10).

[14] M. van Gerven, J. Farquhar, R. Schaefer, R. Vlek, J. Geuze, A. Nijholt, N. Ramsey, P. Haselager, L. Vuurpijl, S. Gielen, and P. Desain. "The brain-computer interface cycle." In: *Journal of Neural Engineering* 6.4 (2009) (cit. on p. 10).

[15] F. Nijboer, D. Plass-Oude Bos, Y. Blokland, R. van Wijk, and J. Farquhar. "Design requirements and potential target users for brain-computer interfaces – recommendations from rehabilitation professionals." In: *Brain-Computer Interfaces* 1.1 (2014), pp. 50–61 (cit. on p. 11).

[16] B. Z. Allison. "Toward ubiquitous BCIs." In: *Brain-Computer Interfaces*. Ed. by B. Graimann, G. Pfurtscheller, and B. Z. Allison. The Frontiers Collection. Springer, 2010, pp. 357–387 (cit. on p. 12).

[17] Wikipedia. *Device Driver*. `http://en.wikipedia.org/wiki/Device_driver`. Last accessed: February 21, 2014. (cit. on p. 12).

[18] F. A. C. Azevedo, L. R. B. Carvalho, L. T. Grinberg, J. M. Farfel, R. E. L. Ferretti, R. E. P. Leite, R. Lent, and S. Herculano-Houzel. "Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain." In: *Journal of Comparative Neurology* 513.5 (2009), pp. 532–541 (cit. on p. 13).

[19] A. L. Hodgkin and A. F. Huxley. "A quantitative description of membrane current and its application to conduction and excitation in nerve." In: *The Journal of physiology* 117.4 (1952), p. 500 (cit. on p. 13).

[20] J. Naish. *Enough*. Hodder & Stoughton, 2009 (cit. on p. 13).

[21] L. A. Farwell and E. Donchin. "Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials." In: *Electroencephalography and clinical Neurophysiology* 70.6 (1988), pp. 510–523 (cit. on p. 14).

[22] N. Birbaumer, N. Ghanayim, T. Hinterberger, I. Iversen, B. Kotchoubey, A. Kübler, J. Perelmouter, E. Taub, and H. Flor. "A spelling device for the paralysed." In: *Nature* 398 (1999), pp. 297–298 (cit. on p. 14).

[23] F. Nijboer, E. W. Sellers, J. Mellinger, M. A. Jordan, T. Matuz, A. Furdea, S. Halder, U. Mochty, D. J. Krusienski, T. M. Vaughan, J. R. Wolpaw, N. Birbaumer, and A. Kübler. “A P300-based brain-computer interface for people with amyotrophic lateral sclerosis.” In: *Clinical neurophysiology* 119.8 (2008), pp. 1909–1916 (cit. on p. 15).

[24] C. S. Herrmann. “Human EEG responses to 1–100 Hz flicker: resonance phenomena in visual cortex and their potential correlation to cognitive phenomena.” In: *Experimental brain research* 137.3-4 (2001), pp. 346–353 (cit. on p. 15).

[25] T. Hinterberger, S. Schmidt, N. Neumann, J. Mellinger, B. Blankertz, G. Curio, and N. Birbaumer. “Brain-computer communication and slow cortical potentials.” In: *Biomedical Engineering, IEEE Transactions on* 51.6 (2004), pp. 1011–1018 (cit. on p. 15).

[26] M. Jahanshahi and M. Hallett. “The Bereitschaftspotential: what does it measure and where does it come from?” In: *The Bereitschaftspotential*. Springer, 2003, pp. 1–17 (cit. on p. 15).

[27] A. A. Karim, T. Hinterberger, J. Richter, J. Mellinger, N. Neumann, H. Flor, A. Kübler, and N. Birbaumer. “Neural internet: Web surfing with brain potentials for the completely paralyzed.” In: *Neurorehabilitation and Neural Repair* 20.4 (2006), pp. 508–515 (cit. on p. 16).

[28] D. J. McFarland, L. A. Miner, T. M. Vaughan, and J. R. Wolpaw. “Mu and Beta Rhythm Topographies During Motor Imagery and Actual Movements.” In: *Brain Topography* 12.3 (Oct. 2000), pp. 117–186 (cit. on pp. 16, 18).

[29] C. Mühl, H. Gürkök, D. Plass-Oude Bos, M. E. Thurlings, L. Scherffig, M. Duvinage, A. A. Elbakyan, S. Kang, M. Poel, and D. K. J. Heylen. “Bacteria hunt.” In: *Journal on Multimodal User Interfaces* 4.1 (2010), pp. 11–25 (cit. on p. 18).

[30] R. J. Barry, A. R. Clarke, S. J. Johnstone, C. A. Magee, and J. A. Rushby. “EEG differences between eyes-closed and eyes-open resting conditions.” In: *Clinical Neurophysiology* 118.12 (2007), pp. 2765–2773 (cit. on p. 18).

[31] G. Pfurtscheller and F. H. Lopes da Silva. “Event-related EEG/MEG synchronization and desynchronization: basic principles.” In: *Clinical neurophysiology* 110.11 (1999), pp. 1842–1857 (cit. on p. 18).

[32] J. C. Shaw. “Intention as a component of the alpha-rhythm response to mental activity.” In: *International Journal of Psychophysiology* 24.1 (1996), pp. 7–23 (cit. on p. 18).

[33] B. L. A. van de Laar, B. Reuderink, D. Plass-Oude Bos, and D. K. J. Heylen. “Evaluating user experience of actual and imagined movement in BCI gaming.” In: *International Journal of Gaming and Computer-Mediated Simulations (IJGCMS)* 2.4 (2010), pp. 33–47 (cit. on p. 18).

[34] B. L. A. van de Laar, H. Gürkök, D. Plass-Oude Bos, M. Poel, and A. Nijholt. "Experiencing BCI control in a popular computer game." In: *IEEE transactions on computational intelligence and AI in games* 5.2 (2013), pp. 176–184 (cit. on p. 18).

[35] H. Gürkök, B. L. A. van de Laar, D. Plass-Oude Bos, M. Poel, and A. Nijholt. "Players' Opinions on Control and Playability of a BCI Game." In: *Universal Access in Human-Computer Interaction. Universal Access to Information and Knowledge - 8th International Conference, UAHCI 2014*. Springer, 2014, pp. 549–560 (cit. on p. 18).

3 WHAT USERS WANT

Current brain-computer interface research focuses on detection performance and bit rates. However, this is only one part of what is important to the user. Other aspects of usability and the user experience may be just as influential. In this chapter I apply the user-centred approach from human-computer interaction, first to discover new potential mental tasks which might be used as an input for brain-computer interfaces, and second to investigate what people consider important aspects of these mental tasks.

The research area of brain-computer interfaces (BCI) traditionally focused on providing fully-paralysed people with a new output channel to enable them to interact with the outside world, despite their handicap. As the technology improves, the question arises whether BCIs could also be beneficial for healthy users, for example by improving quality of life or by providing private, hands-free interaction [2, 3].

Current BCI research concentrates on improving the recognition accuracy and speed, which are two important usability factors. But in order for this technology to be accepted by the general public, other factors of usability and user experience have to be taken into account as well [4, 5, 6].

A couple of other BCI research labs are starting to take note of user-centred design methods as well, such as including an informed lead user throughout the development of a BCI system [7] and evaluating systems for their workload and user satisfaction [8].

In this chapter I apply user-centred development principles from the area of human-computer interaction to BCI design and evaluation. In Section 3.1 I use a user-centred design method to find new mental tasks for shape-shifting in the popular massively-multiplayer online role-playing game World of Warcraft® (developed by Blizzard Entertainment, Inc®). Within the large group of healthy users, gamers are an interesting target group. Fed by a hunger for novelty and challenges, gamers are often early adopters of new paradigms [9]. Besides that, it is suggested that users will be able to stay motivated and focused for longer periods if the BCI experiment can be presented in a game format [10].

This chapter has been presented orally at MMM2011 and as: D. Plass-Oude Bos, M. Poel, and A. Nijholt. "A study in user-centered design and evaluation of mental tasks for BCI." In: *Advances in Multimedia Modeling* (2011), pp. 122–134.

Afterwards, in Section 3.2 and 3.3 the use of the selected mental tasks are evaluated by users. The main research questions I address are: Which mental tasks do the users prefer, and why? How is this preference influenced by the detection performance of the system?

3.1 USER-CENTRED DESIGN

See Section 2.4 for more on these specific mental tasks.

One of the problems facing BCI research is the uncovering of usable mental tasks that trigger detectable brain activity. The tasks (by convention often indicated by the name of the corresponding brain activity) that are currently most popular are: slow cortical potentials, imaginary movement, P300, and steady-state visually-evoked potentials [11, 12]. Users regularly indicate that these tasks are either too slow, nonintuitive, cumbersome, or just annoying to use for control BCIs [13, 9].

Current commercial applications are a lot more complex and offer many more interaction possibilities than applications used in BCI research. Whereas current game controllers have over twelve dimensions of input, BCI games are generally limited to one or two-dimensional controls. Also, the mental tasks that are available are limited in their applicability for intuitive interaction. New mental tasks are needed that could be mapped in an intuitive manner with the system response[1].

One way to discover mental tasks that are suitable from a user perspective is to simply ask the user what they would like to do to trigger certain actions. In World of Warcraft®, the user can play an elf druid who can shape-shift into animal forms. As an elf, the player can cast spells to attack or to heal. When in bear form, the player can no longer use most spells, but is stronger and better protected against direct attacks, which is good for close combat.

In an open interview, I asked four World of Warcraft® players of varying expertise and ages which mental tasks they would prefer to use to shape-shift from the initial elf form to bear, and back again. The participants were not informed about the limits of current BCI systems, but most people did need an introduction to start thinking about tasks that would have a mental component. They were asked to think of using the action in the game, and

1 If 'intuitive' as a definition is reduced to 'familiar' [14], as in: "known from other previous interactions", then it would be easy to say it is impossible: our brains never interact with our environment directly, so how could we be familiar with it? Brain-computer interaction is not natural in the way that gesture-based or speech-based interaction can be, for example. On the other hand, as the brain is a key player in many of our interactions, there should definitely be suitable opportunities for familiar interaction. Perhaps the four themes of reality-based interaction can provide inspiration on how to minimize the gulfs of execution and evaluation [15, 16]. But as brain-computer interaction has no direct reality-based equivalent, we *may* need something else entirely, as Wigdor and Wixon recommend in their book on natural user interfaces: "Create an experience that is authentic to the medium — do not start by trying to mimic the real world or anything else."[17]. Section 8.2 describes a potential direction for future research.

what it meant to them, what it meant to their character in the game, what they thought about when doing it, what they thought when they wanted to use the action, and then to come up with mental tasks that would fit naturally with their gameplay. The ideas that the players came up with can be grouped into three categories. To be able to conduct the user evaluation (see Section 3.2), these three categories needed to be translated into concrete mental tasks. Each category consists of a task, and its reverse, to accommodate the shape-shifting action in the directions of both bear and elf form.

1. *Inner speech*: recite a mental spell to change into one form or the other. The texts of spells subsequently used were derived from expressions already used in the game world. The user had to mentally recite "I call upon the great bear spirit" to change to bear. "Let the balance be restored" was the expression used to change back to elf form.

2. *Association*: think about or feel like the form you want to become. Concretely, this means the user had to feel like a bear to change into a bear, and to feel like an elf to change into an elf.

3. *Mental state*: automatically change into a bear when the situation demands it. When you are attacked, the resulting stress could function as a trigger. For the next step of this research, this had to be translated into a task that the users could also perform consciously. To change to bear form the users had to make themselves feel stressed; to shift into elf form, relaxed.

3.2 USER EVALUATION METHODOLOGY

The goal of the user evaluation was to answer the following question in this game context: Which mental tasks do the users prefer, and why? As it was difficult to predict the influence of the probably not-very-good detection of these novel mental tasks, the participants were split up into two groups. One group would use the mental tasks with an actual BCI, however the detection performance would turn out to be. The other group would pretend to use a BCI with perfect recognition. This split allows further investigation into the effects of the detection performance on the user preference for the selected mental tasks.

PARTICIPANTS

Fourteen healthy participants (average age 27, ranging from 15 to 56; four female) voluntarily took part in this experiment after signing an informed consent form. All but one of the participants were right-handed. Highest finished education ranged from elementary school to a Master's degree. Experience with the application World of Warcraft® ranged from "I never play any games" to "I raid daily with my level 80 druid". Three participants were actively playing on a weekly basis.

CONDITIONS

The general methodology to answer the research questions was as follows. In order to measure the influence of the detection performance of the system the participants were divided in two groups, a so-called "real-BCI" and "utopia-BCI" group. The group that played World of Warcraft® with "utopia-BCI" decided for themselves whether they performed the mental task correctly, and pressed the button to shape-shift when they had. In this way a BCI system with 100% detection performance (a utopia) was simulated[2]. The group that played World of Warcraft® with "real-BCI" actually controlled their shape-shifting action with their mental tasks, at least insofar as the system could detect it.

The participants came in for experiments once a week for five weeks, in order to track potential changes over time. During an experiment, for each pair of mental tasks, the participant underwent a training and game session and filled in questionnaires to evaluate the user experience. The three different categories of mental tasks were tested in random order. Each category consisted of two mental tasks: one to change to bear, and another to change to elf.

To summarize: the measurements were within-subject for the different mental task categories, but between-subject for the "utopia-BCI" and "real-BCI" conditions. The following sections explain each part in more detail.

WEEKLY SESSIONS AND MEASUREMENTS

The participants participated in five experiments, lasting about two hours each, over five weeks. The mental tasks, mentioned above, were evaluated in random order to eliminate any potential order effects, for example, due to fatigue or user learning.

For each task pair, the participant underwent a training session. The purpose of the training session was manifold: (1) it gathered clean data to evaluate the recognizability of the brain activity related to the mental tasks, (2) the user was trained in performing the mental tasks, (3) the system was

---

2 Interesting side note: although 100% detection performance might be optimal from a functional point of view, it does not always create the optimal user experience. See, for example, Laar et al. [18].

trained for those participants who played the game with the real BCI system, and (4) the user experience could be evaluated outside the game context.

A training session consisted of two sets of trials separated by a break to allow the participant to rest. Each set started with four watch-only trials (two per mental task), followed by 24 do-task trials (twelve per mental task). The trial sequence (see Figure 8) involved five seconds watching the character in their start form, followed by two seconds during which the shape-shifting task was presented. After this the participant had ten seconds to perform the mental task repeatedly until the time was up, or just watch if it was a watch-only trial. At the end of these ten seconds, the participant saw the character transform. The character in the videos was viewed from the back, similar to the way the participant would see the avatar in the game.

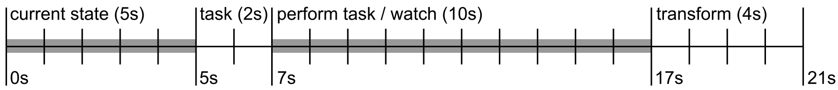


FIGURE 8: **Training session trial sequence:** first the character is shown in their start form, then the task is presented, after which there is a period during which this task can be performed. At the end the animation for the shape-shift is shown.

During the watch-only trials, the participant saw exactly what they would see during the do-task trials, but they were asked only to watch the sequence. The EEG data from these trials were used as a baseline.

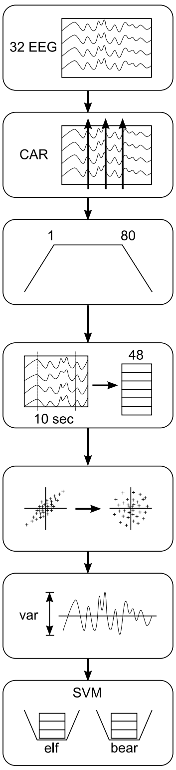
At the end of the training session, the participant was asked to fill out forms to evaluate the user experience. The user experience questionnaire was loosely based on the Game Experience Questionnaire [19]. It contained statements on which the participants had to indicate their amount of agreement on a five-point Likert scale, for example: “I could perform these mental tasks easily”, “It was tiring to use these mental tasks”, and “It was fun to use these mental tasks”. The statements were categorized into the following groups: whether the task was easy, doable, fun, intuitive, tiring to execute, and whether the mapping to the in-game action made sense or not.

After the training session, the participant had roughly eight minutes to try the set of mental tasks in the game environment. The first experiment consisted only of a training session. For weeks two to four, the participants were split up into the “real-BCI” and “utopia-BCI” group. Groups were fixed for the total experiment. In the last week all participants followed a training and played the game with the real BCI system.

The “real-BCI” group received feedback on the recognition of their mental tasks in the form of an orange bar in the game (see Figure 9).

The smaller the bar, the more the system had detected the mental task related to elf form. The larger, the more the system had interpreted the

FIGURE 9: Orange feedback bar with thresholds. The user has to go below 0.3 to change to elf, and above 0.7 to change to bear. In between no action is performed.



The BCI processing steps of this system.

Section 2.3 explains EEG, and Section 2.2 discusses common analysis steps in BCIs.

brain activity as related to bear form. When the thresholds were crossed the shape-shift action was executed automatically.

The “utopia-BCI” participants had to interact with a BCI system with (a near) 100% performance. Unfortunately this was not technically feasible. One option was to use the Wizard of Oz technique, where a human plays the role of the detection system [20]. This ‘wizard’ observes the user, and presses the right buttons when the user task is performed correctly. With *mental* tasks, however, the wizard would have no way of knowing what the user is doing as there is no external expression of the task. The only option left to simulate a perfect system is to let the participants evaluate themselves whether or not they had performed the task correctly. Then they pressed the shape-shift button in the game manually.

At the end of the game session, the user experience questionnaire was repeated, to determine potential differences between the training and game sessions. The game session questionnaire contained an extra question to determine the perceived detection performance of the mental tasks.

At the end of the total experiment for the week, the participants filled out a final form concerning the experiment as a whole. The participants were asked to put the mental tasks in order of preference, and to indicate why they choose this particular ordering.

EEG ANALYSIS

The EEG analysis pipeline, programmed in Python, was kept very general, as there was no certainty about how to detect the selected mental tasks. Common Average Reference was used as a spatial filter, in order to improve the signal-to-noise ratio [21]. The bandpass filter was set to a very wide range of 1–80Hz. The data gathered during the training session was sliced in 10-second windows. These samples were then whitened [22], and the variance of each component was computed as an indication of the power in the window. A support vector machines (SVM) classifier provided different weights for the power of each EEG component. The data from the training session was used to determine the whitening matrix and train the SVM.

MAPPING

The BCI control was made more robust to artefacts with two post-processing methods. These methods affect the mapping of classification results to in-game actions. A short dwelling was required to trigger the shape-shift, so it would not be activated by quick peaks in power. Secondly, hysteresis was applied: the use of two thresholds in order to reduce the effects of quick oscillations. The threshold that needed to be crossed to change into a bear was higher than the threshold required to revert back to elf form, see Figure 9. In between these two thresholds, the avatar remained in the most-recent form. Hysteresis has a stabilizing effect which is also known as debouncing. For an overview of various post-processing methods, see Chapter 6.

3.3 RESULTS

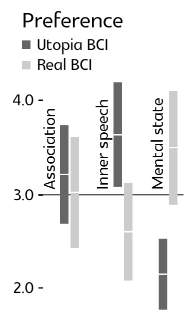
MENTAL TASK PREFERENCE

In the post-experiment questionnaire, the participants were asked to list the mental tasks in order of preference. The place in this list was used as a preference score, which was then rescaled and inverted to match the user experience questionnaire values. As a result, the preference values range from 1 to 5, where 5 indicates first choice, 1 is the least preferable task, and 3.0 is interpreted as a neutral disposition.

Sixty-nine measurements were obtained from 14 participants over five weeks. One week one participant had to leave early and could not fill out his preference questionnaire.

On average, there was a general preference for the association tasks, and the mental state seemed to be disliked the most. But this paints a very simplistic image, as there are large differences between the “real-BCI” and “utopia-BCI” groups.

For a better understanding of the effects of the different aspects, Figure 10 shows the preference and user experience scores for each of the three mental task pairs, separated for the two participant groups. Whereas for the “real-BCI” group the mental state tasks are most liked, the opposite is true for the “Utopia BCI” group, where they are most disliked. Similarly, The “utopia-BCI” group most preferred inner speech, which was least preferred by the “real-BCI” group. Because of these large differences, these two groups need to be investigated as two separate conditions.



There were large differences between the inner speech and mental state preference scores for the Utopia and Real BCI conditions.

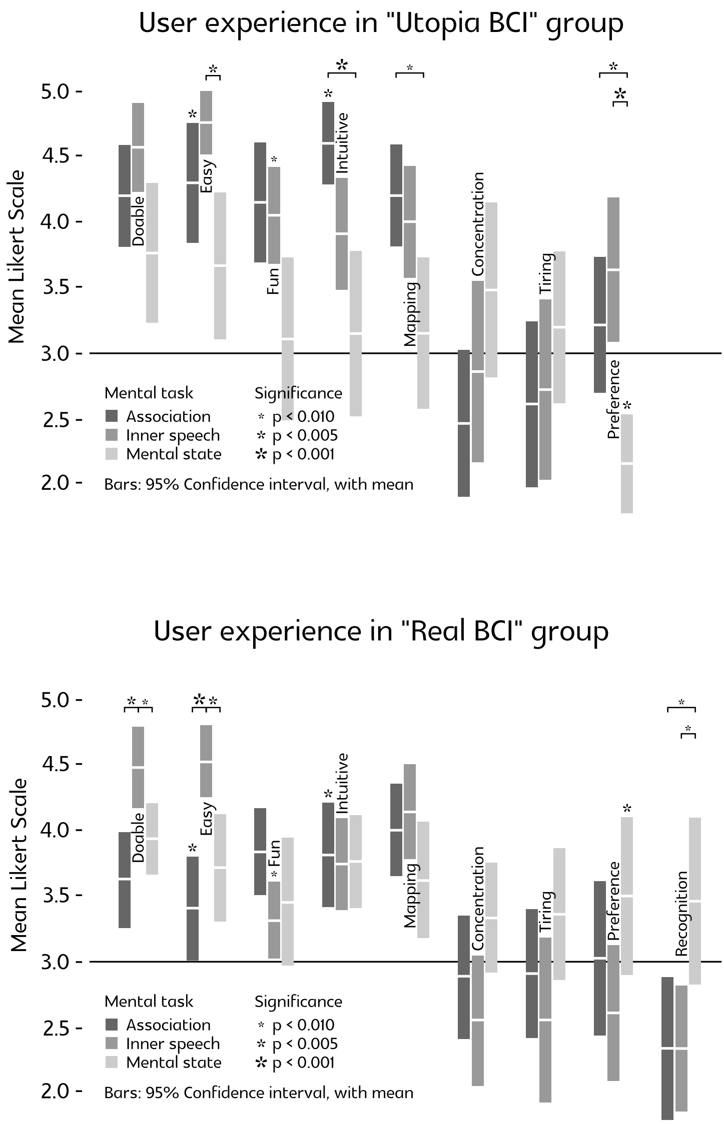


FIGURE 10: **User experience, preference, and perceived performance scores** for the "utopia-BCI" and "real-BCI" groups, separate for the three mental task pairs, averaged for weeks 2 to 4. The plot shows the means with 95% confidence intervals, and is annotated with significant differences between task pairs (association, inner speech, mental state; with a line with star above the two pairs). Significant differences between conditions ("utopia-BCI", "real-BCI") are indicated with a star above the corresponding bars in both the top and bottom plots . The black horizontal line at a Likert scale value of 3.0 indicates the neutral position.

TASK RECOGNITION AND TASK PREFERENCE

It is not possible to completely separate the influence of the recognition performance from other aspects that differ between the participant groups. But based on the user experience scores, recognition perception scores, and the words the participants used to describe their reasoning for their preference, it is possible to explain the discrepancy in preference between the two conditions and get an idea of the influence of recognition performance.

Although the *inner speech* tasks were rated highly positive (doable, easy, intuitive) by both groups, the task recognition by the system had a big impact. Despite the positive ratings overall, it is the least-preferred task pair for the “real-BCI” participants. The *association* tasks are valued mostly for their intuitiveness and the mapping with the in-game task, by the “utopia-BCI” group at least. This time, not only the preference scores, but aspects of the user experience (ease and intuitiveness) are significantly affected by the recognition as well. So far, bad task recognition resulted in low preference scores. For *mental state* we see that good recognition can also do the opposite. This task pair scored low across the board by both groups, yet it was preferred by the “real-BCI” group.

Based on these results, it seems that the recognition performance has a very strong influence on the user preference, as it is the most important consideration for the “real-BCI” group task preference. For the “utopia-BCI” group different considerations emerge, where the ease of execution seems to play a dominant role.

This view is confirmed by the words the participants used to describe their preference ranking, shown in Figure 11. Through the method of content analysis, the words were categorized, and the number of occurrences within each category was used as an indication of how important that category was to the reasoning [23]. To reduce the number of categories, words that indicated a similar concept were clustered into one category. For example, *difficult* was combined with *easy*. Where applicable, the key terms from the user experience questionnaire were used to indicate the word categories.

The “real-BCI” group mostly used the word *recognition* performance ($n = 15$), more than twice as often as any other word category ($n <= 7$). The “Utopia BCI” group mostly referred to the *ease* of executing the task ($n = 12$, where $n <= 5$ for the other word categories). Other issues that were often mentioned were *feels good*, *concentration*, *fun* and *tiring*.

On a side note, the participants appeared to be optimistic about the system recognition. A score of 1 on the Likert scale would have indicated no recognition at all, and even though the recognition for inner speech, for example, would have been minimal, the average rating is still above 2 (see Figure 10). The participants did feel that the system recognized the mental state task pair, and this perceived recognition increased over time.

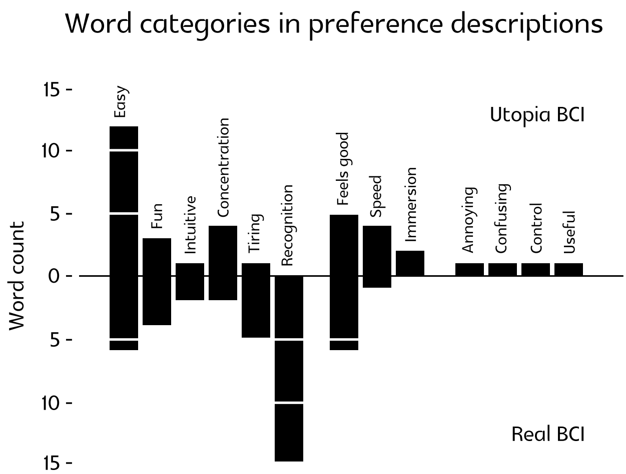


FIGURE 11: **Counts for the categories of words used by the participants to describe the reasoning behind their preference ranking**, total for weeks 2 to 4. The bars for the "Utopia BCI" are on top; "Real BCI" is below. Categories related to the questionnaire are grouped on the left. In the middle are new categories that were not known from the questionnaire, but still mentioned multiple times. On the right are new categories that were only mentioned once.

	doable	easy	fun	intuit	mapp	concnt	tiring	recgn
Utopia r	**-,482**	**-,542**	**-,478**	**-,384**	**-,395**	,273	,254	
p	,000	,000	,000	,002	,002	,032	,047	
Real r	-,218	-,293	-,294	,057	,079	,246	,137	-,316
p	,095	,023	,023	,665	,550	,058	,295	,014
All r	**-,360**	**-,415**	**-,391**	-,203	-,190	**,259**	,199	-,316
p	,000	,000	,000	,025	,036	,004	,028	,014

TABLE 1: **Pearson correlation coefficients and p-values for the correlation of user experience components and perceived recognition rate with the preference scores** for the mental task pairs. Significance annotation: $p <= 0,005$ in bold.

TASK PREFERENCE AND USER EXPERIENCE

Given the fact that participants indicated task *recognition* and *ease* to be the most important considerations for their preference, do these aspects from the user experience questionnaire also show a correlation to the preference scores?

Weeks 1 and 5 were excluded from this analysis as both groups performed the tasks in the same conditions in these weeks ("Utopia" in week 1 and "Real" in week 5). Therefore the number of samples for the correlation tests are 63 (3 weeks, 3 task pairs, 7 participants), except for one case where there are some missing samples due to a participant having to leave early. For the correlation tests with the two conditions ("Real" and "Utopia") combined, there are twice the number of samples. There are no perceived recognition scores for the "utopia-BCI" group.

For the "utopia-BCI" group, the expected correlations with the ease-related aspects *doable* and *easy* were found, as well as correlations with most of the other aspects. After correction for multiple tests, the correlations with *concentration* and *tiring* are not significant.

There were no significant correlations of preference with any of the user experience aspects for the "real-BCI" group, but the most relevant correlation was with the perceived *recognition* performance by the system, which is as expected. Looking at the conditions combined, the most significant correlations are for *doable*, *easy*, and *fun*.

The relation between preference and recognition performance is not that apparent when investigating this correlation. Yet, the correlations do show the importance of easy task execution, expressed in how *easy* and how *doable* it is to perform the task. They also show that other aspects can be important as well, such as *fun*, how *intuitive* it is, and the *mapping* to the in-game action.

3.4 DISCUSSION AND CONCLUSIONS

When evaluating BCIs, current research focuses strongly on task recognition performance, speed, and the derivative: bit rates. Human-computer interaction research shows that for a user to accept and value this new means of interaction other aspects may be important as well, generally summarized as "usability and user experience".

Historically, mental tasks for BCI control are based on neuromechanisms discovered in neuroscientific research, and are selected for their discriminability, to ensure that a BCI can recognize the task being performed based on the user's brain activity. For this research, I involved potential users in the design process of determining which mental tasks to use for certain actions within the application. This resulted in three categories of mental tasks that are not listed among the most frequently-used tasks in current BCI applications: *mental state*, *inner speech*, and *association*. Such a user-centred approach allows us to go beyond the task recognition criterion and look at other aspects which might also be important to users.

These three pairs of mental tasks were then evaluated in a prototype for their resulting user experience. Asking the participants about their experience yielded new insights into what potential users liked and disliked about these particular mental tasks for this BCI system, and why.

In the context of this experiment, the task recognition by the system was so important that its contribution to the users' mental task preference overshadowed all other mental task characteristics for the participants who played with actual BCI detection. Only when the influence of this aspect is removed, by having participants pretend they are using a perfect BCI, do other aspects come to the fore.

Overall, user preference for mental tasks seems to be based on (highest influence first) accuracy of task *recognition* by the system, *ease* of performing the mental task, and lastly by factors such as *fun*, *intuitiveness*, and *suitability* for the task. This is confirmed by the words participants used to describe their preference as well as correlations with items from the user experience questionnaire filled out by the participants.

LIMITATIONS

It is important to keep in mind that these findings come from a limited context: the shape-shifting task in World of Warcraft®. Whether they generalize to other contexts requires further investigation. It is also not possible to see task preference independent of the in-game action as they are inherently linked.

The participants in the "utopia-BCI" group had to evaluate their execution of the mental tasks themselves. How well people can evaluate their own mental task execution, and how this may affect their experience is unknown. There were behavioural indications that the participants did perform their mental tasks seriously. As an additional motivator, their brain activity was recorded, just as with the "real-BCI" participants. These recordings cannot be used to validate task execution as the execution and resulting brain response may be different for the various participants. Besides, it is unsure whether there are clear differences between the brain signals for these task pairs to begin with[3].

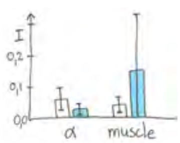
The words in the descriptions of the participants describing the reasons for their preference have been grouped with the questionnaire categories as a guideline where applicable. Other categories could have been possible. For example, the ease of executing a mental task is actually composed of various aspects. Merriam-Webster defines *ease* as "the state of being comfortable", and provides a number of reasons for this experience, such as "lack of difficulty", "effortlessness", and "naturalness" [24]. In other words, a mental task can be easy to do because you understand what you are expected to do (lack of difficulty), because of familiarity (naturalness), or because it requires little concentration to execute (effort). These reasons are addressed in the user experience questionnaire in slightly different words, such as "intuitive" and "tiring".

RECOMMENDATIONS

The fact that the recognition by the system was indicated to be so important to the participants in this experiment seems to validate the current focus of BCI research on speed and accuracy. The importance of good control had already been well established for disabled end-users (for example in [25]), but is now also confirmed for healthy users.

Over time, speed and accuracy of detection can be improved with better hardware and analysis methods. As a result, other aspects of user experience will become more prominent, such as the ease of performing the mental task. In the meantime, it is important to be aware of this distortion of various user experience measurements as a result of the lack of accurate task recognition. Faking well-functioning BCI systems could allow us to see what else we should consider when developing such systems.

When designing and evaluating systems it would be beneficial to take the usability and user experience into account, for patients as well as healthy users. For more information on evaluating the user experience in BCI systems, and some examples of how the user experience can affect BCI performance and vice versa, see van de Laar et al. [26].



The mental state task pair was somewhat discriminable in terms of mutual information, but only for the real BCI group (coloured) and only based on muscle activity, not the brain.

3 I did conduct an unpublished analysis for the mental state task pair, which according to the participants was best recognized. While very limited, there seemed to be some information in there, but only for the real BCI group and only in the higher frequencies where we mainly observe muscle activity, not brain activity.

As this experiment has been conducted with healthy, able-bodied participants, it would be interesting to conduct a similar study with users of assistive technology (AT). I would expect the results to be quite similar: good task recognition and easy task execution are likely to be the two most important aspects for that user group as well. Depending on the underlying reason for the need for AT, sustaining concentration for a longer period of time might be problematic. While the general population requires these characteristics because otherwise they will simply use another interface, in the case of AT users, good task recognition and easy task execution might be a requirement for them to be able to use the interface at all.

The participants appeared to be optimistic about the task recognition by the system. Chapter 4 will describe the investigation into this perception of control.

KEY POINTS

- The field of human-computer interaction has knowledge and methods to offer for developing systems that are accepted and valued by users, which are valuable for brain-computer interfaces as well.
- User-centred design offers a new approach for trying to find new mental tasks for BCI control, which allows us to look beyond recognition performance.
- Task recognition by the system appears to be the most important factor for mental task preference by users. However, the ease of task execution for the user is also highly influential.
- Perceived task recognition performance can significantly affect other user experience measurements. As long as the task recognition is at a critical level, it may be necessary to fake good recognition in order to be able to investigate other aspects of the user experience.
- Participants appeared to be on the optimistic side about the task recognition by the system.

REFERENCES

[1] D. Plass-Oude Bos, M. Poel, and A. Nijholt. "A study in user-centered design and evaluation of mental tasks for BCI." In: *Advances in Multimedia Modeling* (2011), pp. 122–134 (cit. on p. 29).

[2] A. Nijholt, J. van Erp, and D. K. J. Heylen. "BrainGain: BCI for HCI and Games." In: *Proceedings of the AISB Symposium Brain Computer Interfaces and Human Computer Interaction: A Convergence of Ideas*. 2008, pp. 32–35 (cit. on p. 29).

[3] J. R. Wolpaw, G. E. Loeb, B. Z. Allison, E. Donchin, O. F. do Nascimento, W. J. Heetderks, F. Nijboer, W. G. Shain, and J. N. Turner. "BCI meeting 2005 – Workshop on signals and recording methods." In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 14.2 (2006), pp. 138–141 (cit. on p. 29).

