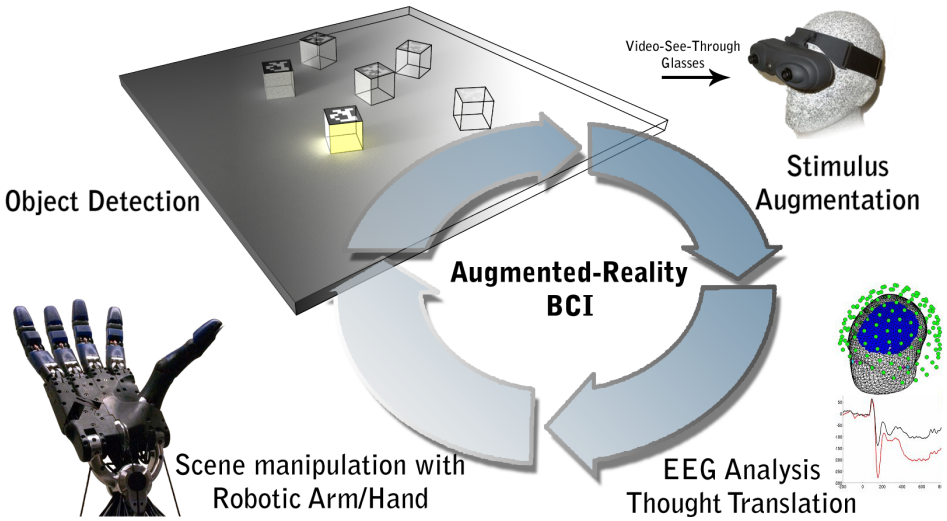
A Brain-Computer Interface for Robotic Arm Control



Alexander Lenhardt

Bielefeld

2011

1. Gutachter: Prof. Dr. Helge Ritter
2. Gutachter: Prof. Dr. Franz Kummert

Vom Promotionsausschuss der Technischen Fakultät
der Universität Bielefeld zur Erlangung des akademischen
Grades Doktor-Ingenieur (Dr.-Ing.) genehmigte Dissertation

Gedruckt auf alterungsbeständigem Papier °° ISO 9706

For my family.

Contents

List of Figures

List of Tables

Abstract

Brain-Computer Interfaces (BCI) are tools that open a new channel of communication between humans and machines. The majority of human input devices for computers require proper functioning of our primary sensors and motor functions like grasping, moving and visual perception. In the case of severe motor disabilities, like amyotrophic lateral sclerosis (ALS) or spinal chord injury (SCI), these pathways are blocked and cannot be used for communication. Since the brain functions of these people are not affected by the disease, BCI can provide an effective way to translate thought into machine readable commands.

The most common method to measure brain activity suitable for BCI are electroencephalographic measurements (EEG) due to their relative cost effectiveness and ease of use. Alternative ways to extract brain signals exist but either require invasive procedures, i.e. opening the skull, or are very costly and bulky (MEG, fMRI) which renders them unusable for home appliance. One of the most popular brain controlled input methods is the *P300-Speller* paradigm [Farwell and Donchin, 1988] which gives the user control over a virtual keyboard to enter text. The term P300 refers to a specific EEG component that can be measured whenever a rare task relevant stimulus is interspersed with many non-relevant stimuli. This method requires the ability to control the visual presentation of stimuli and therefore also requires some sort of computer controlled display. The recognition rates for this type of BCI, yet already quite high with roughly 80-90% accuracy, are still prone to errors and may not be suitable for critical applications like issuing movement commands to a wheelchair in a highly populated environment. Commands to stop the wheelchair might be recognized too late. Further, it is impossible with the standard stimulus matrix to react to external influences like obstacles or select physical objects in a scene which does not allow the user to interact with a dynamic environment.

This work aims to fuse state of the art BCI techniques into one single system to control an artificial actuator like a robot arm and use it to manipulate the physical environment. To achieve this goal, multiple techniques originating from different fields of research as augmented reality, computer vision, psychology, machine learning and data mining have to be combined to form a robust, intuitively to use input device.

1 Introduction

1.1 Aim and Scope

From a historical point of view it was not long ago since computers first appeared on the stage. With the invention of the arguably first programmable computer, the Z1 built by Konrad Zuse around 1938, the number of computing devices is growing at an enormous pace. Clearly the number of computer users is growing to the same extent and nowadays it is a mandatory skill to be able to use a computer with ease. Primarily intended as a tool to automate calculations it has evolved from a simple tool to an all-rounder used in entertainment, multimedia applications, industrial controlling and has even almost replaced pencil and paper. During the evolution of computers numerous devices have been developed to allow for easy communication between humans and computers. These devices are often specifically tailored to a certain task. A prime example is the keyboard which is almost exclusively used to communicate letters to the computer and is less suitable to manipulate graphical objects as they occur in graphical applications like drawing programs, modern graphical user interfaces or games. For the latter task usually 2D-pointing devices like mice or trackballs are used since this type of task demands non-symbolic continuous input data for which pointing gestures are the most intuitive way. Numerous other input devices have been developed for even more specialized tasks, e.g. braille keyboards and displays (figure 1.1) intended for sight impaired people as a replacement for the standard keyboard and computer screen. Yet, all of these input devices require that the user has at least to some extent voluntary control over their limb movement. For severely impaired people as it is the case with amyotrophic lateral sclerosis (ALS) or spinal chord injury, no movement of the muscles in the lower extremities or even the whole body is possible. Therefore none of the preceding input devices is applicable for this group of people. Even though the target group of ALS patients in a late phase is quite small, people with spinal chord injuries are quite common. People who have lost motor control over their body completely and are unable to talk, move or express their feelings in other ways are considered as *locked-in* patients since they are essentially prisoners in their own body. As their brain is usually not severely damaged, one possible way to get access to their world would be to extract relevant information about their intentions directly from their brain-activity. Nowadays we have several methods to measure brain activity of humans, both invasive and non-invasive techniques. Devices exploiting brain-activity data for communication are not even new. Considering the development timeline of the computer mouse, it

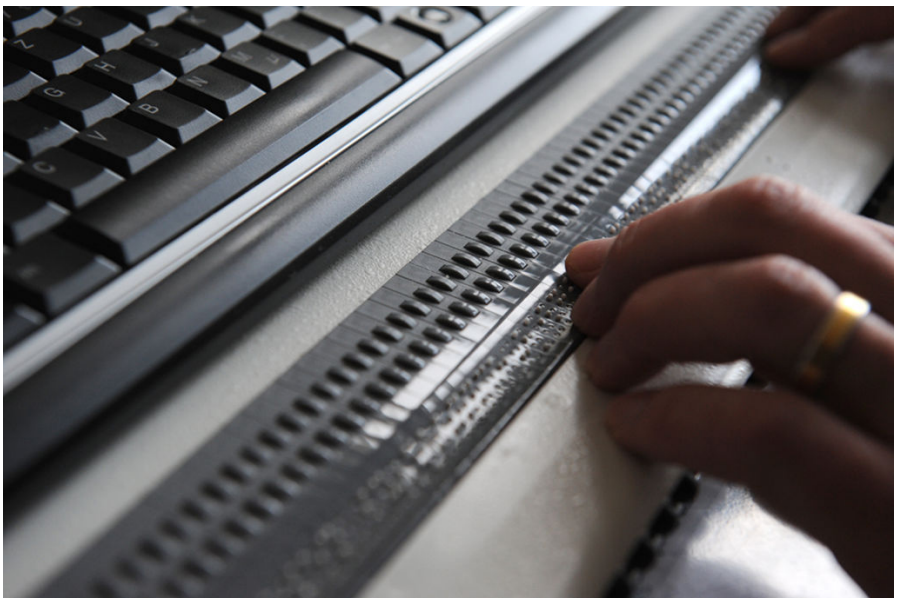


Figure 1.1. Braille-Displays allow visual impaired people to read the content of a computer screen by using tactile-coded alphabets.

was around 1984 when the first mice were shipped with the appearance of Macintosh computers to a broader user base. At first this novel input device was kind of neglected and in the same year it was an article in the San Francisco Examiner which states that "There is no evidence that people want to use these things." [Dvorak, 1984]. Only 4 years later Farwell & Donchin published a paper [Farwell and Donchin, 1988] with their groundbreaking finding that a certain type of event-related brain signal measured by an EEG called P300 component can be used to determine the users intent if coupled with the right stimulus presentation method. They designed an experiment that displayed the letters of the alphabet arranged in a grid structure. The usage is very simple as the subject only has to focus the letter to be communicated. During the experiment each row and column is highlighted in random order while the subject has to mentally count whenever the focused letter is being highlighted. This method is comparable to the keyboard input methology as it selects predefined symbols in an order defined by the subject. The success of this method in BCI research is probably based on the robustness of the involved brain-signals and high recognition rates. It is difficult to see how this input methology, yet event-driven and relying on external stimulation, can effectively be used to control artificial actuators like robot arms which intuitively call for continuous control signals to constrain their movements. There exist approaches based on motor imagery, i.e. imagination of limb movements, that seem to be more suitable for controlling robot arms [Wolpaw et al., 2003a, Blankertz et al., 2006, Pfurtscheller et al., 2003] since they do not rely on external event-triggered stimuli to evoke a brain response. Experiments as in [Leeb et al., 2004] showed that especially the cognitive load introduced with motor imagery

tasks can be a problem because it distracts from focusing on the environment and shifts attention to the imagination of movements. Another drawback appears in time critical settings, i.e. fine motor control since time is needed to reliably detect the intended movement. This in itself would not be a problem but real-time controlling of the movement would be for fine grained movements. As stated in [Graimann et al., 2008], a BCI handling all the intricates of low-level interactions needed to achieve a goal would not only be rather slow it can also be very frustrating for the user which in turn can lead to degraded performance. An alternative are so called *goal-directed* approaches which focus on selection of task relevant goals, i.e. selecting a target location for BCI controlled wheelchair, rather than control of the actuator's motors itself. An autonomous system takes care of the low-level control of the motors are with respect to the selected goal. Therefore, goal-directed control strategies for complex systems like robotic arms or wheelchairs are likely to be more robust than direct control of all available degrees of freedom.

These facts motivate this work which aims at developing a robust Brain-Computer Interface system suitable for controlling object manipulation tasks in real environments using a robotic manipulator. A goal-directed approach will be be used as the basis strategy whereas the motor control itself will be performed in an autonomous way on the robot platform. For goal selection tasks the P300-Speller paradigm is very well suited since the P300 is one of the best investigated components and exhibits stable signal properties which can be classified robustly. And more importantly, a BCI based on P300 evoked potentials does not apply additional cognitive load to the subject since the P300 is evoked subconsciously and no dedicated mental action is required for its evocation which allows for a much more natural communication in agreement with what was already proposed in [Ritter et al., 2007]. Controlling a robot arm to manipulate the physical world also entails that at least the position, size and shape of manipulable object must be recognized by the computer to transform these objects into potential goals, i.e. computer controlled visual stimuli, for the selection task.

To solve this problem, computer-vision and *Augmented-Reality* techniques in conjunction with a stereoscopic *head-mounted display (HMD)* can be used to populate and augment a real-world scene with computer generated stimuli. Figure 1.2 shows how a so called *video see-through* HMD will be used that is equipped with two firewire cameras which record the subject's field of view and projects it stereoscopically on the HMD LCD screens. As a proof of concept, marker-based methods will be employed to extract object information from the scene. The extracted objects will serve as selectable targets that serve as endeffector positions for the robotic arm. The finished system will consist of four distinct parts, each responsible for a specific task within the framework (see Figure 1.2).

Further, possibilities to improve the standard speller paradigm will be investigated in terms of information transfer rates and accuracy. Another important characteristic of a brain-adequate BCI is the possibility for human intervention at any point of an action sequence performed by the robot. This is an important point since wrong movements lead to unintended behavior and in the worst case can even hurt the user or damage the equipment. Recent studies have reported the presence of error-related brain potentials which occur in response to in-

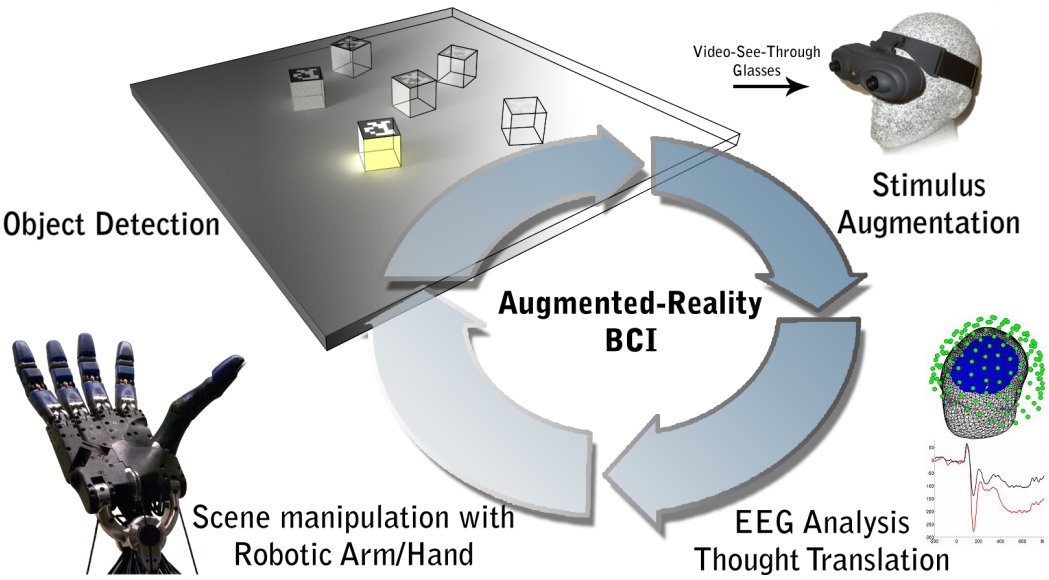


Figure 1.2. Schematic view of an *Augmented-Reality Brain-Computer Interface*.

correct BCI feedback. Of primary interest could be the so called *interaction ErrP* signal as reported in [Ferrez, 2008] which was found to be present in a simulated human-robot interaction task and occurred whenever the robot moved to the wrong side. Within the scope of this work it will be investigated if and how these types of signals can be incorporated into a BCI controlling a real robot arm.

The major difficulty here lies in the reliable detection of erroneous responses in single trials. Usually, as with the P300 detection, multiple trials are averaged to increase the signal-to-noise ratio of the data. This of course is not possible for the error detection method since feedback, i.e. the wrong movement of the robot arm, is only available once and therefore only one short time-series signal must be sufficient to reliably detect an error. There exist only few studies that incorporated ErrP responses into P300 BCIs [Buttfield et al., 2006, Dal Seno et al., 2009] whereas only [Dal Seno et al., 2009] used it in a running online experiment with limited success. The presented work will try to clarify in which cases automatic error correction using error-related potentials are useful and which factors determine the effectiveness of the error correction method. To summarize, the envisioned goals for this thesis can be stated as:

- Develop an augmented-reality based visual interface suitable for P300 BCIs.
- Investigate further performance improvements of the standard P300-speller paradigm.
- Investigate the feasibility of error correction using ErrP for the use in human-robot interaction.
- Implement an online BCI to control a robotic manipulator in realtime.

1.2 Structure

The structure of this work consists of three main parts. In the first part, background information on the topics of human brain anatomy, neurophysiological methods to measure brain activity and brain-computer interfaces is given. A special focus is put on EEG methods and EEG signals that are useful for BCI. This includes event-related potentials with emphasize on the P300 component as well as error-related potentials (ErrP) that can be measured at the occurrence of wrong actions carried out by the BCI.

The second part will describe the existing P300-based online BCI that was implemented during my diploma thesis which served as a basis for this work as well as machine-learning methods and algorithms that were used in the existing BCI along with experiments that aim to improve these algorithms. Along these lines, an error correction method for P300 BCI will be presented and its impact on BCI performance will be evaluated on a theoretical basis.

The third part is devoted to the new augmented-reality BCI paradigm and a final experiment to assess the feasibility of the developed approach. Results of an online experiment will be presented which focuses on grasping objects with a robot arm that is controlled by the BCI.

Chapter 2 - Background This chapter starts with a neuroanatomical overview of the human brain. It continues to describe ways to measure brain activity in terms of metabolic and electrophysiologic activity. A special focus is put on Electroencephalography (EEG) as it is the primary technique in non-invasive BCI research. In this context, types of event-related potentials are explained as well as special kinds of evoked potentials called error-related potentials that occur due to erroneous BCI responses. The chapter closes with an overview of brain-computer interfaces and explains the different types of BCIs and principles involved to deduce intentions from brain activity.

Chapter 3 - Brain-Machine Interfaces The third chapter introduces the concepts and different types of Brain-Machine Interfaces. Among others, the existing P300 BCI that was implemented during my diploma thesis will be briefly described. This system served as a basis for this thesis and the algorithms and machine-learning techniques will be presented in detail.

Chapter 4 - Dynamic Subtrials In this first experiment, the performance of an algorithmical extension called *dynamic subtrials* is evaluated. This extension is intended to adapt the number of stimulus presentations dynamically throughout a running online session. The results have been published in [Lenhardt et al., 2008].

Chapter 5 - Error-Related Potentials In this chapter, the possibilities to integrate a rather new EEG feature called *error-related potentials (ErrP)* is investigated. It will be shown how ErrPs can diminish under certain experimental conditions. Additionally, a method to detect erroneous trials on a single-trial basis will be presented that achieves an accuracy comparable to state-of-the-art P300 detection performances.

Chapter 6 - Augmented-Reality based BCI In this chapter the design of an augmented-reality based P300 BCI is described. It contains details about how brain signals are translated into actions to control a robot arm and how physical objects are translated into computer controlled visual stimuli to elicit P300 potentials. A broad overview of the software design will be given while the implementation details can be found in the appendix.

Chapter 7 - An Asynchronous BCI for Robot Control The final experiment in this chapter evaluates the performance and usability of the new combined P300 and ErrP detection methods. The algorithms will be explained in a non-exhaustive way as they are already described in detail in the previous chapters.

Chapter 8 - Conclusion The conclusion closes with a summary of results and achievements of this work. Further, thoughts about future directions of research, possible extensions to the augmented-reality BCI and existing limitations are discussed.

2 Background

This chapter summarizes the most important aspects of the brain-computer interface research field. Background information about the human brain anatomy will be covered in the first part followed by an overview of measurement methods for brain-activity. Emphasis will be put on electroencephalography (EEG) than on other measurement techniques since this is the most widely used method for non-invasive BCIs and is the one used in this work. Further, brain-signals which are suitable for BCI will be described also with a primary focus on the P300 component.

2.1 Neuroanatomy of the human brain

The human brain is the most complex organ we know on earth. Millions of neurons compressed into a handy bulk of highly complex compound of mass. Even though we might think we understand the neurobiological workings of single units of the brain, the uncountable interconnections and high plasticity of the former makes it a seemingly impossible task to fully understand the working principles of the whole brain. With more than 100 billion neurons in an adult brain it is impossible to investigate the exact paths of information flow at each time instant and decode their meaning. To make the daunting task of analyzing the brain more feasible, neuroscientists approach the problem from a top down perspective which breaks down the whole brain into functionally distinct parts. The late eighteenth century brought up the field of neurophysiology which evolved rapidly and yielded groundbreaking methods to visualize the living human brain. The central nervous system is a bilateral and symmetric structure which is divided into six main parts as can be seen in figure 2.1. Modern experimental methods were able to assign distinct roles and functions to these regions.

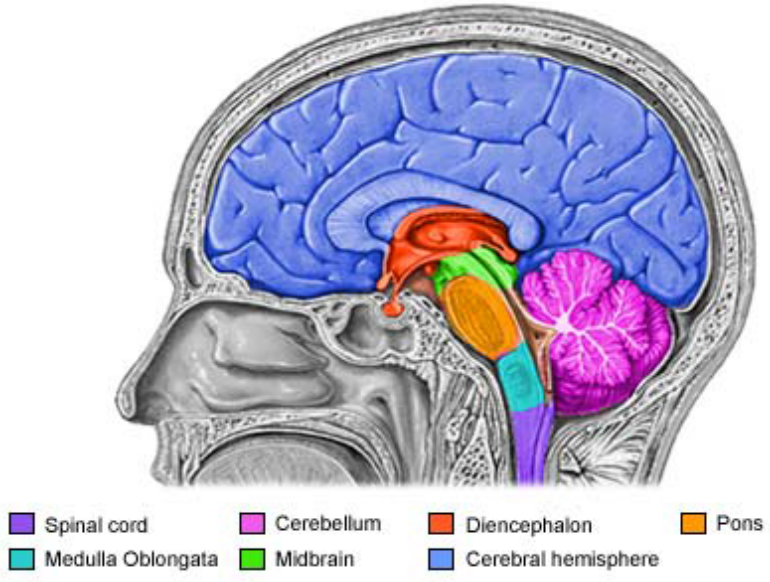


Figure 2.1. Cerebral Hemispheres.

Spinal chord The spinal chord is thought to be partly responsible for the control of limb movement and the trunk. It also processes sensory information from the skin, joints and their associated muscles. Further, specialized nuclei responsible for more specialized functions as hearing, balance and taste are present primarily towards the more rostrally located brain stem, consisting of the medulla, pons and midbrain.

Medulla Oblongata The medulla is located right above the spinal chord and controls vegetative functions like breathing, regulation of heartbeat or digestion.

Pons This structure emerging above the medulla is responsible for providing movement information originating from the cerebral hemisphere to the cerebellum. The cerebellum, which lies behind the pons, plays a major role in the learning of movements as well as controlling force and range of movements.

Midbrain The midbrain is involved in processing and controlling a variety of sensory and motor functions like eye movements and coordinates auditory and visual reflexes.

Diencephalon Two substructures, the hypothalamus which regulates endocrine, visceral and other autonomic functions and the thalamus which is responsible for processing informations reaching the cerebral cortex.

Cerebral hemispheres The cerebral hemisphere is the outermost structure which also contains three deeper lying substructures called basal ganglia, hippocampus and the amyg-

daloid nucleus. In conjunction, these regions are involved in aspects like memory storage, coordination of autonomic and endocrine responses to emotional states. The *cerebrum*, which hosts the cortex, covers the outer surface of the cerebral hemispheres. The cortex is morphologically characterized by its distinctive folds called gyri. These gyri make it possible to fit the whole cortex' surface or roughly $2000cm^2$ into the skull.

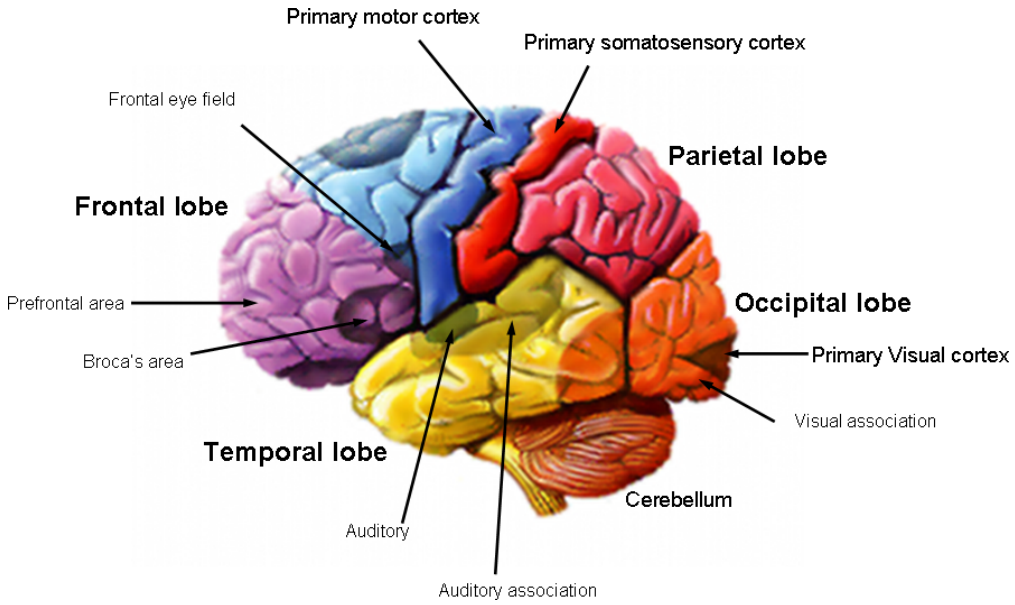


Figure 2.2. Mapping of cognitive functions to localized brain areas.

