Die approbierte Originalversion dieser Diplom-/Masterarbeit ist in der Hauptbibliothek der Technischen Universität Wien aufgestellt und zugänglich.
http://www.ub.tuwien.ac.at

The approved original version of this diploma or master thesis is available at the main library of the Vienna University of Technology.
http://www.ub.tuwien.ac.at/eng

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

BCI-Based Cursor Control Using EEG Sensorimotor Rhythms

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Computational Intelligence

eingereicht von

Sebastian Böhm
Matrikelnummer 0442181

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuer: Ao. Univ.-Prof. Mag. Dr. Horst Eidenberger

Wien, 2.10.2013

(Unterschrift Verfasser) (Unterschrift Betreuer)

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

BCI-Based Cursor Control Using EEG Sensorimotor Rhythms

MASTER'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Computational Intelligence

by

Sebastian Böhm
Registration Number 0442181

to the Faculty of Informatics
at the Vienna University of Technology

Supervisor: Ao. Univ.-Prof. Mag. Dr. Horst Eidenberger

Vienna, 02/10/2013

	(Author's Signature)	(Supervisor's Signature)

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.ac.at

Sebastian Böhm
Hetzendorfer Straße 97/4/1
1120 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

(Ort, Datum) (Unterschrift)

Zusammenfassung

Brain-Computer-Interfaces (BCIs) sind Mensch-Maschine-Schnittstellen, die Benutzern ermöglichen, über eine direkte neuronale Schnittstelle mit einem Computer zu kommunizieren. Brain-Computer-Interfaces sind historisch eng mit Anwendungen aus dem Bereich der assistiven Technologie verknüpft, seit einigen Jahren steigt jedoch die Anzahl von Publikationen, die sich mit BCIs für Menschen ohne Behinderungen auseinandersetzen.

Ziel eines Cursor-Control-BCIs ist, die Position eines Cursors mittels Hirnaktivität steuern zu können. Während sich eine Reihe von Publikationen mit Cursor-Control-BCIs befasst, liegt der Fokus dieser Arbeiten primär auf der Steuerung eigens hierfür entworfener Anwendungen.

Die vorliegende Arbeit beschreibt das Design und die Implementierung eines Cursor-Control-BCI-Prototyps, der ein EEG-Sensomotor-Rhythmus-Signal in eindimensionale Cursorbewegungen übersetzt und dessen Ausgangssignal über Standard-Eingabeschnittstellen abgefragt werden kann. Da das beschriebene BCI über Standard-Eingabeschnittstellen zugänglich ist, können bereits existierende Eingabebibliotheken verwendet werden, um experimentelle BCI-Anwendungen zu implementieren. Weiters kann das vorgestellte BCI dazu benutzt werden, bestehende Anwendungen zu steuern, sofern diese Standard-Eingabeschnittstellen verwenden.

Abstract

Brain-computer interfaces (BCIs) are human-computer interaction systems in which users communicate with a computer via a direct neural interface. Brain-computer interfaces have historically been closely associated with assistive technology for persons with disabilities, but the past few years have seen an increase in publications on human-computer interfaces for people without disabilities.

The goal of a cursor control BCI is to control the position of a cursor via brain signals. While there have been a number of publications on cursor control using BCIs, most of them focus on controlling a cursor in custom-built experimental applications.

The present thesis describes the design and implementation of a cursor control BCI prototype that translates EEG sensorimotor input into one-dimensional cursor movement and can be accessed using standard input interfaces. Since the BCI is accessible via standard input interfaces, existing input libraries can be used to create experimental applications and the BCI can be used to control existing applications.

Contents

Chapter 1

Introduction

Brain-computer interfaces (BCIs) are human-computer interaction systems in which users communicate with a computer via a direct neural interface. Such interfaces have a wide range of applications as diverse as thought-controlled artificial limbs, and BCI-based spellers or games.

While the first BCI prototypes were developed as early as in the 1970s, the number of peer-reviewed publications in the research field has exploded in the last fifteen years (cf. Wolpaw and Winter Wolpaw, 2012).

Origins

Historically, BCIs have been closely associated with assistive technology for people with disabilities, with a special focus on patients with locked-in syndrome (LiS) – people who have lost almost all voluntary movement control, often due to brain hemorrhages, strokes, traumatic brain injuries or neurological diseases, such as late stage amyotrophic lateral sclerosis (cf. Bauer et al., 1979). Because classical LiS encompasses almost complete loss of mobility (ibid.), locked-in patients lose their ability to communicate with other people. Although a number of locked-in patients retains some degree of control over voluntary eye movement, this is often restricted to limited vertical eye movement, which is why conventional assistive technologies, such as pure eye-tracking systems, are usually not reliable for patients with LiS.[1]

Applications

Experimental brain-computer interfaces have been successfully used to provide these locked-in patients with a means of communication (see for example Birbaumer et al., 1999; Kübler et al., 1999; Kennedy et al., 2000). While there are still some major problems that have to be addressed in order for BCIs to become a truly viable alternative means of communication in day-to-day

[1] In addition to bilateral palsy of horizontal gaze, which is the most common oculomotor disturbance in LiS patients, various other oculomotor disturbances are documented in Bauer et al. (1979, p. 86).

use, the last decade's advances already indicate the potential brain-computer interfaces hold for application in assistive technology.

Cursor control interfaces are a well-known type of BCI, whose goal it is to let test subjects control the position of an on-screen cursor via a non-muscular interface. This type of BCI has various applications ranging from neural spellers to artificial limb control, applications that readily lend themselves to the motorimagery tasks often involved in such interfaces.

Motivation While the majority of BCI research still seems to be directed towards potential assistive technology applications, the past few years have seen an increase in publications on recreational brain-computer interface prototypes for people without disabilities, such as BCI-based games. Game control via BCI is an interesting topic because it provides a promising testing ground for relatively low-latency online interactive BCI control paradigms used in general purpose navigation and cursor control tasks.

There have been a number of publications on game control using BCI, but most of them focus either on simple custom-built experimental games or custom BCI control applications merely supplementing conventional input mechanisms for existing games. Plass-Oude Bos et al. (2010) provide an extensive survey of publications on existing BCI-based games and BCI-based game control. As mentioned before, the vast majority of low-latency game control implementations targets either custom-built games or only a small subset of the controls of existing games via proprietary extensions (for instance *AlphaWOW*, qtd. ibid.). By contrast, the focus of the present master's thesis will be the design and implementation of an experimental general purpose cursor control mechanism for existing applications, such as games.

Goals Concretely, the goal of the present master's thesis is to build a BCI that translates EEG input into cursor movement and is accessible via standard input interfaces. This BCI could then be used in order to control existing applications and facilitate various experimental setups. By providing cursor movement signals via standard input interfaces, existing libraries, such as SDL or DirectInput/XInput, could be leveraged when building new BCI applications. The application should provide a simple control interface that allows users to move a cursor on a one-dimensional axis, for instance by imagined movement or resting periods of their left or right hands.

Overview The first part of the thesis provides a discussion of the theoretical background of brain-computer interfaces, covering the fundamentals of EEG as well as the basic principles of BCIs.

The second part focuses on methodological issues of BCI system design, specifically the use of sensorimotor rhythms for cursor control applications and machine learning foundations for brain-computer interfaces.

Finally, the third part discusses the cursor control brain-computer interface developed on the basis of the principles laid out in the preceding chapters and evaluates the software architecture, implementation, and performance characteristics of the resulting interface.

I Theoretical background

Chapter 2

Fundamentals of EEG

Electroencephalography (EEG) is currently probably the most widely used neuroimaging method and is by far the most pervasive neuroimaging method in brain-computer interface contexts. This chapter provides an overview of the fundamentals of EEG.

2.1 Origins

Although the origins of Electroencephalography are inseparably linked to Hans Berger's publications (see for example Berger, 1929), the groundwork necessary for the discovery of EEG dates back even earlier.

19th century

Niedermeyer (1999a) dates the beginnings of electrophysiology as early as the mid-nineteenth century, when William Thompson (Lord Kelvin) refined the galvanometer and Carlo Matteucci and Emil Du Bois-Reymond proposed an electrophysiological basis of the human nervous system, coining the term "negative variation", which was later replaced by Hermann von Helmholtz' concept of the "action current" (ibid.).

Richard Caton

In 1875, Richard Caton conducted experiments to explore electrical activity in the brains of rabbits and monkeys by using optical amplification and a Thompson galvanometer. Using this setup, Caton was able to show that "feeble currents of varying direction pass through the multiplier when the electrodes are placed on two points of the external surface [of the skull]" (Caton, 1875; qtd. in Niedermeyer, 1999a, p. 2). Caton also discovered a form of evoked potential, noting that localised functional activity induces negative currents, thereby laying the groundwork for some of the most important concepts used in current BCI research.

Eastern Europe

Around the same time that Caton and his contemporaries in Western Europe explored electrical activity of the brain, researchers in Central and Eastern Europe pioneered electrical stimulation of the brain. Gustav Fritsch and Eduard Hitzig discovered that the human cerebral cortex could be electrically stimulated after observing spontaneous muscle contractions while attending to a soldier's open brain wound in the Prussian-Danish war (Niedermeyer, 1999a). This discovery prompted various studies on electrical stimulation of the brain, especially at Eastern European universities. Among the studies conducted in Eastern Europe, "Investigations into the Physiology of the Brain", the thesis of Vasili Yakovlevich Danilevsky (Danilevsky, 1877; cited in Niedermeyer, 1999a), stands out insofar, as he documented both electrical stimulation of the brain as well as spontaneous electrical activity of the brain and seems to have been among the first to explore both phenomena in conjunction and to emphasise their close relation. The ability to artificially induce spontaneous muscle contractions led to the realisation that electrical activity is not merely an artefact of mental activity, but constitutes one of the principal mechanisms of the human brain.

20th century

At the beginning of the twentieth century, electrophysiological research began to stagnate in Eastern Europe. Niedermeyer (1999a) attributes this stagnation to the Soviet regime's interest in conditioned reflexes due to the perceived ideological fit of Ivan Petrovich Pavlov's research (noting that Pavlov himself was a critic of the Soviet regime): "The Pavlovian concept was closer to the ideology of dialectic materialism and this maxim with all its intolerant dogmatism outlasted Pavlov's death by two decades. This ideopolitically governed form of neuroscience stifled all progress of customary neurophysiology" (ibid, p. 3). The political governance described by Niedermeyer resulted in a marked decline of neurophysiological research in Eastern Europe in the early first half of the twentieth century, a decline from which Eastern European neuroscience did not recover until after Hans Berger's publications.

Hans Berger

Hans Berger, the researcher who is probably most often associated with the discovery of EEG, originally studied electrical activity in dogs' brains, but failed to produce meaningful results. Thus, from the early 1920s on, Berger studied human EEG, using simple string galvanometers. In 1926, he started using a Siemens double coil galvanometer and nonpolarisable pad electrodes in a setup attaining a sensitivity of roughly 130 μV/cm.

In 1929, Berger published a series of landmark articles, starting with Berger (1929), in which he described a number of experiments in which he had managed to record human EEG tracings on photographic paper. The recordings were between one and three minutes long and used a relatively simple bipolar fronto-occipital montage, in which two chlorinated silver needle electrodes were placed in the front and back of Berger's test subjects' skulls. Berger recorded EEG, ECG and a 10 Hz time marker simultaneously and was able to observe rhythmic oscillations, which he called "alpha waves", as well as rhythmic attenuation of these waves due to mental activity.

In 1932, J.F. Toennies constructed the first differential EEG preamplifier, which he used for one of the first multichannel EEG setups. In 1935, as EEG research efforts began to shift from Europe to the United States, J.F. Toennies was visited in Germany by Frederic Gibbs who immediately recognised the potential of this new type of setup and started to incorporate Toennies' innovations into his own research. Back in the USA, Gibbs commissioned an improved EEG preamplifier that already boasted multichannel support, an ink writer and paper recording. With this step towards modern EEG instrumentation, all important components of the EEG equipment, as it was used in the following decades, were in place. FREDERIC GIBBS

Although a modern digital EEG setup is of course quite different from the amplifier that Gibbs commissioned, and there has been considerable progress in the field of EEG research since Berger's publications,[1] the basic working principles of EEG used in these pioneering works remain unchanged.

2.2 Working principles

EEG measures electric fields caused by ionic currents generated by biochemical sources due to bioelectrical activity in the human brain (Lopes da Silva and Van Rotterdam, 1999). These changes in the electric field can be measured as voltage fluctuations via electrodes placed on the scalp.

While Berger (1929) viewed EEG as a "measure of global cortical activity" (Reilly, 1999, p. 122) and simply placed EEG electrodes at the front and back of his patients' heads (ibid.), in modern applications EEG electrodes are placed in such a way that they allow localised EEG analysis.

In order to be able to compare and reproduce EEG electrode locations, several standards have been proposed, the most prevalent of which is the international 10-20 system (see Figure 2.1) proposed in 1958 by the International Federation of Societies for Electroencephalography and Clinical Neurophysiology (Jasper, 1958; cited in Reilly, 1999). The 10-20 system describes 21 potential electrode locations using bony landmarks as reference locations. Electrode locations are obtained by measuring scalp distances between nasion (N_Z) and inion (I_Z), and 10-20 SYSTEM

[1] Niedermeyer (1999a) provides a more detailed account of historical advances in EEG research.

the two preauricular points (A_1, A_2), respectively, marking sites with a distance of either 10% or 20% of the obtained front-back and left-right measurements[2] – hence the name "10-20 system". Montages with higher density typically use electrode distances of 10% of the total distances for all sites; the resulting placement system is often called the 10-10 system (see Figure 2.2).

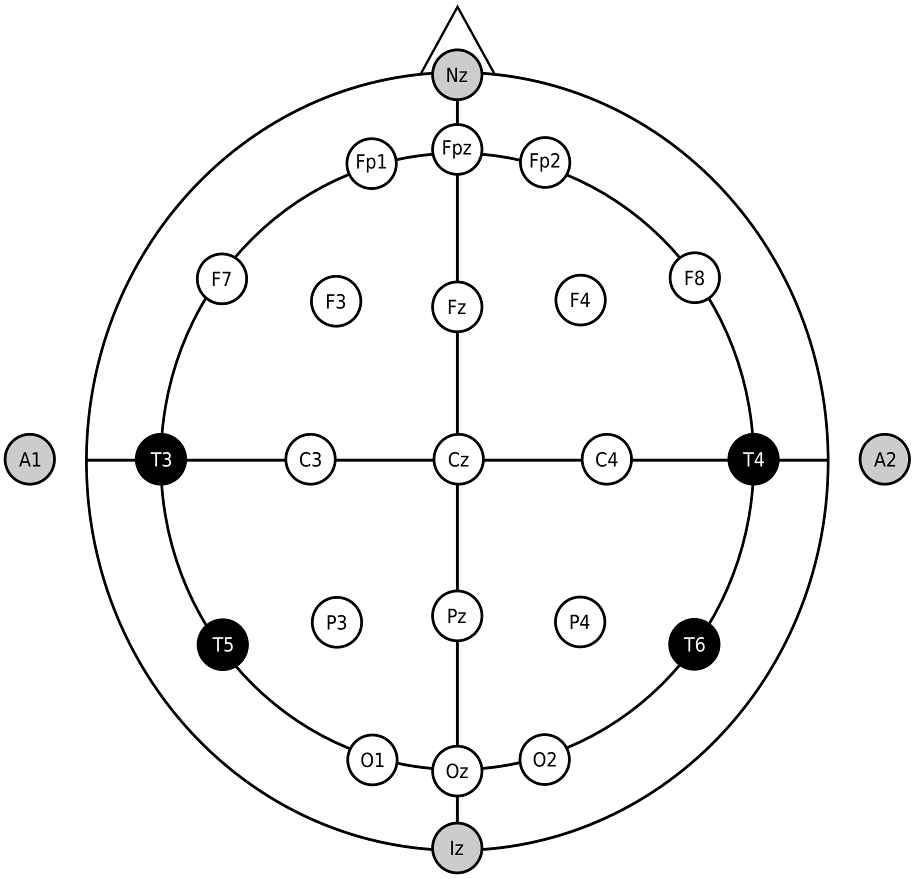

Figure 2.1: Electrode locations of the 10-20 system (from Bernard Marius 't Hart, 2011, modified).

When recording EEG, two major categories of EEG electrode montages are distinguished (cf. Reilly, 1999): *reference montages* and *non-reference montages*.

[2]The exact distances for both, front-back (N_Z-I_Z) and left-right (A_1-A_2) axes are 10%, 20%, 20%, 20%, 20% and 10% of the total distance.