[4] D. Plass-Oude Bos, B. Reuderink, B. L. A. van de Laar, H. Gürkök, C. Mühl, M. Poel, A. Nijholt, and D. K. J. Heylen. "Brain-Computer Interfacing and Games." In: *Brain-Computer Interfaces: Applying our Minds to Human-Computer Interaction*. Ed. by D. S. Tan and A. Nijholt. Springer, 2010. Chap. 10 (cit. on p. 29).

[5] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan. "Brain-computer interfaces for communication and control." In: *Clinical neurophysiology* 113.6 (2002), pp. 767–791 (cit. on p. 29).

[6] B. L. A. van de Laar, H. Gürkök, D. Plass-Oude Bos, F. Nijboer, and A. Nijholt. "Perspectives on user experience evaluation of brain-computer interfaces." In: *Universal Access in Human-Computer Interaction. Users Diversity*. Springer, 2011, pp. 600–609 (cit. on p. 29).

[7] G. Lightbody, M. Ware, P. McCullagh, M. D. Mulvenna, E. Thomson, S. Martin, D. Todd, V. C. Medina, and S. C. Martinez. "A user centred approach for developing brain-computer interfaces." In: *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2010 4th International Conference on-NO PERMISSIONS*. IEEE. 2010, pp. 1–8 (cit. on p. 29).

[8] A. Kübler, E. Holz, T. Kaufmann, and C. Zickler. "A user centred approach for bringing BCI controlled applications to end-users." In: *Brain-Computer Interface Systems-Recent Progress and Future Prospects* (2013) (cit. on p. 29).

[9] A. Nijholt, D. S. Tan, G. Pfurtscheller, C. Brunner, J. del R. Millán, B. Z. Allison, B. Graimann, F. Popescu, B. Blankertz, and K.-R. Müller. "Brain-computer interfacing for intelligent systems." In: *Intelligent Systems, IEEE* 23.3 (2008), pp. 72–79 (cit. on pp. 29, 30).

[10] B. Graimann, B. Z. Allison, and A. Gräser. "New Applications for Non-invasive Brain-Computer Interfaces and the Need for Engaging Training Environments." In: *BRAINPLAY 07 Brain-Computer Interfaces and Games Workshop at ACE (Advances in Computer Entertainment)*. 2007, pp. 25–28 (cit. on p. 29).

[11] B. Reuderink. *Games and Brain-Computer Interfaces: The State of the Art*. Tech. rep. TR-CTIT-08-81. Human Media Interaction, Faculty of EEMCS, University of Twente, 2008 (cit. on p. 30).

[12] O. Tonet, M. Marinelli, L. Citi, P. M. Rossini, L. Rossini, G. Megali, and P. Dario. "Defining brain-machine interface applications by matching interface performance with device requirements." In: *Journal of Neuroscience Methods* 167.1 (2008), pp. 91–104 (cit. on p. 30).

[13] G. G. Molina. "Detection of High-Frequency Steady State Visual Evoked Potentials Using Phase Rectified Reconstruction." In: *16th European Signal Processing Conference EUSIPCO 2008*. 2008 (cit. on p. 30).

[14] J. Raskin. "Viewpoint: Intuitive equals familiar." In: *Commun. ACM* 37.9 (1994), pp. 17–18 (cit. on p. 30).

[15] R. J. K. Jacob, A. Girouard, L. M. Hirshfield, M. S. Horn, O. Shaer, E. T. Solovey, and J. Zigelbaum. "Reality-based interaction: a framework for post-WIMP interfaces." In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM. 2008, pp. 201–210 (cit. on p. 30).

[16] D. A. Norman. *The design of everyday things*. Basic books, 2002 (cit. on p. 30).

[17] D. Wigdor and D. Wixon. *Brave NUI world: designing natural user interfaces for touch and gesture*. Elsevier, 2011 (cit. on p. 30).

[18] B. L. A. van de Laar, D. Plass-Oude Bos, B. Reuderink, M. Poel, and A. Nijholt. "How Much Control Is Enough? Influence of Unreliable Input on User Experience." In: *Cybernetics, IEEE Transactions on* 43.6 (Dec. 2013), pp. 1584–1592 (cit. on p. 32).

[19] W. A. IJsselsteijn, Y. A. W. de Kort, K. Poels, A. Jurgelionis, and F. Bellotti. "Characterising and measuring user experiences in digital games." In: *International Conference on Advances in Computer Entertainment Technology*. 2007 (cit. on p. 33).

[20] D. Salber and J. Coutaz. "Applying the Wizard of Oz technique to the study of multimodal systems." In: *Human-Computer Interaction* 753 (1993), pp. 219–230 (cit. on p. 34).

[21] D. J. McFarland, L. M. McCane, S. V. David, and J. R. Wolpaw. "Spatial filter selection for EEG-based communication." In: *Electroencephalography and clinical Neurophysiology* 103.3 (1997), pp. 386–394 (cit. on p. 34).

[22] G. Dornhege, B. Blankertz, M. Krauledat, F. Losch, G. Curio, and K.-R. Müller. "Combined optimization of spatial and temporal filters for improving brain-computer interfacing." In: *Biomedical Engineering, IEEE Transactions on* 53.11 (2006), pp. 2274–2281 (cit. on p. 34).

[23] S. Stemler. "An overview of content analysis." In: *Practical assessment, research & evaluation* 7.17 (2001), pp. 137–146 (cit. on p. 37).

[24] Merriam-Webster. *Ease*. `http://www.merriam-webster.com/dictionary/ease`. Last accessed: February 21, 2014. (cit. on p. 41).

[25] C. Zickler, A. Riccio, F. Leotta, S. Hillian-Tress, S. Halder, E. Holz, P. Staiger-Sälzer, E.-J. Hoogerwerf, L. Desideri, D. Mattia, and A. Kübler. "A brain-computer interface as input channel for a standard assistive technology software." In: *Clinical EEG and Neuroscience* 42.4 (2011), pp. 236–244 (cit. on p. 41).

[26] B. L. A. van de Laar, H. Gürkök, D. Plass-Oude Bos, F. Nijboer, and A. Nijholt. "Brain-Computer Interfaces and User Experience Evaluation." In: *Towards Practical Brain-Computer Interfaces*. Ed. by B. Z. Allison, S. Dunne, R. Leeb, J. del R. Millán, and A. Nijholt. Springer, 2012, pp. 223–237 (cit. on p. 41).

4 PERCEPTION OF CONTROL

Brain-computer interfaces do not provide perfect recognition of user input (Chapters 1 and 2). However, good task recognition is precisely what users want most (Chapter 3). How well can users really assess their level of control? And how much control do they need? In this chapter we investigate the relation between actual and perceived control.

Like other input modalities based on observations of the body, BCIs do not provide perfect recognition of what the user is trying to convey. This can be problematic, as input is the basis for usable systems in general. In the previous chapter we saw that good recognition was the most important input task characteristic for BCI users.

See Chapters 1 and 2.

See Chapter 3.

Most of the studies in this thesis are focused on game applications specifically. Gamers are a large target audience, and many of them are early adopters [2]. Learning to provide brain-based input can be integrated into the game as part of the challenge [3]. Games can also help experiment participants to stay motivated and focused for longer periods [4].

See Chapter 2.

We have done many demonstrations and experiments in which people could try our brain-computer interface (BCI) games. Sometimes people seem to overestimate their level of control, and sometimes to underestimate it. This made us wonder: how well can people assess how much control they really have? Additionally, what would be the minimum amount of control necessary to operate a given system? In this study, this second question has been limited to: What is the minimum amount of control necessary so people do not give up playing the game.

Previous analysis of data from this experiment has been published in [5], which posed that perfect control may not always result in an optimal user experience. People actually experienced more fun in the experiment game when the control was not perfect. My focus *here* is the *perception* of control. Additionally, I investigate how much control might be necessary so users do not give up.

This chapter has been published as: D. Plass-Oude Bos et al. “Perception and manipulation of game control.” In: *Proceedings 6th International Conference on Intelligent Technologies for Interactive Entertainment (INTETAIN 2014)*. Ed. by D. Reidsma. Vol. 136. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer Verlag, Berlin, 2014, pp. 57–66. It was presented orally at INTETAIN 2014, where it received the best paper award.

4.1 BACKGROUND AND RELATED WORK

Perception of control There are many reasons to suspect that there is no simple linear relationship between the perception of control and how much control people actually have. People overestimate their influence on things with a positive outcome, and underestimate their effect on negative outcomes [6, 7]. Even when people have no control at all, they may experience 'the illusion of control' [8]. Another factor affecting people's conscious perception is how they can rationalize their unconscious behaviours [9].

Additionally, people assess more beautiful systems as being more usable, even if they are not [10]. Norman takes this a step further, arguing that pleasing things (not necessarily through beauty alone) actually work better, as the user is more open to the interaction [11]. One pleasing aspect might be using a novel input modality [12]. People appear to be more lenient towards mistakes made by a brain-computer interface than towards errors made with a non-BCI input (in this case, a variation on mouse selection) [12]. For more on how the user experience may affect BCI performance, see [13].

Controlled simulation of uncertain control Brain-computer interfaces would not allow us to control the amount of user control over the full range from having no control at all to full control. So we needed a substitute where we could be certain that the user has the level of control that we wanted to provide. We looked at various alternatives: simulating imaginary-movement-based BCI input [14], manipulating mouse input [15], or issuing incorrect commands at selection level [16].

Carlson, et al. evaluated the effect of shared control (with an AI) on driving a BCI-controlled wheelchair [17] using another alternative. To make the evaluation less time-consuming, they decided not to use actual BCI control, but to simulate it with input transformation matrices. Such a matrix provides a probability for each input to transform into something else, which can then be used to actually transform input actions into other actions with different system responses. Then an 'expert driver' would *pretend* to control the wheelchair with a brain-computer interface, by simply pressing buttons on a keyboard. In the end, we opted for these *input transformation matrices*. Its simplicity allows us to assess only the control aspect of the input. Additionally, it is easier to implement and adjust.

Minimum amount of control It has been said that for BCIs a selection accuracy of 70% is acceptable [18]. Another BCI research group determined the minimum acceptable level of accuracy to be 77% for four input classes by lowering the selection accuracy step-by-step until the participant indicated meaningful navigation in the application menu had become impossible [16].

Uncertainty in applications There can be uncertainty about whether the input will be interpreted correcty by the input device, but uncertainty can also be purposefully introduced in an application [19]. For example, we observed that uncertainty can increase the sense of fun [5].

4.2 METHODS

EXPERIMENT PROTOCOL

To reach a large number of participants and gather enough data for each level of control, the experiment was run from a web browser, so people could participate from anywhere. Social media was actively used to recruit participants.

The input was provided by keyboard. To manipulate the amount of control, an input transformation matrix would be randomly selected from the database for each run. This randomization had two benefits: (1) It allowed for a distribution of samples over the different levels of control, and (2) if the previous experienced level of control affects the experience of the current level, this avoids order effects in the results.

Each run started with an explanation on how to play the game. During a run, the player tried to guide a laboratory test hamster to freedom through four levels (Figure 12)[1]. After one minute, the player could decide to skip the rest of the run. At the end of each run, a questionnaire would pop up, after which the player would be encouraged to play another round.

MANIPULATION OF THE INPUT

In this game, the amount of control is varied by manipulating the keyboard input, which consists of the four directional arrows. If the user takes no action, a 'no action' input is generated, also known as 'idle state', 'no control', or 'no operation'.

This input is transformed by a matrix which dictates probabilities for each system response (what the hamster you control ends up doing) given a particular provided input. The probabilities for a non-matching output are set equal for all non-matching outputs. The amount of control is thus defined by the probability for correct classification (the hamster obeys), which is equal to the accuracy level.

We generated 15 input transformation matrices, evenly spread out over the whole possible range of *mutual information*. Mutual information is de-

1 The software and additional notes can be found on the following website: `www.dannyplass.nl/control`

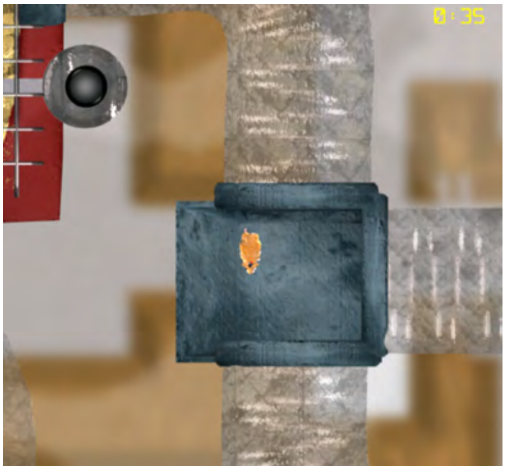


FIGURE 12: **A screenshot of the game** used in the experiment.

fined as the amount of information one sequence provides over another, in bits[2]. In this case, the amount of information that is being shared between the actual input, and the transformed input with a lower accuracy. This measure may be a little less intuitive to interpret compared to accuracy, but it is more suitable, as it is comparable for different numbers of classes and different prior probabilities [20]. The relationship between mutual information and accuracy is logarithmic.

To determine the mutual information $I(X;Y)$ for a given input transformation matrix concerning inputs X and responses Y, we assume a uniform probability function over the input space (verification in the Results section). This probability function is also known as the marginal probability function $p(x)$ for x. The mutual information is then computed as follows:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log_2 \frac{p(x,y)}{p(x)p(y)}$$

$$\text{where } p(x) = \frac{1}{n}, \text{ with } n \text{ the number of input types}$$

2 Say there is no relation between the actual input and the transformed input, then the mutual information is 0. Knowing one does not provide any information about the other. If the actual input completely determines the transformed input (there is a direct translation), then the amount of information in the actual input is the same as the amount of information in the transformed input, and this is the same as the amount of information both sequences provide over each other: the entropy of the input sequence. The entropy is the average amount of information based on the number of different tokens in the sequence and their respective probability distributions.

DATA COLLECTION

The questionnaire consisted of 6 visual analogue scale (VAS) questions [21], and 3 open questions: age, gender, and a field for remarks. The VAS scales went from 0 to 100, and were initialized at 50. For this analysis, the two questions of importance are those related to control: *"I had the feeling that the hamster did what I wanted it to do"* and *"I had the feeling the computer was following my commands"*. Additionally, detailed action logs were maintained for each participant, containing all keyboard inputs and the resulting transformed actions, as well as starting, skipping, finishing, pausing, and resuming levels.

PARTICIPANTS

We could identify 87 individuals based on filled out gender-age answer pairs in combination with the IP addresses. Of these individuals 39% was male, 29% female, and 28% unknown, with an average (provided) age of 24.9 years (in the range of 10-58, with a standard deviation of 7.5).

For the main analysis concerning the perception of control, we excluded runs for which not all the VAS questions were filled out. This filtering resulted in 211 runs, with at least 9 runs and at most 22 runs per level of control.

To determine when people gave up, we looked at the action logs for all started runs, and analysed the final entries for each run, which gives an indication of how it was ended. This resulted in the analysis of 465 runs.

4.3 RESULTS

The two main questions are: (1) How well can people assess their level of control, and (2) How much control is sufficient so users do not give up?

FROM THEORY TO REALITY

The input transformation matrices were computed based on an equal occurrence for each input. In practice, there was indeed a fairly equal distribution among the classes (medians around 20%), but with a preference for 'right' (about 30%, due to level design), and a lower occurrence for 'no action' (around 10%).

How does this affect the amount of control people had? Based on the confusion matrix of observed inputs and into which system responses they were transformed, we computed the *observed mutual information*. The theoretical and observed mutual information are tightly correlated, see Figure 13.

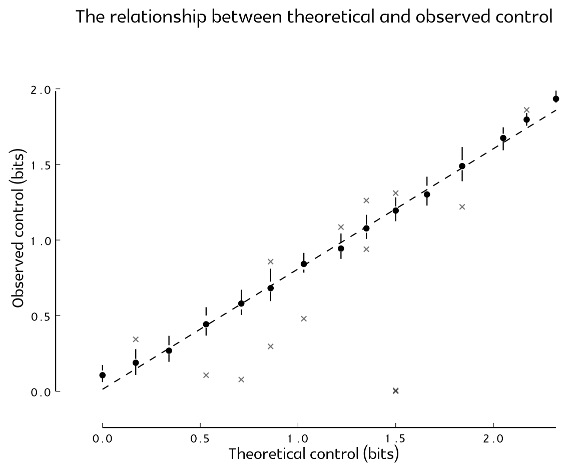


FIGURE 13: **The theoretical amount of control shows a tight linear relationship with the observed amount of control.** *How to interpret Tufte box plots:* Each vertical 'bar' represents the minimum, lower quartile, median, upper quartile, and maximum. The dot indicates the median, and the inter-quartile range is visualized as the vertical white space around this dot. Crosses are outliers, which are beyond 1.5 times the inter-quartile range from the lower and upper quartiles.

We decided to use the medians of *observed* mutual information, instead of its purely theoretical counterpart, to group the data points for each input transformation matrix. These observed medians per matrix are the dots in Figure 13. This grouping of data allows us to provide box plots with more statistical information about the data. The exact details per run are lost in this approximation, but in view of the close relationship between the theoretical and observed values, this effect should be minimal.

SENSE OF CONTROL

The questionnaire contained two questions related to the user's sense of control: "*I had the feeling that the hamster did what I wanted it to do*" and "*I had the feeling the computer was following my commands*". These items averaged together form the combined *sense of control* scale, which was found to be highly reliable (Crohnbach's $\alpha = 0.89$).

Figure 14 shows the sense of control results grouped per input transformation matrix represented by the median observed mutual information related to it. The strong and significant fit of the linear regression analysis between mutual information and sense of control ($\beta = 36.51, p < 0.001$) indicates that people are competent at estimating their level of control.

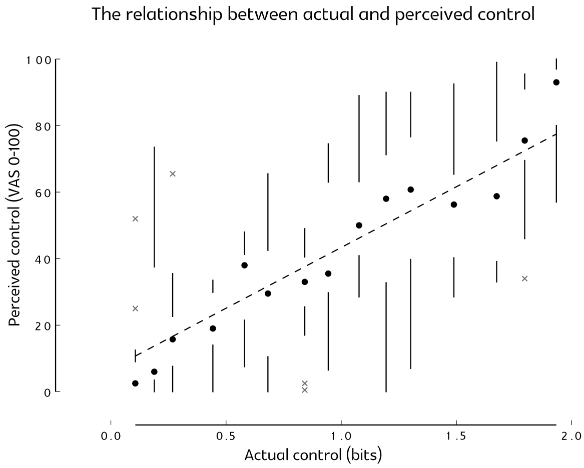


FIGURE 14: **The relationship between actual and perceived control.** The actual control is the median mutual information, representing the input transformation matrices. Perceived control is the combined control scale. A linear regression analysis (the dashed line) showed that actual control is a highly significant predictor of the perceived control ($\beta = 36.51, p < 0.001$), accounting for 72% of the variance. The indicated outliers were included in the regression analysis.

How well does *accuracy* do as a predictor of sense of control? Again, we use the actual accuracy as observed from the interaction logs. Accuracy is a less accurate linear predictor of sense of control than mutual information, explaining 67% of the variance as opposed to 72% (with $p < 0.001$, the same). The medians indicate an exponential relationship, which is to be expected based on the logarithmic relation between mutual information and accuracy.

SUFFICIENT CONTROL

The amount of frustration decreases when the amount of control (in mutual information) increases ($\beta = -23.35, p < 0.001$). However, this does not tell us the minimum amount of control users need. One could put an imaginary boundary at some level of the VAS item, but what level of *frustration* is unacceptable?

Another source of information on how much control is needed is the way runs were ended. Participants could simply leave the website, or they could wait a minute and then skip to the questionnaire by pressing a button, or

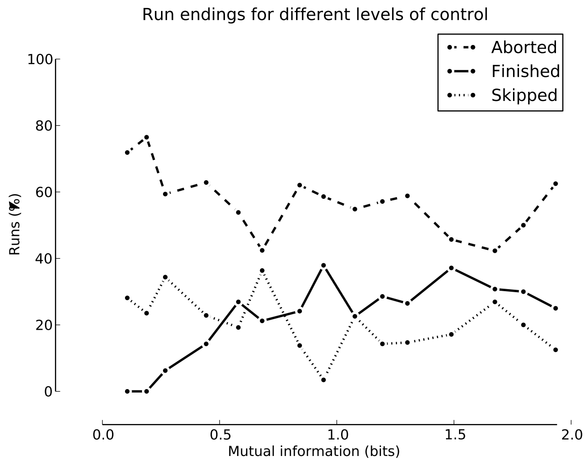


FIGURE 15: **The percentage of different ways of ending a run** for each of the different levels of control.

they could finish the level by bringing the hamster to safety. At the necessary amount of control, more runs should be finished, and fewer runs should be aborted. People can have various reasons not to finish a run, which are not related to control. Perhaps they did not like the game enough, or something more important came up. As long as the reasons are not connected to the amount of control, we can assume that it will have an equal chance to occur at each of the levels of control.

Figure 15 shows the different ways runs were ended for each of the different levels of control. The numbers of aborted and skipped runs slightly decrease with increasing control, and the number of finished runs increases accordingly. Surprisingly, the effect of the amount of control does not seem to be very strong, especially for higher levels of mutual information. However, on the low control side, up to a mutual information of 0.68 bits, there does seem to be a clear effect, with a steep decrease in aborted runs, and a similar increase in finished runs. This seems to indicate that up until this point, the amount of control was the critical reason to stop playing. Beyond this level of control, other unknown, but control-independent, reasons seem to become dominant as the percentages become more steady. The increase in aborted runs after 1.5 bits could be related to the decrease in fun participants experienced when the level of control got (close to) perfect – see our previous data analysis in [5].

This critical point where the amount of control is no longer a critical factor for finishing the game is at about 0.68 bits. This corresponds to 65% observed accuracy for 5 classes. At the 5 inputs per second this game allowed, this is an information transfer rate of 204 bits per minute.

4.4 DISCUSSION AND CONCLUSIONS

SENSE OF CONTROL

People are competent at estimating their level of control over keyboard input in this game. There is a strong and significant linear relationship between people's sense of control and how much control (in mutual information) they actually had, where the actual amount of control explained 72% of the variance in perceived control. This observation may be generalizable to other inputs, other applications, and less immediate effects of the input. To confirm this, further research is required.

I suspect that the key aspect for users to be able to assess their level of control is that they are certain about what input they provide. With brain-computer interfaces, this is not yet the case. Even with relatively simple mental tasks such as focusing on a flickering target in the case of SSVEP, participants can be uncertain whether they are focusing in the right way or with the right intensity. In such situations, the psychological effects on the perception of control might be stronger, which would correspond to what we have informally observed in practice. However, with more practice this uncertainty, and any positive effects from the novelty of this type of interface, will diminish. As a result, the actual amount of control will become more dominant in the perception of control.

Yet, some level of uncertainty will remain even with practice. This opens up a way to make uncertain input modalities more accepted, for example through the psychological phenomena mentioned in the background section, such as the illusion of control. More uncertain input modalities could, for example, be applied to user actions which will have a positive end result, and input modalities with more accurate detection to those actions that prevent a bad outcome from occurring. Input tasks and feedback can also be chosen to enhance these effects. Such deliberate manipulation of perception is particularly applicable for games, as the goals and results of user actions are designed by the game designers instead of following from user goals. Besides, ambiguity can be used as a way to enhance user engagement [19]. Related to this concept is the idea to purposefully introduce a positive bias in the feedback [22].

See Section 8.2 for further musings about this direction of investigation.

SUFFICIENT CONTROL

We observed a critical level of control at 0.68 bits of mutual information, below which the amount of control affects the number of finished runs. In this application, this corresponds to an accuracy of 65% for 5 classes. This is slightly lower than the 70% indicated by BCI research groups (see Background and related work).

Again, this result is based on this one application, with keyboard input. Plus there are different ways of determining what amount of control is sufficient. This concerns just one specific aspect: whether people give up playing the game. Besides, the potential other factors that may affect the sense of control could likely affect the necessary amount of control as well.

All this begs for more research in this area. Not only to increase the amount of information that can be provided through interfaces like BCIs, but also to investigate how this critical amount of control might be reduced. Again, games provide the perfect vehicle for this kind of research, as the goals and results are designed for a specific experience. An example of a concrete next step could be to compare the perception of control in a game where in one condition the control is used to obtain a gain and in another to prevent a loss.

A related direction for investigation is whether to use uncertain input for direct or indirect control [23]. Indirect control is when input affects game mechanics. Direct control is when input affects game objects directly. In the case of multimodal interaction, using any uncertain inputs for indirect control may allow for the more certain direct control to circumvent any problems caused by the uncertain input.

KEY POINTS

- The perception of control in relation to an interface can be split up into three parts: (1) How much control the user has in providing the input; (2) How much the provided input is recognized by the system; (3) Interface and application aspects that appeal to specific psychological phenomena that affect our sense of control.
- If people have full control over the input task, they appear to be good at estimating how well this task is recognized by the system. In our application, the actual level of task recognition explained 72% of the indicated sense of control.
- As the uncertainty over the input task decreases with training, the actual task recognition will become more dominant in the perception of control.
- Uncertainty may be deliberately exploited to create systems that people will perceive as providing more control than they actually do.
- How much control is enough obviously depends on the application, but in this particular case a mutual information of 0.68 bits, or a 5-class accuracy of 65%, seems to be the minimum to prevent people from giving up.

REFERENCES

[1] D. Plass-Oude Bos, B. L. A. van de Laar, B. Reuderink, M. Poel, and A. Nijholt. "Perception and manipulation of game control." In: *Proceedings 6th International Conference on Intelligent Technologies for Interactive Entertainment (INTETAIN 2014)*. Ed. by D. Reidsma. Vol. 136. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer Verlag, Berlin, 2014, pp. 57–66 (cit. on p. 47).

[2] A. Nijholt and D. S. Tan. "Playing with your brain: brain-computer interfaces and games." In: *Proceedings of the international conference on Advances in Computer Entertainment Technology*. ACM. 2007, pp. 305–306 (cit. on p. 47).

[3] A. Nijholt, D. Plass-Oude Bos, and B. Reuderink. "Turning shortcomings into challenges: Brain–computer interfaces for games." In: *Entertainment Computing* 1.2 (2009), pp. 85–94 (cit. on p. 47).

[4] B. Graimann, B. Z. Allison, and A. Gräser. "New Applications for Non-invasive Brain-Computer Interfaces and the Need for Engaging Training Environments." In: *BRAINPLAY 07 Brain-Computer Interfaces and Games Workshop at ACE (Advances in Computer Entertainment)*. 2007, pp. 25–28 (cit. on p. 47).

[5] B. L. A. van de Laar, D. Plass-Oude Bos, B. Reuderink, M. Poel, and A. Nijholt. "How Much Control Is Enough? Influence of Unreliable Input on User Experience." In: *Cybernetics, IEEE Transactions on* 43.6 (Dec. 2013), pp. 1584–1592 (cit. on pp. 47, 49, 54).

[6] L. G. Allan and H. M. Jenkins. "The judgment of contingency and the nature of the response alternatives." In: *Canadian Journal of Psychology / Revue canadienne de psychologie* 34.1 (1980), pp. 1–11 (cit. on p. 48).

[7] S. C. Thompson, W. Armstrong, and C. Thomas. "Illusions of control, underestimations, and accuracy: a control heuristic explanation." In: *Psychological bulletin* 123.2 (1998), p. 143 (cit. on p. 48).

[8] E. J. Langer. "The illusion of control." In: *Journal of personality and social psychology* 32.2 (1975), p. 311 (cit. on p. 48).

[9] M. Buchanan. "Behavioural science: secret signals." In: *Nature* 457.7229 (2009), pp. 528–530 (cit. on p. 48).

[10] N. Tractinsky, A. S. Katz, and D. Ikar. "What is beautiful is usable." In: *Interacting with computers* 13.2 (2000), pp. 127–145 (cit. on p. 48).

[11] D. A. Norman. "Emotion & design: attractive things work better." In: *Interactions* 9.4 (2002), pp. 36–42 (cit. on p. 48).

[12] G. Hakvoort, H. Gürkök, D. Plass-Oude Bos, M. Obbink, and M. Poel. "Measuring Immersion and Affect in a Brain-Computer Interface Game." In: *Human-Computer Interaction - INTERACT 2011*. Berlin/Heidelberg, Germany: Springer-Verlag, 2011, pp. 115–128 (cit. on p. 48).

[13] B. L. A. van de Laar, H. Gürkök, D. Plass-Oude Bos, F. Nijboer, and A. Nijholt. "Brain-Computer Interfaces and User Experience Evaluation." In: *Towards Practical Brain-Computer Interfaces*. Ed. by B. Z. Allison, S. Dunne, R. Leeb, J. del R. Millán, and A. Nijholt. Springer, 2012, pp. 223–237 (cit. on p. 48).

[14] M. Quek, D. Boland, J. Williamson, R. Murray-Smith, M. Tavella, S. Perdikis, M. Schreuder, and M. Tangermann. "Simulating the feel of brain-computer interfaces for design, development and social interaction." In: *Proceedings of the 2011 annual conference on Human factors in computing systems*. ACM. 2011, pp. 25–28 (cit. on p. 48).

[15] F. Cincotti, L. Kauhanen, F. Aloise, T. Palomäki, N. Caporusso, P. Jylänki, D. Mattia, F. Babiloni, G. Vanacker, M. Nuttin, M. G. Marciani, and J. del R. Millán. "Vibrotactile feedback for brain-computer interface operation." In: *Computational intelligence and neuroscience* 2007 (2007) (cit. on p. 48).

[16] M. P. Ware, P. J. McCullagh, A. McRoberts, G. Lightbody, C. Nugent, G. McAllister, M. D. Mulvenna, E. Thomson, and S. Martin. "Contrasting levels of accuracy in command interaction sequences for a domestic brain-computer interface using SSVEP." In: *Biomedical Engineering Conference (CIBEC), 2010 5th Cairo International*. IEEE. 2010, pp. 150–153 (cit. on p. 48).

[17] T. E. Carlson, G. Monnard, and J. del R. Millán. "Vision-based shared control for a BCI wheelchair." In: *International Journal of Bioelectromagnetism* 13.1 (2011), pp. 20–21 (cit. on p. 48).

[18] M. Quek, J. Höhne, R. Murray-Smith, and M. Tangermann. "Designing future BCIs: Beyond the bit rate." In: *Towards Practical Brain-Computer Interfaces* (2012). Ed. by B. Z. Allison, S. Dunne, R. Leeb, J. del R. Millán, and A. Nijholt, pp. 173–196 (cit. on p. 48).

[19] W. W. Gaver, J. Beaver, and S. Benford. "Ambiguity as a resource for design." In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM. 2003, pp. 233–240 (cit. on pp. 49, 55).

[20] D. J. C. MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003 (cit. on p. 50).

[21] M. E. Wewers and N. K. Lowe. "A critical review of visual analogue scales in the measurement of clinical phenomena." In: *Research in nursing & health* 13.4 (2007), pp. 227–236 (cit. on p. 51).

[22] Á. Barbero and M. Grosse-Wentrup. "Biased feedback in brain-computer interfaces." In: *Journal of neuroengineering and rehabilitation* 7.1 (2010), p. 34 (cit. on p. 55).

[23] C. Mühl, H. Gürkök, D. Plass-Oude Bos, M. E. Thurlings, L. Scherffig, M. Duvinage, A. A. Elbakyan, S. Kang, M. Poel, and D. K. J. Heylen. "Bacteria hunt." In: *Journal on Multimodal User Interfaces* 4.1 (2010), pp. 11–25 (cit. on p. 56).

5 POST-PROCESSING IN BCI LITERATURE

From Chapter 3 we know that users want their input to be easy to provide, and well recognized by the system (and users have a good sense of the system detection accuracy, see Chapter 4). These demands seem to be exactly the opposite of what BCIs currently have to offer, especially when talking about consumer-grade hardware used in normal, out-of-the-lab situations (see Chapter 1). Post-classification processing can increase performance and reduce user effort by adjusting the way the input is being used. Previous surveys indicate a lack of attention for the application of post-processing methods in BCIs in the past. In this chapter, I investigate whether this is still the case by conducting a follow-up literature review. To increase awareness of the potential benefits of these methods for the BCI community, this chapter also provides an initial overview of the methods (further extended in Chapter 6) and the performance gains reported in these studies.

Brain-computer interfaces (BCIs) provide systems with input based on the user's mind, so devices and applications can respond to specific mental states. Like other input modalities based on observations of the body, BCIs do not provide perfect recognition of what a user is trying to convey [1, 2, 3, 4]. These inputs suffer from problems related to noise, non-stationarities, and ambiguity [5]. And these problems get worse the more we move towards real-world applications, with more noise, distractions, and multitasking. As input is the basis for usable systems in general, and recognition accuracy is most important to users of BCIs [6], it is only logical that so much BCI research is devoted to improving the detection of mental states, by improving the recording technology, and inventing sophisticated feature extraction and selection methods.

See Chapters 1 and 2.

See Chapters 3 and 4.

But there is another approach that can significantly improve control and reduce user effort: post-classification processing, or *post-processing* in short [7]. Duda, Hart, and Stork roughly define post-processing in their classic text book "Pattern Classification": Post-processing methods can take into account information that goes beyond the narrow scope of the classifier, such as context information or the cost of the action triggered as a result of the classification [8]. In this thesis, I define the term slightly more broadly to include all methods that aid in the translation from logical to semantical control signals, that is from classification (feature translation) results to input with meaning in the application context. This definition is more in line with the literal meaning of the term 'post-processing' indicating the position of these methods in the BCI cycle (see Chapter 2). Other general terms that have been used to indicate post-processing or subsets of post-processing methods are "hybrid BCI", "shared control", "context-aware", "context-sensitive", and "intelligent control" [9, 10, 11, 12, 13].

What post-processing essentially does is add an external cerebellum to the cortex-based control signal, to make it smooth, adaptive, and accurate; the same function our own cerebellum performs when translating motor commands from the motor cortex to the spinal motoneurons which pass it on to our muscles. Wolpaw poses that the variability in BCI performance is inherent to measuring the cortex directly, without the cerebellum and spinal motoneurons for adaptive output [14]. His proposed solution is to develop BCIs that are controlled with goal selection, instead of a step-by-step process control. This chapter will show other methods that can result in more smooth, more adaptive, and more accurate control.

Other input modalities, long before BCI, have found a solution in these post-processing methods. When you press a key on a keyboard, the output signal does not simply go from off to on. There is a so-called 'dirty edge', a period where the signal wavers between on and off, before settling on *on*. A dwell time is added for the system to interpret this as a clean jump. For the mouse, complex transfer functions translate physical mouse movement to movement on-screen, in such a way that it dynamically provides an optimal trade-off between speed and precision. Yet, the BCI community seems hesitant to use these often-simple methods to increase the usability of their systems.

To provide some idea of how post-processing can improve the performance of a BCI system, let us look at a P300 speller [15]. This application shows a matrix of characters that the user may want to type (see Figure 16). The user then counts the number of times the wanted letter is being highlighted, in order to eventually 'type' it.

Most P300 spellers combine multiple repetitions of P300 detections, for example by *averaging*. This means that the user must pay attention to the character to select for multiple flashes for each row and column. The multiple P300 classification probabilities are then combined to yield one detected letter. Krusienski *et al.* summed the distances to the LDA decision plane for the repeated flashes for each row and each column [16]. The row and column with the largest summed distances determined the letter. Where one repetition yields an average classification accuracy of 25%, five repetitions already improves this to 70%, and with fifteen an accuracy of 90% is reached.