This broad mapping of functions to brain regions is not fully satisfying the highly complex and dynamic nature of the brain because the principle of *functional localization* can be extended to an almost infinitely fine grained level. One commonly accepted principle is that neurons serving a similar function are likely to be interconnected with each other. These neuronal assemblies get more and more specialized as one descends from macroscopic to microscopic level of detail. As an example the visual cortex, part of the occipital lobe located of the back of the cerebral hemispheres, contains neuronal assemblies that are specialized to code colors while others focus on shapes or motion. Perceiving an object, assigning semantics to it and using it in creative ways is a result of multiple specialized neuronal groups interchanging and processing our sensory information and thereby eventually creating something we call cognition. Supposedly all cognitive functions are processed in the structures of the cortex which is the outermost layer of the brain. Specialized areas are associated with specific sensory processing. The broad mappings can be extended to different aspects of cognition like perception of auditive stimuli, vision, body perception and other aspects of consciousness. Such a broad mapping is shown in Figure 2.2 for the left hemisphere which shows the localization of different areas responsible for dedicated parts of perception.

Frontal lobes The frontal lobes are located at the front of the brain, anterior to the neighboring parietal lobe and above the temporal lobe. It is thought to play an important role in long-term memory and planning as well as reward and attention. One of the reasons for this belief is the fact that the frontal lobes contain most of the cortex' dopamine-sensitive neurons which have been shown to be largely responsible for the aforementioned cognitive functions.

Parietal lobes Posterior to the frontal lobes, the parietal lobe can be found. The role of this lobe is primarily associated with the integration of multimodal sensory information and object manipulation. This includes mapping the position of visually perceived objects into body space coordinates and knowledge of the relation of different abstract metrics (e.g. based on numbers or spacial distances).

Occipital lobes The occipital lobes are the smallest anatomical region of the cortex and consist mainly of the visual system. They are located at the back of the head and thus are the rearmost of all four lobes. The function of these lobes are almost exclusively restricted to tasks related to vision. Certain sub-areas are specialized to color or edge direction detection while the back of the lobes are arranged in such a way as to reflect the retinal field as a spatial map.

Temporal lobes The temporal lobes are located beneath the parietal and frontal lobes at each side of the brain. Its functional role consists mainly of processing of smell, sound and more complex stimuli like face recognition or semantic processing of visual scenes or speech. Especially the left temporal lobe appears to be specialized to high-level auditory processing like speech. The ventral parts on the other hand are more specialized for high-level visual processing tasks. The roles associated with the medial parts of the temporal lobes consist of memory related functions like episodic and spatial memory, as well as playing a key role in transferring short-term memories into the long-term memory.

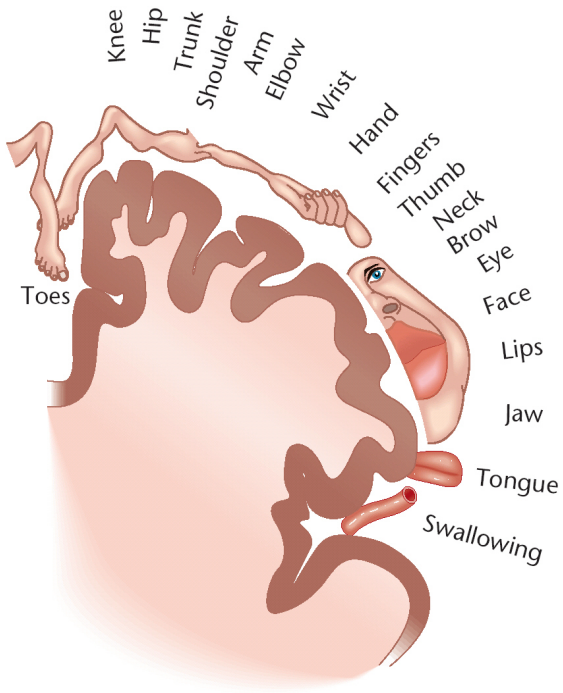


Figure 2.3. Homunculus shows the cortical areas covered by sensory processing of the body.

A specialized area like the somatosensory cortex can be further subdivided into distinct areas associated with different body parts. This can be visualized as a sensory map as seen in Figure 2.3. The size of the warped body parts corresponds to the size of the cortical area they occupy. Especially the face and lips occupy a remarkably large brain area as they are very sensitive.

2.2 Measuring neuronal activity

A variety of methods exist which measure different aspects of brain activity. Some techniques exploit the electrical properties of the brain and measure voltage (**EEG**) or magnetic field changes (**MEG**) at the skull surface and thus measure neuronal activity. Others such as functional imaging methods like **PET**, **fMRI** or **NIRS** measure metabolic processes of the brain. Further, some methods are more invasive than others. The most invasive type requires, in addition to opening the skull, to implant electrodes directly into the cortex whereas with less invasive methods like **ECoG** electrodes are only attached on top of the cortex' gray matter. The reason for the diversity of measurement techniques is simple. Generally speaking, the more invasive the better the spatial resolution. On the other hand, the less invasive the less risky and complex the procedure. Additionally, most non-invasive electrophysiologic tech-

niques lack the ability to measure high frequency content, i.e. the fast temporal dynamics, of the data since the skull attenuates the signals and acts as a lowpass filter. The full information range is only covered by the most invasive techniques like *intracellular recordings* or *single/multi unit recordings* of action potentials (**SUA/MUA**) which in turn are not able to cover the whole cortex. It is only feasible to cover small areas with microelectrodes. Figure 2.4 depicts the techniques that will be described on a two-dimensional map based on their spatial and temporal resolution. The primary method **EEG**, which was used in this work, will be covered in more detail in the next subsection.

Positron Emission Tomography (PET) Positron emission tomography (PET) is a technique that uses radioactive marked substances, called tracers, to indirectly image metabolic processes within the human brain. The tracer is a substance that is metabolized by the brain during activity. Usually fludeoxyglucose (FDG), an analogue of glucose, is used as a tracer. Activity in brain tissues results in increased glucose uptake which can be measured by detecting pairs of gamma rays emitted by the tracer. This type of scan is usually combined with computer tomographic x-ray scans to visualize the anatomic structures of the tissue resulting in a 3-dimensional image of the brain augmented with metabolic activity. The scan itself is rather slow due to the neurovascular coupling, i.e. the slow metabolic responses to neuronal activity. Further, the injection of radioactive molecules, though with a short half-life time, bears a certain risk for the subject.

Functional Magnetic Resonance Imaging (fMRI) Function Magnetic Resonance Imaging (fMRI) is able to measure metabolic changes in living brain tissues, just as the former mentioned technique. In contrast to PET it does not require any sort of radioactive tracer and thus eliminates the risk involved with radiation. Neural activity is thought to be correlated to increased blood deoxygenation since active neurons require a greater amount of energy which is delivered in the form of oxygen. Hemoglobin, the molecule binding and releasing oxygen, has different magnetic properties depending on its state of oxygenation. To measure neural activity, fMRI detects relative changes of oxyhemoglobin and deoxyhemoglobin. This contrast is called Blood-oxygen-level dependent signal (BOLD). As with PET, this technique also requires relatively long scan times and picks up signal frequencies in the range of 4-5Hz.

Near Infrared Spectroscopy (NIRS) Near infrared spectroscopy is a rather new non-invasive technique that can measure metabolic processes of the brain. Similar to fMRI, NIRS employs special properties of oxygenated blood to receive a BOLD signal. In contrast to the former technique, NIRS emits near infrared light of a specific wavelength. This light penetrates the skull and cortex to a depth of 1-2cm . The reflected light is detected at multiple positions. Depending on the position and absorption of the light, blood oxygenation can be deduced from that data. As with with all former methods, the temporal resolution of NIRS is also bound by the neurovascular coupling that leads to signals of 4-5Hz.

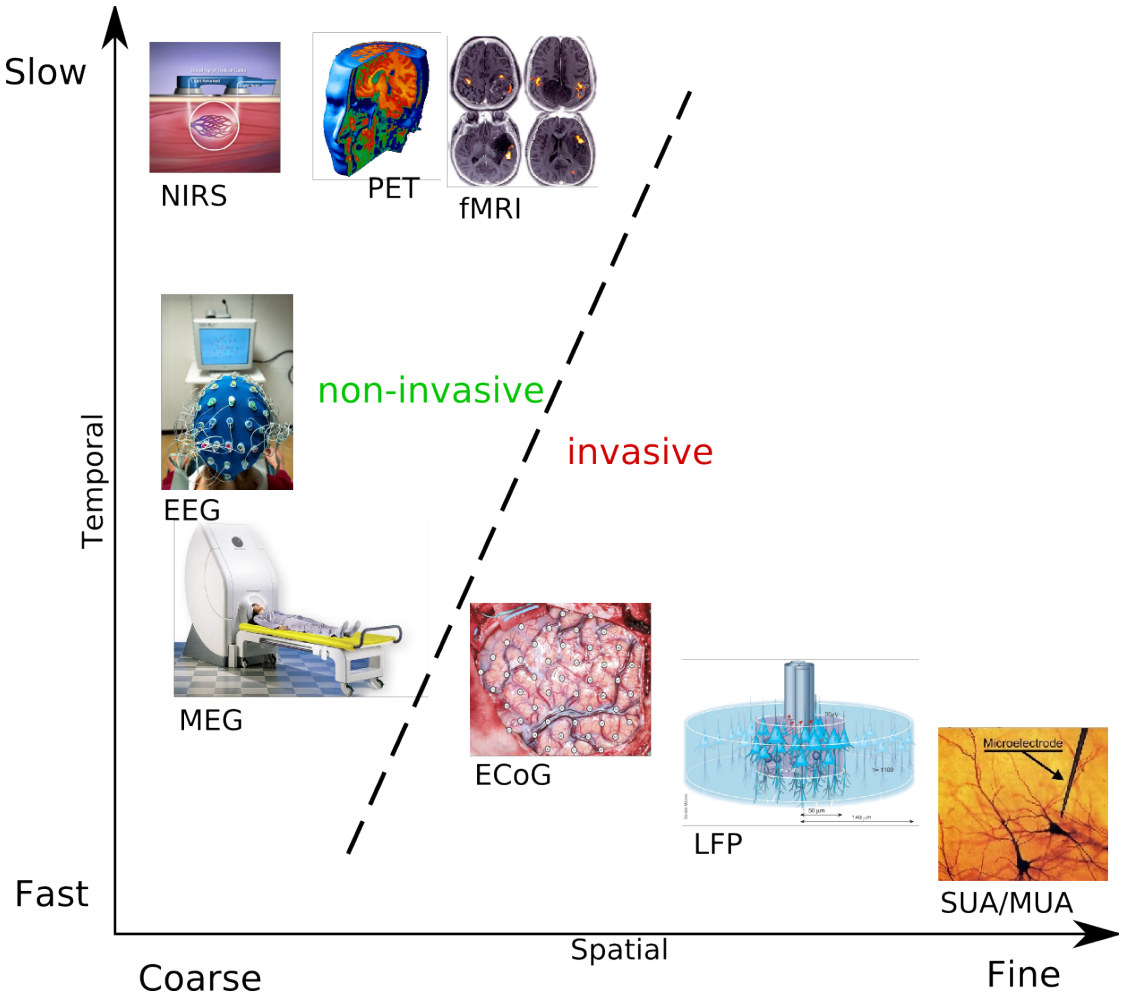


Figure 2.4. Recording techniques ordered by their temporal and spatial resolution. Invasive techniques generally expose better spatiotemporal properties than non-invasive techniques but are more complex to apply and pose higher risks of injuries.

Magnetoencephalography (MEG) This technique measures the magnetic fields that are generated by the electrical currents of neuronal activity. It provides a very high temporal resolution but on the other hand requires huge and costly hardware equipment to acquire the signals. Further, it is very susceptible to movement artifacts which contaminate the signals.

Electrocorticogram (ECoG) As EEG, the electrocorticogram is a technique to record electrophysiological activity where the sensors are directly placed on the surface of the cortex. Since the signals do not have to penetrate the skull and skin anymore the signal quality is vastly improved compared to the non-invasive electroencephalogram. The price to pay for the good temporal and spatial resolution however lies in the invasive natures of the technique. Opening the skull is inevitable since the gray matter of the cortex must be accessible for the electrodes.

Local Field Potentials (LFP) Stepping further down into the microscopic level, the recording of *local field potentials (LFP)* allows an even better spatial resolution. This technique records summed activity from a small cluster of neurons. Thereby a low-impedance microelectrode is implanted close to the neuron cluster but sufficiently far away from individual neurons to prevent that activity of single neurons do not suppress the activity of others in the summed signal. The resulting unfiltered signal reflects the summed neuronal action potentials of a $50-140\mu m$ radius [Buzsaki, 2004].

Single/Multi-Unit Actionpotentials (SUA/MUA) The most accurate in terms of spatial and temporal resolution are single- and multi-unit recordings of action potentials. This type of recording technique is usually only applied to animals since the health risks involved are almost certain to occur. As a result, scars and neuronal injuries will develop during and after the recording since microelectrodes are directly implanted into the cortex and single neuronal cells. It allows for recording of single (or multiple) cell action potentials at their exact firing-rate.

2.3 Electroencephalography (EEG)

Back in 1875 the english physician Richard Caton discovered the presence of electrical potential fluctuations in the cortex of animals. Due to his findings there was no doubt that brain activity can be measured from potential deflections measured from the skull or cortex which result from activity of ganglia cells of the nervous system. The first scientist to exploit his findings was Hans Berger. He is considered the pioneer of electroencephalography (EEG) as he recorded the first electrical brain activity on graph paper which is shown in Figure 2.5. During his research he noticed that prominent rhythmic activity varied with conscious states. Most notably the so called *alpha-rhythm*, also termed *berger wave*, which is clearly visible when the subject closes the eyes. Early on, the potential fluctuations were interpreted as the sum or

superposition of action potentials of ganglia cells firing in synchrony [Adrian and Matthews, 1934]. This theory however could not explain the slow varying components in EEG recordings. Later it could be shown by investigations with microelectrodes, that actionpotentials of ganglia cells almost vanished when moving further away from the cell than $1\mu m$. Therefore, even when measuring activity directly on the cortex, the action potentials are vanished which raises the question what the actual potential sources of EEG are.

2.3.1 Sources of EEG

A revised theory on generators involved in EEG was formulated based on the research of [Brooks and Eccles, 1947, Eccles, 1964] who identified synapses as an additional source of electrical potential changes. Activation of synapses leads to a local potential gradient which is measurable around the center of activity. In contrast to action potentials and their very short duration of 1-2ms, postsynaptic potentials exhibit a slower potential change over time in the range of 10-40ms. The summation of single synaptic potentials, as shown in figure 2.6, at the dendrites exhibit the slow varying characteristics visible in EEG recordings and allow for a manifold of different superpositions. In conclusion, the EEG measures the cortical field potential changes induced by the summation of postsynaptic potentials. Once the strength of these field potentials exceeds a certain strength it can be registered at the skull surface. An important factor in the development of the strength of cortical field potentials is the degree of postsynaptic activation synchronicity. The more synapses are activated at the same time the more potentials sum up which results in a strong field potential. Even though this is an

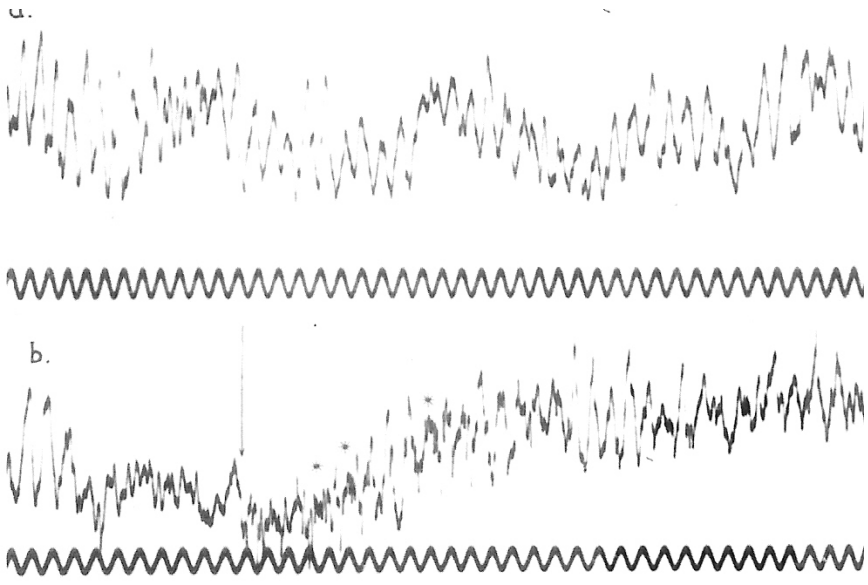


Figure 2.5. First EEG recording by Hans Berger shows a prominent 10Hz alpha-rhythm.

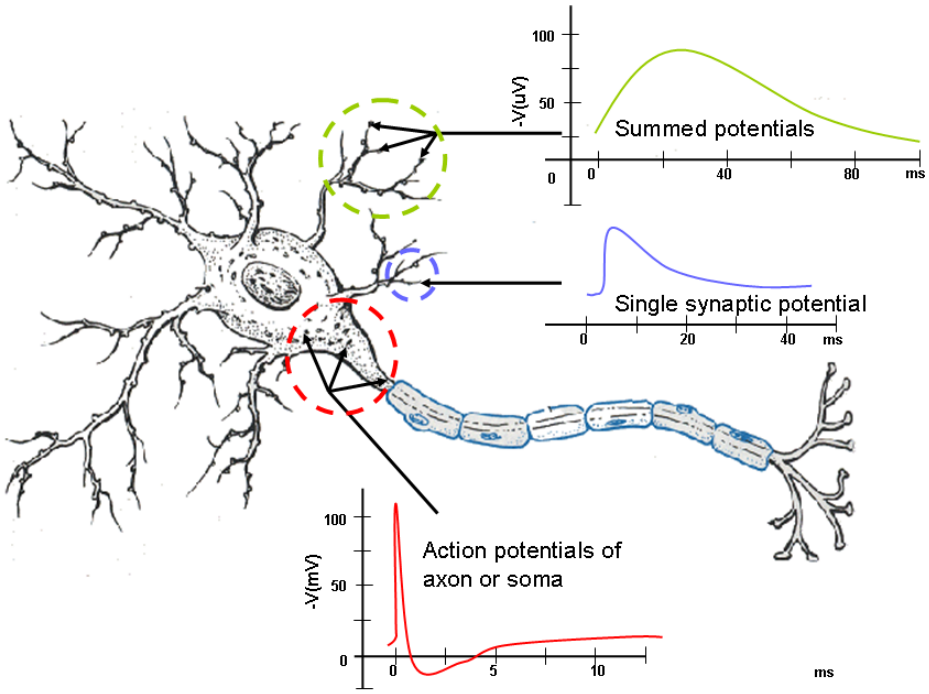


Figure 2.6. Several potential generators can be identified at different locations in and around the cell. The main generator contributing to the EEG are the summed postsynaptic potentials as shown in the top trace.

important factor, not all neurons contribute to the strength of field potentials. The spatial orientation of the cells if also of great importance since these cells can be considered electric dipoles and the direction of the electrical field influences how potentials are summed. Primarily the vertically arranged pyramidal cells are the main contributors to the EEG. Whenever many vertically oriented cells get activated at the same time, many equally directed field potentials can sum up to form a strong field. Only these dipoles are relevant for EEG derivations. Fortunately about one third of the cortex exhibits such an orientation. The remaining cells are oriented in a less consistent way and as such they cancel out their contributing field to a great extent.

2.3.2 Recording techniques

EEG recordings are obtained by placing electrodes on the scalp of the subject. Usually the scalp has to be specifically prepared in order to reduce artifactual influences that result from high impedances between electrodes. Ethanol, light abrasive pastes and conductive gels are

used to improve conductivity between skin and electrodes. EEG measures voltage differences, therefore it is necessary to define pairs of electrodes which are compared to each other. Each pair is connected to a differential EEG amplifier which amplifies the voltage difference up to 100.000 times which equals up to 100dB voltage gain [Niedermeyer and Da Silva, 2004]. Typically the maximum amplitude of a non-pathologic EEG does not exceed $100\mu V$ when measured on the scalp [Thompson and Patterson, 1974]. As mentioned, the derivation of an EEG requires the definition of reference electrodes. There exist four major methods that are used in the majority of EEG experiments.

Common reference In common reference derivations, each *recording electrode* represents the difference in voltage to a single *reference electrode* common to all recording electrodes. The reference electrode should be placed at a neutral place. Here, neutral means a place that reduces influence of the brain and muscle potentials. A common location are the ear lobes or right/left mastoid since these are relatively electrically neutral sites.

Common Average reference As in common reference, *average reference derivations* use a single reference. In contrast to the former however, this method uses the average of all recording electrodes as reference which has the effect of reducing voltage changes common with all electrodes. This can be useful since externally induced artifacts usually appear on all electrodes while retaining the inter-electrode differences for the most part.

Bipolar reference In bipolar EEG derivations, each channel connects two electrodes. Both of these electrodes are considered *active* electrodes, i.e. they are likely to be affected by brain potentials. They are arranged as linked serial pairs since the reference electrode of the first channel is the active electrode of the next channel. This arrangement scheme is extended throughout the entire electrode array.

In conjunction with the mentioned derivation methods, the placement of electrodes is an important issue. Data recorded from one site are required to be comparable among subjects. Since the geometry of the human head is not identical among subjects, the locations of electrodes will be different for each head. Comparable measurements can be achieved by employing the *International 10-20 System* [Jasper, 1958a] which uses two fixed reference locations, the nasion and the inion, that define the size of the head. The distance from nasion to inion is divided into 10% and 20% intervals in the horizontal and vertical plane (see Figure 2.7). Each location has a unique label that is systematically assigned. All labels start with letters, followed by either a number or the letter Z which denotes a central location on the heads midline. Labels ending on odd numbers are located on the left hemisphere while even numbers are on the right hemisphere. Further, the distance of the location to the midline can be determined by its trailing number. The smaller the number the closer the location to the midline. Consequently, the beginning letter denotes their location based on the major underlying brain areas such as occipital (O), temporal (T), parietal (P) or frontal (F) lobes.