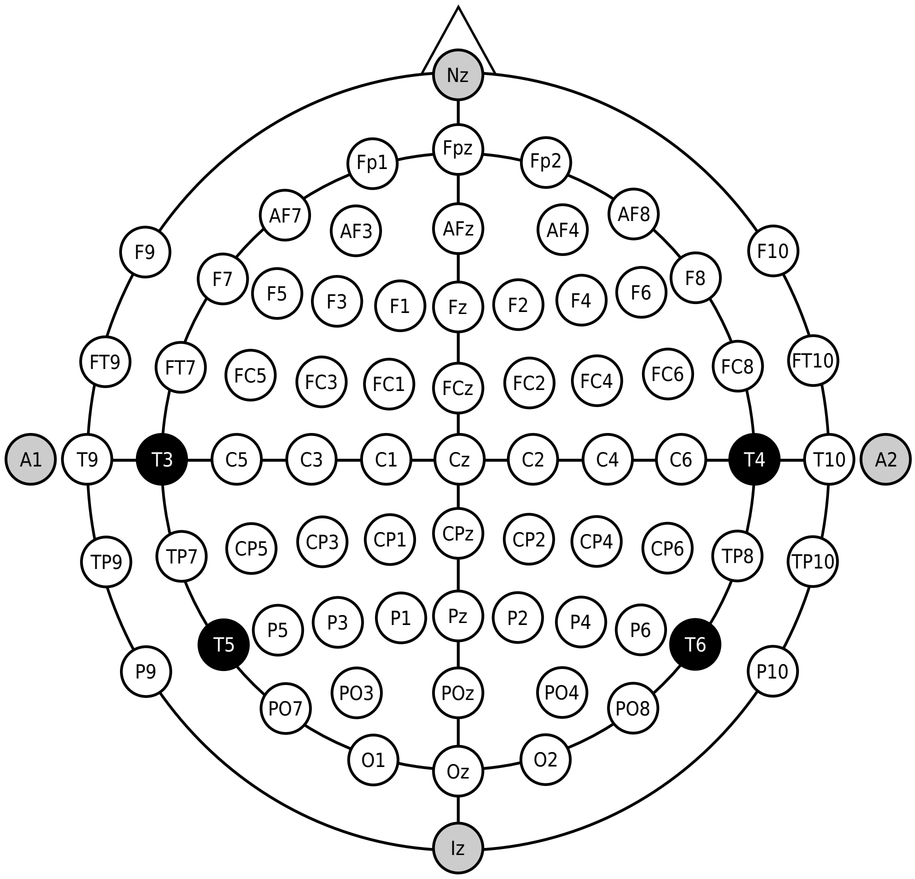

Figure 2.2: Electrode locations of the 10-10 system (from Bernard Marius 't Hart, 2011, modified).

In reference montages, all electrodes are referenced against one of two common electrodes – one on each side of the head. The voltage between each electrode and the common reference electrode is measured. Setups where the reference electrode is placed in a location with minimal brain activity, such as the earlobe, are sometimes called "monopolar" montages in the literature, although Reilly points out that these montages are not strictly speaking "monopolar" because it is almost impossible to obtain totally inactive electrodes, which would be needed for a truly monopolar setup. For more reliable source location, various spatial filters, such as common average reference filters or Laplacian filters, are applied.

Reference montages

In contrast to this, in non-reference montages, or more specifically in bipolar variants thereof, electrodes are placed in pairwise scalp-to-scalp linkage (cf. Reilly, 1999). Typically these electrode pairs are placed relatively close to

Non-reference montages

the location of interest. For example to examine C_3, both electrodes could be placed in a location directly adjacent to C_3 in the international 10-20 system: one electrode might be placed at FC_3, and the other electrode might be placed at CP_3, depending on the specific requirements.

As indicated above, EEG is widely used for both clinical applications and brain-computer interfaces. The next section addresses the characteristics of EEG relevant specifically for BCI applications.

2.3 Applications and limitations

In the past decades, EEG has become a tool that is routinely used in clinical diagnostics and various other clinical applications, as well as experimental applications, such as brain-computer interfaces.

While recording EEG data was very complicated in Berger's days and its use was restricted to research due to the technological constraints of the time, EEG has become a diagnostic commodity in the last decades. Whereas the tools used by Berger and his contemporaries had to be specially adapted in order to achieve the sensitivities needed to record EEG, nowadays EEG electrodes, digital biosignal amplifiers, and EEG recording software are comparatively cheap and widely used for various applications ranging from clinical applications, such as diagnosis of neurological disorders, to research and assistive technology applications, such as brain-computer interfaces.

Generally speaking, EEG is well suited for applications that require high temporal resolution and low-latency measurements, but which do not rely on high spatial resolution.

Advantages

EEG is an interesting tool for BCI researchers not only because of its relative affordability and wide availability but also because of various other desirable properties. Compared to other methods, EEG has a very low latency and a high temporal resolution – while most BCI researchers seem to use sample rates between 128 and 512 Hz, modern biosignal amplifiers are capable of achieving sample rates as high as 20 kHz. Latencies in the millisecond range and relatively high sample rates are especially important when designing online control systems and analysing high frequency rhythms, both of which are priorities to a large proportion of the BCI research community.

In contrast to methods such as fMRI or PET (see Section 2.4), EEG equipment requires little space and is highly portable, allowing for mobile data acquisition. Finally, EEG is completely non-invasive. Although there are several invasive or partially invasive BCI systems, the medical risks concomitant with such interfaces make invasive methods unfeasible for most research groups.

Limitations

As far as the disadvantages of EEG are concerned, the main issues generally are poor spatial resolution and a low signal-to-noise ratio. Spatial resolution is a conceptual issue of EEG, because each channel has to capture the field potential of a whole population of neurons, which is equal to the sum of field potentials of all individual neurons involved (Lopes da Silva and Van Rotterdam, 1999). Therefore, the spatial resolution of EEG is limited by both interelectrode distance and the receptive field size of the individual electrodes. Blankertz and Müller (2009) cite a simulation by Nunez et al. (1997), which predicts that "only about half the contribution to the outer surface (scalp) potential comes from sources within a 3 cm radius of the 'recording' electrode (even assuming no reference electrode contribution)" (ibid., p. 200). These problems can be somewhat mitigated by an appropriate number of electrodes and spatial filtering, but will – to some degree – always be present in conventional EEG-based neuroimaging. Both, spatial resolution and the relatively low signal-to-noise ratio due to artefacts are improved by intracranial electrode placement (intracranial EEG or Electrocorticography/ECoG). However, the surgical intervention required for this type of setup is of course not feasible for applications where noninvasive neuroimaging is desired.

Artefacts

EEG data is typically heavily contaminated by artefacts, voltage fluctuations picked up by scalp electrodes whose origin is not the test subject's brain (Zschocke, 1995). Therefore artefact detection and control is one of the most important problems of clinical EEG. Typically, two types of artefacts are distinguished: biological artefacts, originating from the patient's body, and environmental artefacts, originating from sources outside the patient's body.

Examples of biological artefacts are eye-induced artefacts, such as eye movement and blink artefacts, cardiac (ECG) artefacts, muscular (EMG) artefacts, movement artefacts, and glossokinetic (tongue potential) artefacts. Environmental artefacts include mains hum due to grounding errors and various artefacts caused by faulty equipment, such as defective electrodes or amplifiers.

Avoiding artefacts wherever possible is an important task in the design of EEG-based BCI setups – especially when recording data to be used for training adaptive BCI systems. Although artefact control and correction have a tremendous influence on BCI performance and the quality of neuroimaging data, the procedures used for artefact control are often poorly documented – in many cases because of epoch rejection processes geared towards manual artefact control by qualified EEG professionals, which are not considered a part of the BCI.

While EEG is currently the most prevalent method used for neuroimaging, various other approaches have been used for brain-computer interfaces.

2.4 Other approaches to neuroimaging

Wolpaw et al. (2002) and Tan and Nijholt (2010b) provide an overview of non-invasive neuroimaging methods other than EEG that could and have been used for BCI applications, amongst them magnetoencephalography (MEG), positron emission tomography (PET), single photon emission computed spectography (SPECT), functional magnetic resonance imaging (fMRI), and optical imaging methods, such as functional near-infrared spectroscopy (fNIRS) neuroimaging.

MEG Whereas EEG measures electric potentials of the brain, magnetoencephalography (MEG) measures electric fields generated by the brain's bioelectrical activity. Since the human skull does not significantly influence the power of these magnetic fields, MEG is more sensitive than EEG and allows for much deeper imaging than EEG does. The main disadvantage of MEG is that the equipment requires highly sensitive magnetometers, such as superconducting quantum interference device (SQUID) arrays, and is therefore bulky and very expensive (cf. Tan and Nijholt, 2010b).

PET In positron emission tomography (PET) imaging, a patient's blood flow in the brain is calculated by injecting patients with a positron-emitting radionuclide tracer and measuring indirect gamma ray emissions of the tracer in the patient's body. PET has a much higher spatial resolution than EEG, but since it measures test subjects' blood flow, PET has a much lower time resolution than EEG and the equipment necessary to perform PET scans is – again – quite bulky and expensive. Furthermore, because of the test subject's exposition to radioactive substances, PET is not suitable for regular prolonged use (cf. Tan and Nijholt, 2010b).

SPECT From a functional standpoint, single photon emission computed spectography (SPECT) is very similar to PET,[3] but instead of measuring positron emissions of a radioactive tracer via gamma rays, photomultiplier tubes are used in order to detect single photons generated by gamma rays originating from the tracer. While SPECT has much lower spatial and temporal resolution than PET, compared to PET the observational time window can be widened significantly (Rahmim and Zaidi, 2008) and SPECT equipment is typically slightly less expensive than PET equipment (cf. Tan and Nijholt, 2010b).

FMRI Similar to SPECT and PET, in functional magnetic resonance imaging (fMRI) the subject's blood flow is monitored, but while SPECT and PET require the injection of a radioactive tracer, in fMRI changes in blood magnetisation

[3] Rahmim and Zaidi (2008) provide an in-depth comparison of differences and similarities between PET and SPECT and both methods' respective advantages and limitations.

between oxygenated and deoxygenated blood are used to track the blood flow in the patient's brain (Tan and Nijholt, 2010b). While fMRI has a rather low temporal resolution (5-8 seconds) and relies on superconducting magnets which make fMRI equipment bulky and expensive, it provides spatial resolutions of up to 1mm accuracy, making it one of the most spatially precise functional neuroimaging methods currently available.

Functional near-infrared spectroscopy (fNIRS) is a relatively new method that relies on optical methods to measure the absorption and scattering of near-infrared light on the cortex in order to measure changes in tissue oxygenation or changes in active neurons' neuronal membranes. While the temporal resolution is relatively low when measuring oxygenation changes, time resolution can be increased by measuring event-related responses during neuron firing. Although the spatial resolution of fNIRS is slightly lower than that of fMRI and it can only be used to measure cortical (surface) activity, fNIRS equipment is also by far less expensive than the equipment required by the other neuroimaging methods discussed above (excepting EEG, of course), vastly more portable, and safe for extended use due to its use of non-ionizing light (Tan and Nijholt, 2010b). FNIRS

In summary, while most of the methods above have at some point been used in order to implement experimental brain-computer interfaces, the vast majority of BCIs still relies on EEG due to its hight time resolution, inexpensiveness, easy availability, and portability (cf. Wolpaw et al., 2002). However, it should be noted that fNIRS seems to hold great promise as a neuroimaging technique for future BCIs, even though there are currently relatively few publications utilising fNIRS for BCI applications (see for example Sitaram et al., 2007; Coyle et al., 2007).

Having established that EEG is well suited for BCI, the next chapter focuses on the basic principles of brain-computer interfaces.

Chapter 3

Principles of brain-computer interfaces

Brain-computer interfaces can be implemented in a number of ways. This chapter provides an introduction to the basic working principles of brain-computer interfaces and an overview of current methods.

3.1 Preliminaries

A brain-computer interface (BCI) is a direct neural interface to a computer that "monitors brain activity and detects certain brain patterns that are interpreted and translated to commands for communication or control tasks" (Graimann et al., 2010, p. 22). Wolpaw et al. (2002, p. 769) define BCIs as "communication system[s] 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".

In other words, a brain-computer interface transforms a user's brain activity into a communication or control signal without relying on peripheral nerves and muscles to carry messages. While other approaches have been used, in practice most BCI systems rely on processing EEG data.

Applications

Historically, brain-computer interfaces have been closely associated with assistive technology for people with disabilities. When the neural pathways that control voluntary muscle movement in humans are impaired, BCIs can be used in order to provide affected subjects with alternative means of communication. This is especially important for patients with locked-in syndrome, who cannot communicate with others due to complete paralysis of almost all muscles in their body, and for whom brain-computer interface-based speller applications are sometimes the only means of communication with their environment. BCIs have also been used successfully to implement artificial limb control systems for people with tetraplegia (cf. Hochberg et al., 2012; Collinger et al., 2013).

Training BCI use

However, it is important to bear in mind that the "mind-reading" or "wire-tapping" analogies (cf. Wolpaw et al., 2002) frequently invoked in mainstream media coverage of brain-computer interfaces are misleading at best. Wolpaw et al. criticise this "hyperbolic and often misleading media attention that tends to generate unrealistic expectations in the public and skepticism in other researchers" (ibid., p. 767), noting that even BCI-types that do not rely excessively on subject training (such as P300-based systems) typically exhibit significantly higher information transfer rates when operated by more experienced users, which illustrates how unfitting the aforementioned "mind-reading" and "wire-tapping" analogies actually are.

BCI use is neither "mind-reading" nor the simple act of intercepting random electrophysiological signals and translating them directly into arbitrary commands. In reality, BCI use is a skill that has to be trained and actively acquired in order to be able to successfully use a very limited set of commands (cf. Wolpaw et al., 2002).

While there are some simple brain-computer interfaces for the consumer market, such as the systems developed by Emotiv Systems and NeuroSky, most BCI use – outside of research or lab settings on the one hand and simplistic recreational applications on the other hand – currently seems to be restricted to assistive technologies for people with severe disabilities. This is of course hardly surprising, considering the relative difficulty of BCI use in day-to-day operation and the fact that even locked-in patients often retain voluntary eye movement control and can therefore sometimes use simpler technologies, such as eye-tracking devices.

As suggested by current applications, BCIs are still a developing research field. The next section gives a short overview of the timeline of brain-computer interfaces.

3.2 History

Brain-computer interfaces are a relatively new field of research, but while most BCI systems currently in use were not developed until the 1990s, the foundation for modern BCIs was laid much earlier. The basics of alpha wave desynchronisation were already discussed in Berger (1929) and the principal idea of a BCI seems to have been around since as early as 1938, when Herbert Jasper, a neuroscientist colleague of Hans Berger's, sent Berger a holiday card depicting an early rendering of a BCI (cf. Wolpaw and Winter Wolpaw, 2012, p. 3).

The significance of mu rhythms, on the other hand, was not discovered until Gastaut (1958) and the P300 response was first analysed in the 1960s (Grey Walter et al., 1964; Sutton et al., 1965; Donchin and Smith, 1970).

Early BCI experiments

In the 1970s, a first simple BCI using visually evoked potentials was developed by Vidal (1973, 1977), and Fetz and Finocchio (1975) conducted several invasive studies that demonstrated the possibility of using operant conditioning methods in order to train monkeys to control firing rates of individual cortical neurons.

While these early BCIs mostly concentrated on dependent BCIs (Vidal) or invasive BCIs in animal experiments (Fetz and Finocchio), in the 1980s one of the first modern noninvasive independent BCIs was implemented by Farwell and Donchin (1988), who introduced the modern P300 speller paradigm. Voluntary self-regulated control of slow cortical potentials (SCP) was documented in Elbert et al. (1980) and subsequently used to implement brain-computer interfaces utilising SCPs over the next two decades, resulting in modern interfaces such as those documented in Birbaumer et al. (1999) and Kübler et al. (1999).

Even though the first modern mu rhythm BCI was introduced in the early 1990s (Wolpaw et al., 1991), there were relatively few peer-reviewed BCI papers published per year until the end of the 1990s. Publication numbers only increased slowly around the turn of the century and are now up from a few dozen peer reviewed papers per year in the early 2000s to hundreds of publications per year (cf. Wolpaw and Winter Wolpaw, 2012, p. 3), with recent efforts focusing on creating more robust, higher accuracy brain-computer interfaces.

Historically, there has been a wide range of different methods in BCI research. The next section examines methods used in current BCIs.

3.3 Methods

There is a wide variety of brain-computer interfaces currently being researched. However, most of these interfaces are based on a limited number of methods: based on the electrophysiological signals used, Wolpaw et al. (2002) distinguish five groups of BCIs: Event-related potential (ERP) BCIs, visual evoked potential (VEP) BCIs, slow cortical potential (SCP) BCIs, sensorimotor rhythm (SMR) BCIs, and cortical neuronal BCIs.

Event-related potentials

Event-related potentials (ERPs) are among the best-known mechanisms in BCI research. ERPs are direct time-locked responses to a specific stimulus. A typical example of an ERP is the P300 response, which is observable as a positive peak in a test subject's EEG that occurs with a latency of roughly 300ms (hence the name "P300") and is thought to be a result of stimulus categorisation on the test subject's side. The P300 is usually evoked using the so-called "oddball" paradigm (cf. Farwell and Donchin, 1988), in which subjects are presented with a sequence of stimuli and have to focus on a specific low-probability target stimulus. When this target stimulus is finally presented, test subjects react with a P300 response. The user's task in the P300 speller interface paradigm introduced by Farwell and Donchin is to concentrate on a specific letter of the alphabet. The user is then shown a sequence of random letters and whenever the BCI detects a P300 wave, the corresponding letter is chosen as the next input letter by the speller. ERP-based BCIs typically require little or no user training and are relatively simple to implement, but are relatively inflexible in terms of their experimental setup due to their reliance on time-locked responses.