A second improvement could be to make use of the predictability of language. After writing 'TH', there are not so many options for the next character. An 'E' is a lot more likely than a 'B'. These probabilities can be taken into account when interpreting the next input. Speier *et al.* added a simple trigram model to their P300 speller, which provides the probabilities for a character given the previous two characters [17]. This model not only affected the interpretation of the P300 input, but also reduced the number of

A	G	M	S	Y	5
B	H	N	T	Z	6
C	I	O	U	1	7
D	J	P	V	2	8
E	K	Q	W	3	9
F	L	R	X	4	_

FIGURE 16: **An example of a P300 speller matrix,** with one column 'flashed'. Rows and columns are flashed repeatedly. The user focuses on the character she wants to type, and mentally counts each time her target flashes. After a certain number of repetitions, the system types the character which resulted in the most pronounced P300 responses, which should be the user's target.

necessary repetitions, which were limited by a probability-based threshold. It increased the accuracy by 10%, and improved the bit-rate from 22 to 33 bits per minute.

As post-processing methods affect the translation from logical control to semantic control, they are likely to affect the input required. This is how post-processing can do more than improve recognition accuracy: it can explicitly reduce user effort, as in the above example by reducing the number of repetitions. To take this even further, with auto-completion, for example, words could be finished off automatically, such as 'congrat' to 'congratulations'. Such a feature could add another dramatic reduction of the amount of input needed to type a certain text.

These are large performance increases that make the difference between a barely functional proof-of-concept for the sake of science (25% accuracy), and a system that people will actually want to use in real life (100% accuracy)[1]. Just with the help of a couple of post-processing methods.

In a paper from 2006, Jackson et al. compared 21 BCI systems with a comprehensive framework that included the control interface specifically [18]. Only two of those systems (9.5%) had one.

Bashashati et al. [7] conducted a survey of over 200 BCI papers published before 2006, with a wide search net, including all English conference papers and journal articles with 'BCI', 'BMI', or 'DBI' in the title, abstract, or

1 Although the end result here was 100% detection accuracy, there are contexts where a lower performance actually results in a better user experience. See Laar et al. [5] for an example.

keyword list. They concluded that less than 15%[2] of found systems used some form of post-processing [7]. Their table with the papers with post-processing they found has been reproduced here, see Table 2. The authors summarize: "Of the 30 BCI designs that use post-processing algorithms to reduce the amount of error in the output of the BCI system, 57% use averaging techniques and consider rejecting activations that have low certainty, 27% consider using the debounce block (or refractory period) to deactivate the output for a short period of time when a false activation is detected, and 16% use event-related negativity (ERN) signals to detect error activations."

In the last seven years, one would expect that this number has gone up. The number of BCI systems aimed to be used in real-world situations has increased. Besides, most post-processing methods are extremely simple to implement. Why would you not boost the performance of your system, when performance is still considered to be the main problem of BCIs today?

A recent review paper by Nicolas-Alonso and Gomez-Gil [19] is indicative of the current undervaluation of post-processing methods that still exists in the BCI community. It describes the current state of the art as well as fundamental aspects of BCI system design. It cites over 300 BCI papers, so it attempts to be a good representation of current BCI research. Although it does mention the control interface as a processing step in between classification and the application, this processing step does not get the same attention as the other BCI processing steps. All the other steps get their own section, comparing different implementations. Post-processing methods (also never indicated as such, nor as control interface steps) are muffled away in the descriptions of various applications.

To investigate the *current* use of post-processing in BCIs, I conducted a follow-up literature review to the one from 2007. Where the survey by Bashashati et al. [7] looked at *all* methods used in BCI systems, I focused solely on the post-processing methods. I include an overview of the post-processing methods with the performance gains reported in these studies, as a first step to increasing awareness of the potential benefits, and opening up the dialogue about these methods.

5.1 LITERATURE REVIEW METHODS

How do you search for something that most people do not apply deliberately? Researchers use many different words to describe the post-

2 Thirty BCI papers (the authors seem to use 'paper' and 'design' interchangeably) out of over 200 reviewed constitutes less than 15%. However, if we look at Table 2, it actually contains only 25 unique citations instead of 30. (The original table also mentions Millán and Mourino 2004b, but there was no corresponding citation in the references list.) Besides, it is important to note that many of these citations are from the same research groups. Six have Millán as first author. Five of them have involvement of Birch, and four of Pfurtscheller. It is doubtful each of these papers refers to a distinctive BCI design.

Post-processing	Reference
ERN-based error correction	Bayliss, Inverso, and Tentler [20], Blankertz et al. [21, 22], Parra et al. [23], and Schalk et al. [24]
Successive averaging / rejection option for low posterior probabilities (choice of 'unknown' output state)	Anderson, Devulapalli, and Stolz [25], Bashashati, Ward, and Birch [26], Birch, Bozorgzadeh, and Mason [27], Borisoff et al. [28], Fatourechi et al. [29, 30], Gysels and Celka [31], Millán [32], Millán et al. [33, 34, 35, 36], Millán and Mouriño [37], Müller-Putz et al. [38], Penny et al. [39], Roberts, Penny, and Rezek [40], Townsend, Graimann, and Pfurtscheller [41], and Vidal [42]
Debounce (refractory period)	Bashashati, Ward, and Birch [26], Borisoff et al. [28], Fatourechi et al. [29, 30], Müller-Putz et al. [38], Obeid and Wolf [43], Pfurtscheller et al. [44], and Townsend, Graimann, and Pfurtscheller [41]

TABLE 2: **Reproduction of "Post-processing methods in BCI designs" from Bashashati et al. [7].**

processing methods they use — just for smoothing methods alone we find: filter, antialiasing, averaging —, and although most papers do include specific paragraphs on brain activity acquisition, preprocessing, feature extraction, and feature translation (mostly called ‘classification’), not many papers have an explicit paragraph on ‘post-processing’ or ‘control interface’.

As I could not trust the use of specific terms in the title, abstract, or keywords list (which was the approach by Bashashati et al. [7]) I needed a database that allowed us to search in the article text itself. This eliminated those databases that are most popular for literature reviews, such as Elsevier’s Scopus and Thomson Reuters’ ISI Web of Knowledge. I selected Google Scholar, as it has one of the largest databases, is not limited to specific research areas (important in an interdisciplinary area as brain-computer interfaces), and also allows for the necessary text search.

Searching for BCI papers in general yields too many search results[3], with too many false positives: papers that would not include any information about post-processing whatsoever. But the search should also not be limited to just those methods I already knew about, or to those specific names for those methods. On the other hand, adding any relevant keyword would exclude papers that did not explicitly mention this term.

In the end, I decided to use the search query

bci AND brain AND post-processing

for 2006–2012, in Google Scholar, looking in the entire publication text, excluding patents.

Although adding the term ‘post-processing’ may appear to result in a rather strict search query, it had the highest chance of finding papers with post-processing methods without defining in advance what these methods should be, and how they should be called. At the same time, adding the term did provide a good selection from the total set of papers on brain-computer interfaces published in this time period. The chosen search query yielded 274 search results[4]. These results were further inspected as to whether they fulfilled all the following criteria: (1) it is a journal article or conference paper, and (2) it describes a brain-computer interface, (2) which uses post-classification processing.

5.2 RESULTS

Of the 274 search results, most papers ended up not fulfilling these criteria. Book chapters, theses, presentations, and other non-articles were ig-

3 “bci AND brain” for 2006–2012 gives 13.100 search results in Google Scholar. “bci” alone for the same time period yields 22.700, which includes papers on biology concept inventory and BCI-algebras, for example.

4 Now, in June 2014, about a year after I conducted this literature review originally, the number of results for 2006–2012 has risen slightly to 292.

nored. Three articles could not be accessed. A few articles that showed up in the search results have yet to be published. Some papers were not about brain-computer interfaces (but, for example, about a bladder control index; I did include some cases with only offline brain activity analysis which are officially not considered BCIs either), or about post-classification post-processing. Post-processing is obviously a very generic term, and can in theory be used to denote any processing that happens after something else, such as immediately after recording the brain signals. It is important to note that in such cases I did not simply dismiss the paper. The paper was carefully read to ensure it was not using any post-processing according to the definition provided in the chapter introduction, even if the term was not used as such. In the end 40 papers were actual hits, which comes down to 15% of the original search results.

PUBLICATION INCREASE OVER THE YEARS

To investigate whether the use of post-processing methods is on the increase, as one would expect after the benefits were proven (see the Introduction), I compared the number of post-processing papers I found to the total number BCI papers published. Neither the number of post-processing papers nor the number of general BCI papers will be complete, but they are a subsample that are indicative of the total picture in a relative comparison. The number of samples (7 years), is also quite small. Nevertheless, this data should give some hint towards the use of post-processing in brain-computer interfaces over this time period.

To determine the number of BCI papers published in general, we used Thomson Reuters' Institute for Scientific Information (ISI) Web of Knowledge, just as was done in the bibliometric study into the publications on brain-computer interfaces by Hamadicharef [45]. We also used the same search key, but adjusted the year range to ours: Topic=("brain-computer interface") AND Year Published=(2006-2012), Document Types=(ARTICLE). This yielded 1408 results.

Figure 17 shows both the numbers of general BCI publications and post-processing publications published per year. The number of published papers using post-processing methods is growing. However, relatively to the total number of BCI publications, post-processing does not appear to be on the increase.

METHOD CATEGORIZATION AND OVERVIEW

Some papers concerned the same BCI system. For the categorization, these papers have been combined and are represented by just one of the articles. Fatourechi et al. [75, 76] and Fatourechi, Ward, and Birch [53] all discuss the

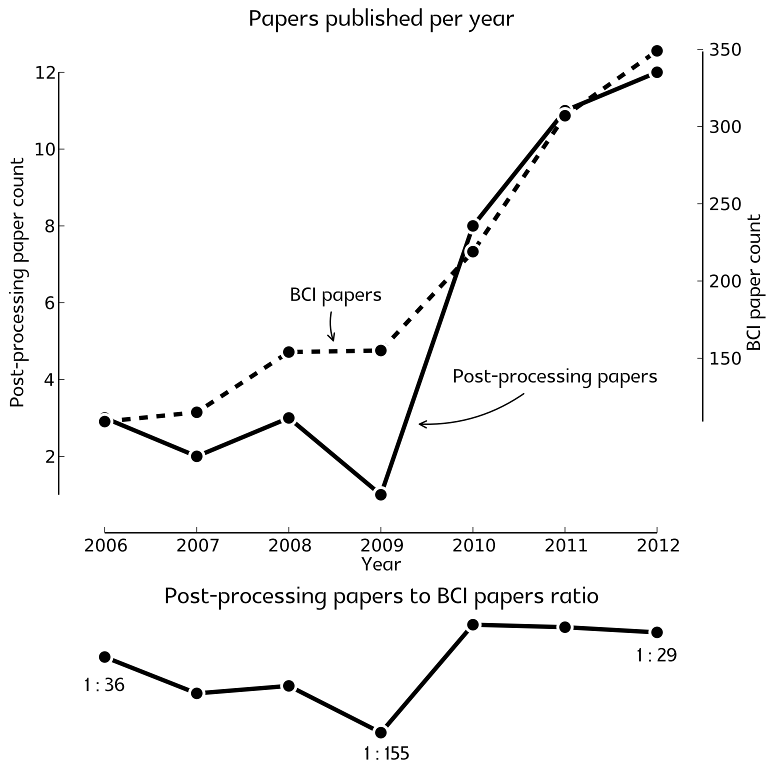


FIGURE 17: **Above: The number of general BCI publications and found post-processing publications** published per year. Both numbers show growth.
Below: the ratio of found post-processing papers to BCI papers in general per year. Although the ratio of found post-processing papers to all BCI papers has increased from 1:36 (2.8%) in 2006 to 1:29 (3.4%) in 2012, there seems to be no steady increase.

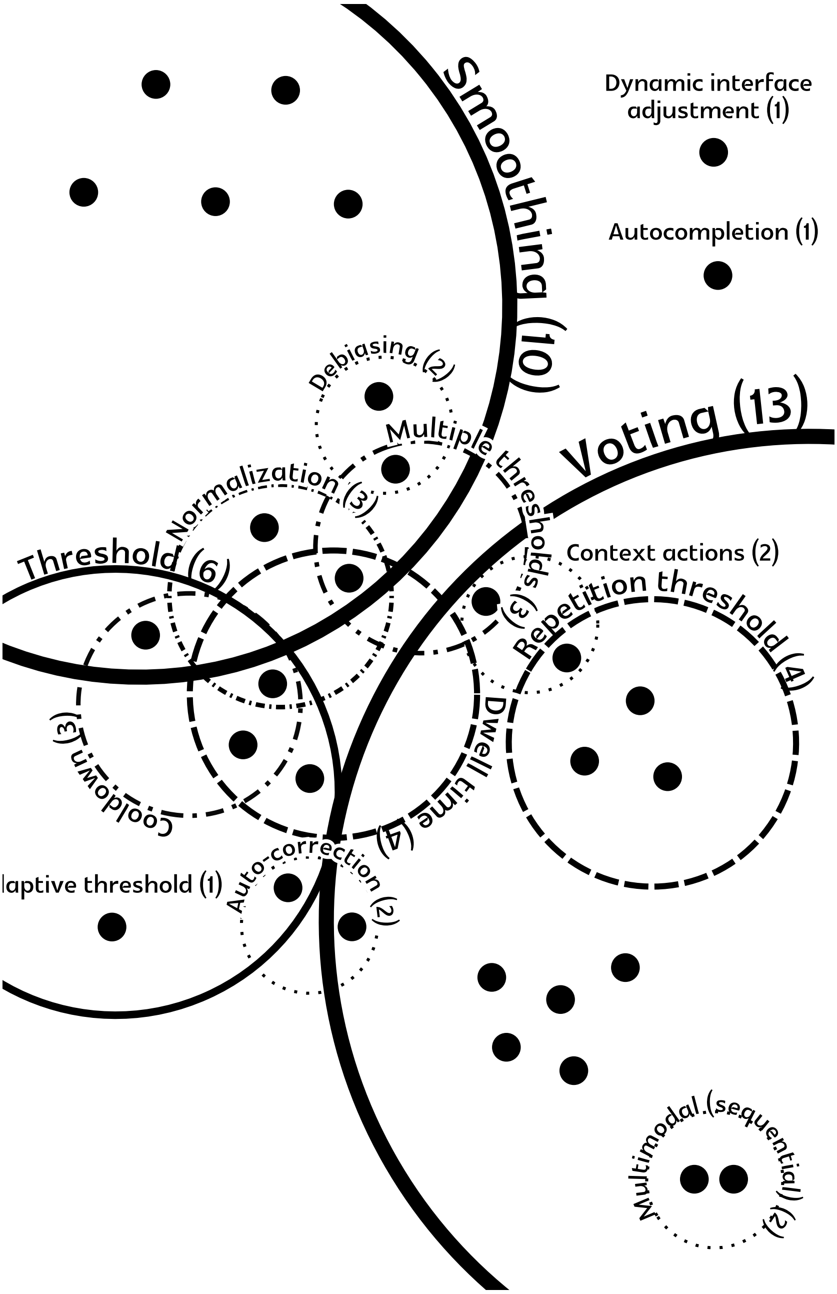


FIGURE 18: Categorization of the post-processing methods used in the 30 distinct systems described in the 40 papers. Category circle radius corresponds to the number of systems assigned to that category.

Post-processing	Systems
Autocompletion	Ashari, Al-Bidewi, and Kamel [46]
Auto-correction	Ferreira et al. [47] and Manoochehri and Moradi [48]
Context actions	Makeig et al. [49] and Satti, Coyle, and Prasad [50]
Dynamic interface adjustment	Jarzebowski, Lakshminarayan, and Coleman [51]
Smoothing	Coyle et al. [52], Fatourechi, Ward, and Birch [53], Heger et al. [54], Lee et al. [55], Liang et al. [56], Perdikis et al. [57], Poli, Salvaris, and Cinel [58], Temko et al. [59], Verwaeren, Waegeman, and De Baets [60], and Plass-Oude Bos et al. [4]
Voting	Dobrea, Dobrea, and Costin [61], Duvinage et al. [62], Ferreira et al. [47], Verschore et al. [63], Liu et al. [64], Makeig et al. [49], Martinovic et al. [65], Orhan et al. [66], Rakotomamonjy et al. [67], Riechmann et al. [68], Satti, Coyle, and Prasad [50], Zhang et al. [69], and Zoughi and Boostani [70]
Debiasing	Coyle et al. [52] and Perdikis et al. [57]
Normalization	Hasan and Gan [71], Temko et al. [59], and Plass-Oude Bos et al. [4]
Threshold	Fatourechi, Ward, and Birch [53], Hasan and Gan [71], Jun [72], Leeb et al. [73], Manoochehri and Moradi [48], and Solis-Escalante et al. [74], and Jun [72] (Adaptive)
Multiple thresholds	Coyle et al. [52], Satti, Coyle, and Prasad [50], and Plass-Oude Bos et al. [4]
Cooldown	Fatourechi, Ward, and Birch [53], Hasan and Gan [71], and Solis-Escalante et al. [74]
Dwell time	Hasan and Gan [71], Leeb et al. [73], Solis-Escalante et al. [74], and Plass-Oude Bos et al. [4]
Repetition threshold	Verschore et al. [63], Liu et al. [64], Makeig et al. [49], and Orhan et al. [66]
Multimodal (sequential)	Duvinage et al. [62] and Riechmann et al. [68]

TABLE 3: **30 systems from 40 papers, categorized by post-processing method.** When systems use multiple methods, they are repeated for each.

LF-ASD, BCI switch controlled by imaginary finger movements. Hasan and Gan [77, 78, 71] detail the same BCI system to play Hangman. Lee et al. [79, 55] are both about the same study on spike trains. Zoughi and Boostani [70] and Zoughi, Boostani, and Deypir [80] are about detecting anaesthesia. Poli, Salvaris, and Cinel [81, 58] describe a P300-based BCI mouse. Solis-Escalante et al. [74] and Leeb et al. [73] discuss the Graz BCI. Temko et al. [59, 82] are about seizure detection. For Perdikis et al. [83], we looked up a follow-up paper with more details, which is Perdikis et al. [57].

The papers for each system were thoroughly analyzed for post-processing methods. When different papers used the same or highly similar methods, but used a different name to indicate the method, one name was chosen to represent the method in both. These names were selected based on ease of interpretation and how well it identified the method category. The development of this categorization has been an iterative process.

Refer to Table 3 for an overview of the papers by post-processing. Figure 18 shows a visual representation of how the different methods are represented in the selected papers.

Auto-correction, autocompletion, context actions, and *dynamic interface adjustment* are methods which are particularly powerful when they have access to the application state, or otherwise have an internal model of certain application-based logic. *Auto-correction* is about adjusting previous input based on the inputs that happened before and after. As a result, you need a kind of roll-back to be able to implement this. An example of this in a BCI spelling application is when the user spells "teh", to automatically correct it to "the"[5]. Such situations can be detected with a relatively simple language model.

Auto-correction

Autocompletion then would be to offer the likely option "the" after the user only provided the "t". With *context actions*, one input can trigger a variety of actions, depending on the context. In a role-playing game, for example, the same action (say hand movement imagery) could trigger a conversation when near a person, or picking something up when near an object. *Dynamic interface adjustment* is mostly used to make it easier to select more likely targets. This can, for example, be done by rearranging menus or increasing target sizes.

Autocompletion

Context actions

Dynamic interface adjustment

The most-used methods are *smoothing* and *voting*, which coincides with the results from Bashashati et al. [7], see *successive averaging* in Table 2. These two methods are very much related, as both combine multiple samples in order to obtain a more robust conclusion. Outliers have less influence on the end result. The difference is that voting takes a specific group of samples, often related to a particular stimulus, whereas smoothing is a contin-

5 Interesting to note: these search results contained no papers using *error-related negativity* (ERN) to correct interpretation mistakes afterwards. One could also argue that this is not really post-processing, but more like a mental backspace. Then it is simply a separate input which is interpreted independently. It does not affect the interpretation of the logical control of another input.

uous process, with a sliding window. To give a concrete example, for a P300 speller, it is common to combine multiple P300 classifications for each possible target, for example by taking the average probability. This is voting. On the other hand, an indication of relaxation might be smoothed by taking the average of a sliding window of data, resulting in a more fluid signal which is less sensitive to short disturbances.

Debiasing

Normalization

Normalization and *debiasing* serve to adjust the range and distribution of the control signal. When the input is not evenly distributed (biased), a transform can adjust this. One of the most simple examples is adding an offset, to adjust the central tendency of the input values. *Normalization* transforms the input to the right range. Such transforms to adjust the *span* and *zero* (the lowest possible value) generally also affect the distribution, so some *debiasing* may be inherent.

Thresholds

Another large category is *thresholds*. Thresholds are a way to transform continuous data into discrete actions, which is essentially a form of *binning*. Data below the threshold can either be rejected, or interpreted as a *low* value. The difference is that a low value could still result in a system response, whereas in the case of rejection, the observation may never be passed on to the listening application. For example, when a brain-sensitive mp3 player receives a control signal declaring that the user is happy (the signal is high, above threshold), the music player can decide to play some happy tunes. When the mp3 player observes *low* happiness, it can play a sad song. If, however, it does not receive any control signal because the data was rejected, there is nothing to respond to. This *no control* (or *idle state*) situation can be a very important feature in BCI systems that listen for user input continuously, for reasons related to the Midas touch problem – not every activity in the brain is meant as a control signal [84, 27, 85]. Such a threshold can be made adaptive, to balance the occurrence of *low* (or *no control*) and *high* [72].

Multiple thresholds

Multiple thresholds, *cooldown*, and *dwell* time are all methods to *debounce*[6] the signal, meaning that it reduces fast, unwanted oscillations. With multiple thresholds, you need to decide how to deal with the new intermediate state. It is either a rejected *no control* state, or retains its previous state. This last option is called *hysteresis*, and has, for example, been used in Plass-Oude Bos, Poel, and Nijholt [6]. In that prototype, the online role-playing game World of Warcraft is adjusted to respond to the user's mental state. When the user is relaxed, her avatar is an elf. When the low threshold (0.25) is crossed, she automatically transforms into a bear. To return to being an elf, she will need to increase her relaxation to cross the high threshold (0.75).

Cooldown

Dwell time

A *cooldown*, also known as *refractory period*, suppresses the output for a set cooldown duration, after the control signal was high. Adding a *dwell time* requires the signal to be high for a set period, to result in a high output. The input for these debounce methods can either be continuous or discrete, but

6 Although in BCI papers, debounce is generally used to indicate a cooldown period, the software and hardware debouncing of physical buttons is generally done by adding a dwell time, implemented through a simple counter or a capacitor.

the output is generally discrete; similar to the behaviour of a switch turning on or off. This makes the output quite different from smoothing, although smoothing also removes fast oscillations. These methods are applied to non-stop input, as opposed to the intermittent type of input which is common for voting.

Repetition threshold

A very different kind of threshold is the *repetition threshold*. Its purpose is to reduce time: the action is triggered as soon as a sufficient probability level has been reached. This is faster than simply going for a (larger) fixed number of repetitions that is known to be on the safe side. These repetitions can then be combined through a voting procedure, for example by averaging. This is why all repetition threshold systems are within the voting circle in Figure 18.

Multimodal

Multiple inputs can be combined for control, which can be done sequentially or in parallel. In BCI research such combinations have been termed "hybrid BCI" [86]. In the research area of human-computer interaction, this concept is known as the more generic "multimodal interface" [87]. The two examples listed here do not actually combine two inputs to arrive at a more certain end result, but use one input to switch off the detection of the other. EOG activity switches off the P300-based BCI in [62], and detected P300 activity turns off ERD-based motor imagery detection in [68].

Most papers only use one post-processing method, but combining them appropriately can be very powerful. One system even combines five methods: Hangman uses debiasing, smoothing, rejection thresholds, dwell time, and cooldown [71].

DETECTORS, APPLICATIONS, AND POST-PROCESSING METHODS

Some post-processing methods are more commonly used with certain BCI detectors[7], such as voting with P300. Similarly, auto-correction or autocompletion based on language models are an obvious benefit to speller applications, which in turn are most often controlled by P300 again. Are any such relations apparent from the papers of this review? Table 4 shows for each detector, which methods have been used.

See Chapter 2.

There is a large diversity in the detectors themselves. It is not just the "classic 4" (P300, SSVEP, motor imagery, and SCP), but papers also describe, for example, emotion detection, person identification, and a set of mental tasks called 'the 5 tasks' which includes tasks like mentally composing a letter and mental rotation of a three-dimensional object [61].

For most BCI detectors, there are no clear preferences for specific post-processing methods. Only motor imagery and P300 show a preference, per-

7 *Detector* is here used as a general term to indicate the processing to detect specific BCI paradigms and mental states, independent of the exact implementation.

BCI detector	Post-processing method
Motor imagery	Threshold (6), Smoothing (4), Cooldown (3), Dwell time (3), Voting (2), Debiasing (2), Thresholds, multiple (2), Auto-correction (1), Context actions (1), Multimodal (classification) (1), Adaptive threshold (1), Normalization (1)
P300	Voting (5), Multimodal (sequential) (2), Threshold, repetition (1), Smoothing (1), Goal selection (1), Dynamic interface adjustment (1)
Emotion	Context actions (1), Threshold, repetition (1), Smoothing (1), Thresholds, multiple (1), Dwell time (1), Voting (1), Normalization (1)
mVEP	Voting (1), Threshold, repetition (1)
RSVP	Voting (1), Threshold, repetition (1)
5 tasks	Voting (1), Smoothing (1)
Seizure detection	Normalization (1), Smoothing (1)
Person identification	Auto-correction (1), Voting (1)
Anesthesia	Voting (1)
Hand speed	Smoothing (1)
Spike trains	Smoothing (1)

TABLE 4: **Overview of the methods used in combination with each BCI detector.** The number after each method indicates the number of systems in the literature review that used this method in combination with this detector. Some systems combined multiple methods.

haps due to the relatively large number of papers describing these detectors.

For motor imagery: threshold and smoothing were the most popular post-processing methods. The applications that were controlled by motor imagery generally required discrete actions, such as 'move right'. This requires a transformation from a probability of motor imagery of the right hand to the discrete decision of whether this motor imagery occurred or not, which is done by applying a threshold. The methods list also contains many methods to make this decision less sensitive to outliers (cooldown, dwell time, smoothing), or to make a fixed threshold more viable (normalization, debiasing).

For P300: the most-used postprocessing method was voting. Voting is a common method for any detector for externally-evoked brain activity, which depends on a given stimulus. This is why it is not only listed for P300, but also for mVEP[8], RSVP[9], and person identification.

For other relations between detector, application, and postprocessing to become apparent, more BCI post-processing papers need to be investigated.

DELIBERATE USE OF POST-PROCESSING

As noted before, most papers that use post-processing will not have included the term post-processing, even though many people in the field should be familiar with the word. The survey by Bashashati et al. [7] features the term and describes it, and, according to Google Scholar, this survey has been cited by 234 papers published in the period between 2006 and 2012.

While the majority of the investigated papers did motivate their use of post-processing, it was often in a non-specific way, such as "to enhance classifier performance" [79, 71, 56, 61]. Other motivations are, for example, to reduce the number of false activations [76], decrease the time per input [51, 63, 66, 46], and to make the output more application-specific [49, 73].

Only for 9 of the 30 systems were any alternatives for their chosen post-processing methods discussed, and for only 5 was the influence of the parameter settings of their post-processing methods evaluated. Deliberate reasoning about which method and which specific implementation to apply should be preferred. Not only does it serve to inform the reader on why a specific decision was made, but it also stimulates discussion and structured evaluation to decide what would be the best method for a given situation.

On the other hand, for 18 of the 30 systems the difference between with and without post-processing was evaluated. Here are their results ordered

8 To explain these more exotic neuromechanisms goes beyond the purpose of this chapter and the scope of this thesis. However, for the sake of completeness: mVEP is a visually-evoked potential that is elicited by making targets move.

9 RSVP stands for Rapid Serial Visual Presentation. As the name implies a sequence of visual stimuli is rapidly shown to the user.

by post-processing method. A few papers had to be excluded as their results were not clearly indicated nor easily derivable.

Autocompletion Ashari, Al-Bidewi, and Kamel [46] reduced the number of necessary selections to select a 7-digit phone number from 7 to 4.

Auto-correction Ferreira et al. [47] reduced the average classification error from 15.7 to 5.1, and Manoochehri and Moradi [48] reduced their false positive rate from 27% to 18%, although at a small cost in true positive rate, which went from 64% to 60%.

Dynamic interface adjustment Jarzebowski, Lakshminarayan, and Coleman [51] reduced their menu depth, and thus the number of steps to arrive at a selection, with a probabilistic model. This increased the information transfer rate by 2.5 bits per minute.

Smoothing Lee et al. [55] used smoothing to increase their accuracy by 20%. For Liang et al. [56] it resulted in a 15% increase in accuracy, and for Perdikis et al. [57] the gain in accuracy was only 3%. On the other hand, in the paper by Poli, Salvaris, and Cinel [58] smoothing actually *increased* the standard error a little, from 3.55 to 3.57.

Voting Dobrea, Dobrea, and Costin [61] increased their 5-class accuracy from 54% to 71% by averaging over 20 segments. Zhang et al. [69] reduced their mean square error by 10% with a neural network.

Debiasing Perdikis et al. [57] gained a 5% increase in accuracy with debiasing.

Adaptive threshold Jun [72] used an adaptive threshold to balance the distribution of the output classes, which reduced the mean square error from 0.59 to 0.30.

Cooldown Fatourechi, Ward, and Birch [53] decreased their false positive rates with an average of 6% by adding a short cooldown period of only 1 decision sample, while reducing the true positive rate by just 1%.

Repetition threshold Verschore et al. [63] decreased the necessary number of repetitions from 12 to only 2.69 on average. Liu et al. [64] managed to increase their information transfer rate from 16 to 26 bits per minute in offline evaluation, and even to 42 bits per minute online.

5.3 DISCUSSION AND CONCLUSIONS

Review articles from 2007 and 2012 show the lack of interest in post-processing methods for brain-computer interfaces [7, 19], even though it has been shown that they can result in systems that are significantly more usable by improving detection accuracy and decreasing user effort (for example, in [17]).

This review is intended as a follow-up to Bashashati et al. [7], focusing on BCI systems using post-processing methods in particular. As post-processing methods are mostly applied without much consideration, there is no consistent use of words that make it easy to identify papers in which such methods are used. Our search query “bci AND brain AND post-processing” yielded 274 results for 2006-2012, of which only 40 actually discussed post-classification processing in brain computer interfaces. There are many more BCI systems with post-processing that were not found using this query, as the authors did not use this term.

It could be interesting to define another search query with all the different names for the various post-processing methods found thusfar, and compare those results with the findings in this chapter. Although such a search query will surely discover more papers with post-processing, it will also be difficult to keep out false alarms, as certain methods are commonly used in other BCI processing steps as well. Adding a keyword like ‘filter’ will not produce much of a selection in the overall pool of BCI papers. On the other hand, it might be possible to construct a search query with other terms to indicate certain subsets of post-processing methods, such as “shared control”, “context-aware”, and “intelligent systems”.

The number of post-processing publications does not appear to be on the increase relative to the total body of publications about brain-computer interfaces. For this comparison, both groups were represented by a subset. The total number of publications with post-processing in BCI is yet unknown, but the papers in this literature study are assumed to be an adequate representation. The same is true for the total body of BCI publications, here based on the results of a specific search query in a specific database.

The various post-processing methods found in this literature review have been described and categorized, thereby extending the categories provided by Bashashati et al. [7]. This overview and categorization provides insight into the methods currently used, but the knowledge it contains is also a first step to facilitate deliberate decision-making about post-processing methods. The categorization is mainly based on function, but it is not an absolute. Some methods could have been grouped together, such as *thresholds* and *multiple thresholds*, for example. However, I think that these methods are functionally different enough to warrant their own category. As for the category names, the term was chosen that seemed to be specific enough for the category but general enough to cover everything in it. Also, simplicity was preferred over complexity. For example: *cooldown* was chosen over *refractory period*.

Some post-processing methods are applied more often in combination with certain BCI detectors, such as voting with stimulus-dependent brain activity detection and threshold with motor imagery. There were not enough papers for the other detectors to derive other relations.

Most of the selected papers did use the term post-processing as meaning post-classification processing, and most of them did motivate their decision to apply post-processing. However, there is very little motivation as to what specific post-processing is applied, and little evaluation of its effect, either compared to other post-processing methods, or compared to no post-processing at all. To enable deliberate use of post-processing methods, it is vital that authors describe their own deliberations, motivate their choices, and evaluate the effects.

It is important to increase the awareness of post-processing methods in the BCI community. Only when these methods are used with the same thorough deliberation as other methods in the BCI pipeline (such as pre-processing, feature selection, and classification), will we learn the best methods for each situation, and what the expected benefits are.

KEY POINTS

- Post-classification processing methods — post-processing methods in short — can drastically improve BCI control and decrease the necessary effort to provide input.
- Post-processing is embraced as a common step in other input modalities.
- Yet, there is little attention for post-processing methods in BCI research, and it does not appear to be increasing relatively to the total body of publications about brain-computer interfaces.
- Application of post-processing methods should be informed through discussion and structural evaluation. Only then we can fully benefit.

REFERENCES

[1] L. Deng and X. Huang. "Challenges in adopting speech recognition." In: *Communications of the ACM* 47.1 (2004), pp. 69–75 (cit. on p. 61).

[2] R. J. K. Jacob and K. S. Karn. "Eye tracking in human-computer interaction and usability research: Ready to deliver the promises." In: *Mind* 2.3 (2003), p. 4 (cit. on p. 61).

[3] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi. "A review of classification algorithms for EEG-based brain-computer interfaces." In: *Journal of neural engineering* 4 (2007) (cit. on p. 61).

[4] D. Plass-Oude Bos, H. Gürkök, B. Reuderink, and M. Poel. "Improving BCI performance after classification." In: *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM. 2012, pp. 587–594 (cit. on pp. 61, 70).

[5] B. L. A. van de Laar, D. Plass-Oude Bos, B. Reuderink, M. Poel, and A. Nijholt. "How Much Control Is Enough? Influence of Unreliable Input on User Experience." In: *Cybernetics, IEEE Transactions on* 43.6 (Dec. 2013), pp. 1584–1592 (cit. on pp. 61, 63).

[6] D. Plass-Oude Bos, M. Poel, and A. Nijholt. "A study in user-centered design and evaluation of mental tasks for BCI." In: *Advances in Multimedia Modeling* (2011), pp. 122–134 (cit. on pp. 61, 72).

[7] A. Bashashati, M. Fatourechi, R. K. Ward, and G. E. Birch. "A survey of signal processing algorithms in brain-computer interfaces based on electrical brain signals." In: *Journal of Neural engineering* 4.2 (2007), R32 (cit. on pp. 61, 63–66, 71, 75, 77).

[8] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern classification*. John Wiley & Sons, 2012 (cit. on p. 61).