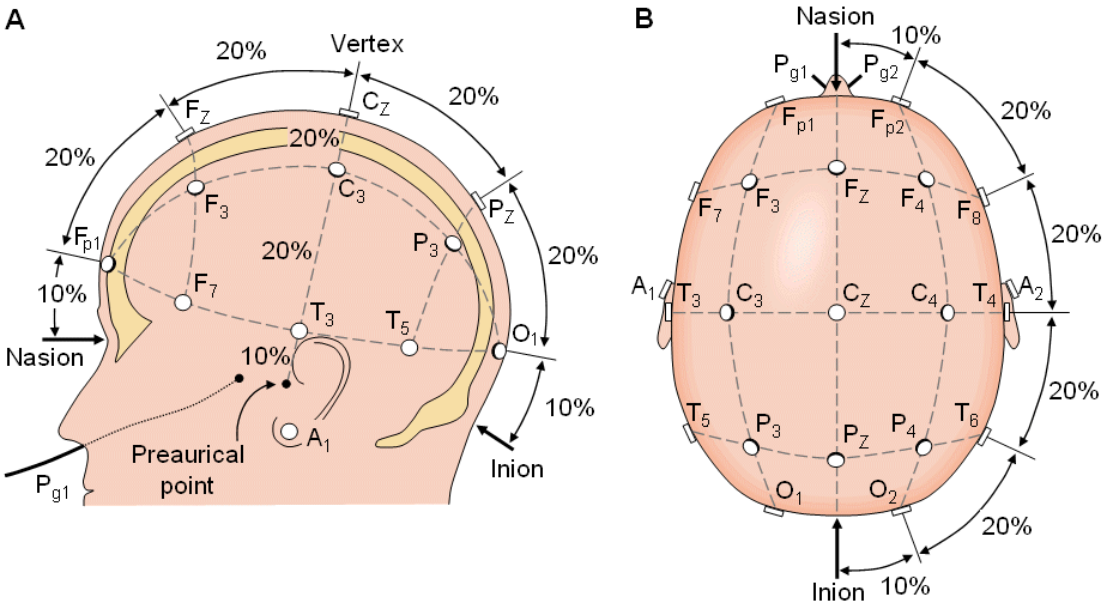


Figure 2.7. The international 10-20 system divides the head into equally sized section in the vertical plane (A) and horizontal plane (B).

2.3.3 Rhythmic activity

When recording an EEG, rhythmic patterns are commonly visible in the traces. Historically, these rhythms were classified by visual inspection whereas in later research these rhythms were classified by their specific frequency ranges. Therefore, no precise mapping exists between the naming conventions and their corresponding frequency range. Figure 2.8 depicts the most common types of EEG rhythmic activity with their associated frequency ranges. Along with these patterns, certain cognitive functions or mental states can be coarsely correlated [Niedermeyer and Da Silva, 2004, Webster, 2007].

Delta waves Delta waves are defined in the frequency band of up to 4Hz and show high amplitudes. They are usually found during slow-wave sleep (SWS) where the EEG consists of 20% delta waves as well as in newborns [Schulz, 2007]. Adults expose a frontal localization for this type while it is located more posterior in children.

Theta waves Residing in the 4-8Hz band, the strong theta waves are associated with drowsiness but also appear in childhood, adolescence and young adults. It is localized over the hippocampal area but its function is not well understood. Research done by [Green and Arduini, 1954] pointed out that these waves occur predominantly along with desynchronized EEG in the neocortex and suggest that it might be associated with arousal. In contrast, [Vanderwolf, 1969] presumed that, due to strong correlation of motor actions and theta, it might be related to sensorimotor processing.

Alpha waves/Mu-rhythm First discovered by Hans Berger and later fully confirmed by [Adrian and Matthews, 1934], the rhythmic activity of 8-12Hz is called the alpha band. It occurs primarily at posterior regions of the head over both hemispheres. Depending on the handedness of the person, the amplitude is slightly higher on the dominant side. The occurrence of alpha waves can be forced by closing the eyes. In general, occipital alpha is associated with relaxed or alerted mental states of consciousness. A similar rhythm called *mu-rhythm* can be observed over the motor cortex and is attenuated while carrying out or just imagining movements which is a very important feature used in certain types of BCI [Wolpaw and McFarland, 2004, Pfurtscheller et al., 2006]. Since alpha and mu expose overlapping frequency bands they are distinguished by their localization. Rhythmic 10Hz activity focused over the visual cortex is called *visual alpha rhythm* while the same activity appearing over the somatosensory cortex is called *mu rhythm*. According to [Niedermeyer and Da Silva, 2004] the generating sources of *mu rhythms* are the thalamocortical neuronal circuits. Studies of [Pfurtscheller and Berghold, 1989] revealed a relation to beta waves in that some of those could be separated either topographically or temporally and thus are independent EEG features. Since *mu rhythm* occurs in cortical areas directly connected to motor output channels, even the imagination/preparation of movements entails a decrease of mu and beta rhythms. This phenomenon has been labeled *"event-related (de)synchronization"* by [Pfurtscheller and Lopes da Silva, 1999].

Beta waves Beta waves of 12-30Hz are associated with normal waking consciousness while low amplitude beta with alternating frequencies within the band are related to active, busy or anxious thinking as well as active concentration. This rhythm is primarily localized frontally and appears symmetric on over both hemispheres.

Gamma waves The last major type are gamma waves with a frequency band of 30-100Hz. The functional role of gamma is not fully known. It is hypothesized that this type of rhythmic activity is related to conscious perception, in particular the *binding problem*[1]. There is however no agreement between researchers on the exact functional role.

2.3.4 Artifacts in EEG recordings

A common problem with EEG is the recording of artifacts that do not originate from the brain [Lindsley et al., 1974]. In the context of BCI, this definition is further extended to not only label non-brain originated data as artifact but also certain types of brain activity which might mask a signal of interest. A prime example are alpha rhythms which expose the same frequency content as the P300 component. With strong alpha activity the P300 diminishes as

[1] The binding problem refers to the phenomenon, how distinct sensations (e.g. color and shape) are combined to form a new perception (e.g. a red circle enclosed in a yellow square).

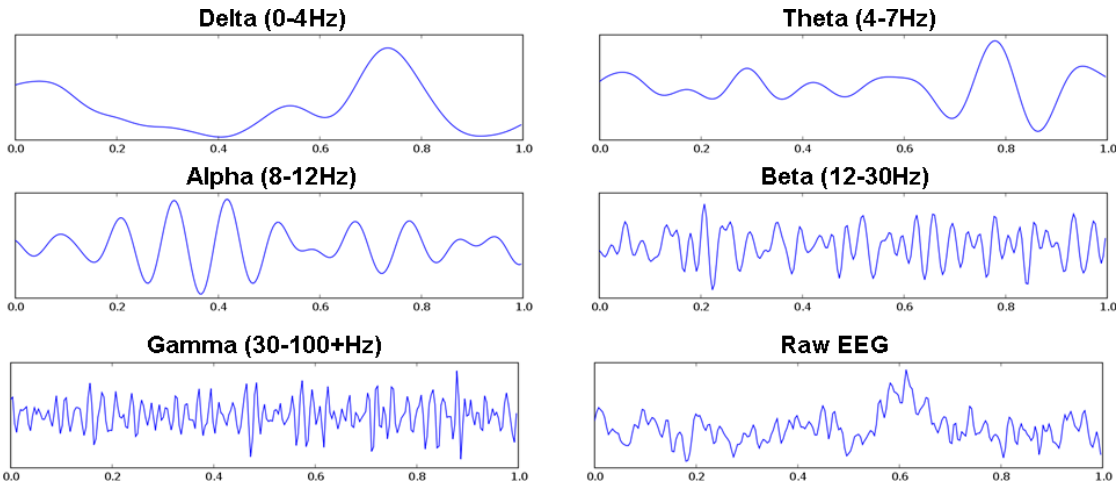


Figure 2.8. The most common types of rhythmic EEG activity. Each type has an associated frequency range.

it gets masked by the higher amplitudes of the alpha activity. In general, artifacts can be classified as subject dependent and subject independent artifacts. According to [Lindsley et al., 1974] subject dependent artifacts may arise from one or more of the following reasons:

- *EMG artifacts.* Electromyographical (EMG) artifacts are a result of increased muscle tension. Especially masticatory, neck and forehead muscles will corrupt the EEG with heavy signal disturbances of high amplitudes. The frequency range of these artifacts is usually higher than for the signals of interest. But high amplitudes of these artifacts can mask possible event-related potentials which are used in BCIs.
- *Cardiac artifacts.* The electrical activity of the heart muscle can also be visible in EEG recordings and appears as a pulsing negative deflection in the EEG.
- *Motion artifacts.* Body and limb movements will cause movement of the electrode cables and can also introduce myographic (EMG) artifacts.
- *Oculographic artifacts.* A very common type of artifact seen in every EEG recording is a result of eye blinks and eye movements. The eye, which can essentially be considered as a dipole, introduces very high amplitude voltage changes on the whole skull.

In contrast to subject dependent artifacts, subject independent artifacts can arise from electronic equipment or environmental electromagnetic noise. An artifact source present in every EEG recording is line noise induced by the EEG amplifiers power supply. The frequency range depends on the country's alternating current frequency which is different depending on the country. In europe a frequency of 50Hz is used whereas in the US a frequency of 60Hz is common. Further sources of noise can include grounding-loops, cell phones, escalators, computers and computer screens or static current of the subject itself [Lindsley et al.,

1974]. Grounding-loops are one of the more common artifact types. They occur whenever any element of a circuit is connected to ground at more than one point. In theory this should not be a problem assuming both points should have the same ground potential. In practice however, two different grounds never have the same potential due to slightly different resistances and therefore result in interferences or even electrical shocks which can damage the equipment or the subject. A properly grounded subject will use the same ground point as the internal circuits of the amplifier. In addition to ground-loops, electromagnetic interferences of the environment can pose another problem in EEG recordings. The electrodes act like an antenna picking up even the slightest electromagnetic signals of the surrounding and amplifying them greatly. To reduce this influence, it is necessary to properly prepare the subjects scalp in order to reach good transition resistances from reference electrodes to active electrodes. The reason for this is that electromagnetic influences are less likely to get picked up when the impedances of the electrodes (in DC recordings impedance equals resistance) are low. Besides achieving low electrode impedances, reducing the amount of electronic equipment is usually the best advice to improve recording quality. When signal quality is of the highest concerns, electromagnetically shielded rooms are another option since they reduce EM influences in the environment to an absolute minimum. This however is not practical in home usage of BCI which is why this option can be safely rejected for practical work with BCIs.

2.3.5 Limitations of EEG

The non-invasiveness, high temporal resolution, ease of use and relative cost effectiveness of EEG comes at a price. The spatial resolution of EEG is very limited since for each electrode, even though positioned above specific brain areas, electrical activity of locations in close proximity gets mixed into the recorded signal. Additionally, it is most sensitive to specific post-synaptic potentials which emerge at the crests (see section 2.3.1). Deeper neuronal layers and opposingly ordered dipole cells do not contribute to the EEG at all. Therefore, a precise localization and determination of the underlying brain dynamics is not possible. Specifically, for neuronal activity to be measurable it must fulfill certain constraints. Dipole neurons that generate electrical activity must be oriented in a perpendicular axis towards the scalp and their dendrites must be aligned in parallel to sum their potentials. Only summed potentials are strong enough to penetrate the skull and be detectable by EEG. The claim for summation also entails the constraint for synchronicity. Cells that do not fire in synchrony do not sum up their potentials to the maximum extent. In conclusion, EEG is only able to measure a very limited subset of the full brain dynamics and is not able to localize the origins and contributing components precisely. Another limitation is the very low signal-to-noise ratio (SNR) which is the relation of the signal of interest's amplitude to the noise amplitude present in every EEG recording.

2.4 EEG features for BCI

EEG based brain-computer interfaces rely on reproducible electrophysiological signals that allow to infer what command the subject intended to communicate. Over the years, different types of BCIs have been developed which use various kinds of electrophysiological properties to translate thoughts into commands. This section will give a short overview about the different methologies and their associated electrophysiological basis.

2.4.1 Motor Imagery

As briefly mentioned in section 2.3.3, an 8-12Hz mu-rhythm over the somatosensory cortex can be measured in awake people who are not processing any sensory input or producing motor output. The frequency range is identical to the visual alpha rhythm that occurs primarily over the visual cortex. It could be shown by [Pfurtscheller and Berghold, 1989] that this somatosensory 8-12Hz mu-rhythm is present in most adults. The actual rhythm is comprised of a variety of frequencies within this band which can be separated by their spatial location and characteristic frequency. These rhythms share correlations with 18-26Hz beta rhythm in such a way that a motor action, or even the imagination of such actions, will lead to a decrease of the mu- and beta-rhythms contralateral to the side of motor activity. This decrease has been labeled *Event-Related Desynchronization (ERD)* while the opposite *Event-Related Synchronization (ERS)* occurs with relaxation after the imagined or conducted motor action. In order to obtain a clear quantification on how much the power of the specific frequency band changed, a reference power is needed. This is usually obtained by measuring a baseline signal prior to any imagined movement which serves as a reference value in the following classification task. The time course of such an ERD/ERS phenomenon is depicted in

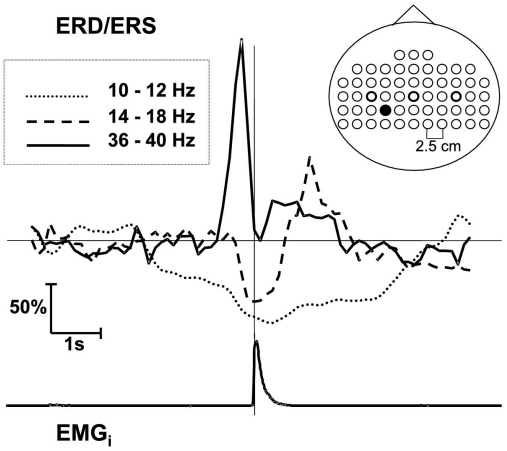


Figure 2.9. Rhythmic activity in the beta and mu band decrease (ERD) during imagination of motor actions and increase (ERS) after the imagination of motor actions. (Illustration adopted from [Durka, 2003]).

Figure 2.9 for a carried out movement. It is remarkable that already 1-2 seconds before the actual movement, a desynchronization in the alpha band is visible. This is commonly referred to as *Readiness Potential (RP)* or *Pre-Motor Potential*. This potential can be observed prior to a voluntary muscle movement and is associated with pre-motor cortical structures that plan the actual movement. The alpha ERD reaches its peak shortly after the movement onset and lasting for a few seconds until it approaches baseline level. The central beta activity in contrast shows a short lasting ERD quickly followed by an ERS that falls back to baseline level quicker than the alpha activity. Additionally, 40Hz gamma activity shows a sharp peak right before the movement onset to quickly reapproach baseline level right at movement onset. A BCI utilizing these features along with the topographical mapping of the measured signals can at least operate on two degrees of freedom since ERD/ERS are primarily measured on the contralateral side of the limb movement. This technique however requires subject training to some extent. Even though recent studies improved greatly in this topic (e.g. [Krauledat et al., 2008]) there remains the need for the subjects to be able to concentrate on the task and with no doubt trained subjects will perform better than untrained. It is however a highly popular method employed by numerous research groups (e.g. [Wolpaw et al., 1991, Blankertz et al., 2006, Pfurtscheller et al., 1997]).

2.4.2 Slow cortical potentials

Slow cortical potentials (SCP) expose the lowest frequency range of all EEG features suitable for BCIs. Positive deflections of this potential are associated with reduced cortical activity as suggested in [Rockstroh et al., 1984, Rockstroh, 1989, Birbaumer et al., 1990] while a negative deflection usually indicates increased cortical activity as it appears during movements or other activity involving cortical structures. A special property about SCPs are that subjects can learn to control the exposition of this signal voluntarily which has been shown in many studies of Birbaumer and his colleagues. The signal is used to control a 2D cursor on a computer screen to select visual goals. As a result, Birbaumer implemented a BCI commonly referred to as *Thought Translation Device (TTD)* which can be used to select letters by moving the cursor into the appropriate position. It is also remarkable that this kind of signal works well for the original target group of BCI users, namely patients with late-stage amyotrophic lateral sclerosis (ALS) [Kübler, 2000]. In contrast to motor imagery BCIs this method requires extensive user training of weeks or months to be able to achieve recognition rates of >75% [Wolpaw et al., 2002]. The recognition phase itself consists of a 2 second baseline measurement and a 4 second recognition phase during which the subjects have to control their SCP.

2.4.3 Visual evoked potentials

Visual evoked potentials (VEP) occur in response to visual stimuli and can be measured above the visual cortex. In BCI a special form of these potentials called steady-state visual evoked

Figure 2.10. Pole balancing task controlled by SSVEP. The stimuli for left and right movements are phase inverted checkerboard patterns, flickering at a specific rate.

potentials (SSVEP) can be utilized to determine the gaze direction of the subject. The term steady-state refers to the periodic property of the VEP when visual stimuli are presented at a rapid rate of 6-8Hz or more. The presentation of such flickering stimuli leads to overlapping measurable VEPs above the visual cortex which match the presentation frequency of the attended stimulus. In a study, Lalor et al. [Lalor et al., 2005] employed this feature in a game scenario with the aim to balance a pole.

2.4.4 Event-Related Potentials

Voltage changes timelocked to sensory, motor or cognitive events are commonly referred to as *event-related potentials (ERP)*. These voltage changes in ongoing EEG that may be stimulus related are usually in the order of microvolts and thus are hard to detect within a single presentation epoch. A common way to improve detectability is to average over multiple stimulus epochs which improves the *signal-to-noise ratio (SNR)* significantly. Any brain activity that is not time-locked to the stimulus onset will most likely disappear due to the averaging process and only the time-locked components are retained. An ERP consists of multiple components which are labeled according to their polarity (P or N) and latency. A component occurring at a latency of 100ms exposing a positive deflection is called P100 (or P1) whereas a component at a 200ms latency with a negative deflection is called N200 (or N2). Clearly, the P300 component which this work is mainly based on can be found approximately 300ms after a certain stimulus presentation showing a positive peak which only appears under very specific circumstances. Using this classification scheme, an ERP can be segmented by its components.

Common time-locked components that can be observed due to a sensory stimulus are the *N100*, *N200*, *P300* and the *N400*. In neurophysiological literature these components are further subdivided into subcomponents whereas only the *P3a* and *P3b* are of greater interest for this work. Therefore, a further subdivision of the remaining components will be omitted. A more extensive overview of ERPs and their respective components can be found in [Patel and Azzam, 2005, Rugg and Coles, 1995].

N100 The N100 is a time-locked ERP peaking at 80-120ms in response to a stimulus and can be measured over the fronto-central scalp region. The elicitation of this component is highly sensitive to unpredictable auditory stimuli in the absence of task demands. Yet, it is also present when presenting visual, olfactory or somatosensory stimuli [Warnke et al., 1994, Pause et al., 1996, Wang et al., 2008]. A strong correlation between stimulus repetition rate and N100 amplitude has been shown by [Schafer and Marcus, 1973] such that randomly occurring stimuli produce higher N100 amplitudes while repetitive stimuli expose a much lower amplitude or even disappear in the case when subjects are allowed to control the presentation of stimuli themselves.

N200 The N200 component, also called *Mismatch Negativity (MMN)*, appears in response to an odd stimulus in a long sequence of similar stimuli. The component's maximum peak spans from parietal to fronto-central sites. The MMN has been first studied by [Näätänen et al., 1978] in the context of auditory stimuli. The component was elicited by presenting a sequence of equal sounds with an interspersed deviant sound stimulus. Their study showed that an N200 is elicited even when the subject was not consciously paying attention to the sound sequence. A subcomponent, the *N2b*, usually precedes the P300 component and in contrast to the *N2a* only occurs when the subject actively attends the stimulus sequence. There exists no commonly accepted theory on the origins and functional role of the MMN. One of the more recent theories hypothesizes that sensory afferent neuronal elements that are tuned to the frequent stimulus respond less vigorous with each repetition of the standard stimulus. The MMN might then be a result of different neuronal elements tuned to the deviant which respond stronger upon deviant presentation [Näätänen, 1992].

N400 In 1978, Kutas and Hillyard conducted experiments in order to investigate whether sentence contexts in word recognition tasks can be measured using ERPs. They predicted that unexpected words in a sentence would elicit a certain ERP component nowadays called P3b which occurs usually after the presentation of an unpredictable stimulus interspersed in a sequence of predictable stimuli. It turned out that sentences with physically unexpected endings as in "'I shaved off my mustache and beard"' indeed elicited a P3b. To their surprise, sentences with a semantic anomaly as in "'I shaved off my mustache and city"' were followed by a large negative peak at 400ms over central-parietal sites after the semantically wrong ending [Kutas and Hillyard, 1980, Kutas and Hillyard, 1983, Kutas and Hillyard, 1984]. Factors that influence the N400 am-

plitude are repetition rate, semantic/associative priming, expectancy/cloze probability and attention. Each stimulus that in the first place elicited a N400 will also elicit it at the second repetition with the difference that the N400 amplitude is significantly decreased as shown by [Rugg, 1985]. A similar effect could be shown for experiments that incorporated semantic priming, i.e. experiments presenting a pre-stimulus that rendered the upcoming target stimulus more predictable (see e.g. [Harbin et al., 1984,Stuss et al., 1988]). Another very important determinant of N400 amplitude is the *cloze probability* (i.e. the proportion of people who give a particular word as the most likely completion of a sentence fragment), whereas according to [Kutas and Hillyard, 1984] high cloze probability correlates with decreased N400 amplitudes and vice versa. A controversial factor that could influence N400 amplitudes is attention. Studies seem to contradict on whether the processes involved in the N400 elicitation are automatic or of controlled nature. The component has been observed for masked stimuli [Misra and Holcomb, 2003] and even during sleep [Brualla et al., 1998]. On the other hand, experiments that controlled the extent to which attention was allocated to the eliciting stimulus showed a correlation between amplitude and allocated attention [Holcomb, 1988] and thus suggest that the N400 is at least not fully automatic.