Visual evoked potentials

Visual evoked potentials (VEPs) are potential changes in the visual cortex resulting from exposition of test subjects to visual stimuli flashing at different rates. The resulting potential changes in the brain have similar frequencies as those of the stimuli the test subject is exposed to and can therefore be used in order to detect the test subjects gaze direction. However, this type of BCI depends on its users ability to voluntarily change their eye gaze towards specific targets and is therefore not a truly independent brain-computer interface. Furthermore, simple eye trackers can be used for similar purposes without introducing most of the complexities of a BCI system.

Slow cortical potentials

Slow cortical potentials (SCPs) are slow potential shifts originating in the human cortex. They occur over time spans as long as 0.5-10s. According to Wolpaw et al. (2002), negative SCPs are usually associated with cortical activation (for example movement), whereas positive potentials are associated with reduced cortical activation. In brain-computer interface research, this effect can be utilised by letting test subject learn to control their SCPs in order to control an on-screen cursor for selection tasks. While SCP-based BCIs are suitable for selection tasks where the cursor only has to be moved over a time span of a few seconds, the associated potential shifts cannot be maintained indefinitely, which is why SCPs are usually not used for pure cursor control tasks where the cursor has to be controlled over longer time spans.

Sensorimotor rhythms (SMRs) are brain wave rhythms of the sensorimotor cortex which are associated with motor tasks. SMR are idle rhythms, that is they are attenuated when test subjects move their hands or feet or imagine these movements. Since the rhythmic attenuation is localised and usually stronger in the brain hemisphere contralateral to the side on which the movement occurs, SMR can be used to distinguish between different types of movement, such as "left hand", "right hand" or "feet". Because SMR-BCIs do not necessarily rely on external stimuli or fixed timing, their use can be quite flexible. While SMR are only present in some humans, most test subjects can learn to control their sensorimotor rhythms. However, this type of operant conditioning usually requires a significant amount of training time on the user's side, which is discussed in more detail in Chapter 4.

Sensorimotor rhythms

Cortical neuronal BCIs are a type of invasive BCI for which cone electrodes, which can detect action potentials of single cortical neurons, are implemented in the patient's outer neocortex. Users of these BCIs can learn to control the firing rates of the neurons adjacent to the sensor electrodes (Kennedy et al., 2000). In BCI applications this type of interface is typically used for cursor control tasks or letter selection. While cortical neuronal BCIs are quite flexible and seem to perform quite well, they require surgery to implant electrodes in the user's cortex and are therefore not suitable for simple research applications with healthy human test subjects.

Cortical neuronal BCIs

Most current BCIs use one of the methods discussed above. However, in order to fully describe a brain-computer interface system, simply stating the method it is based on is not sufficient. In addition to the method itself, the exact setup and experimental paradigms underlying the BCI have to be considered as well. The next section tries to establish a simple taxonomy of these BCI setups and paradigms.

3.4 Taxonomy

When comparing different types of brain-computer interfaces, there are a number of different ways to categorise systems, each of which focuses on a different set of aspects of BCI design or operation.

For example, a rather obvious categorisation would be to characterise BCIs by degree of invasiveness. Typically BCIs are classified as *invasive*, *noninvasive* or sometimes *partially invasive*. The term "partially invasive" is used to describe procedures where data is captured from the surface of the brain (as opposed to the scalp), but the implants reside outside to the brain itself. Capturing EEG data via scalp electrodes is a typical example of noninvasive

Invasiveness

BCI approaches, whereas the BrainGate interface by Hochberg et al. (2006), for which an implant was placed directly inside the patient's motor cortex, is an example for an invasive interface. Electrocorticography-based BCIs are the partially invasive equivalent to noninvasive EEG-based BCIs, where data is captured directly from the surface of the cortex. Thus, ECoG is sometimes also called "intracranial EEG". Invasive methods are usually associated with higher spatial resolution, higher signal to noise ratio, and overall better signal quality than noninvasive methods but due to ethical and medical reasons, most BCI research nowadays concentrates on noninvasive BCI systems. Invasive or partially invasive methods are mostly found either in highly specialised BCIs which are actively used by severely handicapped people or in research prototypes used in animal experiments.

Dependent vs. independent

Another simple way of classifying brain-computer interfaces is to group them by neural pathway activity and thus to categorise them into *dependent* and *independent* systems (cf. Wolpaw et al., 2002). In an independent BCI neural pathway activity is needed neither for message transmission nor for control signal generation. Currently most brain-computer interfaces are independent BCIs, that is they do not depend in any way on any of the brain's normal output pathways. Typical examples of independent interfaces include SCP, ERP and SMR BCIs. Dependent BCIs, on the other hand rely on some form of neural pathway activity for message generation, but not for message transmission. The only dependent BCI type that is currently widely used is the SSVEP BCI, where users have to focus on stimuli flashing with different frequencies. Because test subjects have to move their eyes in order to use this type of BCI, their regular neural output pathways have to be active, even if they are not needed for message transmission. SSVEP BCIs are therefore classified as dependent interfaces.

Control paradigm

Finally, brain-computer interfaces can be classified by the basic control paradigms underlying their design, such as in Moore Jackson and Mappus (2010).

Moore Jackson and Mappus classify brain-computer interfaces according to their control task paradigm into *exogenous* (evoked) paradigm BCIs and *endogenous* (self-generated) paradigm BCIs. Exogenous paradigm BCIs rely on external stimuli in order to cause brain signal changes. A typical example would be P300 spellers or SSVEP interfaces, where the user has to be exposed to specific stimuli in order for their brain signal to change in a significant way. Endogenous paradigm BCIs do not rely on external stimuli but instead require the user to perform some kind of mental task in order to change their brain signals.[1]

Another kind of paradigmatic classification of BCIs given in Moore Jackson and Mappus (2010) is that according to the BCI's dialog initiative paradigm, where interfaces are categorised as either *synchronous* interfaces or *asynchronous* interfaces. Synchronous BCIs are systems in which interaction with the BCI is only possible in fixed time windows, for example when triggered by a cue. By contrast, in an asynchronous BCI the user determines timing or initiates the control signal. This distinction is examined in some more detail in Chapter 4, where asynchronous and synchronous SMR BCIs are compared. It should be noted that the terms "synchronous" and "asynchronous" are not always used consistently in the literature. In some publications the term "asynchronous" is only used for interfaces where the user actually initiates BCI control, whereas in others it is used for all interfaces where control is not restricted to fixed time windows. In this thesis, the latter definition is used unless stated otherwise.

Of course there is no such thing as the best type of BCI or the best BCI paradigm, merely an appropriate choice of BCI for a certain application (or perhaps equally often an appropriate application for a given BCI). A number of considerations have to be taken into account when choosing a BCI, some of which are explained in more detail in the course of Chapter 4.

Apart from the advantages and disadvantages of various methods discussed in Section 3.3 and the characteristics accompanying certain interface paradigms, the concept of BCI illiteracy is an important factor when comparing different BCI approaches.

3.5 BCI illiteracy

Brain-computer interfaces allow communication and control for many users, but some test subjects never achieve meaningful BCI control. This phenomenon is called *BCI illiteracy*.

There are a lot of trivial reasons why a BCI might not work for a given user that can be easily solved by more training or by switching to different mental tasks for BCI operation. However, in a significant number of cases users cannot use a BCI that relies on a particular brain signal or even any BCI at all.

[1] Typically these BCIs present users with some kind of feedback indicator. While such a feedback indicator is of course an external stimulus, it only helps the user to enhance self-generated signal changes instead of providing evoked signal changes by itself, as would be the case with exogenous paradigm interfaces.

For example, Allison and Neuper report that about 10% of test subjects do not produce a robust P300 and therefore do not profit from training, alternate P300 tasks or better signal processing in any way. Instead, researchers should try to use alternative brain signals to supply these users with effective BCIs. For example, P300-illiterate test subjects might be able to use SMR-based BCIs and vice versa.

Allison and Neuper (2010) point out that while all humans share the same cortical processing systems with similar functional subdivisions and locations, there are strong interpersonal variations in brain structure. If some users cannot achieve control with a certain type of BCI, this might be due to the fact that the neuronal systems they use for control simply do not produce signals detectable by the neuroimaging methods used in the BCI. This indicates neither health problems on the users side nor inactivity of the corresponding neural populations; the signal produced might simply not be detectable by the chosen neuroimaging technology. For instance, "[t]he key neural population may be located in a sulcus, or too deep for EEG electrodes, or too close to another, louder group of neurons" (ibid., p. 36).

The problem of BCI illiteracy again demonstrates how misleading the aforementioned mainstream media "mind-reading" analogy actually is. Surely, if BCIs were actually capable of "reading test subjects' minds" the way portrayed in the media, researchers wouldn't have to put so much effort into identifying those thoughts that they can actually tap into.

Having outlined the basic working principles of brain-computer interfacing and some problems of current BCIs, the next chapter turns to sensorimotor rhythms and their use in cursor control tasks.

II Methodological approach

Chapter 4

Sensorimotor rhythms for cursor control

Sensorimotor rhythm-based brain-computer interfaces are one of the principal approaches used to implement neural cursor control interfaces. This chapter provides an introduction to SMR and their use in cursor control tasks.

4.1 Characteristics of SMR

LOCATION AND FREQUENCY

Sensorimotor rhythms, such as the Rolandic mu rhythm, are brain wave rhythms with a frequency and amplitude similar to that of classical posterior alpha rhythms as described by Berger (1929). Whereas Berger's alpha rhythms are typically most prominent in posterior locations, mu rhythms are localised over the sensorimotor cortex which corresponds to the electrode locations C_3 and C_4 in the international 10-20 system (see Section 2.2).

Mu rhythm frequency usually ranges from $8 - 12$ Hz, but since sensorimotor activity is sometimes also accompanied by less pronounced beta activity in the range of $18 - 26$ Hz, SMR frequency is often simply defined as the frequency band from $8 - 30$ Hz.

FUNCTION

Functionally, sensorimotor rhythms play an important role in voluntary movement control and planning. While Rolandic mu rhythms have been known since the late 1930s as "precentral alpha rhythms" (Jasper and Andrews, 1938; cited in Niedermeyer, 1997) or "high voltage rolandic alpha" (Schütz and Müller, 1951; cited in Niedermeyer, 1997), their significance for sensorimotor functions was not discovered until the late 1950s (Gastaut, 1958).

Sensory stimuli as well as hand movement planning and execution can result in an attenuation of sensorimotor rhythms (Pfurtscheller and Lopes da

Silva, 1999). These synchrony changes are called event-related synchronisation (ERD) and event-related desynchronisation (ERS).

Event-related SMR desynchronisation during externally paced movement was described as early as in Pfurtscheller and Aranibar (1978), where test subjects were instructed to press a button when cued by an auditive stimulus. During movement initiation, Pfurtscheller and Aranibar observed a decrease in mu activity in some test subjects and a decrease of occipital alpha band power in all test subjects.

Applications In more recent work, this event-related desynchronisation of mu rhythms has been used to distinguish between movements of the left and the right hand: typically, hand movement leads to contralateral ERD and ipsilateral ERS (cf. Pfurtscheller et al., 1997). That is to say, if test subjects move their left hand, mu band power over the central left sensorimotor area increases while mu band power over the central right sensorimotor area decreases. The reverse is true for right hand movement, where mu band power on the right sensorimotor area increases while mu band power on the left sensorimotor area decreases. This effect seems to be more pronounced for the dominant hand and brisk rather than slow movement (cf. Stancák and Pfurtscheller, 1996).

Interestingly, these shifts in mu band power are not only observable for actual physical movement, but to a slighter extent also for imagined movement (cf. Pfurtscheller and Lopes da Silva, 1999; Pfurtscheller et al., 2006), making SMR-based methods suitable for endogenous control task paradigm brain-computer interfaces.

While mu rhythms cannot reliably be detected in every human adult (cf. Niedermeyer, 1999b), Wolpaw et al. (2003) have successfully implemented a mu response BCI that requires users to learn to control their mu and – to a lesser degree – lower beta rhythm amplitudes via motor imagery and which gives them appropriate feedback to help them to do so.

Although this approach entails a significantly higher amount of subject training than typical exogenous control task paradigms, such as P300 response-based BCIs, SMR-based interfaces exhibit much lower latency and are suitable for self-paced BCI applications, which makes SMR-based features a valuable tool for cursor control tasks.

4.2 Cursor control tasks

The aim of cursor control tasks is to let test subjects change the position of a cursor on a computer screen via a continuous position control BCI. Hence, cursor control interfaces can be classified as active steering BCIs according to the taxonomy introduced in Zander et al. (2010).

When using a cursor control interface, users are given immediate feedback in order to be able to evaluate the efficiency of their control strategies and change or adjust these strategies if necessary.

While two-dimensional cursor control has been shown to be attainable for at least some users in Wolpaw and McFarland (2004) and other publications, due to the high amount of training necessary for users to achieve proficiency with this type of interface, most researchers have so far focused their efforts on one-dimensional cursor control.

Other applications

In the past, continuous position control interfaces were associated primarily with cursor control, yet they have various other applications, such as BCI-based navigation tasks, where they can be used for vehicular control. Other possible fields of applications for this type of active steering BCI include assistive technology for people with disabilities, where wheelchair and artificial limb control via brain-computer interfaces are active research areas (cf. Wolpaw et al., 2002).

One of the most prevalent SMR-based methods in cursor control interfaces is the use of ERD features in cue-based BCIs, the working principles of which are discussed in the next section.

4.3 Cue-based BCIs

Cue-based cursor control BCIs – also known as synchronous BCIs – are externally or system-paced brain-computer interfaces, where for each trial a cue denotes when control is transferred to the user.

Reference period

A single trial in a cue-based motor imagery BCI is typically a few seconds long and can be roughly divided into a rest period in which the test subject is instructed to relax, a stimulus presentation period, where the test subject is told which target to focus on next, a motor imagery period, during which the test subject has to imagine the target movement, and finally a short recovery period.

The presence of a rest period in this paradigm is of particular importance insofar as the rest period serves as a reference period, which is a prerequisite for ERS/ERD calculation, since ERS and ERD are defined as changes with regard to a reference interval. Therefore, in order to be able to measure event-related synchronisation and event-related desynchronisation, some sort of baseline signal is needed. It is precisely because there are reference periods in cue-based setups that ERS and ERD can be calculated.

Most approaches of ERD calculation can be categorised as either frequency domain methods or time domain methods.

Frequency domain methods

In frequency domain methods, the signal is transformed from the time to the frequency domain, for example by computing the discrete-time short-time Fourier transform (STFT) for each EEG channel. By comparing the resulting power spectral densities of a given time window with those of a reference period, each frequency range is assigned an ERS/ERD measure.

Because sensorimotor activity is usually restricted to the frequency range from $8 - 30$ Hz, only the relevant frequency bins are used to build a subject specific model from labelled ERS/ERD training data, which is subsequently used to classify new EEG data.

Time domain methods

In contrast to this, the more commonly used time domain methods do not transform the signal to the frequency domain. Instead, ERD is calculated using time domain features, such as the change of the intertrial variance in relation to the reference interval.

In Kalcher and Pfurtscheller (1995), the intertrial variance IV is defined as

$$\mathrm{IV}_{(j)} = \frac{1}{N-1}\sum_{i=1}^{N}(x_{f(i,j)} - \overline{x}_{f(j)})^2$$

where N is the total number of trials, $x_{f(i,j)}$ is the j-th sample of the i-th band-pass filtered trial, and $\overline{x}_{f(j)}$ is the mean of the j-th sample averaged over all band-pass filtered trials.

The ERD can then be quantified (ibid.) for each sample point or time window as

$$\mathrm{ERD}_{(j)} = \frac{R - \mathrm{IV}_{(j)}}{R}$$

where R is the average intertrial variance in the reference interval, averaged over k samples:

$$R = \frac{1}{k}\sum_{j=n_0}^{n_0+k}\mathrm{IV}_{(j)}$$

While these simple ERD-based approaches are well suited for system-paced experiments, their dependence on periodic reference periods and relatively high recovery times preclude their use in self-paced interfaces.

Therefore, reference period-based ERD methods are not suitable for continuous cursor control such as it is required for the practical part of this thesis.

4.4 Self-paced BCIs

Self-paced cursor control BCIs – sometimes also called asynchronous BCIs – are user-paced brain-computer interfaces, where sensor data is continuously converted into a control signal.[1] In contrast to cue-based BCIs, where control is time-locked to a system-paced cue, in self-paced BCIs the user can choose when to generate a control signal (cf. Pfurtscheller and Scherer, 2010).