[9] G. Pfurtscheller, B. Z. Allison, C. Brunner, G. Bauernfeind, T. Solis-Escalante, R. Scherer, T. O. Zander, G. Müller-Putz, C. Neuper, and N. Birbaumer. "The hybrid BCI." In: *Frontiers in neuroscience* 4 (2010) (cit. on p. 62).

[10] L. Tonin, R. Leeb, M. Tavella, S. Perdikis, and J. del R. Millán. "The role of shared-control in BCI-based telepresence." In: *Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on*. IEEE. 2010, pp. 1462–1466 (cit. on p. 62).

[11] B. Z. Allison, R. Leeb, C. Brunner, G. R. Müller-Putz, G. Bauernfeind, J. W. Kelly, and C. Neuper. "Toward smarter BCIs: extending BCIs through hybridization and intelligent control." In: *Journal of neural engineering* 9.1 (2012), p. 7 (cit. on p. 62).

[12] R. Scherer, E. C. V. Friedrich, B. Z. Allison, M. Pröll, M. Chung, W. Cheung, R. P. N. Rao, and C. Neuper. "Non-invasive brain-computer interfaces: Enhanced gaming and robotic control." In: *Advances in Computational Intelligence*. Springer, 2011, pp. 362–369 (cit. on p. 62).

[13] A. S. Royer, A. J. Doud, M. L. Rose, and B. He. "EEG control of a virtual helicopter in 3-dimensional space using intelligent control strategies." In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 18.6 (2010), pp. 581–589 (cit. on p. 62).

[14] J. R. Wolpaw. "Brain-computer interfaces as new brain output pathways." In: *The Journal of Physiology* 579.3 (2007), pp. 613–619 (cit. on p. 62).

[15] L. A. Farwell and E. Donchin. "Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials." In: *Electroencephalography and clinical Neurophysiology* 70.6 (1988), pp. 510–523 (cit. on p. 62).

[16] D. J. Krusienski, E. W. Sellers, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw. "Toward enhanced P300 speller performance." In: *Journal of neuroscience methods* 167.1 (2008), p. 15 (cit. on p. 62).

[17] W. Speier, C. Arnold, J. Lu, R. K. Taira, and N. Pouratian. "Natural language processing with dynamic classification improves P300 speller accuracy and bit rate." In: *Journal of neural engineering* 9.1 (2012) (cit. on pp. 62, 77).

[18] M. M. Moore Jackson, S. G. Mason, and G. E. Birch. "Analyzing trends in brain interface technology: a method to compare studies." In: *Annals of biomedical engineering* 34.5 (2006), pp. 859–878 (cit. on p. 63).

[19] L. F. Nicolas-Alonso and J. Gomez-Gil. "Brain computer interfaces, a review." In: *Sensors* 12.2 (2012), pp. 1211–1279 (cit. on pp. 64, 77).

[20] J. D. Bayliss, S. A. Inverso, and A. Tentler. "Changing the P300 brain computer interface." In: *CyberPsychology & Behavior* 7.6 (2004), pp. 694–704 (cit. on p. 65).

[21] B. Blankertz, C. Schäfer, G. Dornhege, and G. Curio. "Single trial detection of EEG error potentials: A tool for increasing BCI transmission rates." In: *Artificial Neural Networks — ICANN 2002* (2002), pp. 138–138 (cit. on p. 65).

[22] B. Blankertz, G. Dornhege, C. Schäfer, R. Krepki, J. Kohlmorgen, K.-R. Müller, V. Kunzmann, F. Losch, and G. Curi. "Boosting Bit Rates and Error Detection for the Classification of Fast-Paced Motor Commands Based on Single-Trial EEG Analysis." In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 11.2 (2003), pp. 127–131 (cit. on p. 65).

[23] L. C. Parra, C. D. Spence, A. D. Gerson, and P. Sajda. “Response error correction-a demonstration of improved human-machine performance using real-time EEG monitoring.” In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 11.2 (2003), pp. 173–177 (cit. on p. 65).

[24] G. Schalk, J. R. Wolpaw, D. J. McFarland, and G. Pfurtscheller. “EEG-based communication: presence of an error potential.” In: *Clinical Neurophysiology* 111.12 (2000), pp. 2138–2144 (cit. on p. 65).

[25] C. W. Anderson, S. V. Devulapalli, and E. A. Stolz. “EEG signal classification with different signal representations.” In: *Neural Networks for Signal Processing V. Proceedings of the 1995 IEEE Workshop*. IEEE. 1995, pp. 475–483 (cit. on p. 65).

[26] A. Bashashati, R. K. Ward, and G. E. Birch. “A new design of the asynchronous brain computer interface using the knowledge of the path of features.” In: *Neural Engineering, 2005. Conference Proceedings. 2nd International IEEE EMBS Conference on*. IEEE. 2005, pp. 101–104 (cit. on p. 65).

[27] G. E. Birch, Z. Bozorgzadeh, and S. G. Mason. “Initial on-line evaluations of the LF-ASD brain-computer interface with able-bodied and spinal-cord subjects using imagined voluntary motor potentials.” In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 10.4 (2002), pp. 219–224 (cit. on pp. 65, 72).

[28] J. F. Borisoff, S. G. Mason, A. Bashashati, and G. E. Birch. “Brain-computer interface design for asynchronous control applications: improvements to the LF-ASD asynchronous brain switch.” In: *Biomedical Engineering, IEEE Transactions on* 51.6 (2004), pp. 985–992 (cit. on p. 65).

[29] M. Fatourechi, A. Bashashati, J. F. Borisoff, G. E. Birch, and R. K. Ward. “Improving the performance of the LF-ASD brain computer interface by means of genetic algorithm.” In: *Signal Processing and Information Technology, 2004. Proceedings of the Fourth IEEE International Symposium on*. IEEE. 2004, pp. 38–41 (cit. on p. 65).

[30] M. Fatourechi, A. Bashashati, R. K. Ward, and G. E. Birch. “A hybrid genetic algorithm approach for improving the performance of the LF-ASD brain computer interface.” In: *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*. Vol. 5. IEEE. 2005, pp. v–345 (cit. on p. 65).

[31] E. Gysels and P. Celka. “Phase synchronization for the recognition of mental tasks in a brain-computer interface.” In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 12.4 (2004), pp. 406–415 (cit. on p. 65).

[32] J. del R. Millán. "On the need for on-line learning in brain-computer interfaces." In: *Neural Networks, 2004. IEEE International Joint Conference on*. Vol. 4. IEEE. 2004, pp. 2877–2882 (cit. on p. 65).

[33] J. del R. Millán, F. Renkens, J. Mouriño, and W. Gerstner. "Non-invasive brain-actuated control of a mobile robot." In: *Proceedings of the 18th international joint conference on Artificial intelligence*. 2003, pp. 1121–1126 (cit. on p. 65).

[34] J. del R. Millán, F. Renkens, J. Mouriño, and W. Gerstner. "Noninvasive brain-actuated control of a mobile robot by human EEG." In: *Biomedical Engineering, IEEE Transactions on* 51.6 (2004), pp. 1026–1033 (cit. on p. 65).

[35] J. del R. Millán, F. Renkens, J. Mouriño, and W. Gerstner. "Brain-actuated interaction." In: *Artificial Intelligence* 159.1 (2004), pp. 241–259 (cit. on p. 65).

[36] J. del R. Millán, J. Mourino, M. G. Marciani, F. Babiloni, F. Topani, I. Canale, J. Heikkonen, and K. Kaski. "Adaptive brain interfaces for physically-disabled people." In: *Proc. 20th Annual Int. Conf. of the IEEE Engineering in Medicine and Biology Society (Hong Kong)*. Citeseer. 1998, pp. 2008–11 (cit. on p. 65).

[37] J. del R. Millán and J. Mouriño. "Asynchronous BCI and local neural classifiers: an overview of the adaptive brain interface project." In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 11.2 (2003), pp. 159–161 (cit. on p. 65).

[38] G. R. Müller-Putz, R. Scherer, G. Pfurtscheller, and R. Rupp. "EEG-based neuroprosthesis control: a step towards clinical practice." In: *Neuroscience letters* 382.1 (2005), pp. 169–174 (cit. on p. 65).

[39] W. D. Penny, S. J. Roberts, E. A. Curran, and M. J. Stokes. "EEG-based communication: a pattern recognition approach." In: *Rehabilitation Engineering, IEEE Transactions on* 8.2 (2000), pp. 214–215 (cit. on p. 65).

[40] S. J. Roberts, W. Penny, and I. Rezek. "Temporal and spatial complexity measures for electroencephalogram based brain-computer interfacing." In: *Medical & biological engineering & computing* 37.1 (1999), pp. 93–98 (cit. on p. 65).

[41] G. Townsend, B. Graimann, and G. Pfurtscheller. "Continuous EEG classification during motor imagery-simulation of an asynchronous BCI." In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 12.2 (2004), pp. 258–265 (cit. on p. 65).

[42] J. J. Vidal. "Real-time detection of brain events in EEG." In: *Proceedings of the IEEE* 65.5 (1977), pp. 633–641 (cit. on p. 65).

[43] I. Obeid and P. D. Wolf. “Evaluation of spike-detection algorithms fora brain-machine interface application.” In: *Biomedical Engineering, IEEE Transactions on* 51.6 (2004), pp. 905–911 (cit. on p. 65).

[44] G. Pfurtscheller, G. R. Müller-Putz, J. Pfurtscheller, and R. Rupp. “EEG-based asynchronous BCI controls functional electrical stimulation in a tetraplegic patient.” In: *EURASIP Journal on Applied Signal Processing* 2005 (2005), pp. 3152–3155 (cit. on p. 65).

[45] B. Hamadicharef. “Brain-Computer Interface (BCI) literature-a bibliometric study.” In: *Information Sciences Signal Processing and their Applications (ISSPA), 2010 10th International Conference on*. IEEE. 2010, pp. 626–629 (cit. on p. 67).

[46] R. B. Ashari, I. A. Al-Bidewi, and M. I. Kamel. “Design and simulation of virtual telephone keypad control based on brain computer interface (BCI) with very high transfer rates.” In: *Alexandria Engineering Journal* 50.1 (2011), pp. 49–56 (cit. on pp. 70, 75, 76).

[47] A. Ferreira, C. Almeida, P. Georgieva, and A. Tomé. “Person identification using VEP signals and SVM classifiers.” In: *Neural Networks (IJCNN), The 2010 International Joint Conference on*. IEEE. 2010, pp. 1–8 (cit. on pp. 70, 76).

[48] M. Manoochehri and M. H. Moradi. “The new post processing method for self-paced BCI system.” In: *Biomedical Engineering (ICBME), 2011 18th Iranian Conference of*. IEEE. 2011, pp. 152–155 (cit. on pp. 70, 76).

[49] S. Makeig, G. Leslie, T. Mullen, D. Sarma, N. Bigdely-Shamlo, and C. Kothe. “First demonstration of a musical emotion BCI.” In: *Affective Computing and Intelligent Interaction*. Springer, 2011, pp. 487–496 (cit. on pp. 70, 75).

[50] A. R. Satti, D. Coyle, and G. Prasad. “Self-paced brain-controlled wheelchair methodology with shared and automated assistive control.” In: *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2011 IEEE Symposium on*. IEEE. 2011, pp. 1–8 (cit. on p. 70).

[51] J. Jarzebowski, S. Lakshminarayan, and T. P. Coleman. “Using stochastic control with data compression perspectives to enhance P300 neural communication prostheses.” In: *Information Theory Workshop, 2008. ITW’08. IEEE*. IEEE. 2008, pp. 109–113 (cit. on pp. 70, 75, 76).

[52] D. Coyle, J. Garcia, A. R. Satti, and T. M. McGinnity. “EEG-based continuous control of a game using a 3 channel motor imagery BCI: BCI game.” In: *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2011 IEEE Symposium on*. IEEE. 2011, pp. 1–7 (cit. on p. 70).

[53] M. Fatourechi, R. K. Ward, and G. E. Birch. "Performance of a self-paced brain computer interface on data contaminated with eye-movement artifacts and on data recorded in a subsequent session." In: *Computational intelligence and neuroscience* 2008 (2008) (cit. on pp. 67, 70, 76).

[54] D. Heger, R. Jakel, F. Putze, M. Losch, and T. Schultz. "Filling a glass of water: Continuously decoding the speed of 3D hand movements from EEG signals." In: *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*. IEEE. 2012, pp. 4095–4098 (cit. on p. 70).

[55] Y. Lee, H. Lee, J. Kim, H. C. Shin, and M. Lee. "Classification of BMI control commands from rat's neural signals using extreme learning machine." In: *Biomedical engineering online* 8.1 (2009), pp. 1–10 (cit. on pp. 70, 71, 76).

[56] N.-Y. Liang, P. Saratchandran, G.-B. Huang, and N. Sundararajan. "Classification of mental tasks from EEG signals using extreme learning machine." In: *International Journal of Neural Systems* 16.01 (2006), pp. 29–38 (cit. on pp. 70, 75, 76).

[57] S. Perdikis, H. Bayati, R. Leeb, and J. del R. Millán. "Evidence accumulation in asynchronous BCI." In: *Int. J. Bioelectromagnetism* 13.3 (2011), pp. 131–132 (cit. on pp. 70, 71, 76).

[58] R. Poli, M. Salvaris, and C. Cinel. "A genetic programming approach to the evolution of brain-computer interfaces for 2-D mouse-pointer control." In: *Genetic Programming and Evolvable Machines* 13.3 (2012), pp. 377–405 (cit. on pp. 70, 71, 76).

[59] A. Temko, G. Lightbody, E. M. Thomas, G. B. Boylan, and W. Marnane. "Instantaneous measure of EEG channel importance for improved patient-adaptive neonatal seizure detection." In: *Biomedical Engineering, IEEE Transactions on* 59.3 (2012), pp. 717–727 (cit. on pp. 70, 71).

[60] J. Verwaeren, W. Waegeman, and B. De Baets. "Learning partial ordinal class memberships with kernel-based proportional odds models." In: *Computational Statistics & Data Analysis* 56.4 (2012), pp. 928–942 (cit. on p. 70).

[61] D.-M. Dobrea, M.-C. Dobrea, and M. Costin. "An EEG coherence based method used for mental tasks classification." In: *Computational Cybernetics, 2007. ICCC 2007. IEEE International Conference on*. IEEE. 2007, pp. 185–190 (cit. on pp. 70, 73, 75, 76).

[62] M. Duvinage, T. Castermans, T. Hoellinger, and J. Reumaux. "Human walk modeled by PCPG to control a lower limb neuroprosthesis by high-level commands." In: *Proceedings of the 2nd International Multi-Conference on Complexity, Informatics and Cybernetics (IMCIC'11)*. 2011 (cit. on pp. 70, 73).

[63] H. Verschore, P.-J. Kindermans, D. Verstraeten, and B. Schrauwen. "Dynamic stopping improves the speed and accuracy of a p300 speller." In: *Artificial Neural Networks and Machine Learning – ICANN 2012*. Springer, 2012, pp. 661–668 (cit. on pp. 70, 75, 76).

[64] T. Liu, L. Goldberg, S. Gao, and B. Hong. "An online brain-computer interface using non-flashing visual evoked potentials." In: *Journal of neural engineering* 7.3 (2010), p. 036003 (cit. on pp. 70, 76).

[65] I. Martinovic, D. Davies, M. Frank, D. Perito, T. Ros, and D. Song. "On the feasibility of side-channel attacks with brain-computer interfaces." In: *21st USENIX Security Symp*. 2012 (cit. on p. 70).

[66] U. Orhan, K. E. Hild, D. Erdogmus, B. Roark, B. Oken, and M. Fried-Oken. "RSVP keyboard: An EEG based typing interface." In: *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE. 2012, pp. 645–648 (cit. on pp. 70, 75).

[67] A. Rakotomamonjy, R. Flamary, G. Gasso, and S. Canu. "lp-lq Penalty for Sparse Linear and Sparse Multiple Kernel Multitask Learning." In: *IEEE transactions on neural networks* 22.8 (2011), pp. 1307–1320 (cit. on p. 70).

[68] H. Riechmann, N. Hachmeister, H. Ritter, and A. Finke. "Asynchronous, parallel on-line classification of P300 and ERD for an efficient hybrid BCI." In: *Neural Engineering (NER), 2011 5th International IEEE/EMBS Conference on*. IEEE. 2011, pp. 412–415 (cit. on pp. 70, 73).

[69] H. Zhang, C. Guan, K. K. Ang, and W. Chuanchu. "BCI competition IV–data set I: learning discriminative patterns for self-paced EEG-based motor imagery detection." In: *Frontiers in neuroscience* 6.7 (2012), pp. 1–7 (cit. on pp. 70, 76).

[70] T. Zoughi and R. Boostani. "Presenting a Combinatorial Feature to Estimate Depth of Anesthesia." In: *World Academy of Science, Engineering and Technology, International Science Index 3* 4.1 (2010), pp. 660–664 (cit. on pp. 70, 71).

[71] B. A. S. Hasan and J. Q. Gan. "Hangman BCI: An unsupervised adaptive self-paced Brain–Computer Interface for playing games." In: *Computers in biology and medicine* 42.5 (2012), pp. 598–606 (cit. on pp. 70, 71, 73, 75).

[72] L. Jun. “Adaptive Detection of Idle State in Motor Imagery Based Brain Computer Interface.” In: *Intelligent Computation Technology and Automation (ICICTA), 2010 International Conference on*. Vol. 1. IEEE. 2010, pp. 417–420 (cit. on pp. 70, 72, 76).

[73] R. Leeb, V. Settgast, D. Fellner, and G. Pfurtscheller. “Self-paced exploration of the Austrian National Library through thought.” In: *International Journal of Bioelectromagnetism* 9.4 (2007), pp. 237–244 (cit. on pp. 70, 71, 75).

[74] T. Solis-Escalante, G. Müller-Putz, C. Brunner, V. Kaiser, and G. Pfurtscheller. “Analysis of sensorimotor rhythms for the implementation of a brain switch for healthy subjects.” In: *Biomedical Signal Processing and Control* 5.1 (2010), pp. 15–20 (cit. on pp. 70, 71).

[75] M. Fatourechi, A. Bashashati, G. E. Birch, and R. K. Ward. “Automatic user customization for improving the performance of a self-paced brain interface system.” In: *Medical and Biological Engineering and Computing* 44.12 (2006), pp. 1093–1104 (cit. on p. 67).

[76] M. Fatourechi, S. G. Mason, G. E. Birch, and R. K. Ward. “Is Information Transfer Rate a Suitable Performance Measure for Self-paced Brain Interface Systems?” In: *Signal Processing and Information Technology, 2006 IEEE International Symposium on*. IEEE. 2006, pp. 212–216 (cit. on pp. 67, 75).

[77] B. A. S. Hasan and J. Q. Gan. “Unsupervised movement onset detection from EEG recorded during self-paced real hand movement.” In: *Medical & biological engineering & computing* 48.3 (2010), pp. 245–253 (cit. on p. 71).

[78] B. A. S. Hasan and J. Q. Gan. “Temporal modeling of EEG during self-paced hand movement and its application in onset detection.” In: *Journal of Neural Engineering* 8.5 (2011), p. 056015 (cit. on p. 71).

[79] Y. Lee, H. Lee, Y. Lang, J. Kim, M. Lee, and H. C. Shin. “Classification of BMI Control Commands Using Extreme Learning Machine from Spike Trains of Simultaneously Recorded 34 CA1 Single Neural Signals.” In: *Experimental Neurobiology* 17.2 (2008), pp. 33–39 (cit. on pp. 71, 75).

[80] T. Zoughi, R. Boostani, and M. Deypir. “A wavelet-based estimating depth of anesthesia.” In: *Engineering Applications of Artificial Intelligence* 25.8 (2012), pp. 1710–1722 (cit. on p. 71).

[81] R. Poli, M. Salvaris, and C. Cinel. “Evolution of a brain-computer interface mouse via genetic programming.” In: *Genetic Programming*. Springer, 2011, pp. 203–214 (cit. on p. 71).

[82] A. Temko, W. Marnane, G. Boylan, and G. Lightbody. “A Data-Driven Energy Based Estimator of EEG Channel Importance for Improved Patient-Adaptive Neonatal Seizure Detector.” In: *World Congress*. Vol. 18. 1. 2011, pp. 13770–13775 (cit. on p. 71).

[83] S. Perdikis, M. Tavella, R. Leeb, and J. del R. Millán. “Feedback controller for mental imagery BCI.” In: *Proc. TOBI Workshop 2010: Integrating Brain–Computer Interfaces with Conventional Assistive Technology*. 2010, p. 71 (cit. on p. 71).

[84] R. J. K. Jacob. “Eye movement-based human-computer interaction techniques: Toward non-command interfaces.” In: *Advances in human-computer interaction* 4 (1993), pp. 151–190 (cit. on p. 72).

[85] M. M. Moore. “Real-world applications for brain-computer interface technology.” In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 11.2 (2003), pp. 162–165 (cit. on p. 72).

[86] G. R. Müller-Putz, C. Breitwieser, F. Cincotti, R. Leeb, M. Schreuder, F. Leotta, M. Tavella, L. Bianchi, A. Kreilinger, A. Ramsay, M. Rohm, M. Sagebaum, L. Tonin, C. Neuper, and J. del R. Millán. “Tools for brain-computer interaction: a general concept for a hybrid BCI.” In: *Frontiers in neuroinformatics* 5 (2011) (cit. on p. 73).

[87] H. Gürkök and A. Nijholt. “Brain–computer interfaces for multimodal interaction: a survey and principles.” In: *International Journal of Human-Computer Interaction* 28.5 (2012), pp. 292–307 (cit. on p. 73).

6 POST-PROCESSING GUIDELINES

Chapter 5 showed how helpful post-processing methods can be in improving control and reducing effort. To encourage deliberate reasoning about which methods to apply, this chapter provides an extended overview of post-processing methods, with guidelines for their application.

This chapter provides a description of various post-processing methods, and guidelines for when they could be useful. I also discuss the three most important considerations when applying them, provide an example case that shows how the guidelines could be applied, and end with the answers to some frequently asked questions.

The methods listed here come partly from the findings of the literature review discussed before, but also from discussions with experts in various fields (machine learning, control theory, interaction design, and gaming). The uncovered post-processing methods have been structured into a guide to help find suitable methods for a given system. I will refer to this as *the guidelines*. You can find this guide in Figure 19. The purpose of these guidelines is to show what kind of methods are available and when they are useful, in order to provide new tools to the toolbox of BCI developers.

It is important to note that the overview in this chapter is by no means complete. As already indicated, these methods have been derived from various fields. Although this is the most complete listing I have personally found so far, there are bound to be more post-processing methods than those listed here. It is also possible to develop new methods. Hence the reminder in the figure description to "Be creative!".

The methods in this overview have been split into three categories: value distribution, behaviour, and application levels. This division has to do with the point of view from which the logical control values (the result from the feature translation step) are approached.

Value distribution level This level is about redistributing or normalizing data so they become more meaningful. For example, a relaxation value of 0.0 obtains its meaning from knowing the range, whether it is from 0.0 to 1.0, or from -1.0 to +1.0.

Behavioural level How should the input behave to be appropriate for controlling a specific application? How can we make it less (or more) sensi-

POSTPROCESSING GUIDELINES

VALUE DISTRIBUTION LEVEL

Is the input unevenly distributed? **Debiasing**

Is the input not in the right range? **Normalization**

BEHAVIOUR LEVEL

Is the control signal too inaccurate? ***Multimodal fusion***

CONTINUOUS CONTROL (JOYSTICK)

Is the signal too sensitive to outliers?

- Is the control non-stop? **Smoothing**
- Is the control intermittent? **Voting**
  - Optimize number of repetitions? ***Repetition threshold***

DISCRETE CONTROL

Is the control signal activated too easily? **Dwell time**

Is the control signal reactivated too quickly? **Cooldown**

Is it difficult to set an optimal threshold position? ***Adaptive threshold***

SWITCH BEHAVIOUR **Threshold, rectangular wave**

Is the control signal too unstable? **Multiple thresholds**

Does providing input take too much effort? **Toggle button**

See button behaviour.

BUTTON BEHAVIOUR **Threshold, pulse wave**

Should the control pulse repeat when 'held'? **Repeat**

Is the control signal too sensitive to outliers? **Voting**

- Optimize number of repetitions? ***Repetition threshold***

APPLICATION LEVEL

Can you modify the interface? ***Dynamic interface adjustment***

Does context determine available actions? ***Context actions***

Are next actions already determined? ***Auto-completion***

Are previous actions determined by current? ***Auto-correction***

FIGURE 19: **Post-processing guidelines**

FIGURE 19: (To the left) **Post-processing guidelines.** For three different levels of application, questions concerning possible issues will guide you to a potentially helpful post-processing method or method category. Methods in italics may require external information, such as application context. These guidelines are meant to inspire, not to restrict. Be creative!

tive? This level is divided in *continuous control* (like a joystick), and *discrete control* (like switches and buttons)[1]. Different behaviour options have different potential problems and different accompanying solutions.

Application level Application-level methods take into account the meaning of the input as control commands in the context of the application. The logic of the application creates restrictions on the possible or likely control commands.

6.1 METHOD DESCRIPTIONS

There is quite some overlap between this section, and the method descriptions that are part of the literature review in Chapter 5. The descriptions in this section, however, are meant to accompany the guidelines in this chapter, and have been organized accordingly.

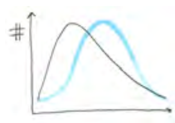
VALUE DISTRIBUTION LEVEL

Is the input unevenly distributed? .... **Debiasing**

Is the input not in the right range? .... **Normalization**

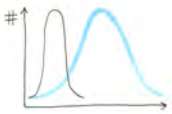
When trying to assign meaning to sensors, these values are often required (or at least expected) to occur within a certain range and with a certain probability distribution. For example, when the value is an indication of relaxation, it could be convenient if it were in the range from 0 to 1 for all users. Then the interpretation could be made that 0 means no relaxation at all is observed, and 1 indicates complete relaxation. Another reason for a fixed range and distribution is so you can apply a meaningful fixed threshold. For optimal functional separation, the following methods are best implemented in the control interface.

See Section 2.2.



Debiasing When the input is not evenly distributed — for example when

1 The terms *continuous* and *discrete control* should not be confused with *non-stop* and *intermittent control*, the equivalents in the time domain. Also, buttons on modern gaming controllers and some high-end keyboards are analogue, so they provide continuous output instead of discrete.



the brain activity characteristics during training are different from during actual use [1, 2] — a transform can adjust this. One of the most simple examples is adding an offset to adjust the central tendency of the input values.

Normalization When the input is not in the right range, again a transform can fix this. Transforms to adjust the *span* and *zero* (the lowest possible value) also affect the distribution, and may affect the central tendency, so some *debiasing* may be inherent. Instead of a simple linear transformation $ax + b$, you could, for example, use a sigmoid function, which provides smooth boundaries on the outer ends of the span.

See also Section 2.5.

AlphaWoW uses adaptive z-score normalization (subtract the mean, divide by the standard deviation, with the mean and standard deviation based on recent observations) to automatically adjust the system to the user, and to prevent the user from getting stuck in one state or the other [3].

BEHAVIOUR LEVEL

Is the control signal too inaccurate? ······················ ***Multimodal fusion***

Post-processing forms the bridge between the detections and the controls required by the application. When we start to reason about the way the application input should behave, we enter behaviour-level reasoning. Like the value-distribution-level methods, the behaviour-level methods are generally best implemented in the control interface.

Behaviour-level methods have been split up into methods that result in continuous control and those for discrete control. Discrete control is again subdivided into methods that are applicable for switch-like behaviour and methods for button-like behaviour.

Multimodal fusion If additional information is available through other input modalities or sensors, this information can be fused with the brain-based detections, to arrive at more accurate interpretations of the input from the user. For example, eye tracking information could be combined with P300 detection to determine what the user is looking at. The strengths are combined to cover the weaknesses of each individual input (dependence on good lighting conditions and low speed). This would be *decision-level fusion*, based on feature translation output. This is a post-processing method, as opposed to data-level fusion (based on pre-processing output) or feature-level fusion (based on feature extraction output). Multimodal BCIs are also referred to as “hybrid BCIs” in the BCI community [4]. Read Gürkök and Nijholt [5] to learn more about multimodal BCI principles.

CONTINUOUS CONTROL (JOYSTICK)

Is the signal too sensitive to outliers?

Is the control non-stop? .... **Smoothing**

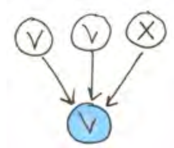
Is the control intermittent? .... **Voting**

Optimize number of repetitions?

.... ***Repetition threshold***

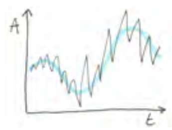
One clear split in input behaviour is whether the control should consist of continuous values (over a range, like a joystick) or whether the control should be discrete (from a selection of options, such as 'on' and 'off') in the case of a button or switch. Here are the methods that are relevant for continuous control.

Voting When the control is intermittent, voting can be used to combine multiple repetitions of the input into one control signal. This reduces sensitivity to outliers, at the cost of increasing the amount of time required to issue one control command (See the trade-offs in Section 6.2). One of the simplest ways to implement voting is by *averaging*. We used voting to combine multiple classifications of covert attention in Wild Photoshoot [6], and to combine SSVEP classifications in Bacteria Hunt [7].



Repetition threshold When you combine multiple repetitions through voting, the exact number of repetitions can be optimized with a repetition threshold, set to a certain probability level for example [8]. See also trade-offs mentioned in Section 6.2.

Smoothing When the control should be continuous, but also non-stop (as opposed to intermittent), then smoothing is the way to reduce sensitivity to brief disturbances. Smoothing is often applied to a sliding sample window, with a set length and interval. Instead of gathering a set of samples which then result in one new output, as with voting, each new input can result in a new output if the interval for the moving window is one sample. This allows for non-stop system response and feedback. As the name implies, the result is a more smooth version of the signal. Liang *et al.* showed how smoothing significantly improved mental task classification for three different classifiers [9].



There are many methods that provide a smoothing effect, and consequently there are also many different words used in papers to indicate smoothing: averaging, antialiasing, low-pass filter, or even simply 'filter'. In online BCI systems, you can only use causal filters, which take into account just past and current samples, not samples from the future. For a comparison of a number of smoothing implementations, see Figure 20.

DISCRETE CONTROL

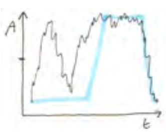
Is the control signal activated too easily? ················ **Dwell time**

Is the control signal reactivated too quickly? ··········· **Cooldown**

Is it difficult to set an optimal threshold position? ······ ***Adaptive threshold***

When the application asks for discrete control, or perhaps the mental task input is most suitable to provide this type of input, there are basically two options: switch-like behaviour or button-like behaviour. The difference is that with a switch, it is the state that matters. The control signal is a rectangular wave. With button behaviour, it is about the activation, represented by a pulse wave. Compare a light switch with typing 'A'. Both types of control are often the result of applying one or more thresholds to a continuous signal. Such a discretization of continuous values into a set of classes is also called *binning*. Before going into switch and button-specific methods, I will first detail three methods that are applicable for discrete control in general.

Cooldown Adding a cooldown period, also known as a refractory period, is a form of *debouncing*. When a threshold is crossed again too quickly, causing false positives, re-activation can be suppressed for a set time period. The downside of this method, is that it increases re-activation time [10, 11]. While most-often used in combination with button behaviour, it can also be applied to prevent quick state changes in the case of switch behaviour. In this case it would be the state change action that is on cooldown, instead of just activation (going from off to on). A BCI example using this method is the system from Pfurtscheller et al. [12], which allows a paralyzed patient to grasp a drink with his hand. Cooldown prevented switching between the different grasping states too quickly.



Dwell time When the threshold is set to a good value considering the value distribution, but the threshold is still crossed too easily, you can add a dwell time. The threshold then has to be crossed for a set time duration before the control signal becomes active [13, 3]. This method also has a *debouncing* effect. The control becomes more deliberate, but also takes more time (Again, see the part on trade-offs in Section 6.2). When applied to switch behaviour, the user has to remain in the state range for the (possibly state-specific) dwell duration in order to activate the state.

From eye movement-based control we know that the exact dwell duration is critical to the flow of interaction [14]. For brain-computer interaction, the optimal duration is likely to be different and will probably depend on the specific mental tasks used for input. By applying a different dwell time for each state, dwell time allows for separate finetuning of the number of false positives and negatives.

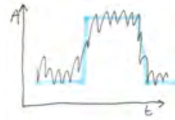
Adaptive threshold Thresholds can either be fixed or adaptive. In the adaptive case, aside from the current value, additional information can be taken into account to make it either easier or more difficult to cross the threshold. For example, when a Bayesian classifier indicates a high confidence in the output, the threshold could be lowered making it easier to cross. Or when application context makes it less likely for the user to provide certain input, the threshold for that input could be increased. When such an adjustment takes place on the application side, it is a form of *dynamic interface adjustment*; see the dynamic interface adjustment paragraph further on.

SWITCH BEHAVIOUR ········ **Threshold, rectangular wave**

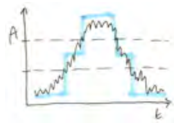
Is the control signal too unstable? ········ **Multiple thresholds**

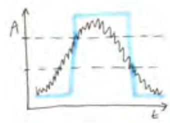
Does providing input take too much effort? ········ **Toggle button**

See button behaviour.

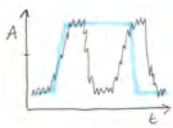


When a threshold results in rectangular wave output, dividing the output into specific states, the result is switch-like behaviour.





Multiple thresholds If the switch-like behaviour resulting from the application of one threshold is too unstable, multiple thresholds or hysteresis can be used to *debounce* the control signal. Multiple thresholds normally create a deadband in between the thresholds in which no action is triggered [15]. This ‘no action’ is often referred to as the ‘no control state’ or ‘idle state’. With *hysteresis*, the previous state is maintained instead of this ‘no action’ state [3]. As the upper threshold can be set at a different level than the lower threshold, such methods allow you to fine-tune the expected number of false alarms and misses independently; see the part on trade-offs in Section 6.2.



Toggle button Maintaining a certain input can take a lot of effort. This can be reduced with the toggle button strategy for state selection. The input then only requires a short activation, like a button. The resulting pulse wave then toggles between the states. Section 6.3 provides an example of how this method could be applied. When converting to this toggle button strategy, do not forget to check out the methods that might be applicable to button behaviour.

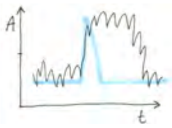
BUTTON BEHAVIOUR ······················ **Threshold, pulse wave**

Should the control pulse repeat when 'held'? ········ **Repeat**

Is the control signal too sensitive to outliers? ······· **Voting** ··········

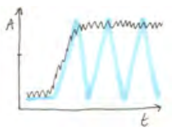
Optimize number of repetitions?

·········· ***Repetition threshold***



Instead of a rectangular wave, a threshold can also result in a pulse wave, which is activated when the threshold is crossed upwards. This creates button-like behaviour. It is just the activation that matters, as opposed to staying within a certain value range, as with the switch.

Voting, repetition threshold Button behaviour is intermittent, and therefore also suitable for voting to make the control signal less sensitive to outliers. And, as we saw with the application of voting for continuous control signals, we can again apply a repetition threshold to adaptively optimize the number of repetitions based on some indication of confidence.