P300 The P300 evoked potential has been discovered in 1965 by Samuel Sutton and colleagues. The component was found to be correlated to unpredictable stimuli interspersed in a sequence of uniform stimuli (an experimental paradigm called *oddball task*). In this paradigm, frequent *background stimuli* are shown in rapid succession and at a random position in that sequence a less frequent *target stimulus* is interspersed. Due to this target stimulus, a strong positive peak at around 300ms can be observed in the EEG which is localized above fronto-central to parietal regions. In addition to the traditional *two-stimulus oddball task* described above (cf. Figure 2.11 (middle)), a P300 can also be elicited by variations of this paradigm. A *single-stimulus task* (c.f. Figure 2.11 (left)) requires the user to mentally or physically respond to a target stimulus which is presented at random times with no other stimulus between target presentations. The second variation is a *three-stimulus oddball task* (c.f. Figure 2.11 (right)) that adds a distractor stimulus in addition to the background and target stimuli. The subject however is required to only respond to the target stimulus. This type of experiment is able to show that the P300 consists of two subcomponents, the *P3a* and *P3b* [Snyder and Hillyard, 1976]. In response to the distractor stimulus, which can be considered as a *novel stimulus* interspersed into the sequence, a P3a is elicited which is located towards central-parietal sites and shorter peak latency, whereas the P3b occurs as a response to task relevant stimuli/target stimuli at spatially more parietal sites than the P3a. Historically these components were thought to be distinct entities but more recent studies as in [Katayama and Polich, 1996, Katayama and Polich, 2001, Brocke, 2004] suggest that these components stem from the same ERP only varying topographically with attentional and task demands. The functional role of the P300 is controversially discussed

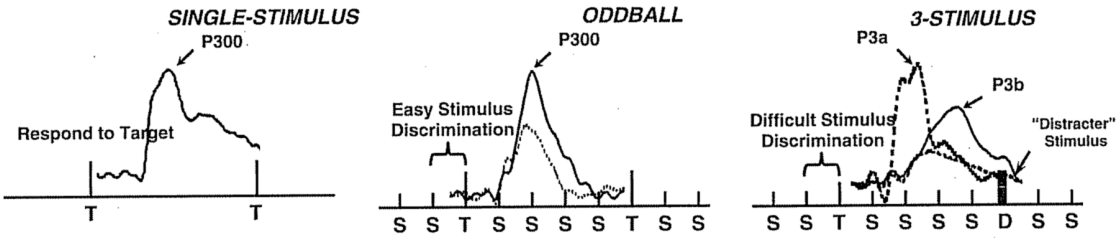


Figure 2.11. Illustration of 3 different experimental paradigms that are used to elicit a P300. The **single-stimulus (left)** requires the subject to focus on an infrequently appearing target marked **T** in the absence of other stimuli between consecutive targets. The traditional **2-stimulus oddball (middle)** requires target focus as well but between target presentation, frequent background stimuli appear. The **3-stimulus oddball (right)** adds a third stimulus class called *distractor*. It also appears less frequent than background stimuli but the subject must not respond to them. In this paradigm, a P3a is elicited for distractors and a P3b for targets. The illustration is a modified version of [Polich and Criado, 2006].

by researchers. A theory of *context-updating*, i.e. the adaption of a mental model to the oddball stimulus, was formulated by [Donchin, 1981]. Other studies as [Verleger, 1988] criticize this theory and suggest the P300 is involved in *context-closure* where a process waiting for an anticipated stimulus terminates. A more recent study of [Polich, 2007] speculates that the P300 might be an inhibitory process of ongoing neuronal activity to facilitate the transmission of stimulus information from frontal (P3a) to parietal (P3b) regions. The theory is based on the assumption that focal attention needed during the detection of stimuli which are resolved using the working memory. Therefore, inhibition of additional stimulus processing mechanisms might improve memory operation since this inhibition facilitates the information transfer from frontal to parietal regions. This hypothesis is supported by neurophysiological results and experiments [Friedman et al., 2001, Nieuwenhuis et al., 2005, Birbaumer et al., 1990]. Depending on the modality of the stimulus and several other aspects, the shape and the latency of the P300 can vary. The use of a visual oddball paradigm will prolong the latencies of the respective components. The same effect of prolonged latencies can be observed for tasks with varying task difficulty. The more difficult it is for the subject to discriminate between background and target stimuli, the more the latencies are prolonged. In a visual paradigm, stimuli with similar color intensities and contrast are harder to discriminate which result in longer latencies. Another factor that influences the appearance of the P300 is the target probability. It has been mentioned that targets must appear less frequent than background stimuli. A study of [Coles et al., 1990] showed the relationship of target probability and P300 amplitude for an auditory oddball paradigm (see Figure 2.12). The subjects were instructed to mentally count the high tones while ignoring the low tones which appeared as background stimuli. The presentation probability for targets was successively increased with the result that the P300 for the targets decreased with increasing target probability. A presentation probability of 90% let the P300 for

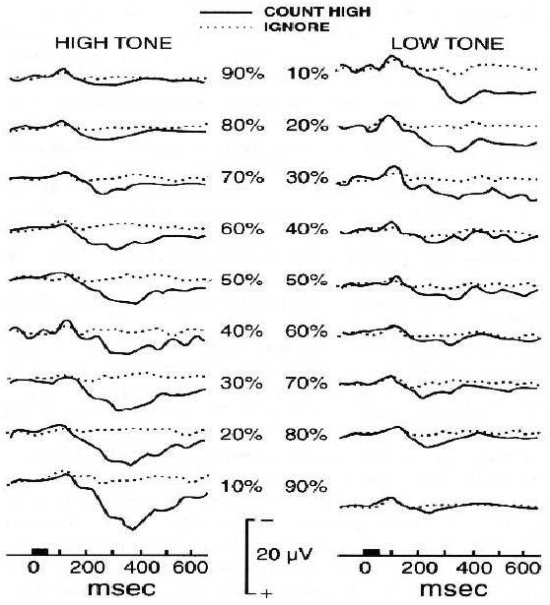


Figure 2.12. Target frequency effect on P300 amplitude. Infrequent targets evoke a P300 with larger amplitudes [Coles et al., 1990].

targets vanish almost completely. A similar effect as target probability emerges from variation of *inter-stimulus intervals (ISI)*. The ISI is the elapsed time between two consecutive stimuli. It was shown by the pioneers of BCI [Farwell and Donchin, 1988], that P300 amplitude is decreased with shorter ISI while latency is prolonged. An important finding was that discriminative potentials can still be obtained with an ISI of 125ms. The ability to evoke a P300 with such a low ISI makes it possible to design P300-based BCIs with rather fast communication rates as it will be described in chapter 3.

3 Brain-Machine Interfaces

These days, a life without computers or electronic devices is unconceivable. To facilitate the control of these devices, intuitive human-computer interfaces (HCI) had been developed to allow for easy communication between the user and the computer. Prime examples for such input devices are keyboards and mice. Since these devices seem to resemble a very natural way of communication they are only suited for healthy users who have complete control over their motor and visual functions. Blind persons would not be able to take advantage of such common input methods which led to new interfaces like Braille displays and keyboards specifically adapted for the needs of sight impaired people. For similar reasons, Brain-Machine Interfaces emerged for cases where common input devices are not viable. This new kind of input interface forms a direct bridge between the brain and a computer or computer controlled output device. Since these devices receive input directly from the brain, voluntary control over muscles is not necessary to operate them. This is especially useful for cases in which patients are in a completely paralyzed state but with a wake mind, a state referred to as *locked-in syndrome*. Patients with late *amyotrophic lateral sclerosis (ALS)* or a stroke with resulting lesions in the pons, which are the most common reasons for this syndrome, are unable to communicate with the outside world. At best they are only able to carry on an internal monologue [Patterson and Grabois, 1986]. Therefore it is even more remarkable that a man called Jean-Dominique Bauby, who was put into a locked-in state after mid-brain stroke, has written a whole book called *Le scaphandre et le papillon*. The book was dictated by himself to his speech therapist by using only eye-blink codes for each letter which took him over one year to complete. In the light of cases like this, it seems imperative to research new ways of communication and this is what gave rise to the invention of brain-computer interfaces. A pioneer in this field, Prof. Niels Birbaumer, devoted his research to the translation of brain-potentials to commands which resulted in his famous *thought-translation device (TTD)* (see e.g. [Birbaumer et al., 2003]). With this device, patients were able to operate a letter spelling device by mere thought after multiple initial training sessions. He applied this technique to the originally intended target group of locked-in patients with great success.

Techniques to translate thoughts into commands however is only a sub-field of the broader field of brain-machine interfaces. Generally, Brain-Machine Interfaces can be subdivided into invasive and non-invasive methods which mostly aim at different levels of control as depicted in Figure 3.1. Non-invasive methods are usually limited in their ability to decode brain patterns as the recording techniques involved are either temporally or spatially inaccurate

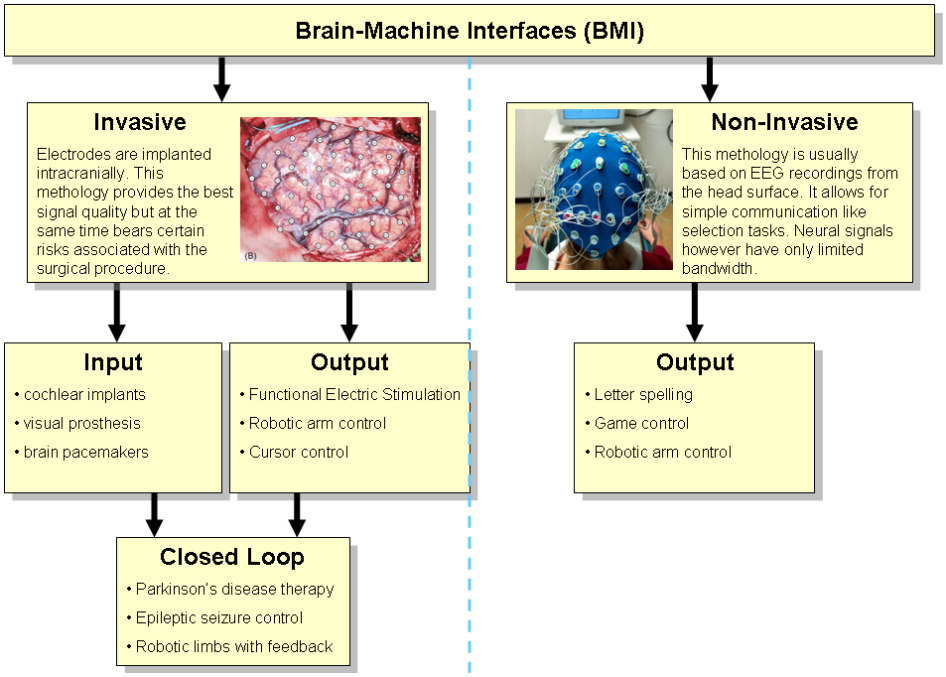


Figure 3.1. Classification of brain-machine interfaces into invasive and non-invasive methods.

(c.f. Figure 2.4). Therefore they are best suited for cases where fine-grained control is not necessary. Further, stimulation of the brain is either not possible at all or only with a very low spatial resolution as is the case with *Transcranial-Magnetic Stimulation (TMS)* where whole brain areas can be muted by large magnetic pulses. Invasive methods on the other hand offer superior signal quality and spatial resolution which in turn results in finer control of the output device. This chapter will give an overview of both, invasive and non-invasive brain-machine interfaces with a stronger focus on non-invasive methods based on EEG.

3.1 Invasive Methods

The first experimental demonstration in 1999 with rhesus macaques showed how signals of neuronal ensembles can be employed to directly control a robotic arm [Chapin et al., 1999]. Recent research concentrates on the improvement of implantable electrodes and signal acquisition modules to reduce side effects like scars in the brain tissue and long-term applicability. In a review of human brain-machine interfaces [Nicolelis, 2001], Nicolelis described how these methods can be employed to generate artificial neuronal signals triggered by brain-activity. This can be useful in cases like epilepsy where seizure activity can be terminated once it gets detected by stimulating cranial nerves. Besides the apparent benefits of these methods, the health risk involved should be evaluated carefully. In most cases the risk

of bioincompatibility and tissue scars will outweigh the benefit for most patients. Further, these approaches are not suitable for healthy subjects at all which renders them less interesting for non-therapeutical applications like gaming or assistive devices for healthy subjects.

Input-Invasive BMI

Input-invasive BMIs transform external stimuli into bioelectric activity that is being processed by the brain. A very popular and well established input-invasive BMI is the cochlear implant. This device records external sound sources by small microphones similar to a hearing aid. The implant though is directly connected to the auditory nerve fibers and translates the recorded sound into bioelectric activity which in turn serves as input for the auditory nerve. The implant is not able to reconstruct the natural hearing since the number of electrodes used to connect to the nerve fibers does not even exceed 0.001% of the normal nerve connectivity. Yet, even with a low number of electrodes, a rudimentary hearing reconstruction is possible.

Similar to methods that reconstruct basic hearing senses, visual prostheses are able to reconstruct lost vision or amplify existing vision. These neural prostheses rely on externally worn cameras that transmit pictures to a translation device which transforms them into bioelectric signals suitable for neuronal processing. Several different types of visual prostheses exist, one of the most interesting from a technical point of view are visual cortical implants (e.g. [Coulombe et al., 2007]). This type is implanted directly above the visual cortex and communicates via a wireless connection with the image acquisition hardware. The implanted part of the system consists of a central module which drives a multitude of micro-stimulators to properly induce sensing of bright spots in the subject's field of view. One of the first to report a successful implantation of such an device was Dobelle in 1974. The patient was able to recognize simple patterns using this device.

A slightly different application emerged in the context of pain relief. In 1965, Melzack and colleagues proposed a new theory called *Gate control theory* [Melzack and Wall, 1965]. It was hypothized that nerves carrying electrically coded tactile and vibrational sensations terminate in the pain fibers which are located at the dorsal horn of the spinal chord. Thereby, two types of cells are involved in the processing. The *transmission (T)* and *inhibitory* cells. Signals from the pain and sensory fibers excite the T cells until they reach a critical level at which time pain sets in. Inhibitory cells on the other hand counteract to the T cells by dampening the T cell excitation, similar to closing the gate to pain (as formulated by Melzack and Wall). The theory concludes that sensory fiber activation excited the inhibitory cells while at the same time, pain fiber activation impedes inhibitory cells and thus leaves the gates to pain open. Therefore, if sensory fiber activation is high compared to pain fiber activation, less pain will be sensed. Due to this finding a spinal chord stimulator (SCS) was built by [Shealy et al., 1967] in order to aid in chronic pain therapy. An SCS uses electrodes implanted on top of the spinal chord and a pulse generator which drives the excitation signal. The sensory fiber excitation can be controlled on demand by the patient and activates the inhibitory cells as described

above. Recent research in this field has extended the applicability of this type if device also to the treatment of Parkinson's disease [Fuentes et al., 2009].

Output-Invasive BMI

Compared to the former input invasive methods, the flow of information is reversed for output-invasive brain-machine interfaces. Neuronal signals originating from the brain are being interpreted by a device that translates them into actions that would normally require muscle activity. One of the most widely used brain signals are neuroelectric potentials arising from motor cortical activity. These signals can be translated into commands to control either computer cursors or even limb prostheses. This field was pioneered by Chapin who succeeded to train rats to control a robotic arm along a 1D trajectory using only their brain signals. This was accomplished by implanting microwire electrodes into the rats primary motorcortex and the ventrolateral thalamus. The information contained in spike signals of the neuronal ensembles were mathematically transformed into neuronal population functions that could accurately predict the lever trajectory. Similar work was conducted by [Wessberg et al., 2000] who trained monkeys to precisely control a lever along a 1D and 3D trajectory. In this study, simultaneously recorded cortical ensemble data of the monkey was used to train a linear model and an artificial neural network (ANN). The recording was realized with implanted microwire electrodes over multiple cortical sites but primarily over the motorcortex. After the implantation of the electrodes, the monkeys were trained for 12 and 24 month to move a lever either to the left or to the right. Subsequently, a more complex task requiring three-dimensional trajectories was trained. This task involved controlling a robot arm to reach small pieces of food which were placed at one out of four random positions on a tray. Neuronal data was recorded simultaneously to the actual movement and served as input to either a linear model or an ANN. The hand trajectories predicted by the two methods revealed highly significant correlations to the real trajectories with average correlation coefficients of 0.71 for the linear model and 0.66 for the ANN. Even though the starting position and movement velocities were not constant for the individual trials, the predictions remained highly correlated to the real trajectories. This experiment showed that it is possible to decoded motorneuron data and translate it into a control signal that can be used to steer robotic devices. An extended and more complex variant of this experiment was conducted by [Carmena et al., 2003]. Two macaque monkeys were trained to perform a 3D movement and grasping task. The goal of the first task was to move a computer cursor to a given position in a 3D virtual environment. The second task consisted in not only moving to a specific location but also grasping a virtual disc located at a random position in the virtual environment. During the first phase of the experiment the monkeys performed motor actions using a pole which controlled the location of a computer cursor in a virtual 3D environment. The cursor served as visual feedback for their actions. Grasping was performed by measuring grip force of the pole. During the training phase the position of the cursor was controlled by the position of the pole. After a few training trials the control signal for the cursor was switched to the brain signals of the monkey. While in this

brain-controlled mode, the monkeys started to perform the same motor actions as learned in the training session. After a few trials however they stopped performing an actual movement in some trials and controlled the cursor merely by thought. In response to this outcome, the researchers removed the pole object completely and continued the experiment. The electromyographic recording of muscle activity (EMG) acquired at the wrist flexors/extensors and biceps did not reveal any sort of muscle activity, yet both monkeys were able to control the cursor by thought only without performing any motor actions.

Closed-Loop BMI

In the context of closed-loop control with BMI, Kevin Warwick has drawn much attention in the media. In 2002, Warwick *et al.* implanted a micro electrode array into his left arm which was connected to the median nerve [Warwick et al., 2005]. The array consisted of 100 electrodes with a $4\mu m$ diameter at the tip. Wire bundle and connector were attached externally on the surface of the arm. The signals transmitted through the wires were used to control a hand prosthesis by means of voluntary opening and involuntary closing (VOIC). This means that only opening of the hand could be controlled by Warwick while closing was accomplished in an autonomous way. Pressure sensors in the prosthesis measured the grip force to prevent slipping and fed back tactile sensory information to the nerves and hence closing the loop. Control of the hand required a bit of training until Warwick was able to use it as intended. Warwick himself stated that coupled with the *on-board intelligence* of the prosthetic hand, it is possible to decode the neural activity into distinct control commands. The coupling of the on-board intelligence and brain controlled commands is a topic that has reemerged recently under the name *shared control* (i.e. [Millán et al., 2009]).

3.2 Non-Invasive Methods

In contrast to invasive BMI, non-invasive BMI, also called Brain-Computer Interfaces (BCI), do not require penetration of the subject's skull. This is usually accompanied with a significant loss of measurement accuracy, either becoming evident in a loss of maximum observable frequency, spatial resolution or both. Generally, the least invasive methods also entail inferior signal quality (c.f. figure 2.4). By far the most widely used non-invasive method to measure brain activity in the context of BCI is *Electroencephalography (EEG)* due to its cost effectiveness, ease of use and its high temporal resolution. Several other approaches exist that use Magnetoencephalography (MEG), functional Magnetic Resonance Imaging (fMRI) or Near Infrared Spectroscopy (NIRS) to measure brain activity suitable for BCI. However, in the case of MEG and fMRI, these methods lack the ability to be applied in every-day settings like living rooms or being mounted on mobile platforms like wheelchairs due to their tremendeous size and high technical implications. Further, the subjects have to remain stationary either lying or sitting which limits the flexibility for possible applications. In the case of NIRS, which does not measure bioelectric activity but metabolic activity, communication

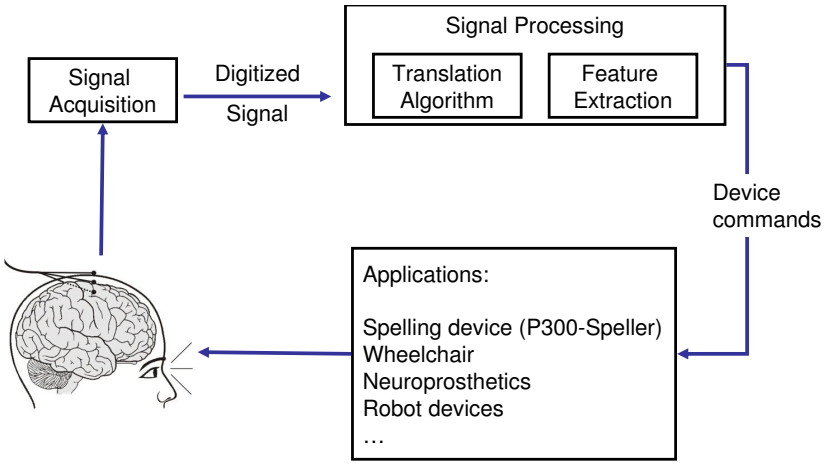


Figure 3.2. Schematic model of a general BCI. Brain signals of the user are digitized with an EEG and in a subsequent step, the intent of the user is extracted using signal processing and machine learning techniques. The output of the signal processing stage serves as commands for an application like spelling devices or other assistive devices like wheelchairs or robots. The action of the output devices can be perceived by the subject which allows him to react accordingly with different intentions.

speed is drastically reduced due to the time constraints of neurovascular coupling. The former reasons might explain the popularity of EEG which does only lack the ability to precisely locate the bioelectric generators involved in the measured EEG potentials. Therefore this section mainly focuses on EEG-based BCI and will give an overview of the existing approaches and their applications.

3.2.1 Design of non-invasive BCIs

The design of EEG-based BCI, as depicted in figure 3.2, consists of a closed-loop interaction between the device and the user's brain. According to [McFarland and Wolpaw, 2008], every BCI contains a signal acquisition module to measure task related brain activity. At the lowest level, these signals carry information which is coded in neuronal spike frequency. In EEG-based BCI this type of feature is not accessible since only a limited type and number of neuronal assemblies penetrates the skull (c.f. section 2.3.1). As a result, the amount information diminishes as it travels through the skull until it is picked up by the EEG electrodes. Yet, there is still enough information contained in the mixed and convoluted bioelectric EEG potentials to infer a meaning if the user is bound to a specific task. This means that the EEG in itself is only useful for a BCI when it is coupled with a specially crafted interface paradigm. In this case, a translation algorithm can decide what action to take since this algorithm has knowledge about the task and the brain signals. It is the type of paradigm which constrains what type of EEG feature is suitable for the BCI. The most common paradigms consist of *steady-state visual evoked potentials (SSVEP), slow cortical potentials (SCP), men-*

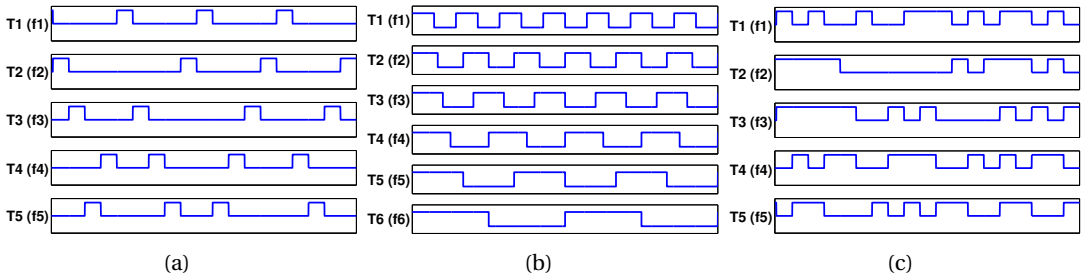


Figure 3.3. Stimulation sequence schemes for a VEP based BCI. Characteristic sequences arise from the different schemes (a) t-VEP, (b) f-VEP, (c) c-VEP.

tal tasks, event-related (de-)synchronization (ERD/ERS) and *P300 speller*. Details of these paradigms will be outlined in the following sections.

One of the main goals in BCI research is to improve communication speed to a level at which the usage of e.g. a brain controlled spelling device feels as natural as typing it on a keyboard. Depending on the task, some paradigms are better suited than others. To measure how well a BCI performs for a given task requires a common performance metric. This however turns out to be a non-trivial task due to the different approaches that the paradigms employ. Many metrics have been suggested, but none of them can be considered as a paradigm independent measure of performance. Usually, the performance of a BCI is expressed in terms of communication speed which can be measured in *bits/sec*. Another popular measure for BCI's which focus on discrete target selections rather than continuous control is *correct selections/min*.

3.2.2 Visual evoked potentials (VEP)

Among EEG based BCIs, visual evoked potentials enjoy a widespread use in the recent decades (e.g. [Prueckl and Guger, 2009, Guo et al., 2008, Lee et al., 2007, Middendorf et al., 2000a]). Visual evoked potentials are generated in response to stimulation of the subject's visual receptive field and can be measured over the primary visual cortical area. The VEPs reflect visual information processing mechanisms of the brain. While stimulation of the central visual field results in large VEPs, a stimulation of the peripheral field leads to smaller potentials. Paradigms based on this EEG feature can identify target stimuli that the user is visually fixating. This method requires that each target is coded by a unique stimulation sequence. Prediction of a target is achieved by analyzing the properties of the VEP in response to the known stimulation sequences. Constructing an appropriate sequence modulation method is a key issue in VEP based BCIs. Depending on the sequence modulation method, VEP based BCIs can be assigned to one of three different categories: *time modulated VEPs (t-VEP)*, *frequency modulated VEPs (f-VEP)* and *pseudorandom code modulated VEPs (c-VEP)*.