Two target task

It should be noted that although these interfaces themselves are not cue-dependent, they are often evaluated in semi-cue-based setups. In one of the best known evaluation setups for one-dimensional cursor control interfaces, test subjects control the vertical movement of a cursor that enters the screen on the left side at the beginning of each trial and moves to the right at a constant speed. The goal of the task is to influence cursor movement in such a way that the cursor hits a target on the right side of the screen. The target region's size directly influences the difficulty of the task and the information transfer rate, both of which are higher when the goal is to hit smaller target regions.

Binary spellers

While this task may appear somewhat contrived, it is interesting insofar as similar interfaces have been used to implement simple binary spellers, where users successively narrow down their selection to match the intended target letter by aiming the cursor at one of two targets. As soon as the user gains proficiency at this task, the number of targets is increased to allow for a higher information transfer rate.

Even if similar evaluation tasks rely on external pacing, self-paced BCIs themselves do not. Clearly, these asynchronous interfaces have considerable advantages outside of laboratory settings, where users have to control a cursor over longer periods of time and where regular recording of reference intervals is typically not possible because of this would entail regular interruptions while the task is being completed by the user.

Wadsworth Center BCI

One approach to realise self-paced BCI cursor control is exemplified by the brain computer-interface developed at the Wadsworth Center by Wolpaw et al. (2003), viz. the "Wadsworth Center BCI" or "Albany BCI". Wolpaw et al. trained test subjects to control their mu- and beta-rhythm amplitudes in such a way that they can asynchronously control a one-dimensional cursor via rhythmic modulation without regular reference periods.

[1] For computational reasons, asynchronous BCIs typically only update the control signal a few times per second, an update frequency that causes a noticeable stutter, but is still high enough not to impair user control.

The users' mu- or beta-rhythm are translated to cursor movement by a linear equation 10 times per second from one or several EEG scalp electrodes (Wolpaw et al., 2003). Users are trained in multiple sessions per week, lasting 40 minutes each and the majority of users gain control in a matter of weeks. EEG data is recorded via 64 electrodes, which are distributed across the entire scalp, for offline analysis purposes as well as in order to minimise non-central nervous system artefacts (for example EMG or EOG activity) and to identify the best electrode locations for each subject. In actual BCI operation, however, only a few channels of data are used.

Wolpaw et al. stress the importance of spatial filtering; according to the authors, a common average reference or large Laplacian filter (6 cm inter-electrode distance) yield significantly better results than a simple monopolar montage without spatial filtering or small Laplacian filtering (3 cm interelectrode distance). At the time of the publication of the paper, the Wadsworth Center BCI used relatively simple band power-based features, although the authors already noted that autoregressive frequency analysis would result in increased resolution for short time segments.

In Wolpaw and McFarland (2004), the Wadsworth Center BCI's support for two-dimensional cursor control is discussed. The authors argue that the introduction of multiple autoregressive frequency band processing and better spatial filtering are the main reasons for the improvement of their results compared to earlier 2D cursor control experiments.

User training Wadsworth Center BCI

While Wadsworth Center-type BCIs typically use relatively simple features and often have similarly moderate hardware requirements[2] as those of the Wadsworth Center BCI itself, their main disadvantages are the training effort necessary for test subjects in order to become proficient users and relatively high illiteracy rates (cf. Allison and Neuper, 2010).

Wolpaw et al. estimate the training effort necessary for a user in order to achieve "significant control" to be as low as several weeks, whereas Krepki et al. (2007) claim that most BCIs of this type require more than 200 hours of training in order for their test subjects to become proficient BCI users.[3]

Fabiani et al. (2004) report that test subjects needed ten sessions of 30 minutes each over the course of several weeks in order to learn controlling a one-dimensional cursor in a two target task. (However, they also state that in addition users "participated for 10–69 additional sessions devoted to a variety

[2]Moderate hardware requirements not only in terms of processing power needed for feature extraction and classification, but also insofar as the Wadsworth Center BCI only relies on two EEG channels.

[3]Krepki et al. point out that the Berlin BCI does not rely as heavily on operant conditioning as other non-VEP, non-P300 BCIs.

of studies", ibid.)

In their study with seven test subjects McFarland et al. (2005) found that two of the subjects never achieved cursor control. For two of the remaining five subjects EMG contamination was "prominent in early sessions", but diminished or disappeared over the course of the first ten training sessions. Interestingly, the three users for which no EMG contamination was present in the beginning achieved two target task accuracies of more than 80% in the very first session, while the two users that could not achieve significant BCI control never passed the 60% mark. While a sample size of seven users is of course too small to draw statistically meaningful conclusions, this result might nevertheless help to understand the seemingly inconsistent results reported by different researchers. Ultimately, how well a motor imagery BCI works for a given user will always depend on how well that user's particular motor imagery strategies match the expectations underlying the BCI's design.

The Berlin BCI (BBCI, see Krepki et al., 2007) is another well-known SMR BCI that tries to reduce this form of user training as much as possible; the authors state their goal as "Let the machines learn!" (ibid.). This reduction of training effort on the user's part is achieved by sampling data from a large number of EEG electrodes (≥128 channels) at high sampling rates (up to 1 kHz) and extracting a multitude of different features from the acquired signal (ibid.). This way, for each user the most appropriate features can be selected and used in order to obtain a control signal without the high amount of training typically associated with traditional self-paced sensorimotor rhythm BCIs. Berlin BCI

From a user's perspective this would be the ideal solution to the problem dealt with in the present paper, but due to constraints of the equipment available and the associated implementation effort this approach is not feasible. The biosignal amplifier used for this project only supports a maximum of 4 EEG channels and sampling rates of up to 256 Hz. Furthermore, the development effort necessary for a BCI with user training periods as short as those of the BBCI would go beyond the scope of this thesis.

Even though Wadsworth Center-type BCIs have a lot of disadvantages regarding the required training effort for users, they still seem to be the interface type best suited for cursor control applications intended for use outside of controlled laboratory settings, at least in situations where the available resources are more limited than the ones presumably required for approaches similar to the BBCI.

The next chapter provides a review of how machine learning concepts can be used in order to implement BCI systems, such as the Wadsworth Center-type BCI discussed above.

Chapter 5

Machine learning for BCIs

Machine learning systems are an important part of every brain-computer interface system. This chapter provides an overview of the use of machine learning methods in BCI applications with a focus on techniques which can be applied in sensorimotor rhythm interfaces.

5.1 Machine learning fundamentals

Machine learning is an area of Artificial Intelligence research that focuses on developing efficient and accurate prediction algorithms that use past information in order to reason about future data (cf. Mohri et al., 2012).

Supervised learning

As regards machine learning in the context of BCI, brain-computer interfaces typically employ so-called supervised machine learning methods for sample classification, where each data point in a set of training samples is assigned a class label, typically by a researcher or user of the system. The resulting set of labelled training samples is then used to construct a model and classify new data points whose class membership is unknown.

In this scenario a training sample consists of an arbitrary media object, such as a piece of text, an audio recording, or an image, and a class label associated with the media object.

Feature extraction

The first processing step in a machine learning system is to extract some type of summary or description from media objects. This process is called feature extraction.

Choosing appropriate features for a given type of media objects and a given application is known as feature selection. While feature extraction is automated, feature selection is usually a manual process undertaken by the researcher designing the machine learning system.

Feature selection and feature extraction are of particular importance because these processes effectively determine which aspects of a given set of media objects the machine learning system "sees". For example, a piece of text might be converted into a vector of word frequencies, an audio recording might be converted into a vector of tempo (beats per minute) and clip length, and an image could be converted to a vector of dominant colour values in different image regions. The selection of features in a given classification task is only appropriate if media objects from different classes can be distinguished by the extracted features alone.

Classification

Once media objects have been transformed into a feature vector, the vectors are then used to build a model.[1] The algorithm or software component that constructs models from sets of sample vectors is called a classifier. Because an n-dimensional feature vector can be regarded as a single point in an n-dimensional vector space, the main task of a binary classifier[2] can be thought of as finding a decision boundary in this vector space, a hypersurface that partitions the vector space into subspaces in such a way that all data points in a given subspace can be classified as belonging to a particular class.

Example

For example, in a very simple hypothetical (and highly contrived) binary classification task, a researcher might be interested in predicting whether an athlete is more likely to be a sumo ringer or a jockey. In order to do this, weight and height are selected as the most important features in the *feature selection* step. In the *feature extraction* process, each athlete's data is transformed into a more compact two-element vector containing that person's weight and height so that each person is represented by a corresponding point in a two-dimensional vector space. In the *model building* phase, the existing athlete's data and class labels ("sumo ringer" or "jockey") are used to find class models or an appropriate decision boundary. Finally, in the *classification* phase, a new athlete without known class label is predicted to be either a sumo ringer or a jockey, depending solely on their weight and height.

Having introduced the two most important tasks of a machine learning system, the focus of the next section is the role of machine learning systems in brain-computer interfaces.

[1] Strictly speaking, class models are only built when using generative classifiers, such as Bayesian classifiers. Discriminative classifiers, such as the support vector machine (see Section 5.5), use the available training data in order to find a decision boundary to discriminate between classes (cf. Lotte et al., 2007).

[2] A binary classifier is a classifier that can distinguish two classes. Binary classifiers can be used to build a k-class classifier by employing a simple "one vs all" classification scheme, where k separate binary classifiers are used instead of a single multi-class classifier.

5.2 The role of machine learning in BCIs

One of the main tasks of machine learning in brain-computer interfaces is to analyse training data in order to be able to infer prototypical patterns (Blankertz and Müller, 2009) in BCI use for a specific test subject and recording session and thus to compensate for intersubject and intrasubject variability.

There are two important types of trial variability in brain-computer interface applications, namely intersubject variability and intrasubject variability (cf. Blankertz and Müller, 2009).

Intersubject variability

The term *intersubject variability* or interpersonal variability denotes test subject-dependent variability. While all humans share the same basic neurophysiology and neuroanatomy, there are still vast differences between individual test subjects' responses in exogenous BCIs, and planning and execution strategies in endogenous BCIs.

Optimal EEG electrode locations vary as much from person to person as frequency components of EEG signals, desynchronisation patterns and latencies and amplitudes in ERD setups.

Intrasubject variability

However, the same aspects of EEG can also vary due to *intrasubject variability*. That is to say that they will differ from session to session or even from trial to trial for a single test subject. A single person's EEG data will vary significantly from trial to trial, even when they are presented with the same stimuli or use the same motor imagery techniques repeatedly.

Objectives

As discussed above, machine learning systems in BCI applications have to extract prototypical patterns for each target class from training data, and – in doing so – compensate for intersubject and intrasubject variability. For example, in a left/right hand motor imagery-task a test subject might generally exhibit above-average rhythmic attenuation in the upper alpha band on the left hemisphere, but this could be masked in a given recording session due to fatigue or visual alpha artefacts. The task of the machine learning component of a BCI would then be to find the most appropriate spatial filters for both the test subject and current recording session and to find the prototypical patterns necessary to be able to distinguish between imagined left hand movements and imagined right hand movements.

A BCI typically undergoes a calibration phase, in which labelled training data is recorded in a supervised process. The machine learning system then uses this data to identify both subject-specific and session-specific characteristics in order to convert EEG data into a control signal.

MAN-MACHINE LEARNING DILEMMA

Both, machine and user have to adapt in order to be able to correctly classify user data and produce a strong control signal, respectively. Users depend on the machine learning system's feedback in order to evaluate the efficiency of their control strategies and the machine learning system depends on consistent training data in order to be able to provide accurate feedback and classification performance. In other words, user and machine are strongly interdependent, but have to be adapted independently. This problem is known as the "man-machine learning dilemma" (Pfurtscheller et al., 1997) and some simple strategies to mitigate part of its effects are discussed in Chapter 6.

The next sections explore some exemplary machine learning methods employed in the conversion process from EEG to control signal, and more specifically spatial filtering via CSP and SVM classifiers.

5.3 Common spatial patterns

Common spatial patterns (CSP) are a method for extracting spatial patterns in the time domain which account maximally for the variance in the EEGs of one population and minimally for the variance in the other population (cf. Koles et al., 1990).

CSP identifies those spatial patterns which contribute most strongly to the discrimination process by finding spatial filters that maximise variance for one class and minimise variance for the other class.

Whereas Koles et al. originally introduced CSP in order to classify the EEGs of a population of healthy subjects and EEGs of a population of subjects with neurological disorders, the method is currently often used to classify EEG data into classes such as "left hand movement" and "right hand movement".

DEFINITION

Given recordings from two distinct populations and C channels, the goal of CSP (cf. Blankertz et al., 2008) is to find a spatial filter matrix $W \in \mathbb{R}^{C\times C}$ that projects the signal $\mathbf{x}(t) \in \mathbb{R}^C$ in the sensor space to $\mathbf{x}_{\text{CSP}}(t) \in \mathbb{R}^C$ in such a way that the difference between the variances is maximised and

$$\mathbf{x}_{\text{CSP}}(t) = W^{\mathsf{T}}\mathbf{x}(t)$$

Let $\Sigma^{(+)} \in \mathbb{R}^{C\times C}$ and $\Sigma^{(-)} \in \mathbb{R}^{C\times C}$ be the covariance matrices of the EEG data for the two given classes, then W can be obtained by solving the following generalised eigenvalue problem (Sannelli et al., 2012):

$$\Sigma^{(+)}W = \left(\Sigma^{(-)} + \Sigma^{(+)}\right) W\Lambda$$

The columns of W are spatial filters w_i, each of which corresponds to an eigenvalue λ_i in the C-dimensional vector Λ.

For dimensionality reduction, N filters can be chosen (where $N < C$) to compose a dimensionality-reduced filter matrix W_R, so that $\mathbf{x}_{\mathrm{CSP}_R}(t) = W_R^{\mathsf{T}}\mathbf{x}(t)$ has a lower dimensionality than $\mathbf{x}_{\mathrm{CSP}}(t)$ and the two classes are still maximally separated (ibid.).

In Figure 5.1, e_1 and e_2 denote electrode potentials from two different EEG populations. The ellipses R_a and R_n denote covariance patterns between these electrode potentials. The principal components of the composite covariance pattern including R_a and R_n are given by b_1 and b_2. While the whitened covariance patterns S_a and S_b share the same principal components u_1 and u_2, u_1 accounts maximally for the variance in S_a and minimally for the variance in S_n, whereas u_2 accounts minimally for the variance in S_a and maximally for the variance in S_n (Koles et al., 1990).

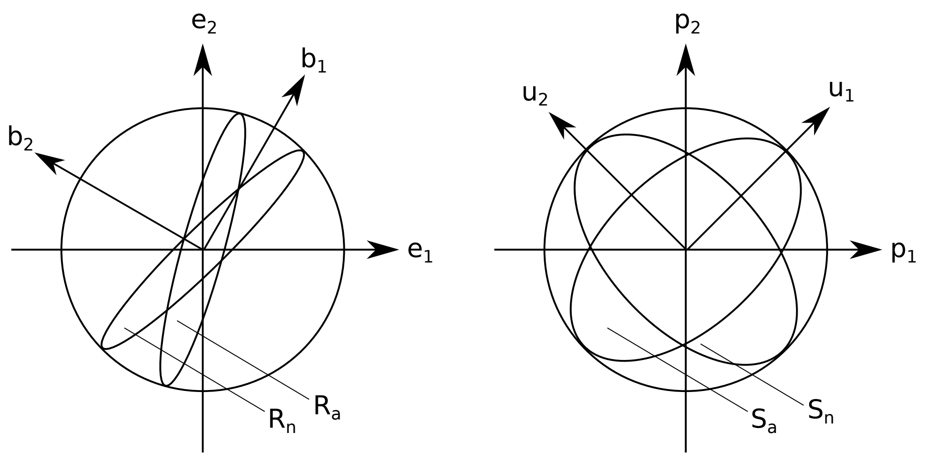

Figure 5.1: Covariance patterns of electrode potentials before and after CSP filtering (from Koles et al., 1990).

Since the filters yielded by CSP are linear spatial filters, their spatial configuration can be easily visualised, which is especially helpful for determining the best electrode locations for a given application and test subject.

Spatial filters are an important component of brain-computer interfaces (see for example Blankertz and Müller, 2009) and the performance, computational efficiency, and simplicity of common spatial patterns make CSP an indispensable tool for spatial filtering in BCI contexts. While the variance of a windowed band-pass- and CSP-filtered signal could in principle be used as a simple feature for ERD calculation, in practice more advanced methods such as autoregressive analysis or Fourier analysis are often more robust. The next section therefore turns to the fundamentals of discrete Fourier transforms.

5.4 Discrete Fourier transform

The discrete Fourier transform (DFT) is the discrete equivalent of the continuous Fourier transform. It is used for Fourier analysis in many practical signal processing applications where the signal is sampled in regular intervals and thus only known for a finite set of points in time (cf. Roberts, 2003).