Repeat The downside of button behaviour is that it only responds when the user crosses the threshold upwards, when going from low to high, or inactive to active. To quickly trigger two times in a row, the user has to go back to inactivity before becoming active again. This can be made more effortless by repeating the pulse when the user remains active for a certain duration. This is the same as pressing and holding the right arrow key to move a cursor in text totally to the right, as opposed to having to press and release it repeatedly.

APPLICATION LEVEL

Can you modify the interface? ················ ***Dynamic interface adjustment***

Does context determine available actions? ·············· ***Context actions***

Are next actions already determined? ····················· ***Auto-completion***

Are previous actions determined by current? ············ ***Auto-correction***

If you have access to the application, just the application state, or can otherwise model some aspects that restrict control possibilities, the following methods can be useful. Whereas the other methods are often best to implement in the control interface, application-level methods are often so interwoven with the application that it is best to implement them in the application directly. Examples of *shared control* often fall into one of the following method categories.

Dynamic interface adjustment If you can modify the application interface, you can use probabilistic knowledge about the input related to the application state to make it easier or more difficult to execute certain actions. Functionally, this method is very much related to the adaptive threshold we saw in the discrete control section. The difference is that we can now use knowledge from the application, and use it to induce changes again in the application. Menu items can be reordered to make more likely targets faster to select. Unlikely targets can even be left out all together [16]. Other ways to make more likely targets easier to select is by representing them with a larger surface making them easier to hit [17], or by creating a gravitation towards them, such as with *snapping*, which is comparable to the dynamic used in Hex-o-Spell [18].

Context actions Often, application context determines which actions are available to the user. In a role-playing game, for example, there is often one generic 'do'-action, which will trigger a conversation when you are next to a friend, and which will make your avatar fight when next to an enemy [13]. This means that a limited number of well-detectable inputs can be used to trigger a wide variety of actions, based on the given context. For more information, see Baldauf, Dustdar, and Rosenberg [19], which discusses various approaches used in existing context-aware systems and presents a design framework. Another example is that of a smart wheelchair, which ignores any commands that lead to a direction where there is no opening [20].

Auto-completion If there are situations where one action determines the next couple of actions, these following actions can automatically be executed. This is known as auto-completion (a function offered by many text editors), but also as the term macro. Auto-completion is often based on language models, such as simple probabilistic models generated from n-grams. Ashari, Al-Bidewi, and Kamel [21] show a combination of auto-completion with adaptive menu adjustment based on the longest-common subsequence in telephone numbers. This method is somewhat related to the application-level decision of whether to use *goal selection* ("Go to the kitchen") or *process control* ("Move one step forward") [22].

Auto-correction As it becomes clear from recent actions that a past action is likely incorrectly interpreted, auto-correction can retro-actively fix that past action. Application of this method does require the possibility to *roll back* the application state to the point of the fix. Auto-correction is also referred to as instant error correction and label correction. Manoochehri and Moradi [23] shows how this can both decrease the false positive rate and increase the true positive rate.

6.2 BEYOND THE GUIDELINES

The guide provides inspiration as to which methods might be suitable for a given system. In this section, I will discuss three other important considerations that come into play when selecting and applying post-processing methods, to give an indication of some of the aspects that are not touched upon by the guidelines.

BE AWARE OF TRADE-OFFS

Every method has its downsides. The most common side-effects are those of the methods that increase accuracy at the cost of effort, mostly in the terms of time [24]. A common example is a P300 speller where multiple P300 repetitions are combined to arrive at one character selection [25]. The more repetitions, the more certain the system will be about what letter the user wants to select. On either side of this *speed-accuracy trade-off*, there is a point where users will give up: either when the accuracy is too low, or when the necessary amount of time to provide one application input (in the case of the speller: one letter) is too high (see also Chapter 4). The sweet spot seems to be where the user can get something done with minimal overall effort (in the context of active BCI input).

A similar issue is the *delay-accuracy trade-off*. Smoothing decreases sensitivity to outliers, which can significantly improve accuracy [9], but comes at the cost of a delay in the signal (see also Figure 20). Although a delay does not require you to provide the input multiple times, it can increase the necessary effort in multiple ways. A delay in feedback on the user input when the input is not perfectly detected either requires the user to wait for confirmation, or can result in a sequence of incorrect commands which will then need to be corrected. This can already be experienced with keyboard input when the system is busy doing something else. At a certain point the receiving program will catch up on what you did, and if everything was typed correctly, there does not seem to be a significant problem in terms of effort. But even then, this lack of intermediate feedback demands additional mental effort. “Provide feedback”, later rephrased as “Visibility of system status”, is one of the core usability heuristics for this reason [26]. Feedback is considered so important that it is number one on Nielsen’s list of Ten usability heuristics [27].

In the experiment described in Chapter 3 we saw that users prefer mental input tasks that are easy to execute. However, if the user concentrates less on the input task, the resulting brain activity may also be less pronounced, resulting in detections that are less accurate [28]. Again, we see a trade-off, this time between concentration and accuracy. Although this specific example is at the level of the input task, and not at the control interface, the control interface can affect the level of concentration as well, by keeping the amount of time the user has to repetitively perform the input task to a minimum. An example of a helpful method in this area is the toggle button to provide switch-like input to an application.

Another typical trade-off is that of *misses* versus *false alarms* (or: false negatives and false positives), when it is the result of the position of a threshold to translate continuous data into discrete categories. Say we apply a threshold to be able to say whether the user is concentrated. A high value yields a positive classification (yes, the user is concentrating), and a low value a negative classification (the user is not concentrating). If a threshold is set higher, then the number of hits (true positives) and also the number of *false alarms* (false positives) will decrease. But it will increase the number of true and false negatives (correct rejections and misses). Then the question becomes: what would be worse? The answer depends on the application. A first-person shooter is an example that could go either way. If you do not shoot, you will definitely miss your target. How detrimental false alarms, such as shooting when you do not want to, are depends on how rare ammunition is in the game, and on how important it is for the gameplay to keep your position hidden. Hysteresis, for example, can provide a way to set separate thresholds for activation and inactivation.

There may be more trade-offs along these lines. Based on the examples mentioned above, a useful guiding principle seems to be to assess the effect of the choice on the overall required user effort.

IMPLEMENTATION MATTERS

When you know which method categories can be applied, it is still very important to consider which exact implementation to use. For example, to implement smoothing you could use a moving average, exponential smoothing, or a causal Savitzky-Golay filter. Refer to Figure 20 to see the difference in results. In this case, the mean square error of the Savitzky-Golay filter is half that of the moving average smoother (0.26 versus 0.54).

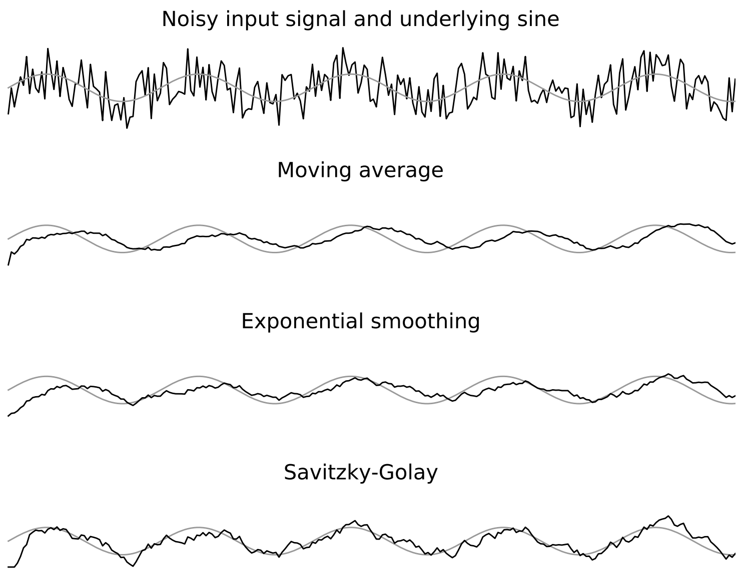


FIGURE 20: **Comparison of smoothing method implementations.** At the top a noisy input signal (240 samples) in black, with the underlying sine wave in grey. In the plots below, the sine has been repeated in grey so it is easy to see the delay caused by the different methods. Moving average (window size is 20 samples) causes the most delay. Then there is exponential smoothing, with the contribution of the new sample set to 0.1. The winner is the causal version of the Savitzky-Golay filter (window size is 21 samples, polynomial order 2, return 17th sample (at 4/5th of the window)). This method can be very responsive.

THE POWER OF COMBINING METHODS

About half of the systems in the literature study of the previous chapter use only one post-processing method. It will, however, be quite unlikely that applying just one method is enough to obtain the desired behaviour for control. Figure 21 shows the power of the combination of threshold, cooldown, and dwell time, which is used by two of the systems in the literature review [29, 30]. Further research is necessary to uncover other valuable combinations that result in behaviour that could be demanded by applications.

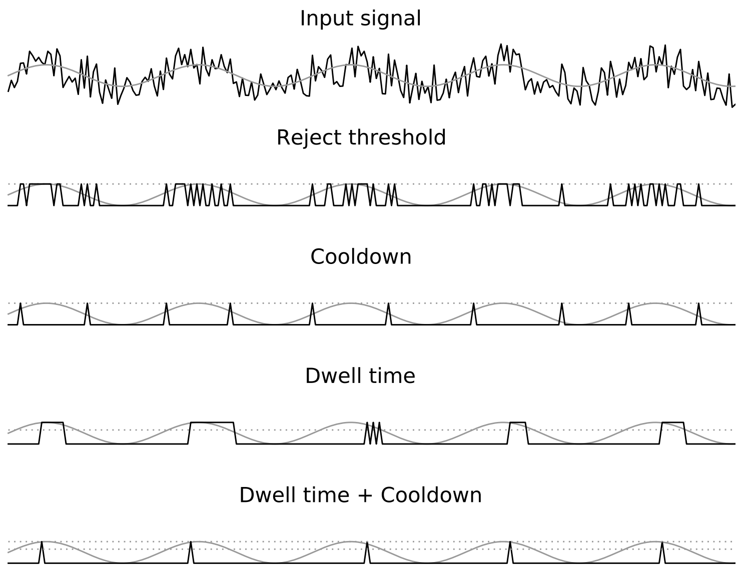


FIGURE 21: **Transforming irregular input to one activation per underlying sine peak.** A high threshold creates a discrete output with a low false positive rate. Adding a long cooldown period is another big step towards creating the control signal behaviour we are looking for, but there is no obvious alignment with the underlying sine peaks. Combining the cooldown with dwell time will result in purposeful interaction, with one activation around each peak.

6.3 EXAMPLE: PAX BRITANNICA

How well the prototype designed in this section holds up in practice is investigated in Chapter 7.

Post-processing transforms the detected brain-based inputs into control commands for the application. Therefore, before I demonstrate the application of the guidelines, first I will explain the task the user will use as input, and the application that will be controlled with it.

USER TASK

For more information on the Emotiv EPOC headset and software, see Section 7.1.

In this case, the brain-based inputs come from the Emotiv EPOC, and are the result of a two-class classification: *rest* versus *hand-tapping* with the main hand. The user's *active* 'mental' task was purposefully chosen to be actual movement, as it is easier to detect than imaginary movement [31], and provides a ground truth for determining the accuracy of the system objectively. For the *rest* task, users were instructed to simply sit in a relaxed manner. During training, users could look at the box shown in the Cognitiv suite training tab of Emotiv's Control Panel.

APPLICATION SOFTWARE

The application that will be controlled through this pair of mental tasks is Pax Britannica, see Figure 22. This is a one-button strategy game, developed by *No Fun Games*[2]. As this game requires just one button for control and its code is easy to adjust (scripted in LUA), it is a very suitable test application for brain-computer interfaces.

Each player is represented by a factory ship which can build smaller ships. These smaller ships can fight and ultimately destroy the opponent ship. The last factory ship left standing wins. What ships can be built depends on the number of resources the factory ship has accumulated. These resources are automatically gathered over time, and spent immediately when a ship is built. Building is instant.

With the standard keyboard controls, holding down the button spins the needle on the radial menu in the middle of the factory ship (see Figure 23). When the button is released, the factory ship will build the option corresponding to the quadrant the needle is pointing to at that time. The four build options, one for each quadrant, are: fighter (effective against bombers), bomber (effective against frigates and factory ships), frigate (against fighters), and upgrade (allows for faster accumulation of resources, so the player can build ships more quickly in the future).

Aside from these descriptions concerning the control of the game, it is important to keep in mind that this is a relatively fast-paced game (one

2 The game is freely available at `http://paxbritannica.henk.ca/`.

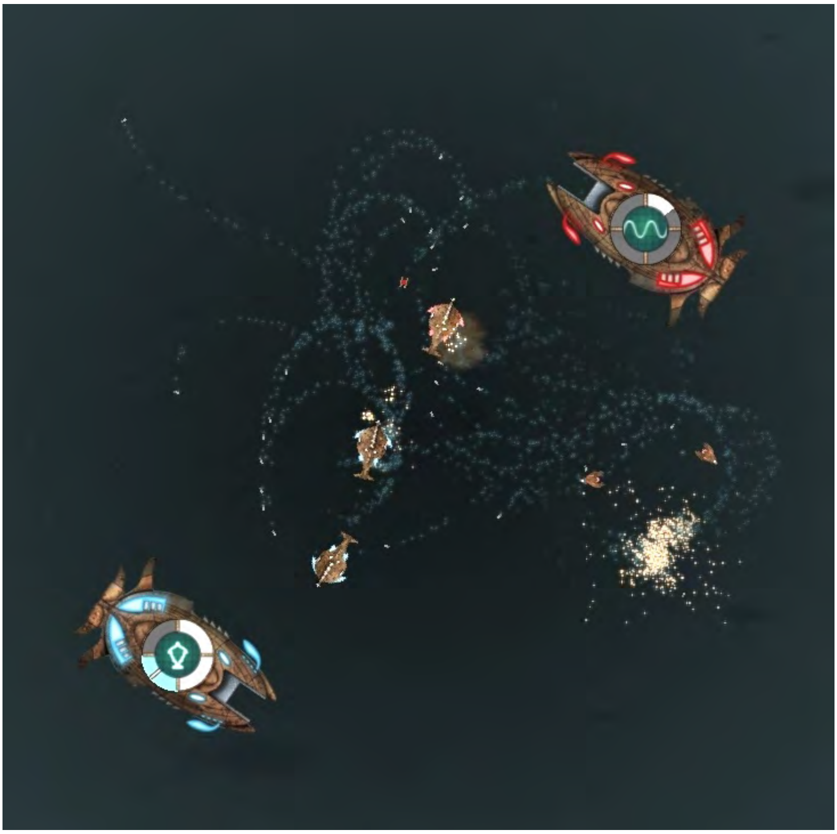


FIGURE 22: **Pax Britannica, a one-button real-time strategy game by *No Fun Games*.** The two big ships on the screen are the *factory ships*. Each colour (here, red and blue) represents one *player*. In the centre of each factory ship is a *selection wheel*. When the player is not in the process of selecting anything, the *selection arrow* is still at the top of the wheel. *Charge* is being built up during this time, shown here in white. The more charge, the faster the selection arrow will move along the wheel. The blue ship shows the selection arrow in the third quartile of the circle. If the player would release now, she would build the ship depicted in the centre of the selection wheel. The four different options are: *fighter*, *bomber*, *frigate*, and *upgrade* to make the factory ship charge faster. Fighters beat bombers. Bombers beat frigates and factory ships. Frigates beat fighters. You can see the video game in action here: `http://www.youtube.com/watch?v=UYZ1GkyvB2s`.

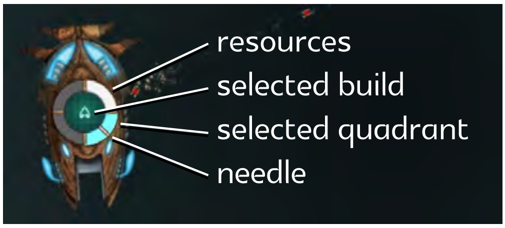


FIGURE 23: **The radial menu in the centre of the player's factory ship**, annotated with the indications for the amount of resources, the selected quadrant and selected build, and the selection needle

game round takes only two to three minutes), and time is important. Timing determines what is built, and spending time doing nothing is a sure strategy for losing.

METHOD SELECTION

The post-processing guidelines will provide an initial indication of which methods could be most suitable given this input pair and the one-button game control.

Value distribution If the output of the Emotiv EPOC is not suitably distributed, *debiasing* and *normalization* could be used to fix this.

The upper plot in Figure 24 shows the distribution of Emotiv EPOC output values. This plot is generated from data of three participants in a small test run prior to the actual experiment, based on 9, 12, and 15 training sets. One training set consists of 30 seconds *rest* and 32 seconds *active* task execution, see Section 7.1. There is not much difference between this overall distribution and the individual distributions for each participant.

This distribution shows a clear bias for the 0.0 bin ([0.00-0.05>), which can be problematic when trying to translate these values into application control commands. Attempts to split up this lower bin into multiple smaller bins did not achieve a more even distribution for these lower values. The strong bias for 0.0 remained. However, the Emotiv Control Panel does provide a sensitivity setting. Adjusting this sensitivity to one 'tick' higher, resulted in a more equal distribution, see the lower plot in Figure 24. This second plot is based on just one participant. However, given that the previous distributions hardly differed between participants, we can assume that the distribution with this slightly higher sensitivity setting is more evenly distributed for others as well. Later on, this is confirmed by the average distribution observed in the experiment described in the next chapter, see Figure 27.

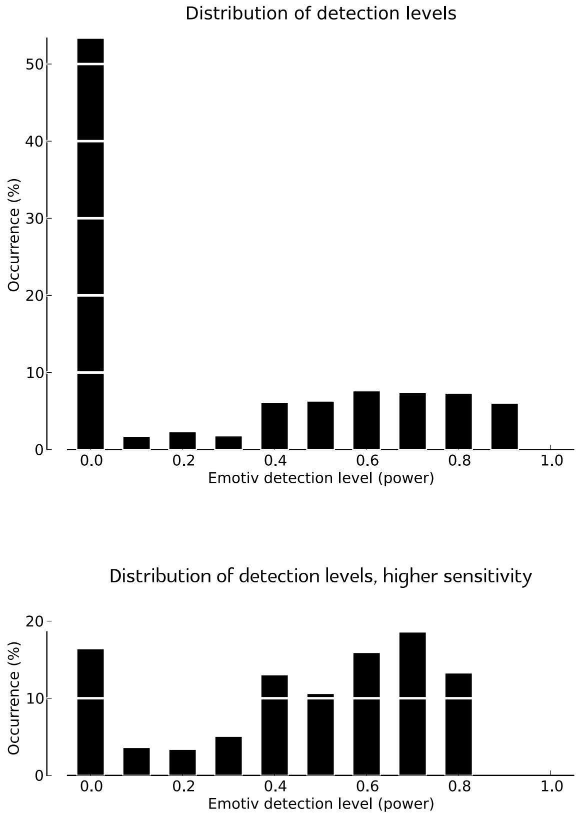


FIGURE 24: **Above: distribution of output values from the Emotiv Cognitiv suite for rest vs. hand tapping.** Values taken from logs of three test run participants, where the users were trying to generate high values for 50% of the time, and low values for 50% of the time. The values are binned around 0.0, 0.1, etc. until 1.0. This means that the outer bins are half the size (0.00-0.05, and 0.95-1.00) of the other bins (ex. 0.05-0.15).
Below: the same but with higher sensitivity, based on one test run participant.

Emotiv EmoKey translates the continuous detection values into key presses, according to self-defined rules.

Behavioural The Emotiv output is continuous, with values ranging from 0.00 to 1.00. Through EmoKey we have access to a discrete version of this output, with 11 bins (each represented by a unique key press) to represent these values. The Pax Britannica game is originally controlled with a key that is pressed or released. So essentially, the game is controlled by a switch that is either *on* or *off*.

To transform our 11-bin input to a switch, the post-processing guidelines indicate as a basic option the application of a *threshold* that yields a *rectangular wave*. Thresholds in general can be improved by having an *adaptive threshold* that incorporates a confidence level, or by adding a *cooldown* or *dwell time* to reduce false positives. EmoKey does not provide access to confidence levels or something similar, making an adaptive threshold less straight-forward to implement. Cooldown and dwell time would slow down the game in a way that is probably undesirable. They would make it very difficult for the player to quickly build a series of fighters, which is actually a common strategy in Pax Britannica.

Consider possible side effects, and the ways users might want to interact with the application.

A switch can be made more stable with *multiple thresholds* or *hysteresis*. An alternative way to obtain switch behaviour is to implement a *toggle button* by using a threshold with pulse-wave output. This way the user does not have to maintain an active state for the entire *on* duration. Instead, the user only has to go from *off* to *on* to toggle states. This method is especially interesting in the case of actual hand movement, as maintaining the *on* state takes a lot of effort, especially compared to the *off* state which only requires inaction.

With this *toggle button* method, there are additional post-processing options available related to button behaviour. *Voting* is unwanted, as it results in less frequent application input. Retriggering the button for a long on state, with *repeat*, could be beneficial, as it could provide the user with a faster way to toggle without first having to return to a rest state and then taking up hand-tapping again.

Application As the game is implemented in a scripting language (LUA), it is easy to adjust and have a peek at the application state. This makes it possible to establish and use correlations between certain state information and control input. Based on given state information, control inputs can then be enhanced or suppressed with an *adaptive threshold*, which could be seen as a form of *dynamic interface adjustment*.

To investigate what the application control should behave like, I analysed game logs of people playing Pax Britannica with keyboard. Here I focus specifically on starting and ending the selection process. There are two strategies for starting the selection process of the factory ship. One is to always start immediately after the previous build. The second strategy is to first wait until enough resources have been collected. There is no real advantage for one over the other, except that the second option requires less button-holding. The strategy where building is always followed by re-initiating the selection procedure allows for *autocompletion*.

The selection process is ended in the quadrant of the option the user wants to build. This is preferably done when the quadrant just begins, as this way the ship or upgrade is available at the earliest moment, at which point it is effective immediately. Such a dependency on a specific game state opens up the possibility of using a probabilistic model for an *adaptive threshold*.

METHOD COMPARISON PRE-TESTS

Having selected a set of potentially relevant methods, I conducted two pre-tests, with one participant each, to get an idea of the effects of these methods in practice. Although it is important to remember that these pre-tests are only based on the experiences of one person — which means the findings may not generalize — the observations can still be informative.

The first participant tested four post-processing combinations based on *switch* behaviour (threshold applied to cause a rectangular wave): *threshold*, *hysteresis*, *adaptive threshold*, and *hysteresis plus adaptive threshold*. The main observation was that the system as a whole was not responsive enough. As a result, the participant had to be always *on*, always performing the active task, to keep on trying to build ships with the overload of resources accumulated in the mean time. This required a lot of effort. The participant felt most in control with the simple *threshold*, but *hysteresis plus adaptive threshold* also resulted in good control. All in all, other factors seemed to have more influence on the usability of the system than the post-processing methods themselves.

The goal for the second pre-test was therefore to significantly reduce the amount of time the user had to be active. The *switch* threshold was adjusted to *toggle button* behaviour, which only requires short bursts of activity. This toggle button allowed for the addition of *autocompletion*, further reducing *active* time. Although this appeared helpful, giving the participant the results she wanted, she actually seemed to feel less in control. The participant could not evaluate the improvement of these methods over the switch threshold, as that was not one of the conditions in her test.

THE SELECTED METHODS

For the purpose of the experiment in the next chapter, I selected three methods to compare. The application of a threshold to obtain a rectangular *on/off* wave, resulting in *switch* behaviour, is the most direct method to obtain a control signal that can be used in the game directly. This setup is therefore considered the baseline condition. The second condition is *toggle button* behaviour. This seems to address one of the most important issues discovered in the pre-tests, and is expected to make a notable difference in the effort required from the user to control the system. The third is the combination of *toggle button* with *autocompletion*. Combinations of post-processing methods can be very powerful, and are therefore important to investigate. Yet, 'good' plus 'good' does not always result in 'better'. Another reason to test this specific condition is the remark of the second pre-test participant indicating a difference between the system doing what you want and having a sense of control. This potential trade-off asks for further inspection.

6.4 DISCUSSION AND CONCLUSIONS

This chapter provides an overview of post-processing methods and guidelines to indicate when they might be useful, with as goal to provide a first step towards deliberate reasoning and discussion when applying these post-processing methods. Sections 6.2 and 6.3 show, however, that these guidelines are just the beginning, and that there are many more decisions to be made before one arrives at a functioning system.

The example in Section 6.3 demonstrates that the guidelines can yield a selection of post-processing methods that could be useful in the given situation. Additionally, it presents a possible approach for post-processing method selection: (1) Investigate the BCI classifier (or regressor) output and the application controls; (2) Go through each of the levels in the guidelines to find methods that match with the issues and opportunities found in step 1; (3) Conduct user tests to make a final selection and finetune parameters.

Future research is further discussed in Section 8.2.

The presented guidelines are also just the beginning in another way. There are likely to be more post-processing methods out there that are not listed in this overview. Further investigation is necessary to make this overview, the guidelines, and the other considerations more complete. This can probably be achieved most quickly as a multidisciplinary endeavour.

An important thing to note, something that is not often mentioned in relation to post-processing, is that it is more than the bridge from detection to application control. As human-computer interaction takes place in a cycle, post-processing will affect the input required from the user, as is exemplified in Section 6.3. This is how post-processing not only affects system accuracy, but user effort as well.

6.5 FREQUENTLY-ASKED QUESTIONS

To end this chapter, I would like to address five questions that I get asked frequently by fellow researchers. The first is why I do not go into detail about which methods are most suitable for continuous versus discrete values. The second is quite similar, but then about the importance of self-paced versus system-paced input on post-processing method selection. The third question concerns how specific these methods are to post-processing, as some of these methods can also be found in other steps of the online BCI cycle. The fourth — and perhaps most frequent — question is if all this is also applicable to passive interfaces, as I talk a lot about 'control'. The last question is important for the development of BCIs as assistive technology. As my own research is applied to healthy people, how does all this translate to assistive applications?

Which methods are most suitable for continuous (or discrete) values? Although some post-processing methods are generally applied to discrete and others to continuous values, no solid guidelines can be given. The main reason is that it is easy to transform continuous values to discrete ones (see the methods for discrete control in the overview). Discrete values can also be represented in continuous form, by assigning classes specific continuous values, such as -1.0, and +1.0, but it is better to use the original soft detection labels, if available. These may indicate a level of certainty or a level of power.

On the application side it is a slightly different story, as the application may require continuous or discrete control for certain actions. But even there, there is more wiggle room than one might think at first glance. For example, on game consoles, movement of the main character is most commonly controlled with joysticks, each of which provides continuous input on two axes. For PC games, however, this type of control is generally provided through the discrete arrow keys, or with the left-hand alternative: the WASD keys.

Two examples of the many BCIs that cross the discrete/continuous border: The P300 mouse uses several discrete P300 targets to control a mouse on a screen, so, effectively on two continous axes [32]. AlphaWoW uses a continuous value of relaxation to trigger a discrete shape change (either to bear, or elf) [33].

Which methods are most suitable for self-paced (or system-paced) systems? When the user can provide input at any time, the system is self-paced, or asynchronous. If input is only observed during specific time slots, the system is system-paced. The latter is also referred to as a synchronous system, as the user and the system have to work in tandem for the user to provide input. BCIs that need to provide some stimulus for the user to react on for input (such as P300 or SSVEP) are often system-paced. The system knows when the stimulus is shown, and only observes brain activity around that time.

But not all self-paced systems listen only for self-induced mental inputs (such as motor imagery), and not all system-paced BCIs listen for just externally-evoked mental inputs. In the case of the P300 mouse, the P300 stimulation is provided non-stop, so the user can provide input almost at any time [32]. Of course, the input still depends on when the target flashes, but this is not limited to a specific period during which information is gathered for one selection, as would normally be the case. On the other hand, it is also possible to only listen for self-induced activity during specific periods in time. These periods during which the system listens for mental input are then not to provide stimuli to trigger mental input, but to support the interaction dynamics of that specific application. To use a videogame example, in fighting games, certain special moves can often only be executed during specific moments in time, for example, when your adversary is down on the floor. The perfect moment to telekinetically throw a giant rock towards your opponent, yoda-style. To conclude, whether the BCI is self-paced or system-paced is not necessarily a function of the input paradigm, but more of the application.

One instance where this distinction *does* affect post-processing methods is when choosing between voting and smoothing. These two methods are two sides of the same coin. Both gather more samples in order to increase certainty, which will introduce some delay. Voting is applied to a set of samples, whereas smoothing is generally applied to a moving window of samples. As such, voting is generally used with intermittent interaction, and smoothing when the interaction takes place non-stop. Another difference is that voting yields one more certain output for a number of inputs. Smoothing results in a new output for each inter-window interval. Many voting methods can easily be turned into smoothing methods by applying them to a sliding window, or vice versa. However, not all methods are interchangeable this way.

Can't these methods also be used in other steps of the online BCI cycle? Yes, the value distribution level methods and some of the behaviour level methods (such as multimodal fusion) are also commonly used in earlier processing steps where they address similar issues as highlighted in the guidelines. The purpose of this chapter, however, is to provide an overview of methods that can be applied after the feature translation step. That is why the methods in this chapter are discussed from that context.

Are these methods also relevant for passive BCIs? With passive systems, the user is not supposed to actively try to manipulate the input for the application. The user is simply observed. These observations can then be used to adjust or control an application, such as a mood-based music player.

This passive input still affects an application, so there are certain application controls. There is also a correct and incorrect interpretation of the user's brain activity, with a corresponding system response. If feedback is provided to the user — any brain-based adjustment of the application visible to the user is already a form of feedback — the user will have or will construct expectations on how it should behave. All in all, for the purpose of applying post-processing methods, the characteristics of passive BCIs are quite similar: the interpretation should be somewhat accurate, match user expectations, and be suitable to control some given application. The main difference is that effort is no longer an issue in passive systems.

It was actually at the workshop on *passive BCIs* at the fifth international BCI meeting that participants suggested using context data to help with the problem of identifying cognitive states [34].

The distinction between passive and active BCI systems is not always as clear as you might assume. For example, in AlphaWoW (see Section 2.5) you can interpret the shape-shifting passively as a representation of your current mental state. After playing a while, most users then start to consciously try to change. If the system does not respond well enough, users tend to give up, and go back to the passive way of using the system. Even in systems fully intended for passive use, users may be tempted to try to consciously affect the system. For an interesting discourse on this idea, applied more broadly to ambient environments, read Nijholt [35].

How does post-processing apply to assistive applications? Post-processing is just as important for assistive technology as it is for applications for the general population. Both user groups want good control for little effort, although for different reasons. The general population has high demands of interfaces because there are sufficient other input alternatives which *do* offer these characteristics. If BCIs do not provide good control for little effort, they simply will not be used by this target population.

For assistive technology, there are also various input alternatives (such as face buttons, sip-and-puff straws, and eye trackers). Brain-computer interfaces are not as easy to use and not as dependable yet as these other options (yet) [36]. Therefore BCIs are currently only adopted by people who have such a severe disability that these other inputs cannot be used, or cannot be used for a longer period of time. Good control for little effort is particularly important to this group. Besides, such characteristics might make BCI also a viable option for the less severely disabled users who now prefer other input devices.

As post-processing can significantly improve control and reduce the necessary user effort, it is an important step in the BCI pipeline, both for the general population and users of assistive technology.

In the introduction of Chapter 5 I already brought up the P300 speller as a concrete example. The basic BCI system can be improved by adding repetitions with averaging [25]. Without repetitions, P300 spellers are highly prone to errors, so methods such as averaging have become common-place to make P300-based systems usable. Because language is not a random sequence of characters, language models can provide probability information on what has been typed so far and what is most likely to come next. Such probabilities can be used in a variety of ways, for example to dynamically reduce the number of repetitions necessary per character [37]. But this can be taken even further. For instance, take a look at Swype®[38]. This is a text entry system for smart phones and tablets which uses personalized language models for auto-correction and auto-completion. The predictions are so accurate that you no longer have to lift your finger to type a particular character. You can simply glide over the various characters in your sentence. Needless to say that if a text entry system can deal with such a low level of preciseness, it might also be a very suitable addition to a BCI speller. Such post-processing methods can eliminate the need for repetitions altogether without changing anything to the underlying mental task recognition. Auto-completion speeds up the text entry process even further.

Another popular solution to create more usable BCI systems is to create so-called hybrid BCIs, which combine multiple input modalities, in parallel as well as sequentially. In this chapter it has been discussed in the part on *multimodal fusion* (the more generic name commonly used in the human-computer interaction community). When applying multimodal fusion in the post-classification processing stage, this can only be fusion of the results after classification, which is termed *decision-level fusion*. So to be exact: only a subset of multimodal fusion can be categorized as post-processing. However, post-processing and multimodal fusion both serve the same purpose: to improve control and reduce effort.

KEY POINTS

- The post-processing methods overview and visual guide offer inspiration as to which post-classification processing methods could be beneficial for a given system.
- Whether the advised post-processing methods are actually helpful will have to be assessed in practice, and will depend on the exact implementation and chosen parameter settings.
- Interaction occurs in a loop. Changing the post-processing therefore will affect not only the system accuracy, but the effort required from the user as well.

REFERENCES

[1] B. Blankertz, M. Kawanabe, R. Tomioka, F. Hohlefeld, V. Nikulin, and K.-R. Müller. "Invariant common spatial patterns: Alleviating nonstationarities in brain-computer interfacing." In: *Advances in Neural Information Processing Systems* 20 (2008) (cit. on p. 92).

[2] B. Reuderink. "Robust brain-computer interfaces." PhD thesis. University of Twente, 2011 (cit. on p. 92).

[3] B. L. A. van de Laar, H. Gürkök, D. Plass-Oude Bos, M. Poel, and A. Nijholt. "Experiencing BCI control in a popular computer game." In: *IEEE transactions on computational intelligence and AI in games* 5.2 (2013), pp. 176–184 (cit. on pp. 92, 94, 95).

[4] G. R. Müller-Putz, C. Breitwieser, F. Cincotti, R. Leeb, M. Schreuder, F. Leotta, M. Tavella, L. Bianchi, A. Kreilinger, A. Ramsay, M. Rohm, M. Sagebaum, L. Tonin, C. Neuper, and J. del R. Millán. "Tools for brain-computer interaction: a general concept for a hybrid BCI." In: *Frontiers in neuroinformatics* 5 (2011) (cit. on p. 92).

[5] H. Gürkök and A. Nijholt. "Brain–computer interfaces for multimodal interaction: a survey and principles." In: *International Journal of Human-Computer Interaction* 28.5 (2012), pp. 292–307 (cit. on p. 92).

[6] D. Plass-Oude Bos, M. Duvinage, O. Oktay, J. F. Delgado Saa, H. Gürüler, A. Istanbullu, M. van Vliet, B. L. A. van de Laar, M. Poel, L. Roijendijk, L. Tonin, A. Bahramisharif, and B. Reuderink. "Looking around with your brain in a virtual world." In: *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2011 IEEE Symposium on*. IEEE. 2011, pp. 1–8 (cit. on p. 93).

[7] C. Mühl, H. Gürkök, D. Plass-Oude Bos, M. E. Thurlings, L. Scherffig, M. Duvinage, A. A. Elbakyan, S. Kang, M. Poel, and D. K. J. Heylen. "Bacteria hunt." In: *Journal on Multimodal User Interfaces* 4.1 (2010), pp. 11–25 (cit. on p. 93).