Time modulated VEPs (t-VEP) Paradigms using t-VEPs require the stimulus activations

to be mutually independent. This constraint is implicitly achieved when either the flash sequences of the selection targets are not overlapping in time [Guo et al., 2008] or the duration of a stimulus flash is randomized [Lee et al., 2007]. In response to a short stimulus flash, a *flash visual evoked potential (FVEP)* can be observed over the occipital area of the skull. These FVEPs which are characterized by a short latency and duration. To achieve a stable prediction, multiple EEG time windows for each target flash can be averaged due to the FVEP being time- and phase locked to the stimulus. The averaging will cancel out background activity while pertaining the components occurring after each flash. Since peripheral stimulation of the visual field results in smaller VEPs compared to those evoked by central field stimulation, the target which produces the largest mean VEP (measured as *valley-to-peak amplitude*) will correspond to the fixated target. However, many repetitions are required for a reliable target identification. Further, the claim for mutually independent stimulus flashes limits the maximum frequency at which stimuli can be flashed (usually <4Hz). This is the main reason why t-VEP based BCIs are only achieving low communication speeds compared to other state-of-the art approaches.

Frequency modulated VEPs (f-VEP) In a frequency modulated VEP approach, each target symbol is flashing at a different frequency. This type of stimulation scheme produces overlapping visual evoked potentials which can be uniquely identified by their frequency spectra which corresponds to the fundamental frequency of the attended flickering stimulus. These flashes usually occur at stimulation rates $> 6Hz$ and produce a periodic elicitation of VEPs which are also called *steady-state visual evoked potentials (SSVEP)* in this context. Such a stimulation sequence is shown in figure 3.3(b). In more detail, the steps required to predict the fixated target symbol consist of

1. Analyze the power spectral densities (PSD) of a short timewindow of EEG data using (short-time) fourier transformations or similar spectral analysis methods like wavelets or PSD estimation using Welch's method [Welch, 1967].
2. Compute the *signal-to-noise ratio* $SNR_{f_i} = \frac{P(f_i)}{\overline{X}}$ for each known stimulus frequency which is the relation of spectral power $P(f_i)$ for a given frequency f_i to the mean power of all other frequencies $\overline{X}$.
3. Predict the target which corresponds to the frequency f_i that produced the highest SNR_{f_i}.

Many studies (e.g. [Muller-Putz and Pfurtscheller, 2008, Bin et al., 2009b, Pfurtscheller et al., 2010]) demonstrated the robustness and high communication rates of SSVEP based BCIs. No specific training of the subject is required to successfully use this type of BCI which can be considered a real advantage over BCIs using *slow cortical potentials* or *motor imagery*. Even though for the latter, the *Berlin Brain-Computer Interface Group* demonstrated how motor imagery tasks can be robustly classified without extensive subject training. However, SSVEP based paradigms do not burden the subject

with additional cognitive load which can be problematic for certain persons using SCP or motor imagery BCIs [Leeb et al., 2004].

Pseudorandom code modulated VEPs (c-VEP) The target decoding of c-VEP based paradigms becomes slightly more complicated. In this paradigm, pseudorandom sequences are used which is shown in figure 3.3(c). The coding scheme is highly related to the *Code Division Multiple Access (CDMA)* method which is widely used in mobile communication. Thereby, a pseudorandom sequence, called *m-sequence*, is generated which has special properties that help to decode a focused target among other seemingly random flashing targets. One of the m-sequence's properties consists of its autocorrelation function which closely approximates the unit impulse function. The orthogonality to its time lag sequence is a second important property. Usually, template matching methods are used to predict the target symbol. The template is computed during an initial training session. According to [Bin et al., 2009a] the following steps are commonly used to implement a c-VEP based BCI:

1. During an initial training phase the user is required to attend to a specific target k_0 for N stimulation runs while EEG data is being collected.
2. A template $T(t)$ is computed based on the previously recorded EEG data which has to be averaged over N trials.
3. To compute the templates for all k targets, $T(t)$ needs to be shifted by the time lag difference between target k and k_0: $T_k(t) = T(t - (\tau_k - \tau_{k_0}))$.
4. For each EEG timewindow, correlation coefficients C_k are calculated between the previously computed templates T_k and the current EEG segment x.
5. The prediction simply consists of selecting the target k for which C_k is maximal.

A c-VEP based BCI developed by [Sutter, 1992] reached very high communication rates of >100 bits per minute which corresponds to approximately 10-12 words per minute. Despite this seemingly promising result, more recent publications could not reproduce the same performance in terms of communication speed. As an example, the BCI developed by Momose [Momose, 2007] took 5 seconds to predict one out of five targets leading to communication rates of less than 20 bits/min.

3.2.3 Slow cortical potentials (SCP)

Using Slow Cortical Potentials (SCP), as described in section 2.4.2, for BCI has the advantage that the expression of the expression of detectable features can be voluntarily controlled by the user which was shown by [Elbert et al., 1980, Birbaumer et al., 1990]. One of the most representative BCIs of this type was developed by the group around Birbaumer who was also the first to demonstrate how operant conditioning can be used to gain voluntary control over slow cortical potentials [Birbaumer et al., 1990]. The study reports that a successful application of this technique does not require continuous feedback but reward at the end of the

Figure 3.4. A BCI based on *slow cortical potentials* operated by a patient suffering from *amyotrophic lateral sclerosis*.

selection. Their system, commonly referred to as *Thought Translation Device (TTD)*, employs SCP features to control an object on a computer screen with one degree of freedom. Contrary to many other BCIs, this system has been extensively tested with patients suffering from late-stage ALS and proved to be able to restore basic communication capabilities of the patients [Kübler, 2000, Kübler et al., 2001]. The associated paradigm for SCPs can be described as a *basket paradigm* in which the user is asked to move a falling ball into one of the target baskets which are visualized as bars at the bottom of the screen as depicted in figure 3.4 (B). A selection round is finished once the ball hits one bar. The speed of the ball limits the maximum communication speed of the system. Prior to each selection the user is asked to relax for 2 seconds while during the following 2 seconds the SCPs have to be self-regulated as desired. The control signal is derived by comparing the SCP voltage level against the baseline period which can be either higher or lower and thus allows for unidimensional control, i.e. left and right movements. Slow cortical potentials have been successfully applied to spelling tasks and control of a web browser [Mellinger et al., 2003, Hinterberger et al., 2001]. To gain control over a web browser, the source code of the application was modified to display colored links as selection targets grouped into stacks of two (see Figure 3.5). The user focuses on regulating her SCP according to the desired selection. The program will keep splitting the links contained in the selection boxes until a unique selection between two choices is possible. In conclusion, the basic methology for using SCPs to control a web browser closely resembles the *basket paradigm* used in their speller application.

Figure 3.5. In the study of [Mellinger et al., 2003] a modified web browser was used to display selecting groups containing links to other pages. By self-regulating SCPs, the selection could be narrowed down to one unique target.

3.2.4 Event-related (de-)synchronization (ERD/ERS)

Brain rhythms related to motor actions or motor action preparations can exhibit either an amplitude decrease, which is called event-related desynchronization (ERD), or an amplitude increase called event-related synchronization (ERS) [Pfurtscheller and Lopes da Silva, 1999]. A BCI based on this feature will exploit the properties of ERD/ERS in either one or multiple frequency bands to allow for a recognition of the 2 distinct states. The associated paradigm can either be cue based (synchronous) or self-paced[1] (asynchronous), i.e. the classification occurs only during fixed timewindows while with self-paced control data processing and classification is continuous. To self-regulate the expression of ERDs, motor imagery tasks are employed. During an initial training session the user repeatedly imagines a movement of a body part like hand, foot, leg or tongue. The computational part consists of finding appropriate electrode locations and distinctive frequency bands which separate the two brain states. The training continues with a visual feedback of the user's sensorimotor rhythm changes with the aim to improve voluntary control over the two states. Thus, the training usually involves training a classifier to discriminate between both states and training the subject to self-regulate these states. A well trained subject is able to reach accuracies in the order of 90% [Scherer, 2008]. To ensure good classification performance, especially the selection of electrode locations is crucial. A standard method to select good combinations of electrodes is the *common spatial patterns (CSP)* method [Müller-Gerking et al., 1999]. Generally CSP extracts event-related desynchronization effects in the power spectrum of an EEG signal, e.g. the frequency attenuation in the μ and β bands. This method was extended by [Dornhege et al., 2003] to extract movement related features in the data which was further extended by [Dornhege et al., 2004] to solve multi-class movement related feature tasks. An asynchronous paradigm involving a visual-rich virtual street (see figure 3.6) was presented by [Leeb et al., 2007]. This paradigm requires the user to control a virtual wheelchair and stop at predefined positions marked by computer avatars. At each stop, which was required to last at least 1 second, the avatar started a short communication with the user. Once the communication ended, the user was allowed to move on to the next avatar. In their experiment, the brain states were mapped to *move forward* and *stop* commands. The experiment was conducted with a male tetraplegic subject who suffered a traumatic injury of the spinal chord with lesions below C4 and C5 (c.f. 2.3.2). In this paradigm, no cue was shown that indicated relaxation and motor imagery (MI) phases but was completely up to the subject when to issue a command and when not. After completing the experiment, the results showed that he was able to stop at 90% of the 150 avatars. These results proved that an application to a real wheelchair in a real environment could be possible since the rather large amount of stimuli in the virtual environment did not seem to influence the classifciation capabilities of the system. Even though this result is very impressive, SCP based BCIs are rather difficult

[1] A more stringent definition of the term *self-paced* also entails that no external stimulation should be required to communicate with the BCI. While the term *asynchronous* usually denotes the absence of cues but allows external stimulation.

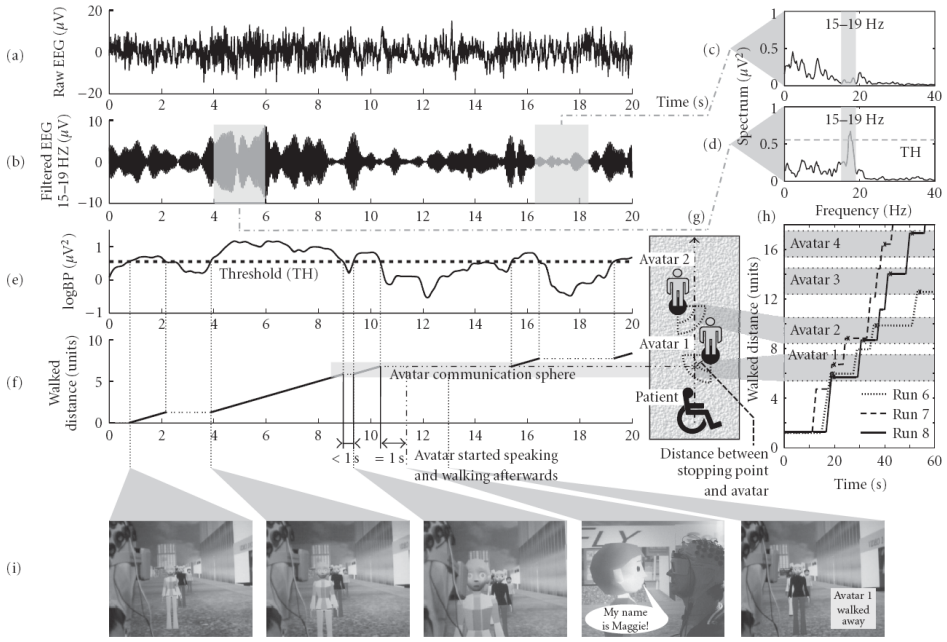


Figure 3.6. (a) Raw EEG data during foot MI and rest. (b) Filtered signal. (c),(d) power spectra of *rest* and *foot* MI. (e) Timecourse of bandpower. (f) Plot of covered distance during movement phases. Missing line segments indicate stops. (g) Placement of the avatars. (h) Time-Distance plot for the first 3 avatars. (i) Image sequence of the experiment.

to operate for a naive user due to the dual task interference. Self-regulation of SCPs as well as concentrating on the task of selecting appropriate targets entails increased cognitive load. With sufficient training however, self-regulation can be automated up to a level at which it does not interfere with the actual selection task anymore as demonstrated by [Neumann and Birbaumer, 2003]. Still, compared to other approaches which utilize subconciously evoked features in the EEG, they cannot be considered as the most natural way of communication since natural communication is effortless and can take place without distraction from a simple concurrent task like spelling a word or visually inspecting the environment. Escpecially SCPs communication rates are highly inferior to any other approach and thus do not seem appropriate for effortless communication.

3.2.5 Mental tasks

Another category of BCIs are related to features emerging while engaged in specific mental tasks. This type of BCI is somehow related to those utilizing movement related features to derive a control signal because they also rely heavily on mu rhythm changes. The difference is that they rely on very specific mental strategies while mu-based BCIs on the other hand do not require an explicit strategy. This technique relies to a great extent on the different spatial locations of brain areas which are activated during the different tasks. It is known that e.g. language processing results in higher activation at left temporal sites while spatial operations like rotating a cube result in right hemispheric activations. The first to present such a system were [Keirn and Aunon, 1990]. They instructed the subjects to engage in one out of five different mental tasks for 10 seconds. This step was repeated five times. As tasks the subjects could choose between relaxation, multiplication, rotating a 3D figure, composing a letter to a friend and visualizing numbers written on a blackboard. In a subsequent step, the power spectrum of the data were analyzed offline and served as input for a Bayes quadratic classifier. With this method they were able to predict at least 70% of the states correctly. When limiting the number of tasks to only two choices, a 90% accuracy could be achieved. Though, translated into *information transfer rate (ITR)* this means that a user operating such a system would only be able to transmit 1-6 bits/min. Compared to other approaches which are ranging up to 90 bits/min [Gao et al., 2003], this method does not seem feasible for most subjects. As a clinical application though it might still be useful if a user fails to gain voluntary control over SCPs or is unable to imagine motor actions.

3.2.6 P300 - Detecting expectation

A P300 is an endogenous event-related potential which is elicited in response to an infrequent or task-relevant stimulus interspersed into a sequence of frequently occurring background stimuli. Thus, all P300 based BCIs rely on the *oddball paradigm* in which target stimuli are infrequently presented during a sequence of background stimuli. The first to implement a P300 online BCI were Farwell and Donchin [Farwell and Donchin, 1988]. Since their system

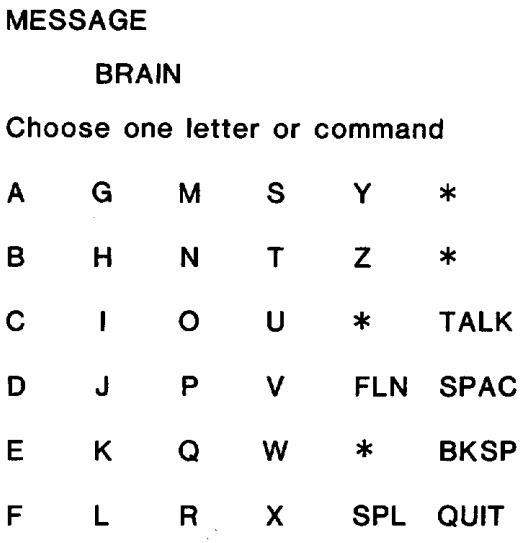


Figure 3.7. Original speller matrix from the experiments of [Farwell and Donchin, 1988].

was intended to be a brain actuated spelling device, the stimuli consisted of the letters of the alphabet and additional symbols for special functions like quitting and deleting. The symbols were arranged in a rectangular 6×6 grid which is commonly known as the *speller matrix* (Figure 3.7). To elicit a P300 in such a speller paradigm it is required that the symbols are flashed in random order. The subject's task is to focus on a target letter until a prediction has been made by the BCI. Stimulus presentation can be conducted in two different ways:

1. Multiple stimuli
2. Single stimulus

Farwell and Donchin employed the multiple stimuli method in their experiments. During the stimulus presentation round, each row and column is highlighted in random order for a brief period until each row and column has been highlighted once. With each flash a short epoch of EEG data is assessed. Usually the full evoked potential can be captured within an 700-800ms interval which determines the length of the EEG epoch. The data is used as input for a classification method that aims to predict whether the epoch contains a P300 or not. An obvious benefit of the multiple stimulus method is that in the case of 36 choices (i.e. symbols in the matrix), once all 12 available stimuli have been flashed, it is theoretically possible to infer the symbol that was focused by the subject by selecting the row and column which elicited a P300. The intersection of this row-column pair will correspond to the target letter. Practically however, this is rarely possible due to the low signal-to-noise ratio (SNR) of EEG. A P300 which was actually present at a certain flash might be occluded by background noise, other brain-activity or artifacts. Thus, a subtrial (i.e. the presentation of the 12 stimuli) is repeated

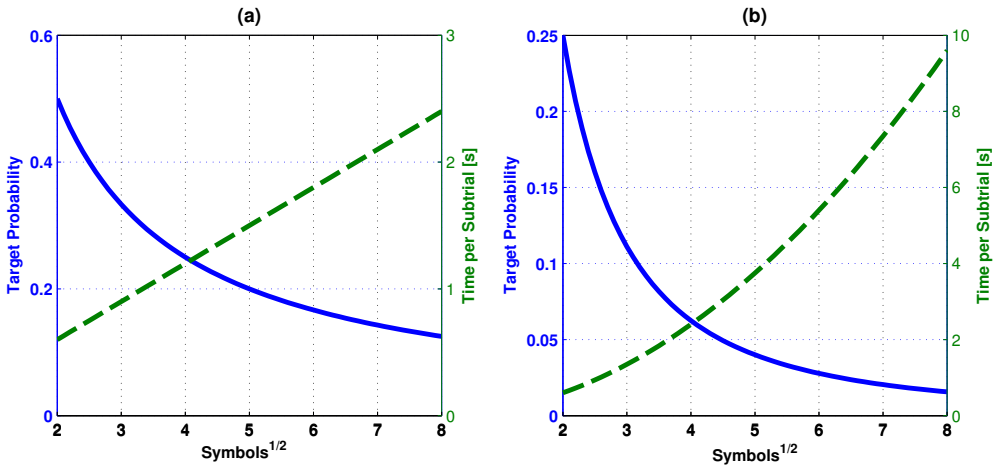


Figure 3.8. Target probability and presentation speed of (a) Matrix vs. (b) Single Symbol paradigms. The blue curves show the ratio of target to background stimuli. The green curves denote the time needed to present a full subtrial given the *inter-stimulus interval (ISI)* is fixed at 150ms. On the x-axis, the number of symbols along one dimension of the stimulus matrix is depicted.

multiple times which allows to average the EEG epochs for the corresponding symbols. The computation of mean epochs effectively increases the SNR which in turn leads to a more accurate prediction of the target letter. An inherent problem with this approach consists of the increased probability of overlapping P300 epochs. For each subtrial in a 6×6 matrix experiment there exist 2 stimuli that will elicit a P300 (i.e. one row and one column). Since the order is randomized its is possible that those two stimuli appear in succession to each other. Depending on the ISI, the brainresponses to the stimuli will be present in both epochs overlapping in time. As shown by [Martens et al., 2009], refractory effects of evoked brain dynamics exist which means that the measured signal in response to time overlapping P300 potentials will be different than those serving as a basis for the classification method. As a result the BCIs performance will suffer from this effect. This problem might not become evident for larger grid sizes since the probability of overlapping P300 epochs within a subtrial decreases with increasing grid size. However, in cases where the number of choices is rather low (i.e. 4-9 choices), flashing multiple stimuli at once might become unfeasible due to the increased probability of a target flash within the measurement interval of the first target flash. Further, with decreasing number of symbols the target probability increases which was has been shown by [Farwell and Donchin, 1988] to have a significantly attenuating effect on the P300 amplitude. The decreased amplitude coupled with refractory effects resulting from overlapping events in the same timewindow can lead to a substantial loss of performance. Therefore flashing on a single stimulus basis might be more beneficial in the case of few symbols. A downside of the single symbol approach stems from the increased amount of time needed to present a full subtrial. Figure 3.8 depicts the issue of speed vs. target probability. The ulti-

mate goal would consist of a paradigm satisfying both *low target probability* and *fast subtrial presentation*. The figure however shows that these are contradicting claims since increasing presentation speed, as accomplished with the matrix paradigm, inevitably results in higher target probabilities due to the fact that the number of stimuli $S = 2 * \sqrt{N_{Symbols}}$ and thus $\left[P(MatrixTarget) = 2/S\right] > \left[P(SingleTarget) = 1/N_{Symbols}\right]$. Generally, for more than 9 choices (i.e. 3×3 matrices) the speed advantage of the matrix paradigm will outweigh the benefit of low target probabilities of the single symbol paradigm since a target probability of 25% does not entail significantly attentuated P300 amplitudes which renders target flash overlaps less of a problem. With even higher number of choices the disadvanteges of the matrix paradigm will almost completely vanish.

3.3 BCI applications in virtual and robotic control

Modern BCIs lost the reputation of being only a tool for the completely paralyzed. Current development strives to make BCIs applicable for healthy persons in the context of entertainment, gaming or assistive technology. Depending on the task to perform with the BCI, a suitable paradigm has to be selected. The main goal is to make the usage of the system as intuitive and reliable as possible for the user. This section will give a non exhaustive overview of BCI applications which employ the modified versions of the paradigms mentioned in the previous sections with the aim to control devices or virtual objects which are usually situated in real-world scenes.

3.3.1 Virtual reality and gaming

In her doctoral thesis, Jessica Bayliss investigated how P300 based BCIs perform in noisy environments with more realistic stimuli and task parameters. The first study consisted of a *virtual stoplight* experiment in which the subjects sat in a modified go-cart as seen in figure 3.9. All subjects were equipped with a *head-mounted display (HMD)* that displayed a virtual town in which the subjects were navigating. The task was to pay attention to the traffic light which could be either red, yellow or green. On every red light, the subjects were required to stop. Bayliss collected EEG data of 100ms prior to the stimulus until 1s after stimulus onset from electrode locations Fz, Pz, CPz, Pz, P3 and P4 with linked references from the left and right mastoids. The ERP grand averages in figure 3.10 show clearly that a P300 response occurs at red lights while no P300 gets evoked by yellow lights. Green lights however, indicating that the user is free to move on, also evoke a P300 with lower amplitude than the red lights. This effect might be explained due to the task relevance, which is a precondition for the evocation of a P300, of the green light. Based on the VR paradigm, Bayliss investigated the effectiveness of three different signal processing methods. Single trial data was processed with either *Independent Component Analysis (ICA)*, *Kalman filter* or a *robust Kalman filter* and subsequently classified using a correlation method which can be considered as a form of template matching. The offline evaluation of the processing methods revealed that robust

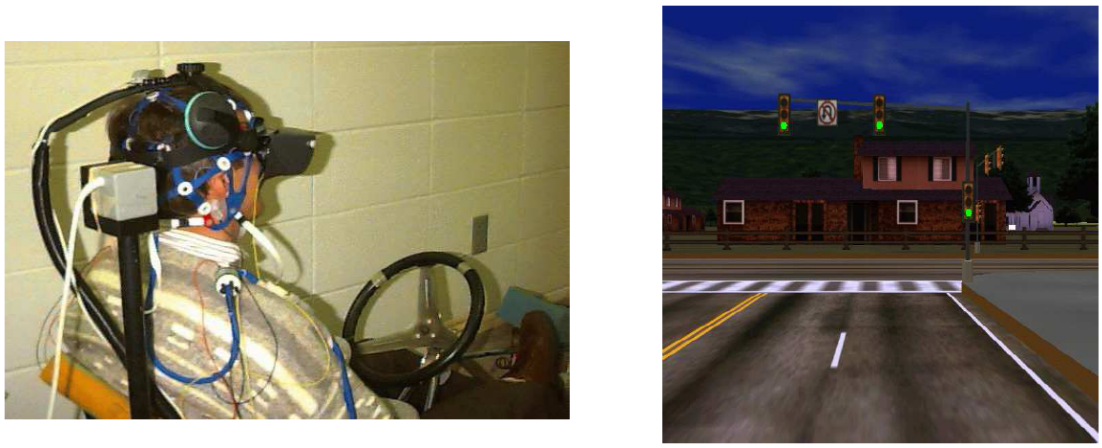


Figure 3.9. The *virtual stoplight experiment* by Bayliss. Subjects were sat in a modified go-cart (left) and were required to attend to red light to stop driving in a virtual street (right).