The continuous Fourier transform of a continuous signal $f(t)$ is

$$F(j\omega) = \int_{-\infty}^{+\infty} f(t)\, e^{-j\omega t}\, \mathrm{d}t$$

Given a sequence of N samples $f[k]$ (where $f[k]$ is the k-th sample and $0 \leq k \leq N-1$), let $T = \frac{1}{f_s}$ be the time interval between two samples. If each sample $f[k]$ is regarded as an impulse of area $f[k]$, then (Roberts, 2003):

$$\begin{aligned} F(j\omega) &= \int_{0}^{(N-1)\,T} f(t)\, e^{-j\omega t}\, \mathrm{d}t \\ &= \sum_{k=0}^{N-1} f[k]\, e^{-j\omega kT} \end{aligned}$$

Although this function could be evaluated for any angular frequency ω, with N samples only N outputs will be meaningful. Because DFT treats the input signal as a periodic signal, the above function is evaluated for the zero-frequency, the fundamental frequency $\frac{2\pi}{NT}$ rad/s, and the fundamental frequency's harmonics $\frac{n2\pi}{NT}$ rad/s (where $n \in \mathbb{N}$ and $2 \leq n \leq N-1$).

The discrete Fourier transform of the sequence $f[k]$ is therefore given by

$$F[n] = \sum_{k=0}^{N-1} f[k]\, e^{-j\frac{n2\pi}{N}k}$$

where $n \in \mathbb{N}_0$ and $0 \leq n \leq N-1$.

Applications

In brain-computer interface systems, the DFT is often used to estimate a signal's power spectral density (PSD), a long-established band power feature that is still used in a lot of BCI applications today, even though more recent publications tend to favour autoregressive parameters.

The DFT can be computed efficiently by means of a fast Fourier transform, such as the Cooley–Tukey algorithm, which reduces the $O(N^2)$ complexity of a naive DFT implementation to $O(N \log N)$, thereby making DFT computationally feasible even on commodity hardware.

In machine learning applications, PSD estimation of a signal via FFT is a typical example of feature extraction. To be able to use a set of examples in order to build a model, which can then be used to classify new samples, a classifier such as the support vector machine is needed.

5.5 Support vector machine

The support vector machine (SVM) is a widely used discriminative supervised classifier first described in its current form – the soft margin SVM – in Cortes and Vapnik (1995).

In the following section, a training set that is linearly separable in the feature space is going to be assumed.

Linear separability

Two sets of points in an n-dimensional vector space are called *linearly separable*, if there is an ($n-1$-dimensional) hyperplane that separates the two sets. For example, data points in a three-dimensional space are linearly separable if there is a (two-dimensional) plane that separates the two sets of data points, whereas in two-dimensional space a (one-dimensional) line has to separate the two sets of data points.

Margin maximisation

In a binary classification task, there are usually many possible different decision boundaries. While a classical perceptron terminates as soon as *any* valid decision boundary is found,[3] support vector machines (SVMs) operate under the assumption that the best boundary is the one that maximises the distance between the decision boundary and the data points closest to the decision boundary (see Figure 5.2). SVMs are therefore said to be margin-maximising classifiers.

Optimising for a large margin is a good choice because models with a large margin generally have a low generalisation error.[4] Therefore, structural risk minimisation learners such as the SVM often perform better than learners which only minimise the empirical error, especially if no problem-domain knowledge regarding the probability distributions of the classes can be incorporated into the classification task.

Computational efficiency

Computationally, support vector machines are rather efficient because they exploit the fact that the decision boundary is typically defined by a very small subset of all training vectors. Because these support vectors are sufficient to describe the decision boundary, all training samples that are not support vectors can be discarded after training.

[3] Although there are extensions of the perceptron, such as the "optimal stability perceptron", which avoid these stability problems, the classical perceptron terminates as soon as any decision boundary is found.

[4] Mohri et al. (2012, pp. 75–83) provide a theoretical justification for the concept of margin maximisation.

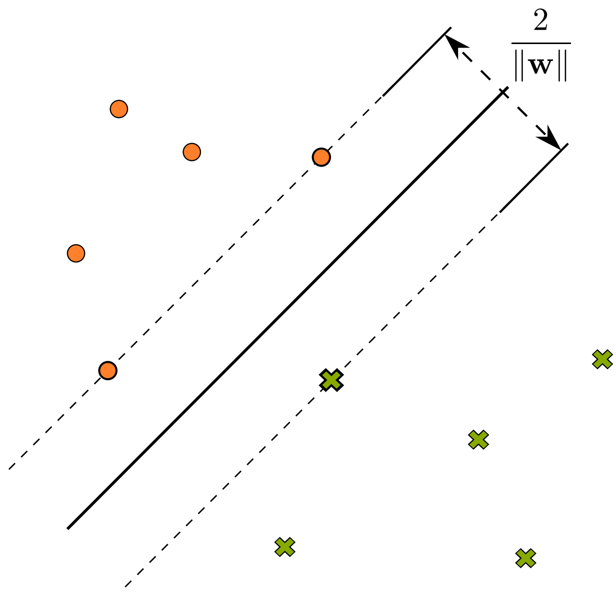

Figure 5.2: Margin maximisation in the SVM.

The following section relies heavily on Bishop (2006), specifically his section on maximum margin classifiers in the chapter on sparse kernel machines (pp. 325–344).

Definition

Formally, binary classification via a linear model can be formulated as

$$y(\mathbf{x}) = \mathbf{w}^{\mathsf{T}}\boldsymbol{\phi}(\mathbf{x}) + b$$

where $\mathbf{w}$ is a weight vector which is orthogonal to every vector lying within the decision surface, $\boldsymbol{\phi}(\mathbf{x})$ is a feature-space transformation, and b is the bias parameter controlling the decision surface's offset from the origin.

The training set consists of N vectors $\mathbf{x}_n$ with corresponding target values $t_n \in \{-1, 1\}$. New data points are classified according to the sign of $y(\mathbf{x})$.

In the following, the training data set is assumed to be linearly separable in the feature space $\boldsymbol{\phi}(\mathbf{x})$, that is $\mathbf{w}$ and b can be chosen such that $y(\mathbf{x}_n) > 0$ for all points for which $t_n = 1$ and $y(\mathbf{x}_n)$ for all points for which $t_n = -1$.[5]

The perpendicular distance of a point $\mathbf{x}$ from a hyperplane $y(\mathbf{x}) = 0$ is given by

$$\frac{|y(\mathbf{x})|}{\|\mathbf{w}\|}$$

[5] It should be pointed out that the data does not necessarily have to be linearly separable in the original feature space – linear separability is a prerequisite only in the transformed feature space $\boldsymbol{\phi}(\mathbf{x})$.

If only solutions for which all data points are classified correctly are considered, $t_n \mathbf{x}_n > 0$ for all n. Therefore the distance between a point $\mathbf{x}_n$ and the decision boundary is

$$\frac{t_n y(\mathbf{x}_n)}{\|\mathbf{w}\|} = \frac{t_n \left(\mathbf{w}^\mathsf{T} \boldsymbol{\phi}(\mathbf{x}_n) + b\right)}{\|\mathbf{w}\|}$$

The margin is defined as the distance between the point closest to the decision boundary and the decision boundary. Hence the weight vector and bias parameter maximising the margin can be found by solving

Optimisation task

$$\arg\max_{\mathbf{w},b} \left\{ \frac{1}{\|\mathbf{w}\|} \min_n \left[t_n \left(\mathbf{w}^\mathsf{T} \boldsymbol{\phi}(\mathbf{x}_n) + b \right) \right] \right\}$$

Rescaling both $\mathbf{w} \to \kappa \mathbf{w}$ and $b \to \kappa b$ preserves the distance between any given point $\mathbf{x}_n$ and the decision boundary.

The rescaling parameter κ can be chosen such that for the point $\mathbf{x}_n$ closest to the surface

$$t_n \left(\mathbf{w}^\mathsf{T} \boldsymbol{\phi}(\mathbf{x}_n) + b \right) = 1$$

and therefore all data points satisfy

$$t_n \left(\mathbf{w}^\mathsf{T} \boldsymbol{\phi}(\mathbf{x}_n) + b \right) \geq 1$$

Because there is always at least one point for which this constraint is active, the optimisation problem is equivalent to maximising $\frac{1}{\|\mathbf{w}\|}$, which is in turn equivalent to minimising $\|\mathbf{w}\|^2$ (compare Figure 5.2):

Primal form

$$\arg\min_{\mathbf{w},b} \frac{1}{2} \|\mathbf{w}\|^2$$

In order to be able to express the model using a kernel function, a Lagrange multiplier $a_n \geq 0$ is introduced for each data point. Reformulating and reducing the primal form yields the dual version of the margin optimisation problem where

Dual form

$$\tilde{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a_n a_m t_n t_m \left(\boldsymbol{\phi}(\mathbf{x}_n)^\mathsf{T} \boldsymbol{\phi}(\mathbf{x}_m) \right)$$

is maximised with respect to $\mathbf{a}$, subject to

$$a_n \geq 0$$

$$\sum_{n=1}^{N} a_n t_n = 0$$

KERNEL TRICK It is important to note that $\boldsymbol{\phi}(\mathbf{x}_n)^\mathsf{T}\boldsymbol{\phi}(\mathbf{x}_m)$ is a simple dot-product of two vectors in the feature space. Therefore it is possible to avoid explicit projections to the feature space and employ a kernel function of the following form instead:

$$k(\mathbf{x}, \mathbf{x}') = \boldsymbol{\phi}(\mathbf{x})^\mathsf{T}\boldsymbol{\phi}(\mathbf{x}')$$

Using this kernel function gives the final dual representation of the original optimisation problem where the explicit feature space transformations are no longer present:

$$\tilde{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m t_n t_m k(\mathbf{x}_n, \mathbf{x}_m)$$

This so-called *kernel trick* is an important feature of the support vector machine because it allows for efficient classification with highly complex decision boundaries by using very high- or infinite-dimensional feature spaces.

CLASSIFICATION Having built a model from training data, new data points can be classified according to the sign of

$$y(\mathbf{x}) = \sum_{n=1}^{N} a_n t_n k(\mathbf{x}, \mathbf{x}_n) + b$$

SUPPORT VECTORS Using the Karush-Kuhn-Tucker conditions, it can be shown that every data point $\mathbf{x}_n$ satisfies either $a_n = 0$ or $t_n\, y(\mathbf{x}_n) = 1$. Because only data points whose corresponding Lagrange multipliers $a_n \neq 0$ contribute to the result of the decision function $\operatorname{sgn} y(\mathbf{x})$, the trained model can be safely represented by only those data points for which $t_n\, y(\mathbf{x}_n) = 1$ holds. These vectors, which lie directly on the maximum margin feature space hyperplane, are called support vectors.

SOFT MARGINS Although support vector machines as described above rely on linear separability in the feature space $\boldsymbol{\phi}(\mathbf{x})$, methods such as the so-called *soft margins* approach extend the classical SVM to allow misclassifications for training sets with overlapping class distributions which are not linearly separable in the feature space. This is achieved by introducing slack variables $\xi_n \geq 0$, where $1 \leq n \leq N$, so that each data point is assigned a slack variable and misclassified points have a corresponding slack variable $\xi_n > 1$. Therefore, the original optimisation problem becomes

$$\underset{\mathbf{w},b}{\arg\min}\left(\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^{N}\xi_n\right)$$

where $C \geq 0$ is a parameter that controls the trade-off between penalisation of misclassified data points on the one hand and model complexity minimisation (and thus margin maximisation) on the other hand.

Soft margin SVMs also perform better than classical SVMs in cases where a small amount of misclassified data points would result in a much larger margin, which helps to improve the generalisation performance of an SVM model.

Discussion

In summary, the support vector machine is a valuable tool for data classification due to its computational efficiency, sparse representation, ability to find complex decision boundaries by using the kernel trick, and good generalisation performance.

One of the most important limitations of the support vector machine is its inherent inability to deal with more than two classes; therefore multiclass classification problems are usually reduced to multiple binary classification problems. Another drawback is that SVM models are relatively hard to interpret – at least when compared to more simple linear classifiers, such as Fisher's LDA.

While traditionally these more simple linear classifiers have been more prevalent than support vector machines in BCI applications, Lotte et al. (2007) analyse the performance and advantages of SVMs compared to other classifiers. Although pure linear classifier models are often easier to visualise in terms of their spatial configuration than SVM models, due to the good classification performance associated with support vector machines, the last few years have seen a steady rise in popularity of SVMs in the BCI community.

Having discussed some machine learning approaches to brain-computer interface design, the next chapters explore the requirements of a cursor control BCI prototype and its resulting software architecture and implementation.

III Practical application

Chapter 6

Software architecture

The aim of the practical part of this thesis is to construct an EEG-based cursor control BCI prototype that can be used in experimental BCI applications via standard input APIs instead of having to adapt each application for use with the BCI prototype. This chapter provides an overview of the resulting software architecture of the cursor control BCI.

6.1 Requirements

The cursor control interface should be usable for applications similar to (albeit simpler than) a regular mouse and via the same interfaces as a regular mouse. Therefore it is essential that the BCI is a self-paced interface. That is, there should be no cued reference periods or external pacing whatsoever; instead users should be able to control the cursor at their own pace. In order to be usable as a mouse replacement in appropriate applications, the interface should have relatively low latency. To improve user feedback, multiple updates per second would be desirable.

Furthermore, the biosignal amplifier available for this project only supports a maximum of four bipolar EEG channels and setup should be easy for inexperienced users, hence the BCI should only require a low number of hardware channels.

While some form of user training is going to be necessary for test subjects to be able to use the BCI effectively, once a user has sufficient experience, session calibration should be relatively fast. Optimally, the calibration process should not take more than fifteen minutes per session.

The biosignal amplifier used for this project currently only supports Windows systems, therefore the BCI system also has to work on Windows systems. It should support standard Windows input APIs, such as WinAPI or DirectX, as well as cross-platform APIs, such as SDL. When using the BCI application, the cursor should be controllable the same way it is usually controlled when using a regular mouse. However, while a regular mouse allows two-dimensional cursor control, the BCI system implemented will – for the time being – be restricted to one-dimensional cursor control.

Computationally, the implemented system should be efficient enough not to consume a significant amount of hardware resources in actual operation on current commodity hardware.

The resulting implementation takes these requirements into account.

6.2 Specifications and experimental setup

The BCI type that meets the requirements discussed above best is a Wadsworth-type sensorimotor rhythm BCI in a self-paced setup. In order to implement such an interface, a bipolar two-channel EEG-montage, power spectral density estimates-based features, and a support vector machine classifier were chosen; see Table 6.1 for a short tabular summary.

BCI type	self-paced cursor control
brain signal	EEG sensorimotor rhythms (SMR)
feature	power spectral density estimates via DFT
classifier	support vector machine (SVM)

Table 6.1: The present BCI's key data.

The EEG signal is recorded using a g.tec g.MOBIlab biosignal amplifier and five passive gold cup electrodes. The ground electrode was placed over F_{PZ}, while the left and right channel's electrodes were placed over FC_3/CP_3 and FC_4/CP_4 respectively. Closely spaced bipolar derivation works similar to unipolar derivation with large Laplacian filtering around C_3 and C_4 (cf. Pfurtscheller et al., 1997), therefore additional Laplacian spatial filtering is not necessary. Table 6.2 provides an overview of the BCI's signal acquisition process.

TRAINING PHASE

Training data is recorded in a cue-based setup with a simple Matlab GUI that presents users with a series of random targets ("left" or "right") and a feedback indicator which is updated using a simple feedback function that is only used in the training phase. A single run in the training phase is structured as depicted in Figure 6.1: after a short pause, the user is shown the target and feedback indicator, and should start to imagine appropriate hand

biosignal amplifier	g.tec g.MOBIlab
sampling frequency	256 Hz
epoch length	128 samples
epochs per second	4 (50% overlap)
EEG electrodes	5 electrodes (2 bipolar channels + ground)
electrode locations	FC_3/CP_3, FC_4/CP_4 (adjacent to C_3 and C_4)

Table 6.2: The present BCI's signal acquisition parameters.

movements. Two seconds later, recording starts.[1] Over a time span of four seconds, four epochs of 128 samples each are recorded per second (for a total of 16 epochs per stimulus) with a sampling frequency of 256 Hz. All epochs have an overlap of 50% – the same overlap that is later used for self-paced online operation. When the recording phase has finished for one stimulus, the next target stimulus is selected and recorded in the same way until the run is complete.

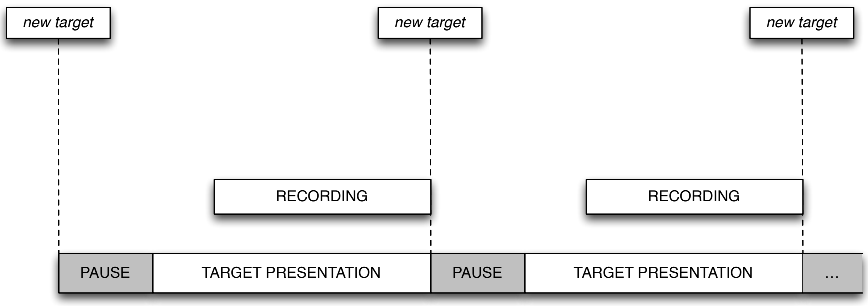

Figure 6.1: The system-paced training phase of the present BCI.