[8] H. Verschore, P.-J. Kindermans, D. Verstraeten, and B. Schrauwen. "Dynamic stopping improves the speed and accuracy of a p300 speller." In: *Artificial Neural Networks and Machine Learning – ICANN 2012*. Springer, 2012, pp. 661–668 (cit. on p. 93).

[9] N.-Y. Liang, P. Saratchandran, G.-B. Huang, and N. Sundararajan. "Classification of mental tasks from EEG signals using extreme learning machine." In: *International Journal of Neural Systems* 16.01 (2006), pp. 29–38 (cit. on pp. 93, 98).

[10] J. F. Borisoff, S. G. Mason, A. Bashashati, and G. E. Birch. "Brain-computer interface design for asynchronous control applications: improvements to the LF-ASD asynchronous brain switch." In: *Biomedical Engineering, IEEE Transactions on* 51.6 (2004), pp. 985–992 (cit. on p. 94).

[11] A. Bashashati, M. Fatourechi, R. K. Ward, and G. E. Birch. "A survey of signal processing algorithms in brain-computer interfaces based on electrical brain signals." In: *Journal of Neural engineering* 4.2 (2007), R32 (cit. on p. 94).

[12] G. Pfurtscheller, G. R. Müller, J. Pfurtscheller, H. J. Gerner, and R. Rupp. "'Thought'-control of functional electrical stimulation to restore hand grasp in a patient with tetraplegia." In: *Neuroscience letters* 351.1 (2003), pp. 33–36 (cit. on p. 94).

[13] R. Scherer, M. Proll, B. Z. Allison, and G. R. Müller-Putz. "New input modalities for modern game design and virtual embodiment." In: *Virtual Reality Short Papers and Posters (VRW), 2012 IEEE*. IEEE. 2012, pp. 163–164 (cit. on pp. 94, 97).

[14] R. J. K. Jacob. "Eye movement-based human-computer interaction techniques: Toward non-command interfaces." In: *Advances in human-computer interaction* 4 (1993), pp. 151–190 (cit. on p. 94).

[15] D. Coyle, J. Garcia, A. R. Satti, and T. M. McGinnity. "EEG-based continuous control of a game using a 3 channel motor imagery BCI: BCI game." In: *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2011 IEEE Symposium on*. IEEE. 2011, pp. 1–7 (cit. on p. 95).

[16] U. Orhan, K. E. Hild, D. Erdogmus, B. Roark, B. Oken, and M. Fried-Oken. "RSVP keyboard: An EEG based typing interface." In: *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE. 2012, pp. 645–648 (cit. on p. 97).

[17] S. A. Wills and D. J. C. MacKay. "DASHER – an efficient writing system for brain-computer interfaces?" In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 14.2 (2006), pp. 244–246 (cit. on p. 97).

[18] J. Williamson, R. Murray-Smith, B. Blankertz, M. Krauledat, and K.-R. Müller. "Designing for uncertain, asymmetric control: Interaction design for brain-computer interfaces." In: *International Journal of Human-Computer Studies* 67.10 (2009), pp. 827–841 (cit. on p. 97).

[19] M. Baldauf, S. Dustdar, and F. Rosenberg. "A survey on context-aware systems." In: *International Journal of Ad Hoc and Ubiquitous Computing* 2.4 (2007), pp. 263–277 (cit. on p. 97).

[20] A. R. Satti, D. Coyle, and G. Prasad. "Self-paced brain-controlled wheelchair methodology with shared and automated assistive control." In: *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2011 IEEE Symposium on*. IEEE. 2011, pp. 1–8 (cit. on p. 97).

[21] R. B. Ashari, I. A. Al-Bidewi, and M. I. Kamel. "Design and simulation of virtual telephone keypad control based on brain computer interface (BCI) with very high transfer rates." In: *Alexandria Engineering Journal* 50.1 (2011), pp. 49–56 (cit. on p. 97).

[22] A. S. Royer and B. He. "Goal selection versus process control in a brain-computer interface based on sensorimotor rhythms." In: *Journal of neural engineering* 6.1 (2009), p. 016005 (cit. on p. 97).

[23] M. Manoochehri and M. H. Moradi. "The new post processing method for self-paced BCI system." In: *Biomedical Engineering (ICBME), 2011 18th Iranian Conference of*. IEEE. 2011, pp. 152–155 (cit. on p. 97).

[24] G. Santhanam, S. I. Ryu, M. Y. Byron, A. Afshar, and K. V. Shenoy. "A high-performance brain-computer interface." In: *Nature* 442.7099 (2006), pp. 195–198 (cit. on p. 98).

[25] D. J. Krusienski, E. W. Sellers, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw. "Toward enhanced P300 speller performance." In: *Journal of neuroscience methods* 167.1 (2008), p. 15 (cit. on pp. 98, 111).

[26] R. Molich and J. Nielsen. "Improving a human-computer dialogue." In: *Communications of the ACM* 33.3 (1990), pp. 338–348 (cit. on p. 98).

[27] J. Nielsen and the Nielsen Norman Group. *Ten usability heuristics*. `http://www.useit.com/papers/heuristic/`. Last accessed: June 13, 2014. 2005 (cit. on p. 98).

[28] E. A. Curran and M. J. Stokes. "Learning to control brain activity: a review of the production and control of EEG components for driving brain–computer interface (BCI) systems." In: *Brain and cognition* 51.3 (2003), pp. 326–336 (cit. on p. 98).

[29] B. A. S. Hasan and J. Q. Gan. "Hangman BCI: An unsupervised adaptive self-paced Brain–Computer Interface for playing games." In: *Computers in biology and medicine* 42.5 (2012), pp. 598–606 (cit. on p. 101).

[30] T. Solis-Escalante, G. Müller-Putz, C. Brunner, V. Kaiser, and G. Pfurtscheller. "Analysis of sensorimotor rhythms for the implementation of a brain switch for healthy subjects." In: *Biomedical Signal Processing and Control* 5.1 (2010), pp. 15–20 (cit. on p. 101).

[31] B. L. A. van de Laar, B. Reuderink, D. Plass-Oude Bos, and D. K. J. Heylen. "Evaluating user experience of actual and imagined movement in BCI gaming." In: *International Journal of Gaming and Computer-Mediated Simulations (IJGCMS)* 2.4 (2010), pp. 33–47 (cit. on p. 102).

[32] R. Poli, M. Salvaris, and C. Cinel. "A genetic programming approach to the evolution of brain-computer interfaces for 2-D mouse-pointer control." In: *Genetic Programming and Evolvable Machines* 13.3 (2012), pp. 377–405 (cit. on p. 109).

[33] D. Plass-Oude Bos, H. Gürkök, B. Reuderink, and M. Poel. "Improving BCI performance after classification." In: *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM. 2012, pp. 587–594 (cit. on p. 109).

[34] J. E. Huggins, C. Guger, B. Z. Allison, C. W. Anderson, A. Batista, A.-M. Brouwer, C. Brunner, R. Chavarriaga, M. Fried-Oken, A. Gunduz, D. Gupta, A. Kübler, R. Leeb, F. Lotte, L. E. Miller, G. Müller-Putz, T. Rutkowski, M. Tangermann, and D. E. Thompson. "Workshops of the fifth international brain-computer interface meeting: Defining the future." In: *Brain-Computer Interfaces* 1.1 (2014), pp. 27–49 (cit. on p. 111).

[35] A. Nijholt. "Playing and Cheating in Ambient Entertainment." In: *Proceedings 6th International Conference on Entertainment Computing*. Sept. 2007, pp. 415–420 (cit. on p. 111).

[36] F. Nijboer, D. Plass-Oude Bos, Y. Blokland, R. van Wijk, and J. Farquhar. "Design requirements and potential target users for brain-computer interfaces – recommendations from rehabilitation professionals." In: *Brain-Computer Interfaces* 1.1 (2014), pp. 50–61 (cit. on p. 111).

[37] W. Speier, C. Arnold, J. Lu, R. K. Taira, and N. Pouratian. "Natural language processing with dynamic classification improves P300 speller accuracy and bit rate." In: *Journal of neural engineering* 9.1 (2012) (cit. on p. 111).

[38] Nuance. *Swype*. www.swype.com. Last accessed: September 20, 2014 (cit. on p. 111).

7 POST-PROCESSING IN PRACTICE

Brain-computer interfaces do not provide perfect recognition of what the user tries to convey, which is inherent in the type of input (Chapter 1). Good task recognition and easy task execution is very important to users (Chapter 3), and users can assess system task recognition quite well, if they have good control over the input task (Chapter 4). In Chapters 5 and 6 I outline a solution in the form of post-classification processing methods, and give an example of how this could be applied. Now it is time to investigate whether practice follows theory.

The experiment in this chapter compares three different post-processing methods on both detection by the system and effort from the user providing this input. The methods were purposefully selected for their effects on the required user input, in an attempt to reduce user effort. Consequently, this is not a simple comparison of post-processing methods, but of the resulting input patterns as well. The main research questions are: Did these methods indeed induce the expected difference in effort? And what is the effect on the user's sense of control?

The contribution of this experiment is two-fold. First of all, it puts the theoretical guidelines from the previous chapter into practice. As such, it builds upon the awareness raised in the previous chapters of the potential value of post-processing methods, not only to increase detection accuracy, but also to reduce user effort. The second contribution is that the surprising results of this experiment point to the importance of evaluating systems such as this one with the user in the loop — not just on pre-recorded datasets, which is a common time-saving method in brain-computer interface development.

7.1 METHODS

Three different ways of interpreting BCI detections have been compared. These interpretations were selected so as to decrease effort. This experiment has been conducted in a randomized within-subjects manner.

PARTICIPANTS

Eighteen people participated in this experiment. Their average age was 39 years (from 18 up to 63), 9 out of 18 were female, and all of them were right-handed. All participants had basic computer proficiency. Six participants never played games (this experiment centres around a simple strategy game). Ten of them had no previous experience with brain-computer interfaces. Of the eight who did, seven had previous experience with imaginary movement as mental task to be recognized by the BCI. None of them had experience with actual movement to control a BCI (which is used in this experiment).

BCI CYCLE

User Task Participants controlled the system by tapping with their main hand (actual movement). The participants were instructed to keep the palm of the hand on their upper leg, to minimize movement artefacts, and to focus on the sense of moving the hand [1]. This kinesthetic approach is supposed to elicit clearer differences in brain activity. This actual movement task was chosen over the more traditional imagined movement (also known as motor imagery), because (1) actual movement should be easier to detect than imaginary movement [1], and (2) because it provides us with a ground truth of what the participant was doing. An important consequence is that besides mental effort, the task now also requires physical effort. As the participant is aware that detection happens through the headset based on activity of the mind, not the body, and that the kinesthetic approach of concentrating on the feeling is vital for successful recognition, the mental aspect is expected to remain dominant. The alternative 'rest' task was explained as sitting relaxed in the chair with the hands rested on the upper legs.

Signal Acquisition Brain activity was recorded with the commercially available Emotiv EPOC headset, which is an electroencephalograph (records EEG). Of all methods to record brain activity, EEG is the only one which is portable, non-invasive (does not require surgery), and provides short response times [2]. The Emotiv EPOC is more affordable and easier to set up than medical or research-grade EEG systems. For a commercial EEG headset, it has a lot of sensors: 14 plus 2 reference electrodes (see Figure 25 for a crude indication of the electrode montage).

See also Section 2.3.

Signal Analysis and Control Interface Emotiv provides three software applications. *Testbench* shows the slightly pre-processed signals per channel and a color indication of sensor connectivity (green is good). *Control panel* shows a similar indication for the sensor connections (see Figure 25),

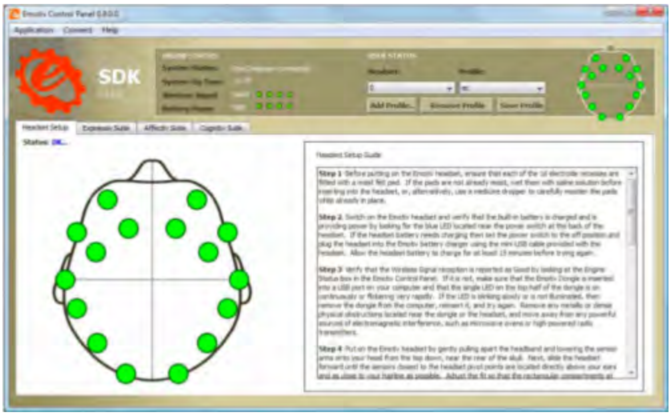


FIGURE 25: **A screenshot the Control panel main screen**, showing a rough indication of sensor positions on the head.

and provides access to the *Cognitiv suite*, which can learn to recognize tasks based on training data. *Emokey* connects to Control panel, and can generate key presses based on user-defined conditional rules. This way, the software applications from Emotiv can handle all the processing steps up until the application.

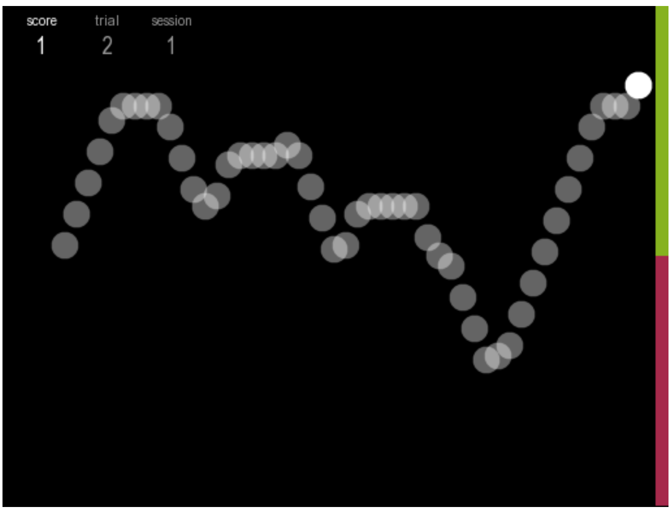


FIGURE 26: **A screenshot of the basket game**, showing the path the ball took with one of the participants. This trace was not visible to the player.

Application There were two applications to be controlled with the BCI. The basket game was used purely to assess the level of control the participant had over the input without any special post-processing. The different post-processing setups were evaluated in a simple strategy game called Pax Britannica.

"The basket game" is a popular tool in BCI research for assessing the amount of control participants have over the BCI input [3, 4]. The player has to direct balls into the correct basket by performing the right task, see Figure 26. After a second of standing still on the left border of the screen, the ball starts to move to the right. The user controls the vertical position of the ball. In our case, the ball was moved up on detected hand tapping, and down in rest. The target would be either at the top or at the bottom half of the screen.

The main application in this experiment was Pax Britannica, a one-button strategy game developed by No Fun Games[1]. This simple control made it very suitable as a test bed for BCI control. A detailed description of this application and its interface can be found in Section 6.3, but to review briefly: Each player is represented by a factory ship which can build smaller ships that can fight, and ultimately destroy the opponent's ship. The last factory ship still standing wins. Which ships can be built depends on the number of resources the factory ship has accumulated. These resources are gathered automatically over time, and spent immediately when a ship is built (building is instant). When the game is played with keyboard controls, holding down the button spins a needle on a radial menu in the middle of the player's factory ship. When the button is released, the factory ship will build the option corresponding to the quadrant the needle is pointing at at that moment.

CONDITIONS

Three different post-processing situations were compared. Each of these resulted in a different way of controlling the game. The logical input from the Emotiv Cognitiv suite ranged from 0.0 to 1.0, with 0.0 meaning the user is likely in the *rest* state, and 1.0 that the user is *tapping*. On the application side, control is either *on* (the button is held) or *off* (the button is released).

Condition A is the baseline condition. Here the game is controlled by applying a simple fixed threshold to the brain-based values. This is the most direct way to translate the continuous logical control values into the two-state semantic control for the application. The resulting behaviour is like a *switch*: the control signal is either *on* or *off*, corresponding to the user *tapping* or *resting*, respectively.

Based on a preliminary inspection of the value distribution from the Emotiv Cognitiv suite, it was decided to set this threshold to 0.4. Post-hoc inspection of the average value distribution over all participants shows that this point is at a cumulative percentage of about 40% of the data samples. As can

1 This game is freely available: `http://paxbritannica.henk.ca`

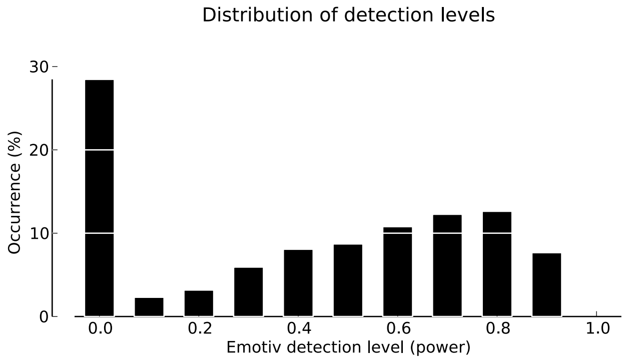


FIGURE 27: **Distribution of output values from the Emotiv Cognitiv suite for rest vs. hand tapping.** Values taken from basket game logs of all 18 participants, so the users were trying to generate high values for 50% of the time, and low values for 50% of the time, to hit the correct targets. The values are binned around 0.0, 0.1, etc. until 1.0. This means that the outer bins are half the size (0.00–0.05, and 0.95–1.00) of the other bins (ex. 0.05–0.15).

be seen in Figure 27, the input values are not normally distributed, but show a clear peak for the bin for 0.00–0.05, which corresponds to complete *rest*, that is to say: no handtapping at all.

One way to reduce the amount of time the user had to perform the active handtapping task, was to simulate *toggle button* behaviour, which was done in Condition B. In this case, the virtual game button was toggled on or off when the brain values went from low to high, so when the user started tapping from a rest position. The user no longer had to keep tapping to 'hold' the virtual button. The threshold was kept the same as in Condition A.

Condition C reduced the amount of active tapping time even more, by automatically setting the virtual in-game button to 'held'. Now the user only had to indicate when to release the button. Similar to Condition B, the button is released when the values go from low to high. The threshold was again kept the same as in the other conditions. Condition C can be described as 'toggle button plus macro', where the macro addition indicates the automatic holding of the virtual button in the game. See Figure 28 for a visualization of the amount of active handtapping and rest time for these different controls.

TEST PROTOCOL

The participants could do part of the experiment in advance. On a website, they could read the informed consent. If they did indeed consent, they could

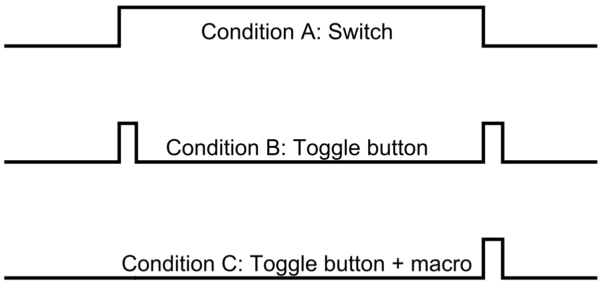


FIGURE 28: **A visualization of the different input patterns induced by the three different post-processing conditions.** Low signal indicates rest task. High signal the active task: hand tapping. Each of these ways of providing input results in the same control command for the application, through different post-processing methods.

fill out a demographics questionnaire. They could read the play instructions for Pax Britannica, and download the game to practice until they figured out a strategy that allowed them to win three times in a row. This step is vital, because it reduces the potential difference between participants with more or less experience with this type of game. Interaction logs from these practice games were e-mailed back to us. Only three participants executed these steps in advance. For the rest, these steps were done at the start of the main experiment.

The experiment session started with a welcome. If necessary, the preparatory steps described above were executed. The Emotiv EPOC is explained, and mounted on the head of the participant. In earlier experiments we observed that the EPOC can cause headaches when worn for longer than 30 minutes, especially for participants with a relatively large head circumference. The headaches seem to be caused by the sensors pressing a little too tightly onto the head. This problem has been confirmed by other research groups [5]. To prevent this from potentially affecting our results, we kept the following part, during which the participant has to wear the EPOC, short.

The training is done through the Control Panel training tab. The *rest* class could be trained in bursts of 30 seconds. The *hand tapping* only in bursts of 8 seconds. These durations were not by choice, but forced by the Emotiv Cognitiv suite. A training set therefore consisted of first one time training of rest (30 seconds), and then four bursts of hand tapping (32 seconds in total). This set was repeated 6 times[2]. After training, each participant played 10 rounds of the basket game, as an initial performance assessment of the

2 In a short assessment with three participants, it was observed that six of such training sets would provide a relatively good performance. At this number there was a peak of 72% average accuracy in the basket game.

Emotiv setup. Control panel also reports a 'skill level', which could serve as another indication of performance.

Then all of the three conditions were tested within-subject. The three conditions were (A) threshold, (B) toggle button, and (C) toggle button plus macro. With 18 participants, each possible ordering of conditions has been tested 3 times in total. The following steps were repeated for each condition. First, the current condition was quickly explained, so the participant knew how to control the game[3]. Then the participant played three game rounds, after which the condition questionnaire was filled out with questions on perceived control and effort. This questionnaire was kept short in order to stay within the time limit we set to prevent headaches.

At the end of the experiment, one last questionnaire followed, where participants specified their order of preference for the conditions and indicated why. Participants were thoroughly thanked for their cooperation, and people not employed by the university received a payment of 6 euros.

MEASUREMENTS

The demographics questionnaire contained questions about age, gender, handedness, and previous experience with brain-computer interfaces.

Initial indications of general BCI performance could be the Emotiv skill level indicated by the Cognitiv suite, and the basket game performance assessment (game score, and percentage on the correct half of the screen).

The analysis for the added value of the selected post-processing methods can be split up into two parts: control (or: detection accuracy), and effort. For the subjective assessment of control, the condition questionnaire contained two questions: "How much control did you feel you had on the game input?" and "Did the game do what you wanted it to?". As an objective measure there was the number of games won (a rather crude indicator). For effort, there was one question in the questionnaire: "How much effort did it take you to control the game?". As an objective measure, the ratio of active (handtapping) time to the total play time was calculated.

3 Not explaining the way the game is controlled, or what post-processing is applied, would allow the evaluation of a certain level of intuitiveness of the game control. Additionally, the explanation might influence what the user *thinks* the effect should be on control and effort, which may in turn affect subjective reports. However, as the participant plays only three rounds of the game, there is very little time to fully explore the control dynamics. Besides, as with these selected methods the required user input is significantly different, a short explanation is warranted.

7.2 RESULTS

OVERALL RECOGNITION

The two main indicators for overall performance were the skill level from the Emotiv Cognitiv suite, and in the basket game the amount of time the ball could be kept on the correct half of the screen. The average obtained Emotiv skill level is 31%. The average basket game performance is 68% of the time on the correct side, with an average game score of 7.9 out of 10. Correlation analysis indicates no relation between the Emotiv skill level and the basket game performance.

EFFORT

Did the different post-processing methods — reflected in the conditions A (switch), B (toggle button), and C (toggle button plus macro) — result in different activity levels, as intended? We computed the percentage of this active hand-tapping time from the total play time per condition. For the comparison between conditions, we assumed the pair-wise differences were normally distributed, which was supported by the Shapiro-Wilk test results. Paired-samples t-tests on these same data sets indicated significant differences between all conditions: ($t(17) = 3.18$, $p < 0.01$) for A ($M = 61.3$, $SD = 13.4$) vs B ($M = 40.9$, $SD = 16.5$), ($t(17) = 4.4$, $p < 0.001$) for A vs C ($M = 31.2$, $SD = 21.5$), and $t(17) = 2.43$, $p < 0.05$) for B vs C. See the left plot in Figure 29 for a visualization of the data. As intended, condition A yielded a higher activity level than B, which in turn resulted in a higher activity level than condition C.

Did this difference in activity level also result in corresponding perceived effort levels? The plot on the right in Figure 29 shows the distribution of the perceived effort levels per condition. Correlation analysis indicated no significant relation. So although our setup did result in the hypothesized reduction in activity levels over conditions A, B, and C, this did not correspond to a similar reduction in perceived effort.

CONTROL

There were two different questions related to control in the condition questionnaire "Did the game do what you wanted it to?" and "How much control did you feel you had over the game input?". The responses per condition can be seen in Figure 30.

The answers to these two questions were strongly correlated ($r(52) = 0.88$, $p < 0.001$). In turn, these answers also correlate with the number of

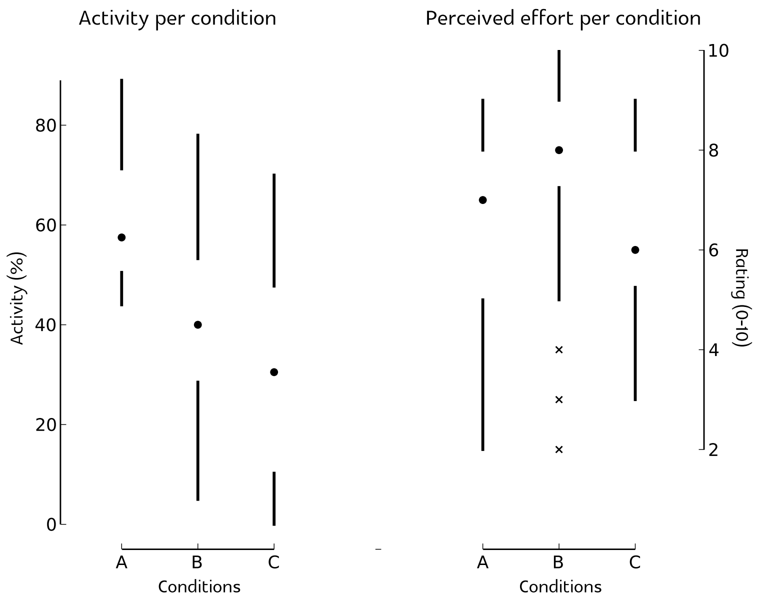


FIGURE 29: **Comparison of the three different conditions on activity levels.** On the left in terms of percentage of play time during which the participants were actively tapping their hands. On the right on perceived effort. The figures show quartile box plots in Tufte style. The line ranges from the minimum to the maximum value, excluding 1.5IQR outliers (indicated with crosses). The whitespace in between is the range from the first quartile to the third quartile. The dot in the centre denotes the median.

game rounds won ($r(52) = 0.43$, $p < 0.005$ and $r(52) = 0.36$, $p < 0.01$ respectively).

The sense of control was significantly higher for A ($M = 5.9$, $SD = 1.8$) than for B ($M = 4.1$, $SD = 2.0$) ($t(17) = 2.97$, $p < 0.01$), and B lower than C ($M = 5.1$, $SD = 2.1$) ($t(17) = -2.29$, $p < 0.05$).

PREFERENCE

In the open questions in the preference questionnaire, participants often indicated that their sense of control was the main driver for their preference in the different conditions (1 is most-preferred, 3 is least-preferred). There is indeed a strong correlation between the reported sense of control and order of preference ($r(52) = -0.44$, $p < 0.001$). Condition A (switch) was the most-preferred with an average preference score of 1.56, followed

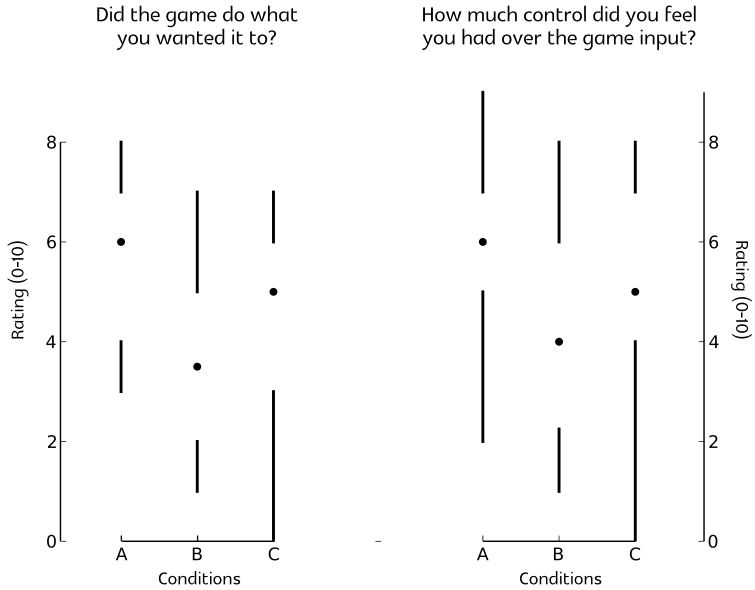


FIGURE 30: Comparison of the three different conditions on subjective control based on the questionnaire ratings.

by Condition C (toggle button plus macro) with 1.88. Condition B (toggle button) was the least preferred, with an average preference score of 2.56.

7.3 DISCUSSION AND CONCLUSIONS

In the example of the previous chapter, the postprocessing guidelines were applied to a simple one-button strategy game. In this chapter, this system was evaluated for its effects on effort and accuracy in practice. Participants did have some measure of control with this system, scoring 7.9 out of 10 points on average in the basket game session, by being on the correct side of the threshold for, on average, 68% of the time.

EFFORT

Although the different post-processing approaches did significantly reduce the amount of active task execution (hand tapping), this did not result in the hypothesized reduction in user effort. Less active does not necessarily mean less perceived effort. Based on the responses of the participants in the open questions and during the experiments themselves, we suspect that the main reason for this is that fast switching between active (handtapping) and passive (rest) states actually takes more mental effort than maintaining the active state. This difference between expected and actual results shows the importance of evaluating BCIs with the user in the loop.

The discrepancy between activity and effort might also point to a difference between physical effort and mental effort. This could be assessed in a follow-up experiment by using a questionnaire that includes this distinction, such as the NASA-TLX [6]. When using longer questionnaires, it could become problematic to stay within the time limit to prevent headaches induced by the pressure of the Emotiv EPOC headset [5]. It would also be interesting to investigate whether the same results would be observed for a body-based input modality for which the mental aspect is less pronounced.

CONTROL

The different post-processing methods affected not only the perceived amount of effort, but also had a significant impact on the perception of control. This perception of control appears to be the main determinant as to why users preferred some conditions over others. The importance of the perception of control in the preference of users for mental tasks in BCIs has been noted before [7].

On a side note: some of our participants indicated that although the condition which required the least user input might have resulted in more wins in the game, they actually felt less in control. Haselager warned us about this potential reduction in agency through the addition of system intelligence [8]. Further research could be helpful here, for example, to see whether this reduction in the sense of agency can be somewhat repaired through better feedback.

GAMERS VS. NON-GAMERS

Our participant pool consisted mostly of gamers, with only 6 people who never play games. There are too few participants to make any claims to any differences between these groups. Both groups did feel most in control in Condition A with a simple threshold. Interestingly, while for non-gamers, Condition A was also most preferred, for the gamers, there was the same level of preference for Condition A (threshold) and C (toggle button with macro). For future research, it might be better to use more homogeneous participant groups, or include more participants to explicitly compare these different user groups.

GENERALIZATION

These results have been obtained with this specific input modality, EEG measurement headset, black-box mental state detector, and application. Whether the observations generalize to other inputs, applications, etcetera, will require follow-up research.

For assistive technology users, the results of the experiment might have been different. If the underlying reason for the disability also affects their ability for sustained effort and concentration, it might actually take more effort to maintain the active task than to switch back and forth to the rest task. Where the balance tips will probably depend on the underlying condition and the mental tasks used to control the system.

Still, this experiment does provide some initial insight into the different aspects that matter when trying to improve the experience of effort and control in systems driven by uncertain user input. To be able to formulate recommendations for researchers and developers, more research into the effects and considerations for various post-processing methods is definitely required.

KEY POINTS

- The effects of post-processing methods should be evaluated not just on improved detection accuracy, but also on other aspects they might affect, such as user effort.
- Post-processing methods can significantly affect the amount of time users have to provide active input, the experienced amount of effort to provide input, and the experienced amount of control.
- It is important to evaluate systems with user tests, as user experience may be different from what one would have expected from theory at first glance.
- Again, the perception of control is indicated as the main determinant as to why the users preferred some conditions to others.

REFERENCES

[1] B. L. A. van de Laar, B. Reuderink, D. Plass-Oude Bos, and D. K. J. Heylen. "Evaluating user experience of actual and imagined movement in BCI gaming." In: *International Journal of Gaming and Computer-Mediated Simulations (IJGCMS)* 2.4 (2010), pp. 33–47 (cit. on p. 120).

[2] L. F. Nicolas-Alonso and J. Gomez-Gil. "Brain computer interfaces, a review." In: *Sensors* 12.2 (2012), pp. 1211–1279 (cit. on p. 120).

[3] D. J. McFarland, W. A. Sarnacki, and J. R. Wolpaw. "Brain-computer interface (BCI) operation: optimizing information transfer rates." In: *Biological psychology* 63.3 (2003), pp. 237–251 (cit. on p. 122).

[4] B. Blankertz, G. Dornhege, M. Krauledat, K.-R. Müller, and G. Curio. "The non-invasive Berlin brain-computer interface: fast acquisition of effective performance in untrained subjects." In: *NeuroImage* 37.2 (2007), pp. 539–550 (cit. on p. 122).

[5] N. Al-Ghamdi, G. Al-Hudhud, M. Alzamel, and A. Al-Wabil. "Trials and tribulations of BCI control applications." In: *Science and Information Conference (SAI), 2013*. IEEE. 2013, pp. 212–217 (cit. on pp. 124, 129).

[6] S. G. Hart. "NASA-task load index (NASA-TLX); 20 years later." In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 50. 9. Sage Publications. 2006, pp. 904–908 (cit. on p. 129).

[7] D. Plass-Oude Bos, M. Poel, and A. Nijholt. "A study in user-centered design and evaluation of mental tasks for BCI." In: *Advances in Multimedia Modeling* (2011), pp. 122–134 (cit. on p. 129).

[8] P. Haselager. "Did I Do That? Brain–Computer Interfacing and the Sense of Agency." In: *Minds and Machines* 23.3 (2013), pp. 405–418 (cit. on p. 129).

8 GENERAL DISCUSSION

When taking a step back, looking at the overall story woven by the results from these experiments, I see three main threads. Even when some of these threads were not originally meant to be the main objects of my investigations, they kept popping up over and over again. After discussing my results along these threads, I identify some of the threads left hanging out of this tapestry, awaiting future research.

8.1 SUMMARY AND DISCUSSION

THREAD 1) THE IMPORTANCE OF CONTROL

Chapter 1 pointed out that brain-computer interfaces do not provide perfect control, and that this problem is inherent in the type of input. BCIs suffer from problems related to noise, non-stationarities, and ambiguity. While we try to move towards real-world applications, these problems only get worse, with more noise, more distractions, and multitasking.

The issues with control touched upon in the introduction already point to the several layers within the concept of control. When looking at it more closely, we see that control is spread throughout all the steps in the BCI cycle. In the end, it is the perception of the user that matters.

The next two chapters provided further motivation to address this problem of control. Chapter 3 showed us that what users consider most important about a mental task, is how well it is recognized by the system, an important aspect of control. Task recognition can significantly affect other measurements of the user experience. When we then varied this level of recognition in Chapter 4, we discovered that users had a good perception of their level of control, at least when they were certain about the input they provide. To translate this to brain-computer interfaces: when users are more certain about the mental input they provide, through training, the actual task recognition will contribute more and more to their perception of control.