Kalman filtering with classification based on template correlation was best suited with correct classification of yellow and red lights ranging from 76-92% (see table 3.1). Two of the five subjects also participated in an online experiment. Accuracies dropped to 80% for the first and 85% for the second subject. A second study was conducted with the aim to immerse the subjects into a virtual apartment and perform tasks like switching the lights on and off or controlling the TV [Bayliss, 2003]. During the experiment, a text at the bottom of the screen indicated the required task to perform. If the subject was asked to switch off the light, a semi-transparent red blob appeared over the light switch. The blob was flashing at 1Hz and the subject was required to silently count how many times the blob appeared. The same electrode configuration as in the stoplight study was used. One of the main questions in this study was the question whether there are any significant differences in the evoked potentials among the following three conditions:

1. VR condition: The scene was viewed through a HMD and head movements were mapped to the scene.

Subjects	**Correlation % correct**			**ICA % correct**			**Kalmann / robust Kalman % correct**		
	Red	Yel	Total	Red	Yel	Total	Red	Yel	Total
S1	81	51	64	76	77	77	54/55	85/86	76/77
S2	95	63	73	86	88	87	82/82	96/94	92/90
S3	89	56	66	72	87	82	61/74	85/85	77/81
S4	81	60	67	73	69	71	63/65	90/91	81/82
S5	63	66	65	65	79	74	78/78	92/92	87/87

Table 3.1. Results for the offline analysis of the virtual stoplight experiment.

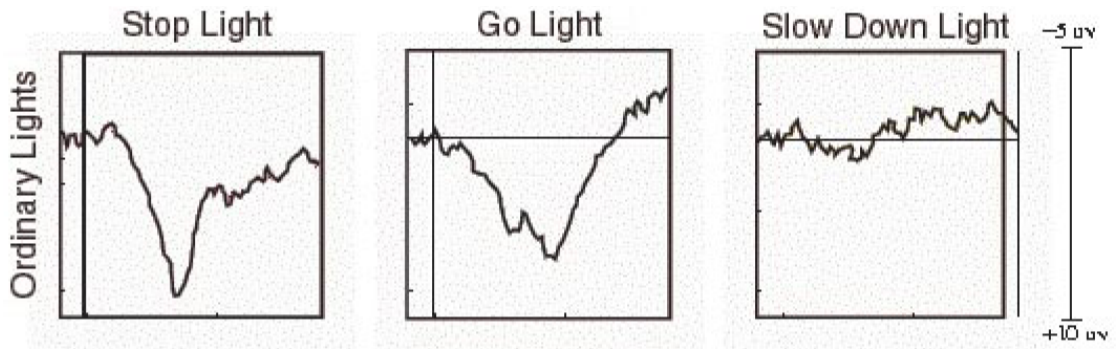


Figure 3.10. Grand average ERPs for red (left), green (middle) and yellow (right) traffic lights.

2. Monitor condition: The same scene was presented at a standard computer monitor without the use of a HMD.

3. Fixed Monitor condition: A virtual monitor was presented through an HMD. This condition is identical with the second one except that the computer screen was a virtual one viewed through the HMD.

Further, differences in classification performance, differences in performance over time as well as the qualitative experience reported by the subjects were analyzed. It turned out that no significant differences in P300 evocation could be found among the three conditions which further testified the robustness of the P300. Slight classification performance differences could be measured between the *fixed monitor* and *monitor* condition. Subjects were overall performing better in the monitor condition than in any other one. While the study confirms minor differences of classification performance over time, none of them was significant enough to confirm a general trend.

With the intention of improving BCI-feedback presentations to obtain a better control over the BCI, Robert Leeb and colleagues also explored the use of Virtual Reality environments. In contrast to the study of Bayliss who used the P300 ERP to toggle discrete actions, Leeb employed a motor imagery paradigm to navigate in a virtual conference hall. Thereby, subjects were imagining left and right hand movements which were mapped to left and right rotations of the avatar which was viewed from a first person perspective. The scene was displayed through a stereoscopic virtual reality helmet coupled with a head tracking device. A cue-based paradigm was chosen where at the beginning of each trial a short beep and a visual stimulus was presented which indicated that the subject should start imagining a movement. During the training phase, the visual cue also indicated what type of movement, either left or right, should be imagined. This information was omitted in the subsequent feedback sessions. The classification phase took at least 2 seconds and in the feedback session, the classification output was presented until 8 seconds after the initial cue. Using bandpass filtering and *linear discriminant analysis (LDA)*, subjects achieved accuracies of 77-100%. In this pre-

liminary study, Leeb showed that it is possible to combine the use of BCI and VR techniques. He stressed the fact, that VR feedback has to be as real as possible and that feedback must not behave in an unknown or unpredicted way. Otherwise it might disturb the subjects attention and eventually break immersion which distracts from the actual task. Further, no disturbing influences on the movement classification from the VR equipment could be identified which is an important factor when trying to combine the highly sensitive EEG-based BCIs with VR equipment. This finding confirms the results of Bayliss who proposed the same in the case of P300 evoked potentials.

3.3.2 Robotic control

Historically, BCI research was focused on developing systems for paralyzed people to give them the opportunity to participate in everyday life. It is not surprising that this effort lead to the development of brain-controlled wheelchairs and prosthetic devices like robot arms. Gaining control over mobile platforms or robot arm trajectories can be a tricky task to master solely with the use of brain activity. To achieve this goal, one can follow two paths. The first would be to develop a system that grants access to a limited set of degrees of freedom the device offers. In the case of a wheelchair, it is enough to control left, right and forward and stop movements. However, since the movement should be a smooth continuous trajectory an event-related approach which requires synchronous or cue based paradigms seems rather complicated. Only discrete commands can be issued which have to be mapped to the motor control in a continuous fashion. A different approach would use an asynchronous paradigm which allows the user to interact with the system at any time without the need of stimulus presentation or cues. A drawback of this approach is that it can be very difficult for the subject to control all the necessary degrees of freedom. Especially for fine grained movements as they would occur while the user wants to move through a passage that is barely wide enough for the wheelchair. In recent years, the concept of *shared control* seems to become a promising alternative. In [Vanhooydonck et al., 2003], an intelligent controller estimates the users intent and corrects the movement of a wheelchair accordingly. In the most recent developments, this concept finds its application in conjunction with both synchronous and asynchronous BCI.

Two developments relevant in the context of this thesis were made by [Iturrate et al., 2009] and [GALAN et al., 2008]. Both constructed a wheelchair that can be moved by brain signals. They differ from each other by means of EEG features and the paradigm they employ. In the case of Iturrate, P300 potentials were used in conjunction with a speller matrix like paradigm. As mentioned, the use of a discrete selection paradigm like the P300 speller matrix might seem counterintuitive for providing a continuous control signal for a wheelchair motor. However, the work of Iturrate uses an autonomic navigation system that takes care of the low level movements using obstacle avoidance and laser odometry to navigate in the environment. The user of the system is only required to select target locations which the wheelchair should move to. This offers the benefit that the user only has to focus on select-

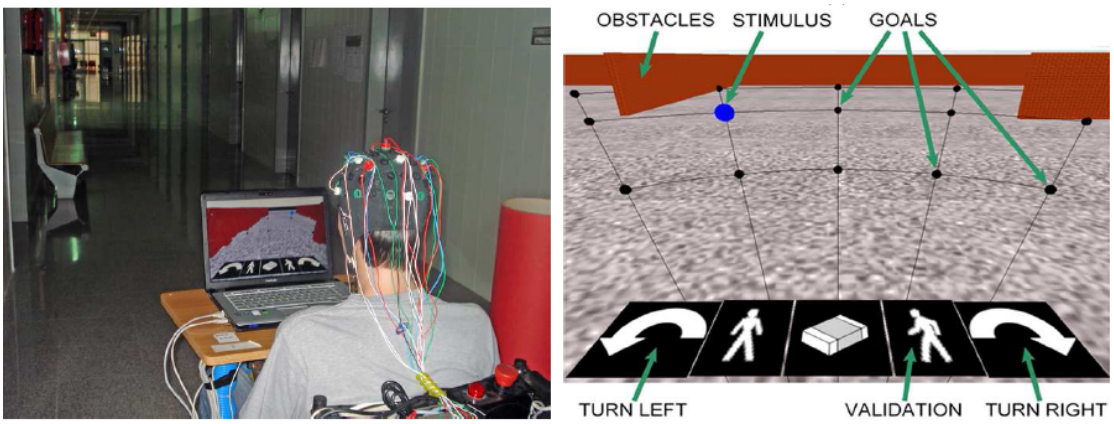


Figure 3.11. Synchronous BCI controlling a wheelchair as presented in [Iturrate et al., 2009]. A virtual-reality paradigm presents circles as stimuli which are aligned in the virtual world to correspond to spatial locations. Actual motor control of the wheelchair is done autonomously by the system.

ing a target location. Once it got recognized he can relax and let the navigation system do the actual work. Such a procedure is much less mentally exhausting than a procedure which requires to continuously give sequential commands to avoid obstacles and correct trajectories. The paradigm is very similar to the well known P300-Speller grid as seen in figure 3.11. The environment is scanned using laser sensors and gets reconstructed in a 3D virtual reality scene on a laptop computer. A stimulus grid consisting of circles connected by lines extends from the first person view into the reconstructed VR scene. Thereby the circles serve as target locations. Visual stimulation follows the P300-Speller paradigm as described in section 3.2.6. The protocol is implemented using a finite state machine. In its initial state, the wheelchair is stopped and awaiting a user command to process. Stimulation begins by flashing the target location stimuli and two additional stimuli for left and right turns. Concurrently, EEG data is recorded which serves as input for the pattern recognition layer. The EEG signals are classified using *stepwise linear discriminant analysis (SWLDA)*. Once a successful classification occurs, the wheelchair switches to the *validation state* which requires the user to confirm the previously predicted goal by selecting the *validation symbol*. The third phase consists of carrying out the path planning and motor control which is completely realized using the intelligent controller and embedded sensors. While in this last *wheelchair moving state*, stimulation is blocked. As a consequence, no further commands like stopping or correcting the goals is possible. Even though the authors report accuracies of up to 94%, possibilities for error correction and to interrupt the driving wheelchair would be benefitial as also remarked by the authors. A similar system was presented by [Rebsamen et al., 2007], though at a different level of control. Instead of selecting abstract target locations, pre-programmed locations could be selected. A context-based menu with flashing menu items served as stimulation matrix similar to the speller paradigm. In contrast to Iturrate and the standard speller paradigm, an asynchronous paradigm was used. This was implemented by flashing all items continu-

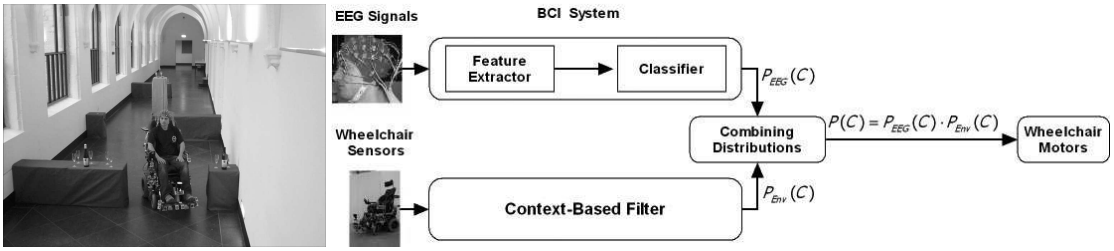


Figure 3.12. Shared-control system design. Probabilities of recognized mental commands are combined with probabilities of expected commands based on the environment. Thereby, shared-control systems act as assistive controllers to modify the user plan.

ously and omitting the use of a predefined number of subtrials. Classification scores for each symbol were accumulated and a selection was made once the score for the highest rated symbol exceeded a manually adjusted threshold value. During wheelchair movement, the BCI is still available for input which is also a strict requirement for this paradigm since the most important command during movements is the `STOP` command. However, the authors mention that for some cases the reaction time of the system, which stems from the time needed to present the stimuli, is not short enough to safely navigate in a dynamic environment. If a human is blocking the path, it is possible that the stop command would be recognized too late. Therefore they implemented a low-cost shared-control component in terms of an infrared proximity sensor. Using this data, an emergency stop command can be issued without the detection of brain potentials. An obvious shortcoming of such a paradigm is the need to create a map of the surrounding, pre-program motion paths and select an appropriate set of actions that serve as stimulation symbols.

In [GALAN et al., 2008], the same goal of controlling a mobile platform was achieved using motor imagery and an asynchronous paradigm. Asynchronous in this context refers to the ability to communicate with the system at any point in time without the need to wait for a cue which signals that the system is now ready to receive input. Their system is built around an intelligent wheelchair which performs low-level control of motors based on estimated brain-states (i.e. mental commands) and the environmental context, a technique called *context-based filtering*. This means that the user does not have full control over the low-level functionality of the wheelchair. Probabilities of imagined commands are interpreted in conjunction with the surrounding which defines the basis for a *shared-control system*. Such a design is depicted in figure 3.12. The combined information is used to control low-level functionalities of the platform. Thereby, the intelligent controller serves as an assistance unit that can modify the user's commands to safely or more efficiently execute their original intent. This can be especially useful for fine maneuvering the wheelchair in environments with many obstacles. Further, recognition of mental commands is not always perfect. If the user is facing a wall a command of moving forward is most likely the result of a wrong recognition. The intelligent controller which is aware of the fact that moving forward would result in

a crash with the wall can modify this plan and maneuver around the wall, since the original intent could be moving to a spot behind the wall. In specific, this environmental awareness is implemented using a laser range finder which continuously monitors the environment. One of three different commands (`LEFT, RIGHT, FORWARD`) is selected based on the highest probability and converted into translational and rotational velocities suitable for the motor control. Combined with the current wheelchair velocity, a smooth trajectory is generated. At this point the intelligent controller uses this motor command in addition to environmental information gathered by the laser scan to generate an assistive behavior. As a proof of concept, Galán conducted a study with 2 subjects which were required to control a simulated VR wheelchair. A predefined goal had to be reached by using the 3 commands `LEFT, RIGHT` or `FORWARD`. The path was split into zones, each of them associated with a command that would allow the user to pass through this zone correctly. Performance was defined as the percentage final goals reached during one session. Each subject was participating for 5 consecutive days with 1 session per day. As far as the limited amount of subject permits, a positive trend in final goals reached could be found over the 5 days. One of the subjects was able to reach all final goals during one session, while the best performance of the second subject did not exceed 70%. At the last day however, performance seemed to decrease for both subjects to 70% and 50% respectively.

In a recent online study [Millán et al., 2009], Millán and colleagues assessed the performance of the system in terms of a function goal radius and targets reached. This time, a real wheelchair was used in contrast to the simulated scenario of Galán. Compared to the simulation scenario, the online results turned out to be much lower since none of them exceeded the 50% mark; not even with a tolerance radius of 2m per goal. However, a simulation with random commands performed only at a 10% level while even a naive user reached almost 30% of the goals. A subject interview revealed that driving the real wheelchair is much harder to control than the simulated one due to the more complex behaviour. Further, the assistance obtained from the shared-control module was not yet perfect for the task as remarked by the authors.

The last relevant field of application in robotics consists of controlling reaching and grasping movements of robotic limbs. These limbs can either be prosthetic devices which are attached directly to the human body (e.g. [Guger et al., 1999]) or external robotic arms that can be "thought controlled". An approach comparable to the one developed in this thesis was presented by [Palankar et al., 2009]. In their paper they describe how they control a 7-DOF robot arm connected to a robotic gripper with the use of a P300 speller-like paradigm. The robot hardware has been attached to a modified wheelchair which could be controlled via the same mechanism as the arm, resulting in a total of 9 DOF. Their stimulus matrix (figure 3.13) contains symbols for positioning the end-effector in the XYZ plane as well as rotation commands to orient the end-effector. A selection of a command results in the desired movement with a predefined velocity (i.e. rotation about the x-axis with 50mm/sec). The authors evaluated the recognition performance of the BCI in a simulation study which revealed that the participants achieved mean accuracies of 70-85%. To improve reaction time of the sys-

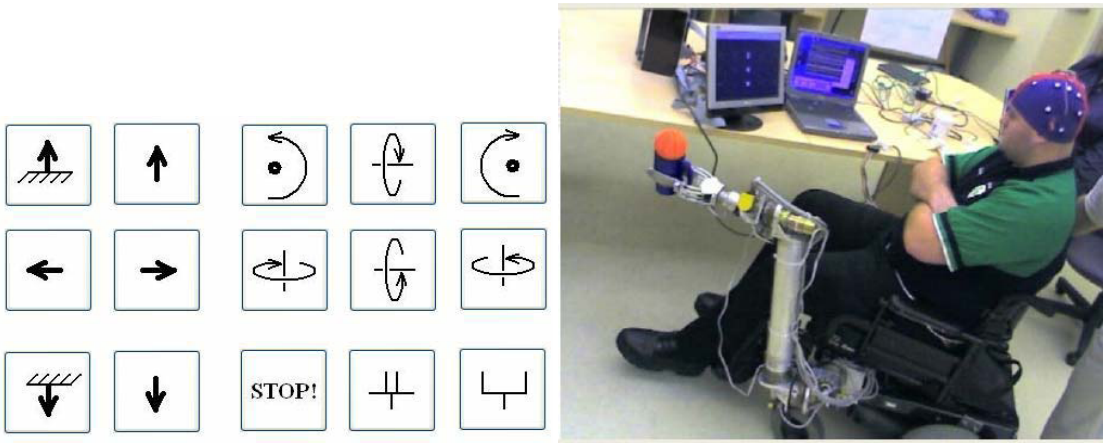


Figure 3.13. Low-level control of a robot arm and gripper with P300 evoked potentials [Palankar et al., 2009]. Translation and rotation are controlled via symbol selection according to the P300-Speller paradigm.

tem, the optimal number of subtrials[1] was determined by evaluating the mean performance of all subjects for different numbers of subtrials. Depending on the desired number of correctly predicted symbols per minute and the desired accuracy, the number of subtrials was set to the appropriate value. Selecting a command of the symbol matrix was equal to switching to a different movement state. While in e.g. *X rotation state*, the gripper might move at 50mm/sec. To bring it into a desired end position, the state must be switch to the *STOP* state in time as it might overshoot if the selection does occur at the right moment. The authors note that it takes 15 seconds to select a symbol which is far too long to be an acceptable reaction time. Approaching stairs in this wheelchair will become a dangerous issue. Further, Palankar reports that the subjects suffered from fatigue after extended periods of use since the user is required to look at the screen the whole time while also paying attention to the robot movement. Command and robot movement have to be precisely timed, otherwise the intended end position will not be reached. This process introduces heavy cognitive load which demands a lot of concentration from the user. A state that can not be maintained for a long time. This dilemma was briefly discussed in the beginning of this section. The work of Palankar makes it clear that low-level control can be a daunting task to achieve with a binary selection based system that has an inherently slow reaction time.

[1] The term subtrial is used in this thesis while [Palankar et al., 2009] use the term *flash sequence* to denote a full round of symbol intensifications.

4 Study 1 - Dynamic Subtrials

This chapter has been published in *IEEE Transactions on Neural Systems and Rehabilitation Engineering* [Lenhardt et al., 2008].

4.1 Introduction

The use of the P300 component of an evoked potential in current brain-computer interfaces (BCIs) is a feasible method to translate the user's intent into commands to control artificial devices. Farwell and Donchin [Farwell and Donchin, 1988] demonstrated how the P300 can be used to spell words by "mere thought". Steady research in this field improved information transfer rates as well as accuracies, which is a crucial issue in terms of usability. In addition to the P300 component in EEG data several other types of signals can be used to translate the user's intent into commands. Examples for the most common types are *slow cortical potentials* [Birbaumer et al., 2003], *mu and beta rhythms* [Wolpaw et al., 2003b], *event related (de)synchronization* [Pfurtscheller et al., 2006] and *steady-state visual evoked potentials* [Middendorf et al., 2000b]. A more detailed discussion of the different types of BCIs can be found in [Wolpaw et al., 2002]. A common way to compare different approaches is to express the performance of a BCI in terms of speed (i.e. bits/min), accuracy and the time needed to train the subject. Therefore, we present an improved P300-based brain-computer interface that allows for simple tuning to be either accurate or fast at detecting letters, and that outperforms similar approaches as presented in [Serby et al., 2005] at comparable accuracies. As already demonstrated by [Kaper, 2006, Kaper and Ritter, 2004], *linear discriminant analysis (LDA)* was used for classification due to its good classification performance and low computational and training requirements compared to more complex classifiers like SVM. Since the dimensionality of one epoch of EEG data was too large to be used with LDA, *principal component analysis (PCA)* was applied to reduce dimensionality. Considering information transfer rates, very competitive results of up to 92.22 bits/min in the first study and 84.6 bits/min in the second study were achieved during online classification, whereas the second study employed a threshold based algorithm [Lenhardt, 2006] which dynamically limits the number of subtrial presentations during a trial. A similar approach was introduced by [Serby et al., 2005] which also adapts to the subjects online performance by limiting the number of subtrial presentations. They demonstrated the usefulness of such techniques and therefore we aim to

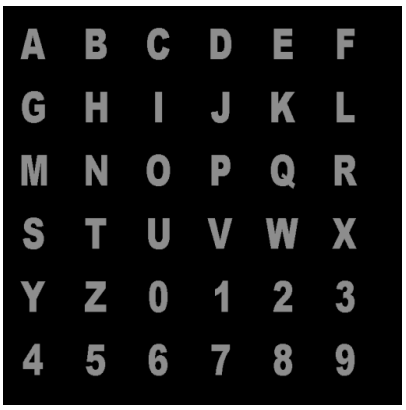


Figure 4.1. Screenshot of the P300-Speller matrix

strengthen those findings by introducing a similar approach and extending the number of involved subjects as well as the amount of data collected per subject.

4.2 Methods

4.2.1 P300 Speller Paradigm

Our BCI is based on a modified *Oddball Paradigm* [Sutton et al., 1965] called *P300-Speller Paradigm*. The experimental setup consists of a computer CRT screen that displays a 6×6 matrix consisting of the letters A-Z and the digits 0-9 (see figure 4.1). According to [Farwell and Donchin, 1988], the term *event* denotes the intensification of either a row or a column of the matrix. The computer controls the presentation of the events, while the basic idea is to intensify each row and column in random order and assess a short time window of EEG data for each event. As the subject is focusing attention to a specific letter, the column and row containing this letter become relevant events. This means, that out of 12 possible events (6 rows and 6 columns) only two of them are relevant. Due to the relative rarity of their occurrence they will elicit a P300 whenever a row or column which contains the cell is highlighted. If the inter stimulus interval (ISI) is large enough along with the low presentation probability of the target events, the P300 is well discriminable from other components, occurring for each background event. Therefore, to infer a letter, the amplitude of the P300 following each intensification is assessed, and the inferred letter is determined by the intersection of the row and column which elicited the highest P300 amplitude. Since EEG typically exhibits a very low signal-to-noise ratio (SNR), the experiment is structured in a special way. Figure 4.2 illustrates the structure of a single subtrial. A subtrial is comprised of 12 events which are displayed in random order. The onset of each event triggers a new EEG time series, called epoch. In our experiments, an ISI of 140ms was used. By averaging over a certain number of subtrials the SNR can be improved.