In order to be able to provide feedback to the user before a user-specific profile has been built, a simple non-parametric feedback function is used (see Section 6.2 for reference). Once enough user data is available, a user-specific model which allows more precise feedback can be built. This feedback can then be used in a second calibration phase, which yields significantly better training data than the primitive user-independent feedback function mentioned before or no feedback at all (cf. Guger et al., 2001). FEEDBACK

When the initial data collection process is complete, the user's session data can be used to build a session- and user-specific classifier model. In

[1]While user feedback is based on a processed signal, the EEG recorder saves the raw EEG signal so that it can be analysed better later on.

order to build this model, power spectral density estimates are computed via a FFT-based method (see Section 7.1 for details) and used to build an SVM model.

Once the final user model has been built, this model can be used to classify new EEG data. While some other BCIs – such as the Wadsworth Center BCI, Wolpaw et al. (2003) – use linear equations to directly translate SMR amplitudes into cursor movement, the present BCI uses a binary SVM classifier in order to predict data labels such as "left" or "right". This label is then sent to a separate process that simulates a simple inertial cursor and calculates the actual cursor movement (see Section 7.3 and Section 7.5 for details).

Having established the BCI paradigm, experimental setup, and some basic parameters, the next section provides a conceptual overview of the software architecture.

6.3 Architectural overview

The present BCI prototype consists of two major components: firstly, a signal acquisition and processing component written in Matlab that captures data from the biosignal amplifier, builds a user-specific model and classifies new data according to that model, and secondly, a cursor server written in C++ that uses the classifier output to simulate a simple inertial cursor and translates classifier data into actual cursor movement. These components use a named pipe and a primitive ad-hoc text protocol for unidirectional communication. Figure 6.2 shows an overview of the basic architecture.

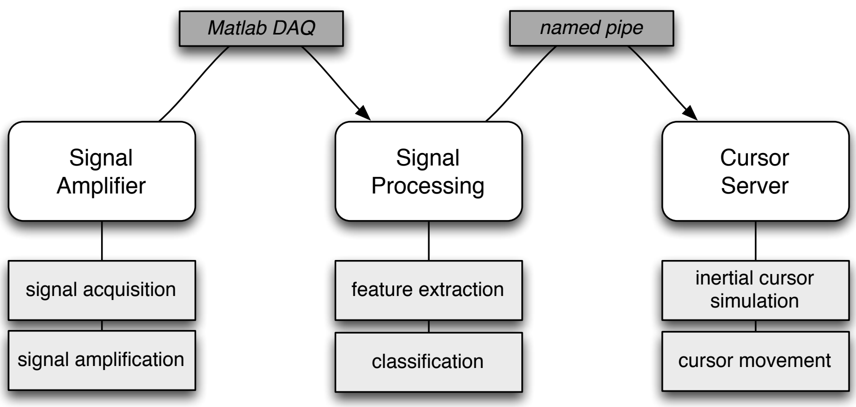

Figure 6.2: The basic architecture of the present BCI.

Concretely, when the classifier component has classified a data window as “left”, it sends a “left” message to the cursor server process over a named pipe. The cursor server receives the message and immediately updates the inertial cursor simulation, accelerating the cursor to the left.

While the classifier only has to run four times per second, the cursor server has to run continuously in order to simulate deceleration due to friction convincingly and move the cursor accordingly. The model used for this simulation incorporates simulated mass, mass inertia and a coefficient of friction; the specifics of the implementation are discussed in Section 7.5.

Whereas the cursor server is only used in actual self-paced BCI operation, the Matlab component also has a simple stimulus presentation and recording mode that is used in the training phase of the BCI. In this mode, users are presented with target stimuli and their response is recorded in order to extract user-specific prototypical responses for each class, as discussed in Section 6.2 above.

The next chapter turns to the implementation of the BCI prototype, with an emphasis on the process of feature extraction, classification and user feedback.

Chapter 7

Implementation

To implement a cursor control brain-computer interface, various components have to be developed. In addition to the typical parts of a classic machine learning application, namely feature extractor and classifier, the choice of the feedback indicator function and cursor function plays an important role for the performance of a cursor control interface. This chapter provides an overview of the implementation of the present BCI.

7.1 Feature extraction

In order to classify EEG epochs, the signal is transformed from the time to the frequency domain by a discrete Fourier transform and the resulting Fourier coefficients are subsequently used to estimate the power spectral densities of the EEG signal.

EEG NON-STATIONARITY

Because EEG is a non-stationary signal, epoch data is normalised using data from previous epochs to estimate the signal's mean and standard deviation. According to Fabiani et al. (2004), the resulting normalised signal can be assumed to be stationary.

AAR ANALYSIS

Some alternatives to the approach used by Fabiani et al. to compensate for EEG non-stationarity are cited in Blankertz et al. (2008) and modern BCI systems tend to employ adaptive autoregressive (AAR) models, which were popularised by Schlögl (2000), instead of using Fourier analysis for spectral analysis. AAR estimation algorithms approach EEG non-stationarity in a more elegant fashion than the simple normalisation described above, and in general seem to be superior to the simple band power Fourier analysis approach used in the present thesis – especially in online single trial analysis applications such as online cursor control. However, due to the complexity of retrofitting efficient AAR model estimation algorithms into the present BCI, discrete Fourier analysis is used for spectral analysis.

Signal filtering

Since the BCI utilises power spectral densities in the sensorimotor rhythm range, the signal is band-pass-filtered using a high order (order 10) Butterworth band-pass filter with a pass band of $\frac{8\times2}{F_s} < \omega < \frac{35\times2}{F_s}$, corresponding to the frequency range of $8 - 35$ Hz.

Spectral analysis

The EEG epoch duration is 500 ms or 128 samples at a sampling rate of 256 Hz, therefore the epoch's discrete Fourier transform yields 64 (meaningful) frequency bins per channel with a bin width of 4 Hz. The corresponding Fourier coefficients are used to estimate the epoch's power spectral densities

$$\frac{|X(f)^2|}{M}$$

where $X(f)$ is the DFT of the original signal and M is the epoch length in samples.

Feature vector

The feature vector thus consists of $2 \times 64 = 128$ power spectral density values corresponding to the frequencies of the 64 Fourier coefficients for both left and right channel,[1] most of which will be close to zero due to the prior band-pass filtering. Thus, while the resulting feature vector has very high dimensionality, intrinsic dimensionality is much lower. In other words, the signal can be represented by a number of variables that is considerably smaller than the dimensionality of the original feature vector. Therefore, the feature vector's dimensionality can be safely reduced.

7.2 Dimensionality reduction

Although the support vector machine classifier used for the present BCI deals relatively well with high-dimensional data, the intrinsic dimensionality of the feature vector is rather low compared to the 128 Fourier coefficients that constitute the feature vector's dimensions.

In fact, using principal component analysis (PCA), the feature vector can safely be projected into a vector space of much lower dimensionality. Retaining as little as 16 principal components preserves more than 95% of a typical feature vector's variance while massively reducing the dimensionality and data volume of the training data.

[1] As discussed in Section 6.2, the present BCI uses two EEG channels roughly corresponding to C_3 and C_4, respectively.

In order to select an appropriate number of principal components, a PCA is performed and the number of dimensions is chosen in such a way that the sum of the corresponding eigenvalues exceeds 95% of the total sum of eigenvalues. Because the eigenvalues are directly proportional to the explained variance, when the data is projected to a feature space of the respective dimensionality 95% of the training data variance are still accounted for – even if the dimensionality of the training data has been significantly reduced.

Having sufficiently reduced the dimensionality of the original feature vector, the resulting feature vector can now be used to build a model from training data, which in turn can be used for classification.

7.3 Classification

There are a variety of classifiers that have been used to classify EEG data in BCI contexts. In the present BCI, a support vector machine classifier (see Section 5.5) was chosen for epoch classification. As detailed in Lotte et al. (2007), SVM classifiers perform quite well in BCI applications. Firstly, regularised classifiers, such as the SVM, deal well with outliers and noise, and generally have better generalisation performance than non-regularised classifiers. Secondly, SVMs can be used with feature vectors of high dimensionality. Thirdly, SVMs are computationally quite efficient (see Section 5.5), making them usable in applications for which low latency is desirable, such as the present BCI.

Lotte et al. (2007) evaluate various other classifiers with regard to their suitability for BCI applications, namely Hidden Markov models, a number of artificial neural network-based classifiers, various Bayesian classifiers, and several combined classifiers, but conclude that “SVM are particularly efficient for synchronous BCI” (ibid., R11), citing the SVM’s regularisation property and immunity to the curse of dimensionality as probable reasons.

SVM IMPLEMENTATION

Therefore, in the present BCI a regularised kernel support vector machine classifier is used. The kernel function used is the radial basis function (RBF), where the kernel function is given by

$$K(u, v) = \exp\left(-\gamma \|u - v\|_2^2\right)$$

The value of γ is chosen to be $\frac{1}{k}$, where k is the dimensionality of the feature vector. The SVM implementation used in the present BCI is the C-SVC SVM implementation by Chang and Lin (2011) with a regularisation parameter $C = 1$.

It is important to note that the actual cursor movement is not based on the epoch classification results alone, but instead derived from the history of classification results via an inertial cursor simulation described in more detail in Section 7.5.

Of course, supervised classification performance highly depends on the quality of the training data used when building the model or computing the decision boundary. Since EEG discriminability seems to be significantly improved by supplying test subjects with appropriate feedback (cf. Guger et al., 2001), feedback is needed when collecting training data.

7.4 Feedback

User feedback is crucial to the performance of a BCI. Appropriate feedback helps BCI users to immediately assess the effectiveness of their control strategies and therefore to choose strategies appropriate for the given BCI. Thus, BCI training data is significantly improved by supplying test subjects with an appropriate feedback indicator (cf. Guger et al., 2001) and a BCI's overall performance in general is highly dependent on user feedback.

In a cursor control BCI, this type of explicit user feedback is typically only necessary in the training phase. Once a user-specific session model has been built from the training data, the BCI itself automatically provides feedback through the cursor movement controlled by its users.

Feedback phases

In training mode, the present BCI runs through two distinct feedback phases. In the first phase, feedback can be either omitted completely or provided by a simple non-parametric feedback function. The non-parametric feedback function was originally devised with the first two test subjects' data in mind, but later tests revealed that its design is not necessarily a good match for all test subjects.

Ad-hoc feedback

Analysis of the first two test subject's EEG data revealed that in a majority of "left" epochs, SMR band powers on the right brain hemisphere increased more (relative to "rest" epochs) than SMR band powers on the left brain hemisphere did, and vice versa for epochs tagged as "right". While this tendency was not pronounced enough to classify epochs accurately, it was strong enough to provided an acceptable feedback measure based on accumulated EEG powers in the SMR band:

$$f = \log_b \frac{\sum_{i=1}^{n} A_{1,n}}{\sum_{i=1}^{n} A_{2,n}}$$

where A is a $2 \times n$ matrix consisting of n Fourier coefficients of the left (first row) and right (second row) EEG channels, and b is a parameter that is hand-tuned to match the respective individual's typical ERD magnitudes. For example, if b is set to a low value, even small SMR band power changes yield large feedback values, whereas if b is set to a very high value, even large changes in SMR band power only yield small feedback changes. Because SMR band powers of the right and left channels per se are not meaningful without a reference, the above measure was baselined against the median SMR band powers of a short rest period in the beginning of the training phase and truncated to $[-1, 1]$.

As mentioned previously, while the above feedback function appears to have worked reasonably well for the first two test subjects, it does not seem to be universally usable for other test subjects. Anecdotal evidence seems to suggest that the present BCI performs even worse when providing bad feedback than it does without providing any feedback at all, therefore this feedback function was only used for the two test subjects for whom it was originally designed.

Classifier feedback

In the second phase, training data from the first phase is used to build a model. This model is subsequently used to classify new epochs with the same method that is used later on in actual BCI operation. This way, new training data can be captured using the same classifier that is used later to provide appropriate feedback. The output of the classifier is smoothed by averaging the last few output labels and displaying the result via the feedback indicator. Figure 7.1 shows a screenshot of the classifier feedback phase in the training GUI.

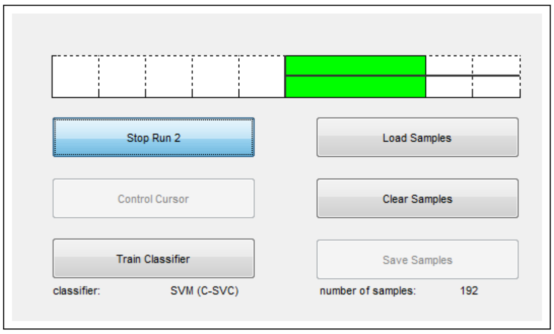

Figure 7.1: Classifier feedback in the training GUI.

While such a relatively simple smoothing function is sufficient for the first phase of user feedback, in order to achieve satisfactory performance in interactive cursor control settings a better cursor function is needed.

7.5 Cursor function

The aim of the cursor function is to convert a time-locked sequence of classification results such as "left" into relative cursor movement commands such as "move the cursor 42 pixels to the left" and, in doing so, to provide a form of signal smoothing. In theory, it would of course be possible to simply move the cursor a little bit to the left every time the classifier classifies an epoch as "left". In practice, however, this is not an acceptable solution for several reasons. Firstly, such an implementation would not result in a continuous smooth movement, but instead in a series of abrupt changes in the position of the cursor. Cursor movement is the most important feedback indicator for users of a cursor control BCI, and since user control is greatly improved by appropriate continuous feedback (cf. Guger et al., 2001), a poor indicator such as a jumping cursor would negatively affect the quality of the training data and therefore the performance of the BCI as a whole. Secondly, from a user's perspective, keeping track of a cursor that does not move smoothly but instead jumps abruptly in short intervals is quite difficult. And thirdly, epoch classification is far from perfect, making cursor movement even harder to predict. Ideally, if the user wants to move the cursor left for four epochs but the fourth epoch is misclassified, the cursor still should not move to the right.

Inertial cursor model

In other BCIs, such as the Wadsworth Center BCI, this problem is often solved by employing a linear translation function instead of a pure binary classifier. This function directly translates brain signals into "move 8 pixels to the right" or "move 40 pixels to the right", depending on the intensity of the processed signal, which helps to mitigate the problem of single misclassified epochs – after all, misclassified epochs are often somewhat ambiguous and therefore translated into a smaller movement anyway. In the present BCI, due to the design of the signal processing and classification component, this approach is not feasible. Instead, it is the task of the cursor function to translate discrete "left" or "right" events into continuous cursor movement.

In order to realise the desired behaviour, a simple inertial cursor model was implemented, where the cursor is given velocity and mass. Furthermore, a friction coefficient is introduced and inertia of (cursor) mass is simulated.

In each simulation step, friction is given by

$$F_f = -v\,m\,\mu$$

where v is the cursor's velocity, m is the cursor's mass, and μ is the coefficient of friction.

Inertia is given by

$$F_i = v\,m\,C$$

where C is the coefficient of inertia.

In order to update velocity, in each simulation step the following update rule is used:

$$\Delta v = F_f + F_i$$

where Δv denotes the change of velocity (acceleration).

When a new epoch is classified, the cursor is simply accelerated into the appropriate direction. Even if an epoch was misclassified by the signal processing component, this usually does not have too much of an effect, as long as the majority of epochs is classified correctly. While this design is rather simplistic, it is quite effective and lets the designers of a BCI tweak cursor behaviour to match the needs of the target application. This is especially helpful in exposing the trade-off between cursor resolution, latency and smoothness inherent to such cursor control applications and letting system designers adjust the configuration appropriately.

The cursor function examined above constitutes the last component of the BCI implementation. The next chapter provides an evaluation of the BCI system as a whole and the cursor function's influence on overall performance.

Chapter 8

Results and discussion

The previous sections have discussed the architecture and design of a self-paced cursor control brain-computer interface. This section provides an evaluation of the present BCI's performance and a discussion of its advantages and limitations.

8.1 Evaluation methodology

In order to evaluate the performance of the present BCI, a simple one-dimensional cursor control two target task-application was implemented using Python and pygame. Users were presented with a standard two target task-setup (see Section 4.4), where a cursor moved to the top of the screen at a fixed speed and users had to move the cursor horizontally via motorimagery in order to hit a randomly chosen target at the top of the screen. Because the BCI converts the user's control signal into actual mouse movement data, the the position of the mouse cursor could simply be queried via the standard input API in pygame. Figure 8.1 shows a screenshot of the evaluation tool.

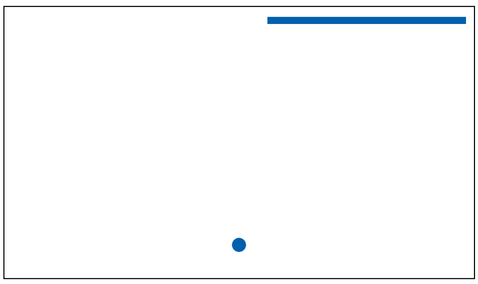

Figure 8.1: A screenshot of the evaluation tool.