The chapters that follow (Chapters 5 and 6) propose post-classification processing as a solution to improve control after the initial task recognition step. The results from literature seem promising. Chapter 7 shows how post-processing does indeed have a significant impact in practice and how the perceived level of control was the deciding factor in user preference for the different post-processing methods that were tested.

Every time I evaluated user preference, aspects of control turned out to be the most important. Task recognition by the system is most important for mental task preference of the user. The resulting perception of control by the user is most important for selecting post-processing methods when designing, developing, and finetuning BCIs. The importance of good control had already been well established for disabled end-users, but is now also confirmed for healthy users. This result provides additional motivation to the already existing focus on control in existing BCI research.

THREAD 2) THE USER IN THE LOOP

I already point out the limited approach common in BCI experiments in Chapter 1, which prefers to keep users and applications out of the equation. In Chapter 3 I deliberately apply user-centred methods, first to select new potential mental tasks, and second to investigate what users prefer in these mental tasks.

Chapter 4 then shows us that actual control and the user's perception of it actually line up pretty well. So there is not always a gap between perception and reality. But this observation also opens up another line of thinking: could we deliberately manipulate user perception of control?

When I evaluate my proposed solution of using post-processing methods to improve control and reduce user effort in Chapter 7, the importance of the user becomes more clear than ever. The results are quite different from what I expected based on theory, and all because of the user experience. Where I thought the experience of effort would come from executing the more active task, it turns out that in practice, switching between the active and inactive tasks was what demanded the most effort. This is something I could never have discovered had I conducted an offline test based on an existing dataset.

In human-computer interaction research, the importance of the user in the whole equation is known and accepted. HCI research has valuable knowledge and methods to offer which are made to develop systems that people *can* and even *want* to use. The BCI community only has to open its doors a bit more widely to benefit. Hopefully the cases in this thesis will provide a helpful nudge.

THREAD 3) PROMISING POST-PROCESSING

This final thread is the one I originally intended to be the main storyline. There is a lot of overlap between Threads 1 and 3, as Thread 1 (the importance of control) supports the main motivation for the importance of control and Thread 3 focuses on a potential solution to the control issues inherent in BCIs.

After establishing the importance of control to the user, I propose to use post-classification processing methods as a solution. Based on the findings in Chapter 5 I can say that these methods *can* be highly beneficial, but have received too little attention in the BCI community. To be able to really benefit from these methods, they need to be applied more deliberately, accompanied by discussion and evaluation. Chapter 6 provides an overview of methods, with guidelines on how to apply them, which will hopefully support BCI researchers and developers to approach these methods more purposively. Chapter 7 confirms the significant effects post-processing methods can have, but also warns us that theory does not always follow practice.

The promise of post-processing methods and lack of current practice is clearly established in this thesis. But this is only a first step into this line of research.

8.2 FUTURE RESEARCH

Research is always limited: in sample sizes, tested user tasks, applications, and any other factors that may be of influence to its outcomes. This means that its findings may not be generalizable to other contexts. To establish such generalizability, further research is necessary. See the discussion sections with each experiment for the details. Besides this, I would like to point out some other interesting directions for future research.

TOWARDS INTUITIVE, INTENT-BASED CONTROL

Originally, my work was to be about the development of intuitive brain-computer interfaces. This goal is recognizable in Chapter 3, for example. After a couple of years I noticed that what I actually had been doing was to make brain-computer interfaces more usable through these post-processing methods, which henceforth became the topic of my thesis.

The most-used mental tasks in brain-computer interfaces are not very intuitive. The user can be trained so these tasks become more familiar, but the tasks are not familiar from other experiences in life, which is one way to interpret the term 'intuitive'. Intuitive mental tasks are easy to learn and easy to remember. As BCIs provide a new input modality, with no established frameworks that new users could make use of (such as swiping, pinching, etc. for touch interfaces) a high level of natural intuitiveness would make this new technology easier to use and accept.

One way of making BCIs more intuitive is the use of naturally-occurring brain activity. Instead of artificial mental tasks, systems could learn to detect brain activity that is naturally related to the user's intent. This is not necessarily passive, as the user can be actively pursuing this intent. When using naturally-occurring brain activity accompanying a user's intent, no translation of the intent to an artificial task and resulting system control would be required. Instead, brain activity could be directly interpreted as reflecting the intention itself. This would allow the user to focus entirely on the task at hand.

To provide an analogy, if you are a proficient computer user, you are no longer conscious of typing on the keyboard when you write an e-mail. Your full attention is on the message you are trying to convey. A similar experience should be achievable with brain-computer interfaces, and with the selection of suitable mental tasks in relation to the corresponding application controls, perhaps it can be achievable without the extensive user training such an experience normally requires.

I still believe this to be a fruitful research direction for brain-computer interfaces especially. Based on the promises in pop-culture references, this is what people have come to expect from brain-computer interfaces: intuitive, intent-based control.

MANIPULATION OF PERCEPTION OF CONTROL

The perception of control can be improved by improving the control itself, or... by *manipulating* the perception of control. In the discussion of Chapter 4 I already point to this interesting avenue for future research, which could have relevant applications for all input modalities which suffer from inherent uncertainty. This manipulation could for example be through the input task, application controls or effects, or the feedback.

Through the application controls and effects of these controls, developers can appeal to specific psychological phenomena which will encourage users to have a more optimistic assessment of the amount of control they have over the system. For example the fact that people overestimate their influence on events that have a positive outcome and underestimate their influence on negative outcomes can be used as follows: by using the BCI input to obtain a gain (instead of to prevent a loss), and using other modalities with better recognition rates for tasks that could have more negative results.

To explain this more clearly, let us take look at one of the prototypes mentioned in this thesis. AlphaWoW already applies the second part of this principle by keeping the standard mouse and keyboard for the primary controls (movement and abilities), while using BCI input on secondary control aspects that are less critical for survival and playability. In this case, the loss prevented is highly inefficient gameplay in the best scenario to virtual death in the worst. The secondary control aspects simply induce a different style of play, which is not always a gain, although it can be situationally beneficial. The user assessment might have been more positive if the BCI had had an explicit benefit, such as a more effective use of the character's abilities when the player is more relaxed.

This line of perception manipulation is particularly suitable for games, as the goals and results of user actions are designed by the game designers instead of following from some external user goals.

Uncertainty can also be introduced, increased, or maintained through the mental task and the provided feedback. For certain mental tasks it is more difficult to assess whether you are performing them correctly, such as motor imagery. Others are already difficult to explain in the first place, such as the type of relaxation related to reduced activity in the parietal cortex. Such tasks have an inherent level of uncertainty. In such cases, users will have to trust the feedback from the system more than their own senses, at least initially. This dependence on feedback from the system creates an opportunity to manipulate the user's perception.

Feedback can be less or more specific in two ways: what feedback information is provided and how it is communicated. To go back to the AlphaWoW example: we show the user their amount of relaxation in a bar chart, one of the easiest to interpret options for visualizing information. A less precise alternative would have been to use the colour of a glow around the avatar to communicate the amount of relaxation. We could also withhold the feedback on relaxation altogether, leaving the user with only the high-level feedback of whether they are in elf-shape or bear. If the detection accuracy is low, the level or preciseness of the feedback should reflect this. This reduces annoyance with incorrect recognitions. On the other hand, when the detection is highly accurate, more detailed feedback better supports learning and expert use.

Such deliberate manipulation of the perception of control could be a useful research direction for all uncertain input modalities, as it has the potential to increase the user acceptance of these inputs. Application will not be straight-forward, as there are some important trade-offs to consider here. Although the task-induced and feedback-induced uncertainties described above may open up opportunities for user acceptance, at the same time it impedes the user in learning to control the input. Such trade-offs may be dealt with dynamically, with more probable or certain detections resulting in more detailed feedback on the fly, or by providing more detailed feedback in specific training situations.

FURTHER DEVELOPMENT OF THE GUIDELINES

The main two implications of this thesis for BCI development is to (1) use HCI knowledge and methods to create usable systems users want, and (2) to use post-processing methods to improve this usability. This second implication has been described in this thesis as the need for deliberate application with structured discussion and evaluation of their advantages and disadvantages.

To enable this, this thesis provides an overview and guidelines, but these are but a first draft. The list of methods can be extended, (a) by widening the search net of Chapter 5 to find more examples of post-processing use in BCI research, (b) by further investigating the post-processing methods used by other input methods, and (c) by continuing the search in other research areas with similar problems, such as machine learning and control theory. An interdisciplinary approach is advised to arrive at a more complete overview and guidelines.

Brain-computer interfaces are not the only inputs which suffer from the various issues that create this inherent uncertainty. Many inputs based on measurements from the body do. The overview and guidelines in Chapter 6 have been set up to be applicable to these other uncertain inputs as well.

It would be highly valuable if through concerted effort from the research areas of each of these modalities this overview and the guidelines could be extended and made up-to-date with current developments. This way, underlying common principles can be identified, and all input modalities could benefit.

AUTHOR PUBLICATIONS

JOURNAL ARTICLES

F. Nijboer, D. Plass-Oude Bos, Y. Blokland, R. van Wijk, and J. Farquhar. “Design requirements and potential target users for brain-computer interfaces – recommendations from rehabilitation professionals.” In: *Brain-Computer Interfaces* 1.1 (2014), pp. 50–61.

B. L. A. van de Laar, H. Gürkök, D. Plass-Oude Bos, M. Poel, and A. Nijholt. “Experiencing BCI control in a popular computer game.” In: *IEEE transactions on computational intelligence and AI in games* 5.2 (2013), pp. 176–184.

B. L. A. van de Laar, D. Plass-Oude Bos, B. Reuderink, M. Poel, and A. Nijholt. “How Much Control Is Enough? Influence of Unreliable Input on User Experience.” In: *Cybernetics, IEEE Transactions on* 43.6 (Dec. 2013), pp. 1584–1592.

B. L. A. van de Laar, B. Reuderink, D. Plass-Oude Bos, and D. K. J. Heylen. “Evaluating user experience of actual and imagined movement in BCI gaming.” In: *International Journal of Gaming and Computer-Mediated Simulations (IJGCMS)* 2.4 (2010), pp. 33–47.

C. Mühl, H. Gürkök, D. Plass-Oude Bos, M. E. Thurlings, L. Scherffig, M. Duvinage, A. A. Elbakyan, S. Kang, M. Poel, and D. K. J. Heylen. “Bacteria hunt.” In: *Journal on Multimodal User Interfaces* 4.1 (2010), pp. 11–25.

A. Nijholt, D. Plass-Oude Bos, and B. Reuderink. “Turning shortcomings into challenges: Brain–computer interfaces for games.” In: *Entertainment Computing* 1.2 (2009), pp. 85–94.

CONFERENCE PAPERS

H. Gürkök, B. L. A. van de Laar, D. Plass-Oude Bos, M. Poel, and A. Nijholt. “Players’ Opinions on Control and Playability of a BCI Game.” In: *Universal Access in Human-Computer Interaction. Universal Access to Information and Knowledge - 8th International Conference, UAHCI 2014*. Springer, 2014, pp. 549–560.

D. Plass-Oude Bos, B. L. A. van de Laar, B. Reuderink, M. Poel, and A. Nijholt. "Perception and manipulation of game control." In: *Proceedings 6th International Conference on Intelligent Technologies for Interactive Entertainment (INTETAIN 2014)*. Ed. by D. Reidsma. Vol. 136. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer Verlag, Berlin, 2014, pp. 57–66.

M. Obbink, H. Gürkök, D. Plass-Oude Bos, G. Hakvoort, M. Poel, and A. Nijholt. "Social interaction in a cooperative brain-computer interface game." In: *Intelligent Technologies for Interactive Entertainment*. Springer Berlin Heidelberg, 2012, pp. 183–192.

D. Plass-Oude Bos, H. Gürkök, B. Reuderink, and M. Poel. "Improving BCI performance after classification." In: *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM. 2012, pp. 587–594.

G. Hakvoort, H. Gürkök, D. Plass-Oude Bos, M. Obbink, and M. Poel. "Measuring Immersion and Affect in a Brain-Computer Interface Game." In: *Human-Computer Interaction - INTERACT 2011*. Berlin/Heidelberg, Germany: Springer-Verlag, 2011, pp. 115–128.

F. Nijboer, B. Z. Allison, S. Dunne, D. Plass-Oude Bos, A. Nijholt, and P. Haselager. "A Preliminary Survey on the Perception of Marketability of Brain-Computer Interfaces and Initial Development of a Repository of BCI Companies." In: (2011). Ed. by G.R. Mueller-Putz, R. Sherer, M. Billinger, A. Kreilinger, V. Kaiser, and C. Neuper, pp. 344–347.

D. Plass-Oude Bos, M. Duvinage, O. Oktay, J. F. Delgado Saa, H. Gürüler, A. Istanbullu, M. van Vliet, B. L. A. van de Laar, M. Poel, L. Roijendijk, L. Tonin, A. Bahramisharif, and B. Reuderink. "Looking around with your brain in a virtual world." In: *Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB), 2011 IEEE Symposium on*. IEEE. 2011, pp. 1–8.

D. Plass-Oude Bos, M. Poel, and A. Nijholt. "A study in user-centered design and evaluation of mental tasks for BCI." In: *Advances in Multimedia Modeling* (2011), pp. 122–134.

B. L. A. van de Laar, H. Gürkök, D. Plass-Oude Bos, F. Nijboer, and A. Nijholt. "Perspectives on user experience evaluation of brain-computer interfaces." In: *Universal Access in Human-Computer Interaction. Users Diversity*. Springer, 2011, pp. 600–609.

D. Plass-Oude Bos, B. Reuderink, B. L. A. van de Laar, H. Gürkök, C. Mühl, M. Poel, D. K. J. Heylen, and A. Nijholt. "Human-computer interaction for BCI games: Usability and user experience." In: *Cyberworlds (CW), 2010 International Conference on*. IEEE. 2010, pp. 277–281.

B. L. A. van de Laar, D. Plass-Oude Bos, B. Reuderink, and D. K. J. Heylen. “Actual and Imagined Movement in BCI Gaming.” In: *Proceedings of the International Conference on Artificial Intelligence and Simulation of Behaviour (AISB 2009)*. 2009.

A. Nijholt, B. Reuderink, and D. Oude Bos. “Turning Shortcomings into Challenges: Brain-Computer Interfaces for Games.” In: *Intelligent Technologies for Interactive Entertainment*. May 2009, pp. 153–168.

F. L. A. Knoppel, A. S. Tigelaar, D. Oude Bos, T. Alofs, and Zs. Ruttkay. “Trackside DEIRA: a dynamic engaging intelligent reporter agent.” In: *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems. 2008, pp. 112–119.

BOOK CHAPTERS

B. L. A. van de Laar, H. Gürkök, D. Plass-Oude Bos, F. Nijboer, and A. Nijholt. “Brain-Computer Interfaces and User Experience Evaluation.” In: *Towards Practical Brain-Computer Interfaces*. Ed. by B. Z. Allison, S. Dunne, R. Leeb, J. del R. Millán, and A. Nijholt. Springer, 2012, pp. 223–237.

D. Plass-Oude Bos, B. Reuderink, B. L. A. van de Laar, H. Gürkök, C. Mühl, M. Poel, A. Nijholt, and D. K. J. Heylen. “Brain-Computer Interfacing and Games.” In: *Brain-Computer Interfaces: Applying our Minds to Human-Computer Interaction*. Ed. by D. S. Tan and A. Nijholt. Springer, 2010. Chap. 10.

WORKSHOPS

H. Gürkök, D. Plass-Oude Bos, M. Obbink, G. Hakvoort, C. Mühl, and A. Nijholt. “Towards multiplayer BCI games.” In: *BioSPlay: Workshop on Multiuser and Social Biosignal Adaptive Games and Playful Applications. Workshop at Fun and Games, Leuven, Belgium*. 2010, pp. 35–48.

C. Mühl, H. Gürkök, D. Plass-Oude Bos, M. E. Thurlings, L. Scherffig, M. Duvinage, A. A. Elbakyan, S. Kang, M. Poel, and D. K. J. Heylen. “Bacteria Hunt: A multimodal, multiparadigm BCI game.” In: *Proceedings of the 5th International Summer Workshop on Multimodal Interfaces (eNTERFACE’09)*. University of Genua, 2010.

D. Plass-Oude Bos, M. Duvinage, O. Oktay, J. F. Delgado Saa, H. Gürüler, A. Istanbullu, M. van Vliet, B. L. A. van de Laar, M. Poel, L. Roijendijk, L. Tonin, A. Bahramisharif, and B. Reuderink. "Looking around in a virtual world." In: *Proceedings of the 6th International Summer Workshop on Multimodal Interfaces (eNTERFACE'10)*. University of Amsterdam, 2010.

D. Plass-Oude Bos, B. Reuderink, B. L. A. van de Laar, H. Gürkök, C. Mühl, M. Poel, A. Nijholt, and D. K. J. Heylen. "Brain-computer interfacing for Games." In: *Brain, Body and Bytes: Psychophysiological User Interaction, Workshop at CHI 2010*. 2010.

ABSTRACTS

H. Gürkök, D. Plass-Oude Bos, B. L. A. van de Laar, F. Nijboer, and A. Nijholt. "User Experience Evaluation in BCI: Filling the Gap." In: *International Journal of Bioelectromagnetism*. Vol. 13. 1. Tampere: International Society for Bioelectromagnetism, July 2011, pp. 54–55.

B. L. A. van de Laar, F. Nijboer, H. Gürkök, D. Plass-Oude Bos, and A. Nijholt. "User Experience Evaluation in BCI: Bridge the Gap." In: *International journal of Bioelectromagnetism*. Vol. 13. 3. Tampere: International Society for Bioelectromagnetism, July 2011, pp. 57–58.

D. Plass-Oude Bos, H. Gürkök, B. L. A. van de Laar, F. Nijboer, and A. Nijholt. "User Experience Evaluation in BCI: Mind the Gap!" In: *International Journal of Bioelectromagnetism*. Vol. 13. 1. Tampere: International Society for Bioelectromagnetism, July 2011, pp. 48–49.

D. Plass-Oude Bos, B. L. A. van de Laar, M. Duvinage, O. Oktay, J. F. Delgado Saa, M. van Vliet, M. Poel, L. Roijendijk, A. Bahramisharif, and B. Reuderink. "Wild photoshoot: Applying overt and covert attention." In: *Neuroscience Letters*. Vol. 500. Supplement. Society of Applied Neuroscience Meeting 2011. Elsevier, 2011, e10.

D. Oude Bos and B. Reuderink. "BrainBasher: a BCI game." In: *Extended Abstracts of the International Conference on Fun and Games*. Eindhoven University of Technology, 2008, pp. 36–39.

SIKS DISSERTATIEREEKS

1998-1 Johan van den Akker (CWI) *DEGAS – An Active, Temporal Database of Autonomous Objects*

1998-2 Floris Wiesman (UM) *Information Retrieval by Graphically Browsing Meta-Information*

1998-3 Ans Steuten (TUD) *A Contribution to the Linguistic Analysis of Business Conversations within the Language/Action Perspective*

1998-4 Dennis Breuker (UM) *Memory versus Search in Games*

1998-5 E.W.Oskamp (RUL) *Computerondersteuning bij Straftoemeting*

1999-1 Mark Sloof (VU) *Physiology of Quality Change Modelling; Automated modelling of Quality Change of Agricultural Products*

1999-2 Rob Potharst (EUR) *Classification using decision trees and neural nets*

1999-3 Don Beal (UM) *The Nature of Minimax Search*

1999-4 Jacques Penders (UM) *The practical Art of Moving Physical Objects*

1999-5 Aldo de Moor (KUB) *Empowering Communities: A Method for the Legitimate User-Driven Specification of Network Information Systems*

1999-6 Niek J.E. Wijngaards (VU) *Re-design of compositional systems*

1999-7 David Spelt (UT) *Verification support for object database design*

1999-8 Jacques H.J. Lenting (UM) *Informed Gambling: Conception and Analysis of a Multi-Agent Mechanism for Discrete Reallocation.*

2000-1 Frank Niessink (VU) *Perspectives on Improving Software Maintenance*

2000-2 Koen Holtman (TUE) *Prototyping of CMS Storage Management*

2000-3 Carolien M.T. Metselaar (UVA) *Sociaal-organisatorische gevolgen van kennistechnologie; een procesbenadering en actorperspectief.*

2000-4 Geert de Haan (VU) *ETAG, A Formal Model of Competence Knowledge for User Interface Design*

2000-5 Ruud van der Pol (UM) *Knowledge-based Query Formulation in Information Retrieval.*

2000-6 Rogier van Eijk (UU) *Programming Languages for Agent Communication*

2000-7 Niels Peek (UU) *Decision-theoretic Planning of Clinical Patient Management*

2000-8 Veerle Coup (EUR) *Sensitivity Analyis of Decision-Theoretic Networks*

2000-9 Florian Waas (CWI) *Principles of Probabilistic Query Optimization*

2000-10 Niels Nes (CWI) *Image Database Management System Design Considerations, Algorithms and Architecture*

2000-11 Jonas Karlsson (CWI) *Scalable Distributed Data Structures for Database Management*

2001-1 Silja Renooij (UU) *Qualitative Approaches to Quantifying Probabilistic Networks*

2001-2 Koen Hindriks (UU) *Agent Programming Languages: Programming with Mental Models*

2001-3 Maarten van Someren (UvA) *Learning as problem solving*

2001-4 Evgueni Smirnov (UM) *Conjunctive and Disjunctive Version Spaces with Instance-Based Boundary Sets*

2001-5 Jacco van Ossenbruggen (VU) *Processing Structured Hypermedia: A Matter of Style*

2001-6 Martijn van Welie (VU) *Task-based User Interface Design*

2001-7 Bastiaan Schonhage (VU) *Diva: Architectural Perspectives on Information Visualization*

2001-8 Pascal van Eck (VU) *A Compositional Semantic Structure for Multi-Agent Systems Dynamics.*

2001-9 Pieter Jan 't Hoen (RUL) *Towards Distributed Development of Large Object-Oriented Models, Views of Packages as Classes*

2001-10 Maarten Sierhuis (UvA) *Modeling and Simulating Work Practice BRAHMS: a multiagent modeling and simulation language for work practice analysis and design*

2001-11 Tom M. van Engers (VUA) *Knowledge Management: The Role of Mental Models in Business Systems Design*

2002-01 Nico Lassing (VU) *Architecture-Level Modifiability Analysis*

2002-02 Roelof van Zwol (UT) *Modelling and searching web-based document collections*

2002-03 Henk Ernst Blok (UT) *Database Optimization Aspects for Information Retrieval*

2002-04 Juan Roberto Castelo Valdueza (UU) *The Discrete Acyclic Digraph Markov Model in Data Mining*

2002-05 Radu Serban (VU) *The Private Cyberspace Modeling Electronic Environments inhabited by Privacy-concerned Agents*

2002-06 Laurens Mommers (UL) *Applied legal epistemology; Building a knowledge-based ontology of the legal domain*

2002-07 Peter Boncz (CWI) *Monet: A Next-Generation DBMS Kernel For Query-Intensive Applications*

2002-08 Jaap Gordijn (VU) *Value Based Requirements Engineering: Exploring Innovative E-Commerce Ideas*

2002-09 Willem-Jan van den Heuvel(KUB) *Integrating Modern Business Applications with Objectified Legacy Systems*

2002-10 Brian Sheppard (UM) *Towards Perfect Play of Scrabble*

2002-11 Wouter C.A. Wijngaards (VU) *Agent Based Modelling of Dynamics: Biological and Organisational Applications*

2002-12 Albrecht Schmidt (Uva) *Processing XML in Database Systems*

2002-13 Hongjing Wu (TUE) *A Reference Architecture for Adaptive Hypermedia Applications*

2002-14 Wieke de Vries (UU) *Agent Interaction: Abstract Approaches to Modelling, Programming and Verifying Multi-Agent Systems*

2002-15 Rik Eshuis (UT) *Semantics and Verification of UML Activity Diagrams for Workflow Modelling*

2002-16 Pieter van Langen (VU) *The Anatomy of Design: Foundations, Models and Applications*

2002-17 Stefan Manegold (UVA) *Understanding, Modeling, and Improving Main-Memory Database Performance*

2003-01 Heiner Stuckenschmidt (VU) *Ontology-Based Information Sharing in Weakly Structured Environments*

2003-02 Jan Broersen (VU) *Modal Action Logics for Reasoning About Reactive Systems*

2003-03 Martijn Schuemie (TUD) *Human-Computer Interaction and Presence in Virtual Reality Exposure Therapy*

2003-04 Milan Petkovic (UT) *Content-Based Video Retrieval Supported by Database Technology*

2003-05 Jos Lehmann (UVA) *Causation in Artificial Intelligence and Law – A modelling approach*

2003-06 Boris van Schooten (UT) *Development and specification of virtual environments*

2003-07 Michiel Jansen (UvA) *Formal Explorations of Knowledge Intensive Tasks*

2003-08 Yongping Ran (UM) *Repair Based Scheduling*

2003-09 Rens Kortmann (UM) *The resolution of visually guided behaviour*

2003-10 Andreas Lincke (UvT) *Electronic Business Negotiation: Some experimental studies on the interaction between medium, innovation context and culture*

2003-11 Simon Keizer (UT) *Reasoning under Uncertainty in Natural Language Dialogue using Bayesian Networks*

2003-12 Roeland Ordelman (UT) *Dutch speech recognition in multimedia information retrieval*

2003-13 Jeroen Donkers (UM) *Nosce Hostem – Searching with Opponent Models*

2003-14 Stijn Hoppenbrouwers (KUN) *Freezing Language: Conceptualisation Processes across ICT-Supported Organisations*

2003-15 Mathijs de Weerdt (TUD) *Plan Merging in Multi-Agent Systems*

2003-16 Menzo Windhouwer (CWI) *Feature Grammar Systems – Incremental Maintenance of Indexes to Digital Media Warehouses*

2003-17 David Jansen (UT) *Extensions of Statecharts with Probability, Time, and Stochastic Timing*

2003-18 Levente Kocsis (UM) *Learning Search Decisions*

2004-01 Virginia Dignum (UU) *A Model for Organizational Interaction: Based on Agents, Founded in Logic*

2004-02 Lai Xu (UvT) *Monitoring Multi-party Contracts for E-business*

2004-03 Perry Groot (VU) *A Theoretical and Empirical Analysis of Approximation in Symbolic Problem Solving*

2004-04 Chris van Aart (UVA) *Organizational Principles for Multi-Agent Architectures*

2004-05 Viara Popova (EUR) *Knowledge discovery and monotonicity*

2004-06 Bart-Jan Hommes (TUD) *The Evaluation of Business Process Modeling Techniques*

2004-07 Elise Boltjes (UM) *Voorbeeldig onderwijs; voorbeeldgestuurd onderwijs, een opstap naar abstract denken, vooral voor meisjes*

2004-08 Joop Verbeek(UM) *Politie en de Nieuwe Internationale Informatiemarkt, Grensregionale politi□le gegevensuitwisseling en digitale expertise*

2004-09 Martin Caminada (VU) *For the Sake of the Argument; explorations into argument-based reasoning*

2004-10 Suzanne Kabel (UVA) *Knowledge-rich indexing of learning-objects*

2004-11 Michel Klein (VU) *Change Management for Distributed Ontologies*

2004-12 The Duy Bui (UT) *Creating emotions and facial expressions for embodied agents*

2004-13 Wojciech Jamroga (UT) *Using Multiple Models of Reality: On Agents who Know how to Play*

2004-14 Paul Harrenstein (UU) *Logic in Conflict. Logical Explorations in Strategic Equilibrium*

2004-15 Arno Knobbe (UU) *Multi-Relational Data Mining*

2004-16 Federico Divina (VU) *Hybrid Genetic Relational Search for Inductive Learning*

2004-17 Mark Winands (UM) *Informed Search in Complex Games*

2004-18 Vania Bessa Machado (UvA) *Supporting the Construction of Qualitative Knowledge Models*

2004-19 Thijs Westerveld (UT) *Using generative probabilistic models for multimedia retrieval*

2004-20 Madelon Evers (Nyenrode) *Learning from Design: facilitating multidisciplinary design teams*

2005-01 Floor Verdenius (UVA) *Methodological Aspects of Designing Induction-Based Applications*

2005-02 Erik van der Werf (UM)) *AI techniques for the game of Go*

2005-03 Franc Grootjen (RUN) *A Pragmatic Approach to the Conceptualisation of Language*

2005-04 Nirvana Meratnia (UT) *Towards Database Support for Moving Object data*

2005-05 Gabriel Infante-Lopez (UVA) *Two-Level Probabilistic Grammars for Natural Language Parsing*

2005-06 Pieter Spronck (UM) *Adaptive Game AI*

2005-07 Flavius Frasincar (TUE) *Hypermedia Presentation Generation for Semantic Web Information Systems*

2005-08 Richard Vdovjak (TUE) *A Model-driven Approach for Building Distributed Ontology-based Web Applications*

2005-09 Jeen Broekstra (VU) *Storage, Querying and Inferencing for Semantic Web Languages*

2005-10 Anders Bouwer (UVA) *Explaining Behaviour: Using Qualitative Simulation in Interactive Learning Environments*

2005-11 Elth Ogston (VU) *Agent Based Matchmaking and Clustering – A Decentralized Approach to Search*

2005-12 Csaba Boer (EUR) *Distributed Simulation in Industry*

2005-13 Fred Hamburg (UL) *Een Computermodel voor het Ondersteunen van Euthanasiebeslissingen*

2005-14 Borys Omelayenko (VU) *Web-Service configuration on the Semantic Web; Exploring how semantics meets pragmatics*

2005-15 Tibor Bosse (VU) *Analysis of the Dynamics of Cognitive Processes*

2005-16 Joris Graaumans (UU) *Usability of XML Query Languages*

2005-17 Boris Shishkov (TUD) *Software Specification Based on Re-usable Business Components*

2005-18 Danielle Sent (UU) *Test-selection strategies for probabilistic networks*

2005-19 Michel van Dartel (UM) *Situated Representation*

2005-20 Cristina Coteanu (UL) *Cyber Consumer Law, State of the Art and Perspectives*

2005-21 Wijnand Derks (UT) *Improving Concurrency and Recovery in Database Systems by Exploiting Application Semantics*

2006-01 Samuil Angelov (TUE) *Foundations of B2B Electronic Contracting*

2006-02 Cristina Chisalita (VU) *Contextual issues in the design and use of information technology in organizations*

2006-03 Noor Christoph (UVA) *The role of metacognitive skills in learning to solve problems*

2006-04 Marta Sabou (VU) *Building Web Service Ontologies*

2006-05 Cees Pierik (UU) *Validation Techniques for Object-Oriented Proof Outlines*

2006-06 Ziv Baida (VU) *Software-aided Service Bundling – Intelligent Methods & Tools for Graphical Service Modeling*

2006-07 Marko Smiljanic (UT) *XML schema matching – balancing efficiency and effectiveness by means of clustering*

2006-08 Eelco Herder (UT) *Forward, Back and Home Again – Analyzing User Behavior on the Web*

2006-09 Mohamed Wahdan (UM) *Automatic Formulation of the Auditor's Opinion*

2006-10 Ronny Siebes (VU) *Semantic Routing in Peer-to-Peer Systems*

2006-11 Joeri van Ruth (UT) *Flattening Queries over Nested Data Types*

2006-12 Bert Bongers (VU) *Interactivation – Towards an e-cology of people, our technological environment, and the arts*

2006-13 Henk-Jan Lebbink (UU) *Dialogue and Decision Games for Information Exchanging Agents*

2006-14 Johan Hoorn (VU) *Software Requirements: Update, Upgrade, Redesign – towards a Theory of Requirements Change*

2006-15 Rainer Malik (UU) *CONAN: Text Mining in the Biomedical Domain*

2006-16 Carsten Riggelsen (UU) *Approximation Methods for Efficient Learning of Bayesian Networks*

2006-17 Stacey Nagata (UU) *User Assistance for Multitasking with Interruptions on a Mobile Device*

2006-18 Valentin Zhizhkun (UVA) *Graph transformation for Natural Language Processing*

2006-19 Birna van Riemsdijk (UU) *Cognitive Agent Programming: A Semantic Approach*

2006-20 Marina Velikova (UvT) *Monotone models for prediction in data mining*

2006-21 Bas van Gils (RUN) *Aptness on the Web*

2006-22 Paul de Vrieze (RUN) *Fundaments of Adaptive Personalisation*

2006-23 Ion Juvina (UU) *Development of Cognitive Model for Navigating on the Web*

2006-24 Laura Hollink (VU) *Semantic Annotation for Retrieval of Visual Resources*

2006-25 Madalina Drugan (UU) *Conditional log-likelihood MDL and Evolutionary MCMC*

2006-26 Vojkan Mihajlovic (UT) *Score Region Algebra: A Flexible Framework for Structured Information Retrieval*

2006-27 Stefano Bocconi (CWI) *Vox Populi: generating video documentaries from semantically annotated media repositories*

2006-28 Borkur Sigurbjornsson (UVA) *Focused Information Access using XML Element Retrieval*

2007-01 Kees Leune (UvT) *Access Control and Service-Oriented Architectures*

2007-02 Wouter Teepe (RUG) *Reconciling Information Exchange and Confidentiality: A Formal Approach*

2007-03 Peter Mika (VU) *Social Networks and the Semantic Web*

2007-04 Jurriaan van Diggelen (UU) *Achieving Semantic Interoperability in Multi-agent Systems: a dialogue-based approach*

2007-05 Bart Schermer (UL) *Software Agents, Surveillance, and the Right to Privacy: a Legislative Framework for Agent-enabled Surveillance*

2007-06 Gilad Mishne (UVA) *Applied Text Analytics for Blogs*

2007-07 Natasa Jovanovic' (UT) *To Whom It May Concern – Addressee Identification in Face-to-Face Meetings*

2007-08 Mark Hoogendoorn (VU) *Modeling of Change in Multi-Agent Organizations*

2007-09 David Mobach (VU) *Agent-Based Mediated Service Negotiation*

2007-10 Huib Aldewereld (UU) *Autonomy vs. Conformity: an Institutional Perspective on Norms and Protocols*

2007-11 Natalia Stash (TUE) *Incorporating Cognitive/Learning Styles in a General-Purpose Adaptive Hypermedia System*

2007-12 Marcel van Gerven (RUN) *Bayesian Networks for Clinical Decision Support: A Rational Approach to Dynamic Decision-Making under Uncertainty*

2007-13 Rutger Rienks (UT) *Meetings in Smart Environments; Implications of Progressing Technology*

2007-14 Niek Bergboer (UM) *Context-Based Image Analysis*

2007-15 Joyca Lacroix (UM) *NIM: a Situated Computational Memory Model*

2007-16 Davide Grossi (UU) *Designing Invisible Handcuffs. Formal investigations in Institutions and Organizations for Multi-agent Systems*

2007-17 Theodore Charitos (UU) *Reasoning with Dynamic Networks in Practice*

2007-18 Bart Orriens (UvT) *On the development an management of adaptive business collaborations*

2007-19 David Levy (UM) *Intimate relationships with artificial partners*

2007-20 Slinger Jansen (UU) *Customer Configuration Updating in a Software Supply Network*

2007-21 Karianne Vermaas (UU) *Fast diffusion and broadening use: A research on residential adoption and usage of broadband internet in the Netherlands between 2001 and 2005*