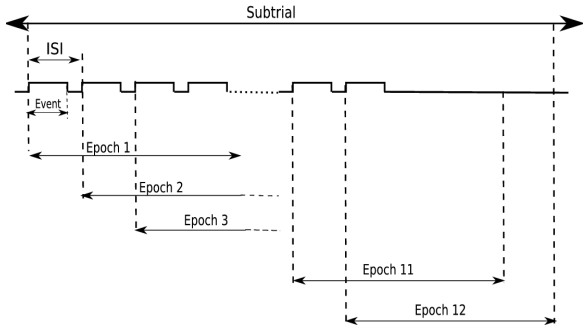


Figure 4.2. Epoching of the continuous EEG data stream.

4.2.2 Experimental Setup and Data Acquisition

Subjects

Data of 19 voluntary, healthy subjects were recorded. The experimenting room was not shielded against noise or electromagnetic influences of the surroundings. Each subject performed a single 30 minute training session which was used to collect EEG data for the training of the classifier. An explicit training of the subject itself was not necessary.

Hardware Setup

All subjects sat on a fixed chair. The gold-cup electrodes were attached to the scalp at the positions Fz,Pz,Cz,Oz,P3,C3,P4,C4,PO7 and PO8 according to the *10-20 system* [Jasper, 1958b]. A Samsung X30 notebook with 512MB Ram was used to record data from the EEG amplifier (Mindset24 - Nolan Computer Systems LLC). Signal quality was monitored with the Mindset24 software in real time, using the traces and the FFT spectrum provided by that software package. Furthermore, a 19" Sony screen with a refresh rate of 100Hz was used to display the visual stimuli.

Experimental Setup

Each experiment consisted of three basic steps:

1. Preparation of the subject
2. Recording of training data and training of the classifier
3. Classifying using the trained classifier.

The preparation involved cleaning of the scalp at the electrode positions with isopropanol. Additionally, an abrasive paste was applied which removed the upper dermal layer to improve impedances. To achieve low-noise signals, only impedances $< 5k\Omega$ were accepted.

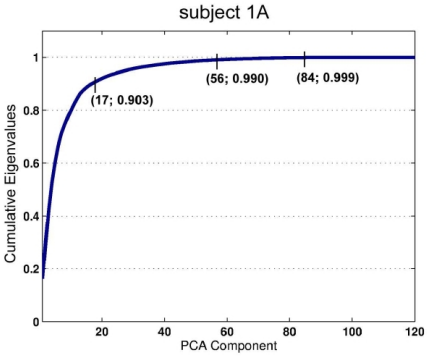


Figure 4.3. Cumulative Eigenvalues for Principal Components from the data of a random subject using 1600-dimensional feature vectors representing time series of 10 electrodes (according to [Kaper, 2006]). Even with only 84 Principle Components 99.9% of the data variance is captured. Therefore, we utilized the first 100 Principal Components in our online experiments

During the recording of training data and the final classification task, the software part of our BCI controlled the presentation of the stimuli and kept track of the stimulus onsets (*triggers*). Using these marks, the continuous EEG data were split into epochs of 800ms length, each corresponding to one single stimulus (see figure 4.2).

Prior to the actual recording of training data, the subjects were offered a few presentation steps (about 2-3 trials) to familiarize themselves with the training procedure. Thereby, the software selected 5 different and randomly chosen letters that were used as *target letters*, i.e. which the subject should pay attention to. A total number of 50 trials were issued. Each *trial* consisted of 5 *subtrials* with each of the 5 letters chosen for a random subset of the 10 trials. Prior to each trial, the target letter was displayed for 2 seconds. During the recording of a trial, the subjects were advised to remain still and try to prevent eye movements and blinking. At the end of a trial the software waited for a keypress to allow the subject to move and blink eyes. After completing all subtrials the data were used to train the classifier.

As a last step, a predefined word should be spelled using the trained classifier. The procedure differs from the training phase in such a way, that no keypresses were issued between consecutive trials. Furthermore, the classified letter is shown at the end of a subtrial for 3 seconds followed by another 3 second delay where the letter matrix was displayed.

4.2.3 Preprocessing and Classification

For the training set, EEG data of the 10 electrodes were collected and split into 800ms time windows, called epochs. All data were sampled at 256Hz (i.e. 256 samples per second) and therefore one epoch of 800ms data consists of

$$\frac{256 \cdot 800}{1000} = 204 \tag{4.1}$$

samples per channel. Each channel was bandpass filtered with upper and lower corner frequencies of 1 and 9Hz (using a DFT of the *FFTW* library [Frigo and Johnson, 2005]). Since 10 channels were used in our experiments, the epochs of each channel were concatenated to form the final input vector with a dimension of $204 \cdot 10 = 2040$. In order to reduce the dimensionality of the data, *Principal Component Analysis (PCA)* was applied.

To calculate this transformation, given $\mathbf{X}$ is the n-dimensional input vector, one has to find the $n \times n$ orthonormal projection $\mathbf{P}$ such that

$$\mathbf{Y} = \mathbf{P}^T\mathbf{X} \tag{4.2}$$

with the constraint that $cov(\mathbf{Y})$ is a diagonal matrix and $\mathbf{P}^{-1} = \mathbf{P}^T$. By substitution and matrix algebra this can be rewritten as

$$\begin{aligned} cov(\mathbf{Y}) &= E[\mathbf{Y}\mathbf{Y}^T] \\ &= E[(\mathbf{P}^T\mathbf{X})(\mathbf{P}^T\mathbf{X})^T] \\ &= E[(\mathbf{P}^T\mathbf{X})(\mathbf{X}^T\mathbf{P})] \\ &= \mathbf{P}^T E[\mathbf{X}\mathbf{X}^T]\mathbf{P} \\ &= \mathbf{P}^T cov(\mathbf{X})\mathbf{P}. \end{aligned}$$

Now we have

$$\mathbf{P}cov(\mathbf{Y}) = \mathbf{P}\mathbf{P}^T cov(\mathbf{X})\mathbf{P} = cov(\mathbf{X})\mathbf{P}$$

Rewriting $\mathbf{P}$ as a $d \times 1$ column vector yields

$$\mathbf{P} = [P_1, P_2, \ldots, P_n]$$

and $cov(\mathbf{Y})$ as

$$\begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix}$$

Substituting into equation above we obtain

$$\begin{aligned} & [\lambda_1 P_1, \lambda_2 P_2, \ldots, \lambda_n P_n] \\ = \; & [cov(X)P_1, cov(X)P_2, \ldots, cov(X)P_n] \end{aligned}$$

For each pair $\lambda_i P_i = cov(\mathbf{X})P_i$ P_i is an Eigenvector of $cov(\mathbf{X})$. Thus finding the Eigenvectors of the covariance matrix a projection $\mathbf{P}$ can be constructed which satisfies (4.2).

Kaper [Kaper, 2006] computed the cumulative eigenvalues for his subjects to get an estimate of the number of principal components that should be used to achieve a strong dimensionality reduction whilst retaining most of the data variance. The distribution of the cumulative Eigenvalues of a exemplary subject is depicted in Figure 4.3 using a 10 channel

time series of 160 samples per channel, sampled at $200Hz$. For this subject, 99.9% of the data variance are captured whithin the first 84 Principal Components, while 99% and 90% are incorporated in 56 and 17 components, respectively. Therefore, we decided to use the first 100 Principal Components in our experiments to construct a dimensionality reduced data set which served as training input for the *Linear Discriminant Analysis (LDA)* classifier. To save computation/preprocessing time, a modified PCA matrix, which we call *general PCA* [Kaper, 2006], was used to eliminate the need to recompute the matrix for each subject. A general PCA matrix can be applied to EEG data of arbitrary subjects, as long as the internal structure of the data (i.e. the order of the channels) remains identical to those data that were used to compute the general PCA. For a detailed explanation the authors are referring to [Kaper, 2006]. Due to the higher sampling rate in our online experiments, the original *general PCA* matrix was linearly extended to be applicable to 2040 dimensions, i.e. 204 samples for each of the 10 channels as explained in the beginning of section 4.2.3 The reduction of dimensionality is essential since the computation of the LDA weight vector

$$\mathbf{w} = \mathbf{S}_W^{-1}(\mathbf{m}_1 - \mathbf{m}_2), \tag{4.3}$$

where $\mathbf{m}_1, \mathbf{m}_2$ are means for class 1 and 2 respectively, requires the inversion of the *within-scatter* matrix $\mathbf{S}_W$. With the initial dimension of 2040 (i.e. 10 channels×204 samples) it is likely that the matrix becomes singular and thus its inverse does not exist.

Using this weight vector, a binary decision can be made by computing the projection

$$s(x) = \mathbf{w} \cdot \mathbf{x} + b, \tag{4.4}$$

where b is a bias and $\mathbf{x}$ is the data vector containing the EEG data for a specific epoch. Based on the sign of $s(x)$, the epoch x is considered to belong to the $\mathscr{P}^+$ class (contains P300) if the sign is positive and to the $\mathscr{P}^-$ class (contains no P300) otherwise. A bias is found using a grid search and validated by a 5-fold cross validation., while the b that showed the best binary classification performance during the cross validation is kept. In this case the grid-search and cross validation to determine the bias is necessary because one can not expect equal covariances for both classes. As seen in figure 4.2 the sample windows (i.e. epochs) are overlapping. Due to this it is clear that several $\mathscr{P}^-$ windows overlap with the two $\mathscr{P}^+$ windows within one single subtrial. Therefore, variance of the $\mathscr{P}^-$ class is influenced by the usual noise and muscle artifacts as well as windowed parts of P300 components (i.e. high amplitudes at varying positions). Consequently the class variance is much higher for the $\mathscr{P}^-$ class than for the $\mathscr{P}^+$ class, which should be rather constant given both class distributions are equally influenced by noise and muscle artifacts, and therefore covariances for both classes are not expected to be equal. As a result, the direct solution for the bias b will be sub-optimal and a grid search has to be conducted to find an optimal value. When running the online letter classification task, the decision on whether a row/column is considered $\mathscr{P}^+$ or $\mathscr{P}^-$ is achieved by averaging a classification over a number of subtrials. Therefore, a score is computed for each epoch $\mathbf{x}_i$ of a trial which is essentially $s(\mathbf{x}_i)$. To infer whether a specific row and column contained the

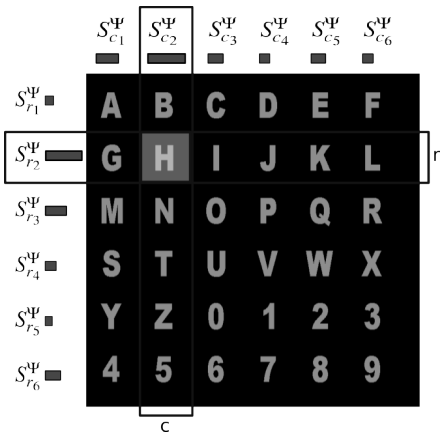


Figure 4.4. The cross-section of the row and column with the highest classification score denotes the classified letter (highlighted letter). The bars near the S_i^Ψ terms indicate the classification score of the corresponding epoch (i.e row/column) as computed by equation 4.5.

target letter (i.e., a P300 was elicited during the intensification of the row/column), the scores

$$S_i^\Psi(\bigcup_j \mathbf{x}_{i,j}) = \sum_{n=1}^{j} s(\mathbf{x}_{i,n}), \tag{4.5}$$

with $x_{i,j}$ denoting the i-th epoch of the j-th subtrial are computed and merged into the score matrix $\mathbf{M}$ whereas each cell $\mathbf{M}_{i,j}$ is defined by

$$\mathbf{M}_{i,j} = S_h^\Psi + S_k^\Psi \tag{4.6}$$

with

$$h = (i-1)\cdot 6 + 1 \tag{4.7}$$

$$k = h + j \tag{4.8}$$

Consequently, the target letter/cell $\mathbf{M}_{i,j}$ is determined by

$$r, c = \underset{i,j\in[1,\ldots,6]}{\arg\max}\, \mathbf{M}_{i,j}. \tag{4.9}$$

This process is also illustrated in figure 4.4.

4.2.4 Measuring the BCI's performance

In order to assess the speed of the BCI we calculated the information transfer rate B in bits/min according to equation 4.10 using the accuracy of the spelling task, denoted by p_{acc}, as dependent variable. Though taking into account the number of possible inferences (i.e. number of

symbols) n_{sym} and the time for spelling a symbol using n_{sub} subtrials with an *Interstimulus Interval* of t_{ISI} seconds, these rates do not account for the inevitable time delay between consecutive trials and thus only reflect a theoretical information transfer rate.

$$B(n_{sym}, n_{sub}, t_{ISI}) = T \cdot L \tag{4.10}$$

where

$$T = \frac{60}{n_{sub} \cdot t_{ISI}}$$

$$L = log_2 n_{sym} + p_{acc} log_2 p_{acc} + (1 - p_{acc}) log_2 \frac{1 - p_{acc}}{n_{sym} - 1}$$

4.2.5 Experiment 1 - Static Subtrials

The first experiment employed a fixed number of subtrials per trial which were systematically modified ranging from 1 to 10 subtrials per trial. After the initial training procedure, the subjects had to spell the word `INTERFACE` ten times, each time with a different number of subtrials per trial, by counting the number of target letter intensification. The experiment started with a fixed number of 10 subtrials while decreasing this number for each subsequent round by one until 1 subtrial was reached. Whenever a wrong classification occurred, the subject continued with the next letter without correcting the last one. Symbol inference was carried out according to equation 4.9 incorporating data of all subtrials.

4.2.6 Experiment 2 - Dynamic Subtrials

While examining single trials of the previous experiment that used a static number of subtrials, we noticed that the majority of the classified letters could have been already classified several subtrials before the predefined limit. A visualization of the row and column scores (figure 4.5) shows the development from a random classification up to a very certain result. Each cell represents a letter of the original letter matrix that was presented to the user. The color intensity of each cell depicts the confidence in a *true positive (TP)* classification for the respective letter. With increasing subtrials the brightness densities are shifted towards the target row and column, resulting in a cross-like shape. As a result, the overall brightness of the matrix $\mathbf{M}$ decreases if M is accumulated over a number of subtrials and then scaled to $[0 \ldots 1]$. A metric to measure the overall brightness is given by

$$TP_{score} = \sum_i \sum_j M_{i,j} \tag{4.11}$$

with

$$\mathbf{M} = \frac{\mathbf{M} - min(\mathbf{M})}{max(\mathbf{M})} \tag{4.12}$$

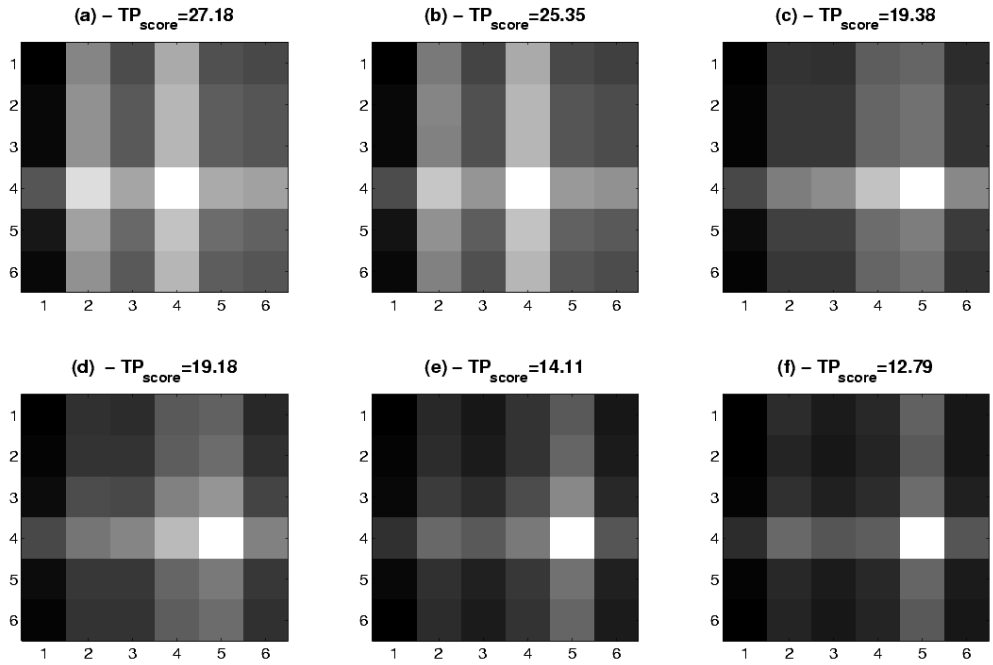


Figure 4.5. The figures (a)-(f) represent the score matrices of a complete trial, consisting of 6 subtrials. Thereby, (a) is the score matrix obtained from the first subtrial and (b)-(f) from the subsequent subtrials. The brightest spot in (a) and (b) would coincide with the target letter that would be inferred using equation 4.9). However, especially the first 2 matrices expose a very broad brightness distribution and thus lead to rather uncertain inferences. With increasing subtrials, the brightest column changes to column 5 (see *(c)*) while the brightest row remains the same. Furthermore, the overall brightness of the matrices (reflected by TP_{score}) decreases with increasing subtrials. Since $M_{4,5}$ denotes the correct target letter, a true positive classification could have been achieved already after the third subtrial (c).

For an infinite number of subtrials TP_{score} will converge towards 1. If $\mathbf{M}$ is processed as stated in 4.12 it represents the sum of all LDA scores received for each cell. Since the classifier was trained to output high values for epochs that match the samples of the $\mathscr{P}^+$ class, the value of the target cell will grow faster than all others. This can be proven using the strong law of large numbers. Let $\overline{X}_i$ be the sample average for i samples of the $\mathscr{P}^+$ class and μ^+ be the $\mathscr{P}^+$ class mean. Then, according to the law of large numbers, the probability that $\overline{X}_i$ will converge towards μ^+ for infinite i equals 1, or more formally speaking

$$P(\lim_{i\to\infty} \overline{X}_i = \mu^+) = 1.$$

The same applies for the $\mathscr{P}^-$ class. Let $\mathbf{R}^i$ be the 6×6 matrix containing only of LDA row scores for the i-th subtrial and $\mathbf{C}^i$ the matrix containing the column scores. Given the target

letter is located at position $(1,1)$ (top left), the following holds:

$$\forall c \quad \lim_{i\to\infty} \frac{1}{i} \sum_{n=1}^{i} R_{1,c}^{n} = \mu^{+} \tag{4.13}$$

$$\forall r \quad \lim_{i\to\infty} \frac{1}{i} \sum_{n=1}^{i} C_{r,1}^{n} = \mu^{+} \tag{4.14}$$

Now let $\mathbf{M}^{i} = R^{i} + C^{i}$ be the combined score matrix (see equation 4.5) for the i-th subtrial, then

$$\lim_{i\to\infty} \frac{1}{i} \sum_{n=1}^{i} m_{1,1}^{n} = 2\mu^{+} \tag{4.15}$$

$$\lim_{i\to\infty} \frac{1}{i} \sum_{n=1}^{i} m_{[2\ldots6],1}^{n} = \mu^{+} \tag{4.16}$$

$$\lim_{i\to\infty} \frac{1}{i} \sum_{n=1}^{i} m_{1,[2\ldots6]}^{n} = \mu^{+} \tag{4.17}$$

and for any other cell r,c

$$\lim_{i\to\infty} \frac{1}{i} \sum_{n=1}^{i} m_{r,c}^{n} = \mu^{-} \tag{4.18}$$

and thus

$$\mathbf{M}^{\Theta} \equiv \lim_{i\to\infty} \frac{1}{i} \sum_{n=1}^{i} M^{n} = \begin{bmatrix} 2\mu^{+} & \mu^{+} & \cdots & \mu^{+} \\ \mu^{+} & \mu^{-} & \cdots & \mu^{-} \\ \vdots & \vdots & \ddots & \vdots \\ \mu^{+} & \mu^{-} & \cdots & \mu^{-} \end{bmatrix}. \tag{4.19}$$

Now $\mathbf{M}^{\Theta}$ contains only three distinct values, namely $2\mu^{+}$, μ^{+} and μ^{-}. If in a next step $\mathbf{M}^{\Theta}$ is scaled to $[0\ldots1]$, the maximum value (i.e. $m_{1,1}^{\Theta} = 2\mu^{+}$) gets mapped to 1, μ^{+} gets mapped to 0.5 and all other values (i.e. μ^{-}) get mapped to 0 since $2\mu^{+} > \mu^{+} > \mu^{-}$. Consequently $\lim_{i\to\infty} TP_{score}(\mathbf{M}^{\Theta}) = 6$, i.e. with increasing subtrials TP_{score} will decrease asymptotically to a minimum of 6.

The training data, incorporating all subjects from experiment 1, were used to calculate a mean TP_{mean} by extracting the first subtrial of each trial which would have resulted in a true positive classification. This value served as a base threshold that corresponds to the fastest setting. During the offline experiments it was systematically modified after each round as stated in table 4.2. During the spelling task, equation 4.12 was calculated after each subtrial and an additional subtrial was issued whenever $TP_{score} < TP_{mean}$. All subjects had to spell the word `BRAINCOMPUTERINTERFACE` four times, each time with a different threshold setting.

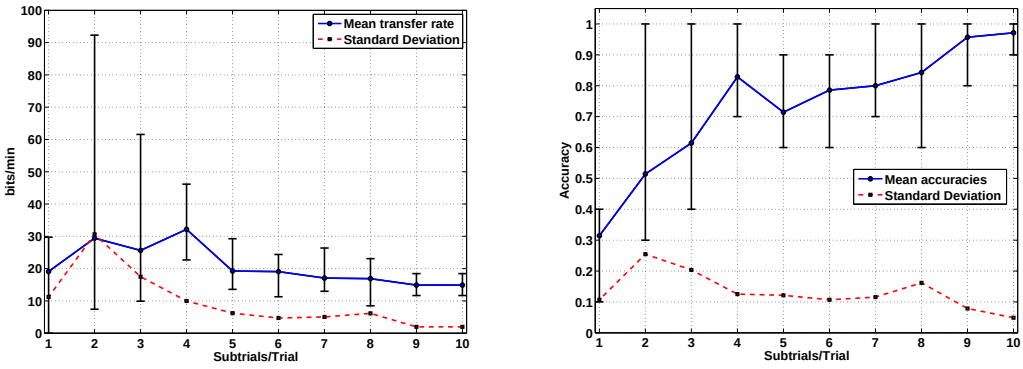
4.3 Results

4.3.1 Experiment 1 - Results

To assess the performance of our BCI, we computed the binary classification accuracy (table 4.1) of the classifier as well as information transfer rates and accuracies of the spelling task, depicted in figure 4.6. The subplot (a) of figure 4.6 shows the transfer rates of each individual subject, that were calculated for the according number of subtrials. Transfer rates up to 92.32 bits/min were achieved by subject 5 who spelled the word `INTERFACE` without errors using only 2 subtrials per letter. The mean transfer rates are depicted in the mid part of the figure, showing a maximum transfer rate of 32.17 bits/min at 4 subtrials, followed by 2 subtrials with an approximate transfer rate of 30 bits/min but at the cost of about 30% accuracy loss which makes effective communication impossible. A further decrease of subtrials did not lead to a better overall performance in transfer rates. Furthermore, the error bars in figure 4.6 (a) and (b), representing the maximum and minimum achieved transfer rate/accuracy for a given subtrial amount, as well as the standard deviation of the results show a generally increasing variance for decreasing subtrials per trial with the exception of 1 subtrial per trial. If only 1 subtrial per trial is issued all subjects performed almost equally bad in terms of accuracy.