Trial setup The test subjects were four healthy individuals without disabilities, who had some experience in use of the present BCI (see Table 8.3 for details), and who were instructed to use left- or right-hand motor imagery in order to move the cursor. Before the start of the evaluation, the test subjects were told to relax in between trials. The intertrial interval of the evaluation task was 2s and the screen was blanked between trials. At the beginning of a new trial, the new target was presented for 1s before the cursor entered the screen horizontally centered and the test subject tried to steer the cursor towards the intended target.

The cursor travelled from the bottom to the top of the screen in 3s. The horizontal position of the cursor was determined by mouse movement relative to the position of the cursor at the beginning of the trial and limited by the left and right edges of the screen. Therefore, test subjects could not miss targets on the left or right edges of the screen by moving the cursor too far.

The cursor was rendered in the form of a filled circle and its diameter measured 3% of the total horizontal screen resolution. The two targets were rendered as filled rectangles at the top of the screen and measured 47% of the total horizontal resolution.

Calibration Before evaluating the system performance for a given test subject, the BCI was calibrated by recording two successive sessions of five runs each. In each of these runs, 960 samples (480 per class) with a length of 250ms and 50% overlap were recorded. The session recordings were used to calibrate the system as described in Section 7.3 and Section 7.4.

Because not every test subject performed best with the same motor imagery class, the most suitable motor imagery class was determined in an offline analysis before the start of the evaluation for each user. In other words, some users were instructed to imagine movement of their left hand in order to move the cursor to the left and to relax in order to move the cursor to the right ("left vs. rest"), while others were instructed to imagine movement of their left hand to move the cursor to the left and movement of their right hand to move the cursor to the right ("left vs. right"). See Section 8.2 for details.

Each test subject's evaluation session consisted of 5 runs comprising 16 trials for a total of 80 trials performed over the course of a time span of approximately 15 minutes (96 seconds per run, pauses of roughly one minute between runs). The next section turns to the results of this evaluation.

8.2 Results

As evidenced by the results in Table 8.1 and Table 8.2, the performance evaluation of the present brain-computer interface revealed strong interpersonal variation. This is not surprising, considering that test subjects A and B were the main test subjects during the design and development of the BCI. Therefore, these test subjects had significantly more training in operating the BCI than the other test subjects. Although it can not be ruled out that the BCI underwent some sort of conceptual overfitting in the implementation phase, the vast differences in user training alone seem to be a relatively plausible explanation for the interpersonal performance differences in the present evaluation.

subject	run 1	run 2	run 3	run 4	run 5
A	81.25%	87.50%	87.50%	81.25%	75.00%
B	87.50%	87.50%	93.75%	81.25%	75.00%
C	50.00%	62.50%	43.75%	68.75%	56.25%
D	81.25%	68.75%	75.00%	62.50%	68.75%

Table 8.1: BCI performance in a two target evaluation task per run.

In order to gain a better understanding of the of the present BCI's performance characteristics, Table 8.2 provides mean accuracies and the accuracy standard deviation between runs for each test subject. Furthermore, because of potentially skewed classes the table includes the mean F1 measure of the "left" and "right" target classes.

Consistency of results

It could reasonably be assumed that experienced BCI users achieve more consistent results than less experienced users, and therefore the accuracy they achieve should have a lower standard deviation between runs than that of less experienced users. While the data in Table 8.2 is of course not sufficient to substantiate this type of behaviour, the evaluated data exhibits similar tendencies, which were also observable over the whole run of the BCI experiment, where the inter-run standard deviation of accuracy decreased as the test subjects gained proficiency in operating the BCI and managed to achieve somewhat more consistent results.

User training

Table 8.3 shows the amount of training each test subject went through and the imagery strategy they used when obtaining the reported results. Data for test subjects A and B are not as representative as desirable because they were the test subjects who participated in the tests during the initial design and development phases of the BCI and therefore had vastly more exposure to the

subject	mean accuracy	SD accuracy	mean F1
A	82.50%	4.68	0.82
B	85.00%	4.43	0.84
C	56.25%	8.84	0.54
D	71.25%	6.37	0.68

Table 8.2: Performance overview of the two target evaluation task.

BCI in its early stages than the other users. Subject C showed no measurable improvements during training and subject C's training was therefore aborted after 10 sessions of about 30 to 40 minutes each. While subject D showed some improvement during the early training sessions, training was again cut short due to time and scheduling constraints.

subject	amount of training	motor imagery classes
A	>40h	left vs. rest
B	>30h	left vs. right
C	<10h	left vs. rest
D	<10h	left vs. rest

Table 8.3: Amount of user training for all test subjects.

IMPACT OF THE CURSOR FUNCTION

As discussed in Chapter 7, the present BCI is highly dependent on the design and performance of its cursor function. The epoch classification process used is relatively unreliable and in order for the user to achieve a sufficient amount of cursor control, the classification results are converted into a control signal via inertial cursor simulation. Table 8.4 shows a comparison of training data classification without cursor function application and subsequent performance in the two target evaluation task using the cursor function described in Section 7.5. The accuracies and F1 scores for BCI performance without a cursor function are average values calculated by performing k-fold cross validation ($k = 10$) on the raw epoch data obtained in the training phase of the BCI.

It is important to note that the performance data for BCI training without a cursor function in Table 8.4 is not representative of the performance of the complete interface, which was designed and implemented with the cursor function as an integral part of the system. Nevertheless, this data provides an interesting insight into the impact of the cursor function on system performance.

subject	without cursor function		with cursor function	
	accuracy	F1 score	accuracy	F1 score
A	70.31%	0.70	82.50%	0.82
B	79.06%	0.79	85.00%	0.84
C	52.08%	0.52	56.25%	0.54
D	64.48%	0.65	71.25%	0.68

Table 8.4: Impact of the cursor function.

The large impact of the cursor function is partly due to the mismatch between a self-paced control paradigm and the supervised learning approach used. A simplifying assumption underlying the design of the present BCI's training component is that every epoch recorded during the training phase is somehow representative of the type of motor imagery the user is supposed to perform at that moment. However, this assumption does not necessarily always hold – in some cases users find it difficult to sustain a particular type of motorimagery long enough, thereby decreasing training data quality and training data classification accuracy. Therefore the cursor function has to compensate for resulting misclassifications.

Motor imagery classes

Table 8.3 shows that the subset of motor imagery classes for which test subjects performed best varied from subject to subject. This was not originally anticipated. Instead, the BCI was initially designed to distinguish the same three target states for each user: "left", "right" and "rest". However, early evaluation revealed that most users were not able to produce SMR signals that had sufficient variation in order for the present classifier to reliably distinguish between all of these classes. Although a resting state would have had some advantages to the current implementation where one class is associated with negative movement whereas the other is associated with positive movement, ultimately this was not feasible using the current methods without sacrificing too much performance in actual BCI operation.

Training progress

As discussed above, test subjects A and B used the BCI from the earliest prototyping stages, therefore no reliable data of their early training progress is available: The system was still too much in flux when A and B started their SMR BCI training, which is why their respective progress is not indicative of the BCI's typical learning curve. Test subject C did not improve significantly during the course of the training. Figure 8.2 provides an overview of subject D's early training progress.

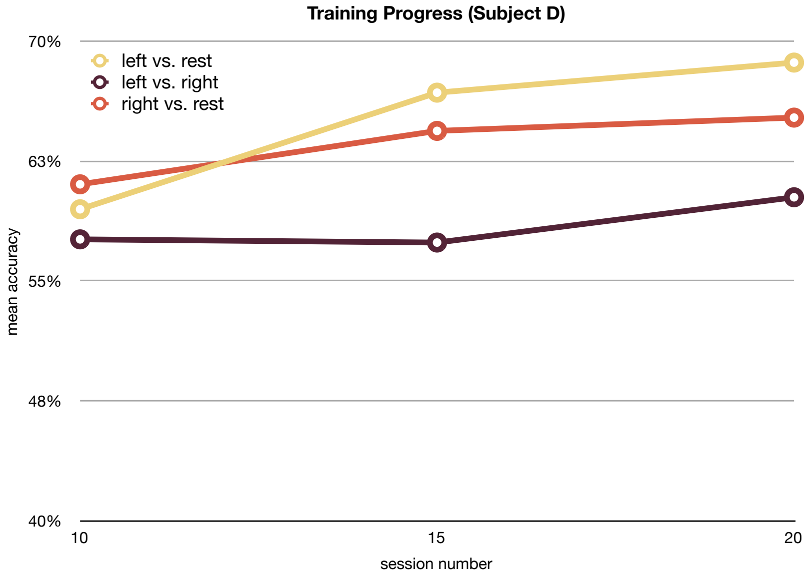

Figure 8.2: Training progress of test subject D.

Information transfer rate

The information transfer rate (ITR) of a brain-computer interface is an important performance measure. Since accuracy and similar measures do not provide any information about the amount of time a BCI user needs in order to select a given target, the information transfer rate (usually given in bits per minute) is a well established complementary measure.

The number of bits transmitted per trial B is given by

$$B = \log_2 N + P \log_2 P + (1 - P) \log_2 \frac{1 - P}{N - 1}$$

where N is the number of targets and P is the probability of hitting the correct target (Pierce 1980; qtd. in McFarland et al., 2003, p. 239). Table 8.5 shows the resulting information transfer rates the test subjects achieved in the evaluation of the present BCI (assuming a trial duration of 6s per trial including pauses; details discussed above).

Having looked at the results in terms of system performance data, the next section discusses the implications of these results for future applications as well as the strengths and limitations of the current approach.

8.3 Discussion

Although the present brain-computer interface does not perform nearly as well as current state of the art non-invasive cursor control BCIs, its accuracy

subject	mean accuracy	mean ITR
A	82.50%	3.31 bits/min
B	87.50%	3.90 bits/min
C	56.25%	0.11 bits/min
D	71.25%	1.35 bits/min

Table 8.5: Information transfer rates in the two target task evaluation.

and information transfer rates are nevertheless in roughly the same order of magnitude as those of other comparable – albeit often far more modern – BCIs.

Comparison

Krepki et al. (2007) report that "[i]n spelling tasks that are truly online with biofeedback, single subjects can reach a level of two to three letters per minute", referencing earlier publications, such as those by Birbaumer et al. (1999) and Wolpaw et al. (1991).[1] Assuming a simple 7-bit character encoding system, such as (non-extended) ASCII, and a naive (non-predictive) binary speller, a user would have to achieve an ITR of $14 - 21$ bits/min in order to select two to three letters per minute. With a more advanced predictive speller, such as *Dasher* (Ward et al., 2000), a FOSS speller employing continuous gestures and language models, the required number of selections could be reduced even further.

McFarland et al. (2003) analyse the influence of trial duration and number of targets on BCI operation accuracy and information transfer rates. For two targets, their eight test subjects achieved ITRs between 0.70 and 8.85 bits/min ($\mu = 5.72$, $\sigma = 2.64$). While a higher number of targets had an adverse effect on the BCI's accuracy, it also increased information transferred on target hits. In other words, while users hit their targets less often, they also transmitted more information when doing so. The best-performing user in McFarland et al. (2003) achieved an ITR of 17.09 bits/min in a four target task and other users achieved as much as 11.82 bits/min in the same task. Of course increasing the number of targets is only a sensible option for users who already have good EEG control in two target tasks and whose information transfer rate is actually limited by the number of targets as opposed to the accuracy achieved (ibid.).

[1] It should be noted that Krepki et al. (2007) also discuss the ITR in studies with "pseudo-online idealized evaluation". In these studies, offline data was analysed as if it was in fact online and Krepki et al. achieved record ITRs as high as 50 bits/min. However, this type of setup is not comparable to the present BCI.

Advantages Even if the present BCI's best ITRs of 3.90 bits/min for two target tasks are significantly lower than those achieved with comparable BCI setups by McFarland et al., Krepki et al., and other research groups, given the present BCI's simplistic design and limited resources these results are quite adequate. The BCI only uses two bipolar EEG channels and a rather simple biosignal amplifier without any additional hardware for artefact control, does not rely on any offline data analysis apart from class selection, and runs on commodity hardware. As expected, being able to access BCI data via standard input interfaces greatly simplifies rapid development of custom BCI applications and the BCI can be used with existing applications which rely on standard mouse input.

The BCI's latency seems to be adequate, although the epoch size of course reflects the inherent trade-off between latency and classification accuracy. While epoch classification of the extracted features alone would be much too inaccurate to provide sufficient control in actual use, the inertial cursor model provides a relatively simple yet remarkably effective workaround to the moderately successful epoch classification process. It is important to note that while the SVM classification itself works relatively well, it is the feature selection and user training which seem to limit the BCI's performance.

Limitations Most modern spectral analysis-based BCIs use more advanced adaptive methods than the simple FFT-based PSD estimator used in the present BCI, such as the adaptive autoregressive models described by Schlögl (2000). In addition, AAR model-based features could have improved the classification performance of the present BCI. Schlögl et al. (2010) provide an in-depth discussion of the motivation for using modern adaptive methods in BCI research. While the current combination of a binary classifier and inertial cursor simulation provide experienced users with a sufficient level of cursor control and the frequent epoch misclassification does not seem to decrease targeting performance significantly, cursor movement tends to be relatively abrupt. This probably contributes to the excessive training needed in order for users to achieve reliable BCI control; if users had better feedback, they would find it easier to evaluate the effectiveness of their control strategies and therefore gain BCI proficiency faster.

The evaluation also revealed that the absence of a proper rest state is a substantial disadvantage of the BCI in its current form insofar, as the epoch classifier does not produce a movement magnitude, its only output being a single movement class.[2] Instead, the system relies exclusively on the cursor function to translate these class labels into actual movement commands. However, during rest periods, consecutive assenting classifications can be a

[2]That is, the classifier's output is either "left" or "right".

problem. For example, when the systems misclassifies the current state as “left” a number of times in a row, this usually accelerates the cursor so much that it can be very hard for the user to re-gain movement control. In fact, most misclassifications in the evaluation occurred due to the fact that rest state misclassifications in between trials often accumulated before the next trial and made it almost impossible for the user to re-gain control and move the cursor towards the intended target in the short time span until the end of the trial. While it would of course have been possible to change the evaluation in order to always start trials without any horizontal cursor movement, this would have significantly distorted the evaluation results. In lieu of an actual rest state, this problem could probably be mitigated by a linear translation function such as the one used in the Wadsworth Center BCI (Wolpaw et al., 2003). Another approach for incorporating rest states into BCI control paradigms is discussed in Fazli et al. (2010).

Although the results presented above help to convey a sense of the relative merits and limitations of the present BCI, it is important to bear in mind the limited scope of this evaluation.

8.4 Scope of this evaluation

This section addresses potential shortcomings in the evaluation of the present BCI, more specifically its number of test subjects and various issues regarding artefact control.

SAMPLE SIZE

The most obvious limitation of the evaluation is the number of test subjects used as BCI operators and therefore the small statistical sample size. While similar sample sizes are far from unusual in BCI contexts, they nevertheless seriously limit the statistical power of an evaluation. Unfortunately, due to the enormous training effort necessary for users in order to achieve significant BCI control in SMR setups and the implications of the man-machine learning dilemma (see Section 5.2) for development, testing, and iterative refinement of brain-computer interfaces, inclusion of a higher number of test subjects simply would not have been feasible for the purposes of the present master’s thesis, and is in fact not feasible for a large number of similar projects.

ARTEFACT CONTROL

Furthermore, various evaluation methods which came to be regarded as standard procedures in BCI research in the past few years were not available for the evaluation of the present project. Concretely, due to a lack of appropriate equipment, neither electrooculography (EOG), nor electromyography (EMG), nor electrocardiography (ECG) could be employed in the evaluation of the present brain-computer interface.

While the inclusion of EOG, EMG, and ECG data in routine BCI operation is very useful for artefact correction and similar purposes, it is strictly optional during this phase. As long as a BCI works well enough in day-to-day use without processing any additional biosignal data to detect artefacts masking signal data, recording EOG, EMG, or ECG data is not necessary. When evaluating the performance of a brain-computer interface, on the other hand, analysing this data is important to ensure that the BCI is in fact controlled via brain signals and to rule out control via EMG artefacts. McFarland et al. (2005) encountered this phenomenon in some users who controlled cursor movement with EMG rather than EEG signals in their first BCI sessions.

However, while it is of course possible that users controlled the present BCI by cranial EMG instead of EEG, McFarland et al. (2005) note that for most users EMG contamination is a problem primarily in early sessions and typically lessens as users attain actual BCI control via EEG. Furthermore, EMG tends to affect frontal regions more strongly than sensorimotor cortex locations and generally seems to play a greater role in unsuccessful rather than in successful trials (ibid.). Therefore it seems unlikely that the test subjects could have developed a meaningful level of BCI control by relying solely on EMG.

Chapter 9

Conclusion

In the present thesis, a self-paced cursor control brain-computer interface utilising EEG sensorimotor rhythms, power spectral density-based features, and a support vector machine classifier was introduced. In order to transform the sequence of class labels into a coherent control signal, namely movement of a mouse cursor, a simple inertial cursor model was proposed.