2007-22 Zlatko Zlatev (UT) *Goal-oriented design of value and process models from patterns*

2007-23 Peter Barna (TUE) *Specification of Application Logic in Web Information Systems*

2007-24 Georgina Ram□rez Camps (CWI) *Structural Features in XML Retrieval*

2007-25 Joost Schalken (VU) *Empirical Investigations in Software Process Improvement*

2008-01 Katalin Boer-Sorbán (EUR) *Agent-Based Simulation of Financial Markets: A modular, continuous-time approach*

2008-02 Alexei Sharpanskykh (VU) *On Computer-Aided Methods for Modeling and Analysis of Organizations*

2008-03 Vera Hollink (UVA) *Optimizing hierarchical menus: a usage-based approach*

2008-04 Ander de Keijzer (UT) *Management of Uncertain Data – towards unattended integration*

2008-05 Bela Mutschler (UT) *Modeling and simulating causal dependencies on process-aware information systems from a cost perspective*

2008-06 Arjen Hommersom (RUN) *On the Application of Formal Methods to Clinical Guidelines, an Artificial Intelligence Perspective*

2008-07 Peter van Rosmalen (OU) *Supporting the tutor in the design and support of adaptive e-learning*

2008-08 Janneke Bolt (UU) *Bayesian Networks: Aspects of Approximate Inference*

2008-09 Christof van Nimwegen (UU) *The paradox of the guided user: assistance can be counter-effective*

2008-10 Wauter Bosma (UT) *Discourse oriented summarization*

2008-11 Vera Kartseva (VU) *Designing Controls for Network Organizations: A Value-Based Approach*

2008-12 Jozsef Farkas (RUN) *A Semiotically Oriented Cognitive Model of Knowledge Representation*

2008-13 Caterina Carraciolo (UVA) *Topic Driven Access to Scientific Handbooks*

2008-14 Arthur van Bunningen (UT) *Context-Aware Querying; Better Answers with Less Effort*

2008-15 Martijn van Otterlo (UT) *The Logic of Adaptive Behavior: Knowledge Representation and Algorithms for the Markov Decision Process Framework in First-Order Domains.*

2008-16 Henriette van Vugt (VU) *Embodied agents from a user's perspective*

2008-17 Martin Op 't Land (TUD) *Applying Architecture and Ontology to the Splitting and Allying of Enterprises*

2008-18 Guido de Croon (UM) *Adaptive Active Vision*

2008-19 Henning Rode (UT) *From Document to Entity Retrieval: Improving Precision and Performance of Focused Text Search*

2008-20 Rex Arendsen (UVA) *Geen bericht, goed bericht. Een onderzoek naar de effecten van de introductie van elektronisch berichtenverkeer met de overheid op de administratieve lasten van bedrijven.*

2008-21 Krisztian Balog (UVA) *People Search in the Enterprise*

2008-22 Henk Koning (UU) *Communication of IT-Architecture*

2008-23 Stefan Visscher (UU) *Bayesian network models for the management of ventilator-associated pneumonia*

2008-24 Zharko Aleksovski (VU) *Using background knowledge in ontology matching*

2008-25 Geert Jonker (UU) *Efficient and Equitable Exchange in Air Traffic Management Plan Repair using Spender-signed Currency*

2008-26 Marijn Huijbregts (UT) *Segmentation, Diarization and Speech Transcription: Surprise Data Unraveled*

2008-27 Hubert Vogten (OU) *Design and Implementation Strategies for IMS Learning Design*

2008-28 Ildiko Flesch (RUN) *On the Use of Independence Relations in Bayesian Networks*

2008-29 Dennis Reidsma (UT) *Annotations and Subjective Machines – Of Annotators, Embodied Agents, Users, and Other Humans*

2008-30 Wouter van Atteveldt (VU) *Semantic Network Analysis: Techniques for Extracting, Representing and Querying Media Content*

2008-31 Loes Braun (UM) *Pro-Active Medical Information Retrieval*

2008-32 Trung H. Bui (UT) *Toward Affective Dialogue Management using Partially Observable Markov Decision Processes*

2008-33 Frank Terpstra (UVA) *Scientific Workflow Design; theoretical and practical issues*

2008-34 Jeroen de Knijf (UU) *Studies in Frequent Tree Mining*

2008-35 Ben Torben Nielsen (UvT) *Dendritic morphologies: function shapes structure*

2009-01 Rasa Jurgelenaite (RUN) *Symmetric Causal Independence Models*

2009-02 Willem Robert van Hage (VU) *Evaluating Ontology-Alignment Techniques*

2009-03 Hans Stol (UvT) *A Framework for Evidence-based Policy Making Using IT*

2009-04 Josephine Nabukenya (RUN) *Improving the Quality of Organisational Policy Making using Collaboration Engineering*

2009-05 Sietse Overbeek (RUN) *Bridging Supply and Demand for Knowledge Intensive Tasks – Based on Knowledge, Cognition, and Quality*

2009-06 Muhammad Subianto (UU) *Understanding Classification*

2009-07 Ronald Poppe (UT) *Discriminative Vision-Based Recovery and Recognition of Human Motion*

2009-08 Volker Nannen (VU) *Evolutionary Agent-Based Policy Analysis in Dynamic Environments*

2009-09 Benjamin Kanagwa (RUN) *Design, Discovery and Construction of Service-oriented Systems*

2009-10 Jan Wielemaker (UVA) *Logic programming for knowledge-intensive interactive applications*

2009-11 Alexander Boer (UVA) *Legal Theory, Sources of Law & the Semantic Web*

2009-12 Peter Massuthe (TUE, Humboldt-Universitaet zu Berlin) *Operating Guidelines for Services*

2009-13 Steven de Jong (UM) *Fairness in Multi-Agent Systems*

2009-14 Maksym Korotkiy (VU) *From ontology-enabled services to service-enabled ontologies (making ontologies work in e-science with ONTO-SOA)*

2009-15 Rinke Hoekstra (UVA) *Ontology Representation – Design Patterns and Ontologies that Make Sense*

2009-16 Fritz Reul (UvT) *New Architectures in Computer Chess*

2009-17 Laurens van der Maaten (UvT) *Feature Extraction from Visual Data*

2009-18 Fabian Groffen (CWI) *Armada, An Evolving Database System*

2009-19 Valentin Robu (CWI) *Modeling Preferences, Strategic Reasoning and Collaboration in Agent-Mediated Electronic Markets*

2009-20 Bob van der Vecht (UU) *Adjustable Autonomy: Controling Influences on Decision Making*

2009-21 Stijn Vanderlooy (UM) *Ranking and Reliable Classification*

2009-22 Pavel Serdyukov (UT) *Search For Expertise: Going beyond direct evidence*

2009-23 Peter Hofgesang (VU) *Modelling Web Usage in a Changing Environment*

2009-24 Annerieke Heuvelink (VUA) *Cognitive Models for Training Simulations*

2009-25 Alex van Ballegooij (CWI) *RAM: Array Database Management through Relational Mapping*

2009-26 Fernando Koch (UU) *An Agent-Based Model for the Development of Intelligent Mobile Services*

2009-27 Christian Glahn (OU) *Contextual Support of social Engagement and Reflection on the Web*

2009-28 Sander Evers (UT) *Sensor Data Management with Probabilistic Models*

2009-29 Stanislav Pokraev (UT) *Model-Driven Semantic Integration of Service-Oriented Applications*

2009-30 Marcin Zukowski (CWI) *Balancing vectorized query execution with bandwidth-optimized storage*

2009-31 Sofiya Katrenko (UVA) *A Closer Look at Learning Relations from Text*

2009-32 Rik Farenhorst (VU) and Remco de Boer (VU) *Architectural Knowledge Management: Supporting Architects and Auditors*

2009-33 Khiet Truong (UT) *How Does Real Affect Affect Affect Recognition In Speech?*

2009-34 Inge van de Weerd (UU) *Advancing in Software Product Management: An Incremental Method Engineering Approach*

2009-35 Wouter Koelewijn (UL) *Privacy en Politiegegevens; Over geautomatiseerde normatieve informatie-uitwisseling*

2009-36 Marco Kalz (OUN) *Placement Support for Learners in Learning Networks*

2009-37 Hendrik Drachsler (OUN) *Navigation Support for Learners in Informal Learning Networks*

2009-38 Riina Vuorikari (OU) *Tags and self-organisation: a metadata ecology for learning resources in a multilingual context*

2009-39 Christian Stahl (TUE, Humboldt-Universitaet zu Berlin) *Service Substitution – A Behavioral Approach Based on Petri Nets*

2009-40 Stephan Raaijmakers (UvT) *Multinomial Language Learning: Investigations into the Geometry of Language*

2009-41 Igor Berezhnyy (UvT) *Digital Analysis of Paintings*

2009-42 Toine Bogers (UvT) *Recommender Systems for Social Bookmarking*

2009-43 Virginia Nunes Leal Franqueira (UT) *Finding Multi-step Attacks in Computer Networks using Heuristic Search and Mobile Ambients*

2009-44 Roberto Santana Tapia (UT) *Assessing Business-IT Alignment in Networked Organizations*

2009-45 Jilles Vreeken (UU) *Making Pattern Mining Useful*

2009-46 Loredana Afanasiev (UvA) *Querying XML: Benchmarks and Recursion*

2010-01 Matthijs van Leeuwen (UU) *Patterns that Matter*

2010-02 Ingo Wassink (UT) *Work flows in Life Science*

2010-03 Joost Geurts (CWI) *A Document Engineering Model and Processing Framework for Multimedia documents*

2010-04 Olga Kulyk (UT) *Do You Know What I Know? Situational Awareness of Co-located Teams in Multidisplay Environments*

2010-05 Claudia Hauff (UT) *Predicting the Effectiveness of Queries and Retrieval Systems*

2010-06 Sander Bakkes (UvT) *Rapid Adaptation of Video Game AI*

2010-07 Wim Fikkert (UT) *Gesture interaction at a Distance*

2010-08 Krzysztof Siewicz (UL) *Towards an Improved Regulatory Framework of Free Software. Protecting user freedoms in a world of software communities and eGovernments*

2010-09 Hugo Kielman (UL) *A Politiele gegevensverwerking en Privacy, Naar een effectieve waarborging*

2010-10 Rebecca Ong (UL) *Mobile Communication and Protection of Children*

2010-11 Adriaan Ter Mors (TUD) *The world according to MARP: Multi-Agent Route Planning*

2010-12 Susan van den Braak (UU) *Sensemaking software for crime analysis*

2010-13 Gianluigi Folino (RUN) *High Performance Data Mining using Bio-inspired techniques*

2010-14 Sander van Splunter (VU) *Automated Web Service Reconfiguration*

2010-15 Lianne Bodenstaff (UT) *Managing Dependency Relations in Inter-Organizational Models*

2010-16 Sicco Verwer (TUD) *Efficient Identification of Timed Automata, theory and practice*

2010-17 Spyros Kotoulas (VU) *Scalable Discovery of Networked Resources: Algorithms, Infrastructure, Applications*

2010-18 Charlotte Gerritsen (VU) *Caught in the Act: Investigating Crime by Agent-Based Simulation*

2010-19 Henriette Cramer (UvA) *People's Responses to Autonomous and Adaptive Systems*

2010-20 Ivo Swartjes (UT) *Whose Story Is It Anyway? How Improv Informs Agency and Authorship of Emergent Narrative*

2010-21 Harold van Heerde (UT) *Privacy-aware data management by means of data degradation*

2010-22 Michiel Hildebrand (CWI) *End-user Support for Access to Heterogeneous Linked Data*

2010-23 Bas Steunebrink (UU) *The Logical Structure of Emotions*

2010-24 Dmytro Tykhonov *Designing Generic and Efficient Negotiation Strategies*

2010-25 Zulfiqar Ali Memon (VU) *Modelling Human-Awareness for Ambient Agents: A Human Mindreading Perspective*

2010-26 Ying Zhang (CWI) *XRPC: Efficient Distributed Query Processing on Heterogeneous XQuery Engines*

2010-27 Marten Voulon (UL) *Automatisch contracteren*

2010-28 Arne Koopman (UU) *Characteristic Relational Patterns*

2010-29 Stratos Idreos(CWI) *Database Cracking: Towards Auto-tuning Database Kernels*

2010-30 Marieke van Erp (UvT) *Accessing Natural History – Discoveries in data cleaning, structuring, and retrieval*

2010-31 Victor de Boer (UVA) *Ontology Enrichment from Heterogeneous Sources on the Web*

2010-32 Marcel Hiel (UvT) *An Adaptive Service Oriented Architecture: Automatically solving Interoperability Problems*

2010-33 Robin Aly (UT) *Modeling Representation Uncertainty in Concept-Based Multimedia Retrieval*

2010-34 Teduh Dirgahayu (UT) *Interaction Design in Service Compositions*

2010-35 Dolf Trieschnigg (UT) *Proof of Concept: Concept-based Biomedical Information Retrieval*

2010-36 Jose Janssen (OU) *Paving the Way for Lifelong Learning; Facilitating competence development through a learning path specification*

2010-37 Niels Lohmann (TUE) *Correctness of services and their composition*

2010-38 Dirk Fahland (TUE) *From Scenarios to components*

2010-39 Ghazanfar Farooq Siddiqui (VU) *Integrative modeling of emotions in virtual agents*

2010-40 Mark van Assem (VU) *Converting and Integrating Vocabularies for the Semantic Web*

2010-41 Guillaume Chaslot (UM) *Monte-Carlo Tree Search*

2010-42 Sybren de Kinderen (VU) *Needs-driven service bundling in a multi-supplier setting – the computational e3-service approach*

2010-43 Peter van Kranenburg (UU) *A Computational Approach to Content-Based Retrieval of Folk Song Melodies*

2010-44 Pieter Bellekens (TUE) *An Approach towards Context-sensitive and User-adapted Access to Heterogeneous Data Sources, Illustrated in the Television Domain*

2010-45 Vasilios Andrikopoulos (UvT) *A theory and model for the evolution of software services*

2010-46 Vincent Pijpers (VU) *e3alignment: Exploring Inter-Organizational Business-ICT Alignment*

2010-47 Chen Li (UT) *Mining Process Model Variants: Challenges, Techniques, Examples*

2010-48 *Withdrawn*

2010-49 Jahn-Takeshi Saito (UM) *Solving difficult game positions*

2010-50 Bouke Huurnink (UVA) *Search in Audiovisual Broadcast Archives*

2010-51 Alia Khairia Amin (CWI) *Understanding and supporting information seeking tasks in multiple sources*

2010-52 Peter-Paul van Maanen (VU) *Adaptive Support for Human-Computer Teams: Exploring the Use of Cognitive Models of Trust and Attention*

2010-53 Edgar Meij (UVA) *Combining Concepts and Language Models for Information Access*

2011-01 Botond Cseke (RUN) *Variational Algorithms for Bayesian Inference in Latent Gaussian Models*

2011-02 Nick Tinnemeier(UU) *Organizing Agent Organizations. Syntax and Operational Semantics of an Organization-Oriented Programming Language*

2011-03 Jan Martijn van der Werf (TUE) *Compositional Design and Verification of Component-Based Information Systems*

2011-04 Hado van Hasselt (UU) *Insights in Reinforcement Learning; Formal analysis and empirical evaluation of temporal-difference learning algorithms*

2011-05 Base van der Raadt (VU) *Enterprise Architecture Coming of Age – Increasing the Performance of an Emerging Discipline.*

2011-06 Yiwen Wang (TUE) *Semantically-Enhanced Recommendations in Cultural Heritage*

2011-07 Yujia Cao (UT) *Multimodal Information Presentation for High Load Human Computer Interaction*

2011-08 Nieske Vergunst (UU) *BDI-based Generation of Robust Task-Oriented Dialogues*

2011-09 Tim de Jong (OU) *Contextualised Mobile Media for Learning*

2011-10 Bart Bogaert (UvT) *Cloud Content Contention*

2011-11 Dhaval Vyas (UT) *Designing for Awareness: An Experience-focused HCI Perspective*

2011-12 Carmen Bratosin (TUE) *Grid Architecture for Distributed Process Mining*

2011-13 Xiaoyu Mao (UvT) *Airport under Control. Multiagent Scheduling for Airport Ground Handling*

2011-14 Milan Lovric (EUR) *Behavioral Finance and Agent-Based Artificial Markets*

2011-15 Marijn Koolen (UvA) *The Meaning of Structure: the Value of Link Evidence for Information Retrieval*

2011-16 Maarten Schadd (UM) *Selective Search in Games of Different Complexity*

2011-17 Jiyin He (UVA) *Exploring Topic Structure: Coherence, Diversity and Relatedness*

2011-18 Mark Ponsen (UM) *Strategic Decision-Making in complex games*

2011-19 Ellen Rusman (OU) *The Mind's Eye on Personal Profiles*

2011-20 Qing Gu (VU) *Guiding service-oriented software engineering – A view-based approach*

2011-21 Linda Terlouw (TUD) *Modularization and Specification of Service-Oriented Systems*

2011-22 Junte Zhang (UVA) *System Evaluation of Archival Description and Access*

2011-23 Wouter Weerkamp (UVA) *Finding People and their Utterances in Social Media*

2011-24 Herwin van Welbergen (UT) *Behavior Generation for Interpersonal Coordination with Virtual Humans On Specifying, Scheduling and Realizing Multimodal Virtual Human Behavior*

2011-25 Syed Waqar ul Qounain Jaffry (VU)) *Analysis and Validation of Models for Trust Dynamics*

2011-26 Matthijs Aart Pontier (VU) *Virtual Agents for Human Communication – Emotion Regulation and Involvement-Distance Trade-Offs in Embodied Conversational Agents and Robots*

2011-27 Aniel Bhulai (VU) *Dynamic website optimization through autonomous management of design patterns*

2011-28 Rianne Kaptein (UVA) *Effective Focused Retrieval by Exploiting Query Context and Document Structure*

2011-29 Faisal Kamiran (TUE) *Discrimination-aware Classification*

2011-30 Egon van den Broek (UT) *Affective Signal Processing (ASP): Unraveling the mystery of emotions*

2011-31 Ludo Waltman (EUR) *Computational and Game-Theoretic Approaches for Modeling Bounded Rationality*

2011-32 Nees-Jan van Eck (EUR) *Methodological Advances in Bibliometric Mapping of Science*

2011-33 Tom van der Weide (UU) *Arguing to Motivate Decisions*

2011-34 Paolo Turrini (UU) *Strategic Reasoning in Interdependence: Logical and Game-theoretical Investigations*

2011-35 Maaike Harbers (UU) *Explaining Agent Behavior in Virtual Training*

2011-36 Erik van der Spek (UU) *Experiments in serious game design: a cognitive approach*

2011-37 Adriana Burlutiu (RUN) *Machine Learning for Pairwise Data, Applications for Preference Learning and Supervised Network Inference*

2011-38 Nyree Lemmens (UM) *Bee-inspired Distributed Optimization*

2011-39 Joost Westra (UU) *Organizing Adaptation using Agents in Serious Games*

2011-40 Viktor Clerc (VU) *Architectural Knowledge Management in Global Software Development*

2011-41 Luan Ibraimi (UT) *Cryptographically Enforced Distributed Data Access Control*

2011-42 Michal Sindlar (UU) *Explaining Behavior through Mental State Attribution*

2011-43 Henk van der Schuur (UU) *Process Improvement through Software Operation Knowledge*

2011-44 Boris Reuderink (UT) *Robust Brain-Computer Interfaces*

2011-45 Herman Stehouwer (UvT) *Statistical Language Models for Alternative Sequence Selection*

2011-46 Beibei Hu (TUD) *Towards Contextualized Information Delivery: A Rule-based Architecture for the Domain of Mobile Police Work*

2011-47 Azizi Bin Ab Aziz(VU) *Exploring Computational Models for Intelligent Support of Persons with Depression*

2011-48 Mark Ter Maat (UT) *Response Selection and Turn-taking for a Sensitive Artificial Listening Agent*

2011-49 Andreea Niculescu (UT) *Conversational interfaces for task-oriented spoken dialogues: design aspects influencing interaction quality*

2012-01 Terry Kakeeto (UvT) *Relationship Marketing for SMEs in Uganda*

2012-02 Muhammad Umair(VU) *Adaptivity, emotion, and Rationality in Human and Ambient Agent Models*

2012-03 Adam Vanya (VU) *Supporting Architecture Evolution by Mining Software Repositories*

2012-04 Jurriaan Souer (UU) *Development of Content Management System-based Web Applications*

2012-05 Marijn Plomp (UU) *Maturing Interorganisational Information Systems*

2012-06 Wolfgang Reinhardt (OU) *Awareness Support for Knowledge Workers in Research Networks*

2012-07 Rianne van Lambalgen (VU) *When the Going Gets Tough: Exploring Agent-based Models of Human Performance under Demanding Conditions*

2012-08 Gerben de Vries (UVA) *Kernel Methods for Vessel Trajectories*

2012-09 Ricardo Neisse (UT) *Trust and Privacy Management Support for Context-Aware Service Platforms*

2012-10 David Smits (TUE) *Towards a Generic Distributed Adaptive Hypermedia Environment*

2012-11 J.C.B. Rantham Prabhakara (TUE) *Process Mining in the Large: Preprocessing, Discovery, and Diagnostics*

2012-12 Kees van der Sluijs (TUE) *Model Driven Design and Data Integration in Semantic Web Information Systems*

2012-13 Suleman Shahid (UvT) *Fun and Face: Exploring non-verbal expressions of emotion during playful interactions*

2012-14 Evgeny Knutov(TUE) *Generic Adaptation Framework for Unifying Adaptive Web-based Systems*

2012-15 Natalie van der Wal (VU) *Social Agents. Agent-Based Modelling of Integrated Internal and Social Dynamics of Cognitive and Affective Processes.*

2012-16 Fiemke Both (VU) *Helping people by understanding them – Ambient Agents supporting task execution and depression treatment*

2012-17 Amal Elgammal (UvT) *Towards a Comprehensive Framework for Business Process Compliance*

2012-18 Eltjo Poort (VU) *Improving Solution Architecting Practices*

2012-19 Helen Schonenberg (TUE) *What's Next? Operational Support for Business Process Execution*

2012-20 Ali Bahramisharif (RUN) *Covert Visual Spatial Attention, a Robust Paradigm for Brain-Computer Interfacing*

2012-21 Roberto Cornacchia (TUD) *Querying Sparse Matrices for Information Retrieval*

2012-22 Thijs Vis (UvT) *Intelligence, politie en veiligheidsdienst: verenigbare grootheden?*

2012-23 Christian Mühl (UT) *Toward Affective Brain-Computer Interfaces: Exploring the Neurophysiology of Affect during Human Media Interaction*

2012-24 Laurens van der Werff (UT) *Evaluation of Noisy Transcripts for Spoken Document Retrieval*

2012-25 Silja Eckartz (UT) *Managing the Business Case Development in Inter-Organizational IT Projects: A Methodology and its Application*

2012-26 Emile de Maat (UVA) *Making Sense of Legal Text*

2012-27 Hayrettin Gürkök (UT) *Mind the Sheep! User Experience Evaluation & Brain-Computer Interface Games*

2012-28 Nancy Pascall (UvT) *Engendering Technology Empowering Women*

2012-29 Almer Tigelaar (UT) *Peer-to-Peer Information Retrieval*

2012-30 Alina Pommeranz (TUD) *Designing Human-Centered Systems for Reflective Decision Making*

2012-31 Emily Bagarukayo (RUN) *A Learning by Construction Approach for Higher Order Cognitive Skills Improvement, Building Capacity and Infrastructure*

2012-32 Wietske Visser (TUD) *Qualitative multi-criteria preference representation and reasoning*

2012-33 Rory Sie (OUN) *Coalitions in Cooperation Networks (COCOON)*

2012-34 Pavol Jancura (RUN) *Evolutionary analysis in PPI networks and applications*

2012-35 Evert Haasdijk (VU) *Never Too Old To Learn – On-line Evolution of Controllers in Swarm- and Modular Robotics*

2012-36 Denis Ssebugwawo (RUN) *Analysis and Evaluation of Collaborative Modeling Processes*

2012-37 Agnes Nakakawa (RUN) *A Collaboration Process for Enterprise Architecture Creation*

2012-38 Selmar Smit (VU) *Parameter Tuning and Scientific Testing in Evolutionary Algorithms*

2012-39 Hassan Fatemi (UT) *Risk-aware design of value and coordination networks*

2012-40 Agus Gunawan (UvT) *Information Access for SMEs in Indonesia*

2012-41 Sebastian Kelle (OU) *Game Design Patterns for Learning*

2012-42 Dominique Verpoorten (OU) *Reflection Amplifiers in self-regulated Learning*

2012-43 *Withdrawn*

2012-44 Anna Tordai (VU) *On Combining Alignment Techniques*

2012-45 Benedikt Kratz (UvT) *A Model and Language for Business-aware Transactions*

2012-46 Simon Carter (UVA) *Exploration and Exploitation of Multilingual Data for Statistical Machine Translation*

2012-47 Manos Tsagkias (UVA) *Mining Social Media: Tracking Content and Predicting Behavior*

2012-48 Jorn Bakker (TUE) *Handling Abrupt Changes in Evolving Time-series Data*

2012-49 Michael Kaisers (UM) *Learning against Learning – Evolutionary dynamics of reinforcement learning algorithms in strategic interactions*

2012-50 Steven van Kervel (TUD) *Ontologogy driven Enterprise Information Systems Engineering*

2012-51 Jeroen de Jong (TUD) *Heuristics in Dynamic Sceduling; a practical framework with a case study in elevator dispatching*

2013-01 Viorel Milea (EUR) *News Analytics for Financial Decision Support*

2013-02 Erietta Liarou (CWI) *MonetDB/DataCell: Leveraging the Column-store Database Technology for Efficient and Scalable Stream Processing*

2013-03 Szymon Klarman (VU) *Reasoning with Contexts in Description Logics*

2013-04 Chetan Yadati (TUD) *Coordinating autonomous planning and scheduling*

2013-05 Dulce Pumareja (UT) *Groupware Requirements Evolutions Patterns*

2013-06 Romulo Goncalves (CWI) *The Data Cyclotron: Juggling Data and Queries for a Data Warehouse Audience*

2013-07 Giel van Lankveld (UvT) *Quantifying Individual Player Differences*

2013-08 Robbert-Jan Merk (VU) *Making enemies: cognitive modeling for opponent agents in fighter pilot simulators*

2013-09 Fabio Gori (RUN) *Metagenomic Data Analysis: Computational Methods and Applications*

2013-10 Jeewanie Jayasinghe Arachchige (UvT) *A Unified Modeling Framework for Service Design.*

2013-11 Evangelos Pournaras (TUD) *Multi-level Reconfigurable Self-organization in Overlay Services*

2013-12 Marian Razavian (VU) *Knowledge-driven Migration to Services*

2013-13 Mohammad Safiri (UT) *Service Tailoring: User-centric creation of integrated IT-based homecare services to support independent living of elderly*

2013-14 Jafar Tanha (UVA) *Ensemble Approaches to Semi-Supervised Learning Learning*

2013-15 Daniel Hennes (UM) *Multiagent Learning – Dynamic Games and Applications*

2013-16 Eric Kok (UU) *Exploring the practical benefits of argumentation in multi-agent deliberation*

2013-17 Koen Kok (VU) *The PowerMatcher: Smart Coordination for the Smart Electricity Grid*

2013-18 Jeroen Janssens (UvT) *Outlier Selection and One-Class Classification*

2013-19 Renze Steenhuizen (TUD) *Coordinated Multi-Agent Planning and Scheduling*

2013-20 Katja Hofmann (UvA) *Fast and Reliable Online Learning to Rank for Information Retrieval*

2013-21 Sander Wubben (UvT) *Text-to-text generation by monolingual machine translation*

2013-22 Tom Claassen (RUN) *Causal Discovery and Logic*

2013-23 Patricio de Alencar Silva(UvT) *Value Activity Monitoring*

2013-24 Haitham Bou Ammar (UM) *Automated Transfer in Reinforcement Learning*

2013-25 Agnieszka Anna Latoszek-Berendsen (UM) *Intention-based Decision Support. A new way of representing and implementing clinical guidelines in a Decision Support System*

2013-26 Alireza Zarghami (UT) *Architectural Support for Dynamic Homecare Service Provisioning*

2013-27 Mohammad Huq (UT) *Inference-based Framework Managing Data Provenance*

2013-28 Frans van der Sluis (UT) *When Complexity becomes Interesting: An Inquiry into the Information eXperience*

2013-29 Iwan de Kok (UT) *Listening Heads*

2013-30 Joyce Nakatumba (TUE) *Resource-Aware Business Process Management: Analysis and Support*

2013-31 Dinh Khoa Nguyen (UvT) *Blueprint Model and Language for Engineering Cloud Applications*

2013-32 Kamakshi Rajagopal (OUN) *Networking For Learning; The role of Networking in a Lifelong Learner's Professional Development*

2013-33 Qi Gao (TUD) *User Modeling and Personalization in the Microblogging Sphere*

2013-34 Kien Tjin-Kam-Jet (UT) *Distributed Deep Web Search*

2013-35 Abdallah El Ali (UvA) *Minimal Mobile Human Computer Interaction*

2013-36 Than Lam Hoang (TUe) *Pattern Mining in Data Streams*

2013-37 Dirk Börner (OUN) *Ambient Learning Displays*

2013-38 Eelco den Heijer (VU) *Autonomous Evolutionary Art*

2013-39 Joop de Jong (TUD) *A Method for Enterprise Ontology based Design of Enterprise Information Systems*

2013-40 Pim Nijssen (UM) *Monte-Carlo Tree Search for Multi-Player Games*

2013-41 Jochem Liem (UVA) *Supporting the Conceptual Modelling of Dynamic Systems: A Knowledge Engineering Perspective on Qualitative Reasoning*

2013-42 Léon Planken (TUD) *Algorithms for Simple Temporal Reasoning*

2013-43 Marc Bron (UVA) *Exploration and Contextualization through Interaction and Concepts*

2014-01 Nicola Barile (UU) *Studies in Learning Monotone Models from Data*

2014-02 Fiona Tuliyano (RUN) *Combining System Dynamics with a Domain Modeling Method*

2014-03 Sergio Raul Duarte Torres (UT) *Information Retrieval for Children: Search Behavior and Solutions*

2014-04 Hanna Jochmann-Mannak (UT) *Websites for children: search strategies and interface design – Three studies on children's search performance and evaluation*

2014-05 Jurriaan van Reijsen (UU) *Knowledge Perspectives on Advancing Dynamic Capability*

2014-06 Damian Tamburri (VU) *Supporting Networked Software Development*

2014-07 Arya Adriansyah (TUE) *Aligning Observed and Modeled Behavior*

2014-08 Samur Araujo (TUD) *Data Integration over Distributed and Heterogeneous Data Endpoints*

2014-09 Philip Jackson (UvT) *Toward Human-Level Artificial Intelligence: Representation and Computation of Meaning in Natural Language*

2014-10 Ivan Salvador Razo Zapata (VU) *Service Value Networks*

2014-11 Janneke van der Zwaan (TUD) *An Empathic Virtual Buddy for Social Support*

2014-12 Willem van Willigen (VU) *Look Ma, No Hands: Aspects of Autonomous Vehicle Control*

2014-13 Arlette van Wissen (VU) *Agent-Based Support for Behavior Change: Models and Applications in Health and Safety Domains*

2014-14 Yangyang Shi (TUD) *Language Models With Meta-information*

2014-15 Natalya Mogles (VU) *Agent-Based Analysis and Support of Human Functioning in Complex Socio-Technical Systems: Applications in Safety and Healthcare*

2014-16 Krystyna Milian (VU) *Supporting trial recruitment and design by automatically interpreting eligibility criteria*

2014-17 Kathrin Dentler (VU) *Computing healthcare quality indicators automatically: Secondary Use of Patient Data and Semantic Interoperability*

2014-18 Mattijs Ghijsen (VU) *Methods and Models for the Design and Study of Dynamic Agent Organizations*

2014-19 Vincius Ramos (TUE) *Adaptive Hypermedia Courses: Qualitative and Quantitative Evaluation and Tool Support*

2014-20 Mena Habib (UT) *Named Entity Extraction and Disambiguation for Informal Text: The Missing Link*

2014-21 Kassidy Clark (TUD) *Negotiation and Monitoring in Open Environments*

2014-22 Marieke Peeters (UU) *Personalized Educational Games – Developing agent-supported scenario-based training*

2014-23 Eleftherios Sidirourgos (UvA/CWI) *Space Efficient Indexes for the Big Data Era*

2014-24 Davide Ceolin (VU) *Trusting Semi-structured Web Data*

2014-25 Martijn Lappenschaar (RUN) *New network models for the analysis of disease interaction*

2014-26 Tim Baarslag (TUD) *What to Bid and When to Stop*

2014-27 Rui Jorge Almeida (EUR) *Conditional Density Models Integrating Fuzzy and Probabilistic Representations of Uncertainty*

2014-28 Anna Chmielowiec (VU) *Decentralized k-Clique Matching*

2014-29 Jaap Kabbedijk (UU) *Variability in Multi-Tenant Enterprise Software*

2014-30 Peter de Kock Berenschot (UvT) *Anticipating Criminal Behaviour*

2014-31 Leo van Moergestel (UU) *Agent Technology in Agile Multiparallel Manufacturing and Product Support*

2014-32 Naser Ayat (UVA) *On Entity Resolution in Probabilistic Data*

2014-33 Tesfa Tegegne Asfaw (RUN) *Service Discovery in eHealth*

2014-34 Christina Manteli (VU) *The Effect of Governance in Global Software Development: Analyzing Transactive Memory Systems*

2014-35 Joost van Oijen (UU) *Cognitive Agents in Virtual Worlds: A Middleware Design Approach*

2014-36 Joos Buijs (TUE) *Flexible Evolutionary Algorithms for Mining Structured Process Models*

2014-37 Maral Dadvar (UT) *Experts and Machines United Against Cyberbullying*

Controlling things directly with your mind sounds like magic, yet brain-computer interfaces (BCIs) promise us just that. After decades of research, we have prototypes that show that this form of telekinesis through technology is possible, but not easy...

Most of the current research on BCIs is focused on improving the detection of the 'thoughts' that control these interfaces. This book goes beyond that, looking at the system as a whole. This fresh point of view opens up new ways to dramatically improve BCIs, making them more accurate and more easy to control.

The first half of the book tackles the basic questions: What do users want from BCI control? Do they even know how much control they really have? The second half provides simple post-processing methods. If you are looking for a way to patch up brain-computer interfaces to provide the user better control for less effort, look no further.

About the author

After receiving a BSc degree in computer science, Danny Plass-Oude Bos got introduced to the research area of brain-computer interfaces (BCIs) at the Human Media Interaction group at the University of Twente. For her MSc thesis, she evaluated how the looks of an application affected the performance of people using a BCI. She then went on to earn a PhD in human-computer interaction, continuing to improve the usability of BCIs. Together with Boris Reuderink, she started the company Senzing, which offers software to make it easier for developers to bring quality BCI-controlled applications to the people. As a lecturer, she teaches Software Engineering at Saxion University of Applied Sciences.

CTIT

CTIT Ph.D. Thesis Series No. 14-312
ISSN: 1381-3617

ISBN 978-90-365-3779-7