The BCI was evaluated over multiple runs using a standard system-paced two target task. Although the accuracy and information transfer rates achieved by the test subjects in this evaluation were significantly lower than that of current state of the art interfaces, the BCI's performance was in the same order of magnitude as that of more advanced interfaces. In summary, the BCI performed quite acceptably for an experimental prototype.

While the present BCI's performance and requirements regarding user training effort are far from ideal, being able to query the BCI via standard input interfaces proved quite useful in practice and could allow users of similar interfaces to control existing applications without the target software adaptations currently required by other BCIs. Furthermore, the inertial cursor model employed in the cursor function to convert discrete class labels into continuous cursor movement was relatively effective and could possibly be adapted to match the cursor functions of more modern interfaces better and supplement the linear translation functions typically used in these interfaces.

Given the extensive changes that would be necessary in order for the present BCI to achieve performance comparable to that of more modern brain-computer interfaces, further improvements in this regard do not seem to be feasible. Instead, a more advanced open source BCI framework such as BCI2000 (Schalk et al., 2004) could be extended in order to be accessible via standard input interfaces.[1] In addition, the performance characteristics of

[1]Brunner et al. (2013) provide an extensive and – as of September 2013 – up-to-date

the inertial cursor simulation in combination with linear translation functions, such as the one employed by the Wadsworth Center BCI, could be evaluated.

Brain-computer interfaces are a comparatively new field of research, and today, BCIs are still mainly used in experimental applications rather than in day-to-day use. The present diploma thesis presents a way to facilitate brain-computer interfacing by exposing BCI data via standard input interfaces, yet the amount of training necessary to successfully control a computer via a brain-computer interface, the cumbersome setup procedures, and performance considerations still stand in the way of more widespread adoption of BCIs. However, the vast number of BCI-related publications in the past two decades and the advances of the past few years suggest a promising future for brain-computer interfaces.

survey of open source BCI software platforms.

Bibliography

Brendan Z. Allison and Christa Neuper. Could Anyone Use a BCI? In Tan and Nijholt (2010a), chapter 3, pages 35–54.

Gerhard Bauer, Franz Gerstenbrand, and Erik Rumpl. Varieties of the Locked-in Syndrome. *Journal of Neurology*, 221(2):77–91, Aug 1979.

Hans Berger. Über das Elektrenkephalogramm des Menschen. *Archiv für Psychiatrie und Nervenkrankheiten*, 87:527–570, 1929.

Niels Birbaumer, Nimr Ghanayim, Thilo Hinterberger, Iver Iversen, Boris Kotchoubey, Andrea Kübler, Juri Perelmouter, Edward Taub, and Herta Flor. A Spelling Device for the Paralyzed. *Nature*, 398(6725):297–298, Mar 1999.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, New York, 2006.

Benjamin Blankertz and Klaus-Robert Müller. Machine Learning and Signal Processing Tools for BCI. `http://videolectures.net/bbci09_blankertz_muller_mlasp/`, Aug 2009. accessed 24 Apr 2013.

Benjamin Blankertz, Ryota Tomioka, Steven Lemm, Motoaki Kawanabe, and Klaus-Robert Müller. Optimizing spatial filters for robust EEG single-trial analysis. *IEEE Signal Processing Magazine*, 25(1):41–56, Jan 2008.

Clemens Brunner, Giuseppe Andreoni, Lugi Bianchi, Benjamin Blankertz, Christian Breitwieser, Shin'ichiro Kanoh, Christian A. Kothe, Anatole Lécuyer, Scott Makeig, Jürgen Mellinger, Paolo Perego, Yann Renard, Gerwin Schalk, I Putu Susila, Bastian Venthur, and Gernot R. Müller-Putz. *BCI Software Platforms*. Biological and Medical Physics, Biomedical Engineering. Springer, Berlin, 2013.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–

27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Jennifer L. Collinger, Brian Wodlinger, John E. Downey, Wei Wang, Elizabeth C. Tyler-Kabara, Douglas J. Weber, Angus J.C. McMorland, Meel Velliste, Michael L. Boninger, and Schwartz Andrew B. High-performance neuroprosthetic control by an individual with tetraplegia. *The Lancet*, 381 (9886):557–564, Feb 2013.

Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. *Machine Learning*, 20(3):273–297, 1995.

Shirley M. Coyle, Tomás E. Ward, and Charles M. Markham. Brain-computer interface using a simplified functional near-infrared spectroscopy system. *Journal of Neural Engineering*, 4(3):219–226, Sep 2007.

Emanuel Donchin and D. B. D. Smith. The contingent negative variation and the late positive wave of the average evoked potential. *Electroencephalography and Clinical Neurophysiology*, 29(2):201–203, Aug 1970.

Thomas Elbert, Brigitte Rockstroh, Werner Lutzenberger, and Niels Birbaumer. Biofeedback of slow cortical potentials I. *Electroencephalography and Clinical Neurophysiology*, 48(3):293–301, Mar 1980.

Georg E. Fabiani, Dennis J. McFarland, Jonathan R. Wolpaw, and Gert Pfurtscheller. Conversion of EEG activity into cursor movement by a brain-computer interface (BCI). *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 12(3):331–338, Sep 2004.

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

Siamac Fazli, Márton Danóczy, Florin Popescu, Benjamin Blankertz, and Klaus-Robert Müller. Using Rest Class and Control Paradigms for Brain Computer Interfacing. In Tan and Nijholt (2010a), chapter 4.

Eberhard E. Fetz and Dom V. Finocchio. Correlations between activity of motor cortex cells and arm muscles during operantly conditioned response patterns. *Experimental Brain Research*, 23(3):217–240, 1975. doi: 10.1007/BF00239736.

Henri Gastaut. Some Aspects of the Neurophysiological Basis of Conditioned Reflexes and Behaviour. In G. E. W. Wolstenholme and Cecilia M. O'Connor, editors, *Neurological Basis of Behaviour*, chapter 15, pages 255–276. Churchill, London, 1958.

Bernhard Graimann, Brendan Allison, and Gert Pfurtscheller, editors. *Brain-Computer Interfaces: Revolutionizing Human-Computer Interaction*. The Frontiers Collection. Springer, Berlin, 2010.

William Grey Walter, Ray Cooper, V. J. Aldridge, W. C. McCallum, and A. L. Winter. Contingent negative variation: An electric sign of sensori-motor association and expectancy in the human brain. *Nature*, 203(4943):380–384, Jul 1964.

Christoph Guger, Alois Schlögl, Christa Neuper, Dirk Walterspacher, Thomas Strein, and Gert Pfurtscheller. Rapid prototyping of an EEG-based brain-computer interface (BCI). *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 9(1):49–58, Mar 2001.

Bernard Marius 't Hart. Illustration of electrode locations of the 10-20 system. `http://www.mariusthart.net/downloads/eeg_electrodes_10-20.svg`, Oct 2011. accessed 19 Sep 2013, CC BY-SA 3.0.

Leigh R. Hochberg, Mijail D. Serruya, Gerhard M. Friehs, Jon A. Mukand, Maryam Saleh, Abraham H. Caplan, Almut Branner, David Chen, Richard D. Penn, and John P. Donoghue. Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature*, 442(7099):164–171, Jul 2006.

Leigh R. Hochberg, Daniel Bacher, Beata Jarosiewicz, Nicolas Y. Masse, John D. Simeral, Joern Vogel, Sami Haddadin, Jie Liu, Sydney S. Cash, Patrick van der Smagt, and John P. Donoghue. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485:372–377, May 2012.

Joachim Kalcher and Gert Pfurtscheller. Discrimination between phase-locked and non-phase-locked event-related EEG activity. *Electroencephalography and Clinical Neurophysiology*, 94(5):381–384, May 1995.

Philip R. Kennedy, Roy A. E. Bakay, Melody M. Moore, Kimberly Adams, and John Goldwaithe. Direct control of a computer from the human central nervous system. *IEEE Transactions on Rehabilitation Engineering*, 8(2): 198–202, Jun 2000.

Zoltan J. Koles, Michael S. Lazar, and Steven Z. Zhou. Spatial Patterns Underlying Population Differences in the Background EEG. *Brain Topography*, 2:275–284, Jun 1990.

Roman Krepki, Benjamin Blankertz, Gabriel Curio, and Klaus-Robert Müller. The Berlin Brain-Computer Interface (BBCI) — towards a new communication channel for online control in gaming applications. *Multimedia Tools and Applications*, 33(1):73–90, Apr 2007.

Andrea Kübler, Boris Kotchoubey, Thilo Hinterberger, Nimr Ghanayim, Juri Perelmouter, Margarete Schauer, Christoph Fritsch, Edward Taub, and Niels Birbaumer. The Thought Translation Device: A Neurophysiological Approach to Communication in Total Motor Paralysis. *Experimental Brain Research*, 124(2):223–232, Jan 1999.

Fernando Lopes da Silva and Ab Van Rotterdam. Biophysical Aspects of EEG and Magnetoencephalogram Generation. In Niedermeyer and Lopes da Silva (1999), chapter 5, pages 93–109.

Fabien Lotte, Marco Congedo, Anatole Lécuyer, Fabrice Lamarche, and Bruno Arnaldi. A review of classification algorithms for EEG-based brain–computer interfaces. *Journal of Neural Engineering*, 4(2):R1–R13, Jun 2007.

Dennis J. McFarland, William A. Sarnacki, and Jonathan R. Wolpaw. Brain-computer interface (BCI) operation: optimizing information transfer rates. *Biological Psychology*, 63(3):237–251, Jul 2003.

Dennis J. McFarland, William A. Sarnacki, Theresa M. Vaughan, and Jonathan R. Wolpaw. Brain-computer interface (BCI) operation: signal and noise during early training sessions. *Clinical Neurophysiology*, 116(1):56–62, Jan 2005.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. Adaptive Computation and Machine Learning series. The MIT Press, Cambridge, Massachusetts, Aug 2012.

Melody Moore Jackson and Rudolph Mappus. Neural Control Interfaces. In Tan and Nijholt (2010a), chapter 2, pages 21–33.

Ernst Niedermeyer. Alpha rhythms as physiological and abnormal phenomena. *International Journal of Psychophysiology*, 26(1-3):31–49, Jun 1997.

Ernst Niedermeyer. Historical Aspects. In Niedermeyer and Lopes da Silva (1999), chapter 1, pages 1–14.

Ernst Niedermeyer. The Normal EEG of the Waking Adult. In Niedermeyer and Lopes da Silva (1999), chapter 9, pages 149–173.

Ernst Niedermeyer and Fernando Lopes da Silva, editors. *Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*. Lippincott Williams & Wilkins, Philadelphia, 4th edition, 1999.

Paul L. Nunez, Ramesh Srinivasan, Andrew F. Westdorp, Ranjith S. Wijesinghe, Don M. Tucker, Richard B. Silberstein, and Peter J. Cadusch. EEG coherency: I: statistics, reference electrode, volume conduction, Laplacians, cortical imaging, and interpretation at multiple scales. *Electroencephalography and Clinical Neurophysiology*, 103(5):499–515, Nov 1997.

Gert Pfurtscheller and Alberto Aranibar. Occipital rhythmic activity within the alpha band during conditioned externally paced movement. *Electroencephalography and Clinical Neurophysiology*, 45(2):226–235, Aug 1978.

Gert Pfurtscheller and Fernando H. Lopes da Silva. Event-related EEG/MEG synchronization and desynchronization: basic principles. *Clinical Neurophysiology*, 110(11):1842–1857, Nov 1999.

Gert Pfurtscheller and Reinhold Scherer. Brain-computer interfaces used for virtual reality control. In *Proceedings of the 1st International Conference on Applied Bionics and Biomechanics (ICABB 2010)*, Venice, Oct 2010.

Gert Pfurtscheller, Christa Neuper, Doris Flotzinger, and Martin Pregenzer. EEG-based discrimination between imagination of right and left hand movement. *Electroencephalography and Clinical Neurophysiology*, 103(6):642–651, Dec 1997.

Gert Pfurtscheller, Clemens Brunner, Alois Schlögl, and Fernando H. Lopes da Silva. Mu rhythm (de)synchronization and EEG single-trial classification of different motor imagery tasks. *Neuroimage*, 31(1):153–159, May 2006.

Danny Plass-Oude Bos, Boris Reuderink, Bram Laar, Hayrettin Gürkök, Christian Mühl, Mannes Poel, Anton Nijholt, and Dirk Heylen. Brain-Computer Interfacing and Games. In Tan and Nijholt (2010a), chapter 10, pages 149–178.

Arman Rahmim and Habib Zaidi. PET versus SPECT: strengths, limitations and challenges. *Nuclear Medicine Communications*, 29(3):193–207, Mar 2008.

Edward L. Reilly. EEG Recording and Operation of the Apparatus. In Niedermeyer and Lopes da Silva (1999), chapter 7, pages 122–142.

Stephen Roberts. The Discrete Fourier Transform. `http://www.robots.ox.ac.uk/~sjrob/Teaching/SP/l7.pdf`, May 2003. accessed 12 Jun 2013.

Claudia Sannelli, Carmen Vidaurre, Klaus-Robert Müller, and Benjamin Blankertz. Common Spatial Pattern Patches: online evaluation on BCI-naive users. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2012:4744–4747, 2012.

Gerwin Schalk, Dennis J. McFarland, Thilo Hinterberger, Niels Birbaumer, and Jonathan R. Wolpaw. BCI2000: A general-purpose brain-computer interface (BCI) system. *IEEE Transactions on Biomedical Engineering*, 51 (6):1034–1043, Jun 2004.

Alois Schlögl. *The Electroencephalogram and the Adaptive Autoregressive Model: Theory and Applications*. PhD thesis, Technische Universität Graz, Graz, Apr 2000.

Alois Schlögl, Carmen Vidaurre, and Klaus-Robert Müller. Adaptive Methods in BCI Research - An Introductory Tutorial. In Graimann et al. (2010), pages 331–355.

Ranganatha Sitaram, Haihong Zhang, Cuntai Guan, Manoj Thulasidas, Yoko Hoshi, Akihiro Ishikawa, Koji Shimizu, and Niels Birbaumer. Temporal classification of multichannel near-infrared spectroscopy signals of motor imagery for developing a brain-computer interface. *Neuroimage*, 34(4): 1416–1427, Feb 2007.

Andrej Stancák and Gert Pfurtscheller. Event-related desynchronisation of central beta-rhythms during brisk and slow self-paced finger movements of dominant and nondominant hand. *Cognitive Brain Research*, 4(3):171–183, Oct 1996.

Samuel Sutton, Margery Braren, Joseph Zubin, and E. R. John. Evoked-potential correlates of stimulus uncertainty. *Science*, 150(3700):1187–1188, Nov 1965.

Desney S. Tan and Anton Nijholt, editors. *Brain-Computer Interfaces*. Human-Computer Interaction Series. Springer, London, 2010a.

Desney S. Tan and Anton Nijholt. Brain-Computer Interfaces and Human-Computer Interaction. In Tan and Nijholt (2010a), chapter 1, pages 3–19.

Jacques J. Vidal. Towards direct brain-computer communication. *Annual Review of Biophysics and Bioengineering*, 2:157–180, Jun 1973.

Jacques J. Vidal. Real-time detection of brain events in EEG. *Proceedings of the IEEE*, 65(5):633–641, May 1977.

David J. Ward, Alan F. Blackwell, and David J. C. MacKay. Dasher—a data entry interface using continuous gestures and language models. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*, UIST '00, pages 129–137, New York, 2000. ACM.

Jonathan Wolpaw and Elizabeth Winter Wolpaw. Brain-computer interfaces: Something new under the sun. In Jonathan Wolpaw and Elizabeth Winter Wolpaw, editors, *Brain-Computer Interfaces: Principles and Practice*, chapter 1, pages 3–12. Oxford University Press, Oxford, 2012.

Jonathan R. Wolpaw and Dennis J. McFarland. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the National Academy of Sciences of the United States of America*, 101(51):17849–17854, Dec 2004.

Jonathan R. Wolpaw, Dennis J. McFarland, Gregory W. Neat, and Catherine A. Forneris. An EEG-based brain-computer interface for cursor control. *Electroencephalography and Clinical Neurophysiology*, 78(3):252–259, Mar 1991.

Jonathan R. Wolpaw, Niels Birbaumer, Dennis J. McFarland, Gert Pfurtscheller, and Theresa M. Vaughan. Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, 113(6):767–791, Jun 2002.

Jonathan R. Wolpaw, Dennis J. McFarland, Theresa M. Vaughan, and Gerwin Schalk. The Wadsworth Center brain-computer interface (BCI) research and development program. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):204–207, Jun 2003.

Thorsten O. Zander, Christian Kothe, Sabine Jatzev, and Matti Gaertner. Enhancing human-computer interaction with input from active and passive brain-computer interfaces. In Tan and Nijholt (2010a), chapter 11, pages 181–199.

Stephan Zschocke. *Klinische Elektroenzephalographie*. Springer, Berlin, May 1995.