Table 4.1. Binary classification results for all 7 subjects as obtained after the initial training of the LDA classifier. These percentages the training accuracy based on single subtrial classifications. Thereby only the sign of the resulting projection of the data vector onto the weight vector was used to discriminate between the $\mathscr{P}^+$ and $\mathscr{P}^-$ classes.

S1	S2	S3	S4	S5	S6	S7	Mean
80%	82%	71%	74%	88%	79%	83%	$80\% \pm 5.6\%$



(a) Experiment 1 - Mean bits/min

(b) Experiment 1 - Mean accuracy

Figure 4.6. Performance for a word spelling task of *experiment 1*: The subjects had to spell the word `INTERFACE`. Plot a) represent the speed and b) the accuracy for a given number of subtrial presentations.

4.3.2 Experiment 2 - Results

Twelve subjects participated in the final experiment block to evaluate the performance of the BCI using dynamic subtrials. To assess performance more precisely, the word that had to be spelled was elongated to BRAINCOMPUTERINTERFACE. An initial threshold value of TP_{mean} was used in the first session, while this parameter was decreased for each subsequent session as shown in table 4.2. Fastest transfer rates were achieved at a threshold of 34 which revealed a maximum performance of 84.60 bits/min (or 19.48 theoretical symbols/min) at 81.81% accuracy. Mean transfer rates (table 4.3) ranged from 29.35 to 50.61 bits/min with mean accuracies ranging from 65.53% to 87.50% (table 4.4). Since high bit rates coupled with low accuracy are of limited use for an online BCI a more practical measure to evaluate the actual performance is given in table 4.3. This measure shows the number of accurate symbol selections per minute (SPM) calculated by

$$\text{SPM} = \frac{N_{correct}}{t},$$

whereas $N_{correct}$ denotes the number of correctly classified letters and t the time used for the whole spelling task. The results presented in table 4.3 show theoretical symbol communication rates ranging from 13.13 at the fastest setting to 6.37 at the most accurate setting. These rates however, did not account for the inter-trial delay that results from displaying the classified letter to the user. Since this is an inevitable issue of online BCI's the actual "real-world performance" was included in the table (*SPM 2*) which incorporates the 6 seconds inter-trial delay in the calculation. Taking the latter results as a basis it can be stated that our BCI is able communicate about 4 letters per minute in a real-wold scenario. The theoretical transfer rates however, (those not incorporating the inter-trial delay) are still useful as they serve as reference values to other BCI systems which usually omit the time between trials in their performance measures. Compared to the fastest mean transfer rate of the static subtrial approach even the second most accurate setting (thr=28.9) of this new approach outperforms the initial one by about 4 bits/min while maintaining a mean accuracy of 86%. For two of the 12 subjects accuracies of 100% were achieved and 7 of the subjects achieved more than 90% accuracy. A comparison of the maximum mean transfer rates shows an improvement of 72.9% (i.e. 21.46 bits/min difference) over the static subtrial approach. At this fastest setting, accuracies dropped to 65.53% in the mean. As figure 4.7 depicts, the mean accuracies are clearly decreasing with increasing threshold, whereas mean transfer rates are almost linearly increasing. The standard deviation of the results however decreases with increasing accuracy settings as expected. While results differ more than 20 bits/min at the fastest setting, less than 10 bits/min variance is present at the most accurate setting (a threshold of 27.2).

4.3.3 Offline Analysis of the Scoring Function

The dynamic subtrial method we employed in our last experiment was solely based on the overall brightness intensity of the normalized score matrix and therefore was based on only

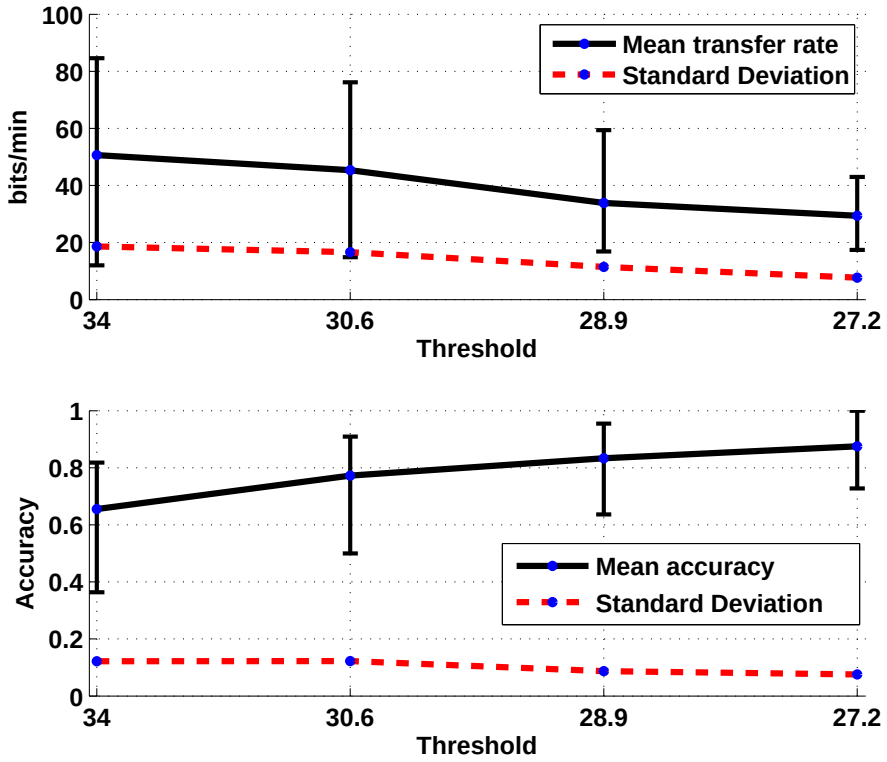


Figure 4.7. Mean transfer rates and accuracies obtained from the second experiment. The error bars depict the maximum and minimum rates achieved by the subjects for a given threshold. The standard deviation (dotted line) steadily decreases with decreasing thresholds, i.e. increasing accuracy.

Session	Threshold	% of Start Threshold
1	34	100%
2	30.6	80%
3	28.9	70%
4	27.2	60%

Table 4.2. Thresholds for TP_{score} that were investigated in experiment 2. An initial threshold of 34.00 was applied for all subjects. For each subsequent session, the threshold was decreased by the certain amount, according to table 4.2.

	Threshold			
	34	**30.6**	**28.9**	**27.2**
BPM (theoretical mean)	50.61	45.33	33.88	29.35
SPM 1 (theoretical mean)	13.13	10.57	7.62	6.37
SPM 2 (real mean)	4.33	4.41	3.91	3.65
Avg. theoretical time (min)	1.17	1.74	2.58	3.16
Avg. real time (min)	3.37	3.94	4.78	5.36

Table 4.3. Transfer rates measured in **bits/min (BPM)** as well as **correctly predicted letters per minute (SPM)** for the 22 letter word spelling task using the dynamic subtrial scheduling scheme. The *theoretical* values omitted the 6 second inter trial delay that occurred between consecutive letters in the study, whereas the *real* values incorporate this delay in the calculation. Additionally, the average time needed by the subjects to spell a 22 letter word presented in the lower part of the table.

	Threshold			
Subject	**34**	**30.6**	**28.9**	**27.2**
1	72.72	81.81	81.82	90.91
2	77.27	86.36	90.91	86.36
3	72.72	50.00	77.27	72.72
4	68.18	90.90	86.36	81.82
5	68.18	77.27	86.36	90.91
6	63.63	77.27	86.36	100.00
7	63.63	77.27	72.72	86.36
8	50.00	72.72	81.82	86.36
9	81.81	90.90	95.45	86.36
10	36.36	59.09	63.63	81.82
11	68.18	86.36	86.36	100.00
12	63.63	77.27	90.91	86.36
Mean	65.53	77.27	83.33	87.50

Table 4.4. Accuracies in percent as achieved by the 12 subjects for each threshold setting. Mean accuracies increase with decreasing threshold values.

a single criterion. This, however, is not the only criterion which can serve as an indicator of whether to repeat or end the current subtrial. We expected results to improve by adding a second criterion which must be met in addition to the original brightness condition. An investigation of false classified epochs showed that the difference between the best and the second best cell of the score matrix can be a suitable secondary criterion since similar values indicate uncertain classification. Therefore, we added the secondary criterion to the subtrial repetition decision function to improve accuracy of the decision function. A trial will be ended if both of the following criteria are met:

C_1) *Sum Threshold* - Overall brightness is **lower** than (or equals) threshold t_1 (i.e. the original TP_{score} criterion)

C_2) *Ratio Threshold* - Score ratio of highest scored cell to second highest scored cell (i.e. percental difference of the two highest ranked cells) is **greater** than (or equals) threshold t_2.

Thereby, C_1 corresponds to the TP_{score} measure which was already introduced in section 4.2.6. Both conditions can be encapsulated into one single function of the score matrix $\mathbf{M}$. Defining the conditions C_1 and C_2 as

$$C_1(\mathbf{M}) = TP_{score} = \sum_i \sum_j M_{i,j} \quad , \tag{4.20}$$

$$C_2(\mathbf{M}) = 1 - \frac{k_2}{k_1} \tag{4.21}$$

and defining

$$k_1 = M_{i_1,j_1} \quad \text{with } i_1, j_1 = \arg\max_{h,k} M_{h,k} \in \mathbf{M} \tag{4.22}$$

$$k_2 = M_{i_2,j_2} \quad \text{with } i_2, j_2 = \arg\max_{h,k} M_{h,k} \in (\mathbf{M} \backslash M_{i_1,j_1}) \tag{4.23}$$

allows the construction of a boolean subtrial repetition decision function $\mathscr{D}$ as

$$\mathscr{D}(\mathbf{M}, t_1, t_2) = \begin{cases} \text{true} & \text{if } (C_1(\mathbf{M}) \leq t_1) \wedge (C_2(\mathbf{M}) \geq t_2) \\ \text{false} & otherwise \end{cases} . \tag{4.24}$$

The basic main loop of the BCI is outlined by Algorithm 1. Clearly, a trial is ended whenever both criteria are met, i.e. $\mathscr{D}$ = true.

Since there exist no optimal a priori values for the thresholds t_1 and t_2, a grid search was conducted to find good values. To evaluate the extended score function, data of a random subject (subject 2 of the second experiment) was used to carry out an offline experiment which incorporates the improved decision function as defined by equation 4.24. These data

Algorithm 1 BCI main loop

```
1: M ← 0 {initialize score matrix M with zeros}
2: repeat
3:    Present subtrial
4:    Preprocess epochs
5:    Classify epochs
6:    Add classification score to M
7:    T ← M
8:    Normalize T to 1
9: until 𝒟(T, t1, t2) == true
10: return Infer letter according to equation 4.9
```

contained 10 subtrials per trial for 22 letters. Figure 4.8 illustrates the grid search result carried out for this subject. Thereby, $81 \times 81 = 6561$ different threshold combinations were tested and shown in the plots a)-d). In addition to the overall performance (measured in bits/min) that would have been achieved with certain threshold pairs, the number of subtrials and the errors/accuracies for each threshold pair are depicted in plots b)-d).

Further a comparison of both decision functions is implicitly shown in *a)*. Since the initial decision function that was used in experiment 2 only employed the *sum threshold criterion* C_1, which is equivalent to setting the *ratio threshold* $\mathscr{C}_2$ to 0, the first horizontal line of plot a) reflects the performance of the subject that would have been achieved with the BCI used in experiment 2. To clarify this equivalency, consider the function C_2 whose output is the interval $[0,\ldots,1]$. In fact, those numbers only describe the percental difference between the highest scored cell and the second highest scored cell of $\mathbf{M}$ and therefore $C_2(\mathbf{M}) = 0$ necessarily requires $k_1 = k_2$ as this equates to

$$C_2(\mathbf{M}) = 1 - \frac{k_2}{k_1} = 1 - \frac{k}{k} = 1 - 1 = 0 \tag{4.25}$$

with

$$k_1 = k_2 = k. \tag{4.26}$$

Since k_2 denotes the value of the second highest scored cell and thus is either less than or equal to k_1, equation 4.26 shows that the lowest possible value for C_2 is 0. Now given that C_2 will at least yield a value of 0 it is clear that the right hand side of the boolean term $(C_1(\mathbf{M}) \leq t_1) \vee (C_2(\mathbf{M}) \geq t_2)$, as defined in equation 4.24, is always true for $t_2 = 0$ and therefore $\mathscr{D}$ is only dependent on the left hand side (i.e. the first condition C_1). Thus, the first line of plot a) shows the performance that would have been achieved by our current online system and all subsequent lines depict the performance that could have been achieved if the ratio threshold (i.e. condition C_2) is incorporated into the decision function. The maximum

transfer rate could be increased from 58.54 bits/min (taking only C_1 into account) to 70.27 bits/min (taking C_1 and C_2 into account) which is an improvement of 19.73 bits/min (i.e. 20.4%) over the decision function which was used in experiment 2.

An important advantage of this method is that the BCI can be optimized for speed at a given accuracy constraint or it can be optimized for accuracy at a given speed constraint. Given **B** is a $n \times m$ matrix with $B_{i,j}$ denoting the bitrate achieved with the i-th C_2 threshold and the j-th C_1 threshold. Further let **A** be the $n \times m$ matrix containing the accuracies for the corresponding thresholds, $T^{C_2} = \{t_1 \dots t_n\}$ be the set of tested C_2 threshold values (ratio threshold) and $T^{C_1} = \{t_1 \dots t_m\}$ be the set of tested C_1 threshold values (sum threshold). The threshold pair $(T_i^{C_2}, T_j^{C_1})$ can be optimized for accuracy by choosing

$$I_1 = \{n | A_{n,m} \geq \tau_{acc}\} \tag{4.27}$$

$$I_2 = \{m | A_{n,m} \geq \tau_{acc}\} \tag{4.28}$$

$$\left\{ (T_i^{C_2}, T_j^{C_1}) | i, j = \underset{n \in I_1, m \in I_2}{\arg\max} \left(B_{n,m}\right) \right\} \tag{4.29}$$

with τ_{acc} denoting the lower bound of the desired accuracy. If $\tau_{acc} = max(\mathbf{A})$, the threshold pair which results in best bit rates for the given maximum accuracy will be chosen. In turn, the BCI can be optimized for pure speed by choosing

$$I_1 = \{n | B_{n,m} \geq \tau_{bps}\} \tag{4.30}$$

$$I_2 = \{m | B_{n,m} \geq \tau_{bps}\} \tag{4.31}$$

$$\left\{ (T_i^{C_2}, T_j^{C_1}) | i, j = \underset{n \in I_1, m \in I_2}{\arg\max} \left(A_{n,m}\right) \right\} \tag{4.32}$$

with τ_{bps} denoting the lower bound of the desired bitrate. Utilizing equation 4.32, the threshold pair which results in highest accuracy and delivers at least a bit rate of τ_{bps} is being selected. One possible drawback of this proposed threshold selection scheme is the possible overfitting. As it can be seen in plot a), c) and d), variance increases in the upper right part. Therefore, if the selected threshold pair lies within a area of high variance it might lead to sub optimal results. Nevertheless, accuracy of the maps should increase with the amount of data used to calculate them.

4.4 Conclusion

In this paper we showed how classification speed of a P300 based BCI can be improved by employing a dynamic subtrial scheduling scheme. Taking the offline BCI of Kaper that was ranked at first place in [Kaper et al., 2004] as our basis, the classification system was modified to an online version and a first experiment to assess its performance was conducted. As the

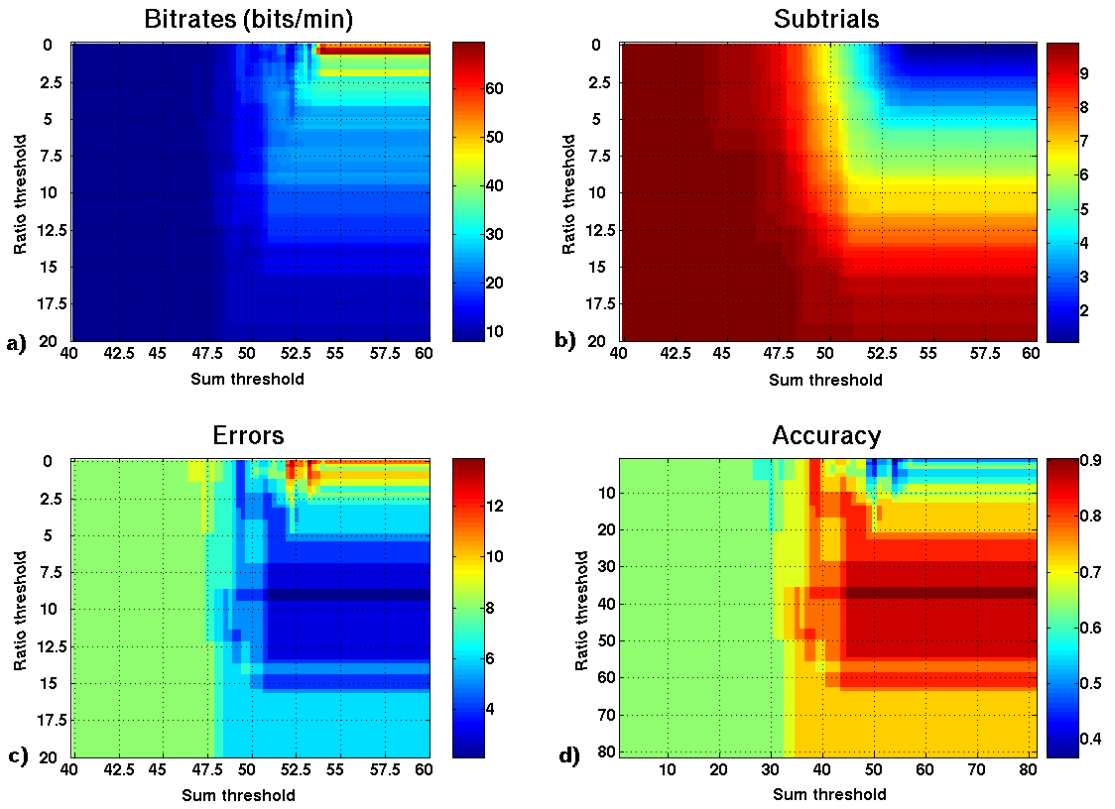


Figure 4.8. Visualization of the grid search results using the extended decision function $\mathscr{D}$. Each point of the picture reflects the performance that would have been achieved with the corresponding thresholds. In total, results of $81 \times 81 = 6561$ offline experiments each with a different threshold combination are shown in the plots. Data for 22 letters with 10 subtrials per trial were provided by subject 2 who participated in the second experiment. Top left: Performance measured in bits/min for each threshold pair. Bottom Left: Amount of subtrial used for the corresponding threshold pair. Right: Accuracy for each threshold pair. .

results were lower than in the offline version, further adaptions were introduced to match the transfer rates achieved by the offline version.

Following the example of Serby [Serby et al., 2005] who already introduced a method to adapt to the users performance online, a similar method has been presented that limits the number of subtrial repetitions according to the users online performance. To strengthen the findings of Serby, who only used 6 subjects and 5 letters in their experiments, that dynamic subtrial limiting according to the users performance is a viable method for boosting bitrates and maintaining accuracy, we elongated the word for the spelling task to 22 letters and incorporated 12 subjects in the second study. Therefore, our mean results are less likely to be contaminated with outliers. As experiment 2 shows, the maximum mean transfer rates could be improvement from 32.17 bits/min to 50.61 bits/min (57.32%), though at a substantial loss of 26% accuracy. At comparable accuracies of around $80\% \pm 5\%$, mean bitrate could be boosted from 32 bits/min to 45 bits/min with a slight accuracy loss of 5%. Compared to the BCI introduced by Serby [Serby et al., 2005], who reached a mean accuracy 79.5% and a communication rate of 4.51 symbols/min, our approach showed a mean communication rate of 7.62 symbols/min at 83.33% mean accuracy. When incorporating the 6 seconds inter-trial delay in our calculation, we still achieve a communication rate of 3.91 symbols/min at 83.33% accuracy. Since the delay is inevitable for an online BCI as it gives the user time to perceive the predicted symbols, this rate should be taken as the practical communication rate whereas the former rates should be considered theoretical communication rates. A peak practical communication rate of 5.82 symbols/min with an accuracy of 90.9% was achieved by subject 9 which equals an information transfer rate of 84.60 bits/min. We also showed with an offline analysis, that it is possible to further increase transfer rates by choosing better subtrial repetition decision functions. An example of such an improved decision function was given in section 4.3.3 that extended the initial decision function by a secondary criterion. Furthermore, a method to select accuracy or bitrate optimized thresholds was presented in conjunction with the extended decision function $\mathscr{D}$. Though not tested in an online scenario, the maximum transfer rate of 58.54 bits/min (of subject 2) obtained by the initial approach (offline in this case) could be increased to 70.27 bits/min which is an improvement of 20.4%. These results show that a dynamic subtrial scheduling scheme can dramatically improve performance of an online BCI and therefore strengthen the findings of [Serby et al., 2005]. Additional research should investigate if a dynamic subtrial approach also applicable for locked-in patients.

5 Study 2 - Error-Related Potentials

5.1 Introduction

Recent research has brought up a vast amount of sophisticated classification methods which are able to predict the user's intent with relatively high accuracy and minimal training data. Nevertheless, BCIs are prone to classification errors. Some efforts have been made to incorporate error-related brain responses to improve accuracy of the system which mainly rely on a certain type of error-negativity (ERN/Ne) occurring after the onset of an erroneous trial followed by a later positivity called *Pe* which occurs at 300-400ms after the stimulus onset. The idea of using additional information contained in EEG signals that can give information about the correctness of the command inferred by the BCI has already been used by some groups [Ferrez et al., 2007, Ferrez and del Millan, 2005, Schalk et al., 2000]. For the most part these studies used the ERN to predict errors in a selection task using imagined motor commands. These methods are based on previous error-potentials (ErrP) research which identified mainly 3 different types of error-potentials.

The *response ErrP* can be observed during choice reaction tasks and is especially pronounced at fast stimulus presentation speeds. The second type is called *feedback ErrP* which occurs during reinforcement learning tasks. That is, the BCI response to an actively executed action is delayed in order to measure only brain activity related to the response. This is the type which coincides with most paradigms employed by BCIs. The last type is called *observation ErrP* and it occurs in pure *action-response observation tasks* where the subject does not execute an action but can observe the action and response of another person. As with the former types of ErrP, a negative deflection is visible 250ms after an erroneous response was perceived by the subject [van Schie et al., 2004]. The latencies of the Ne and Pe components strongly correlate with the peaks found in oddball experiments and therefore, as mentioned by [Schalk et al., 2000], it is not clear whether the ErrP measured after erroneous trials is in fact only a P300. Recent studies tried to incorporate the ErrP into an error correction system for a P300 Speller BCI. For the discrimination task between ErrP and non-ErrP epochs they reported overall correct rates ranging from 60-80% [Visconti et al., 2008, Dal Seno et al., 2009].

The aim of this study is to assess EEG data of erroneous trials and investigate properties of the error-related potential. In a second step, a classification method will be developed which reliably discriminates between correct and erroneous trials. The experiments presented in this chapter will serve as a basis for a general error correction framework. Section 5.5 will in-

troduce a framework which models a general error correcting BCI with Markov chains. This probabilistic model allows to compute the theoretical communication time per correct symbol. A particular useful feature of this method is the ability to optimize classifier thresholds to yield an optimal global system performance. Using this novel method, the feasibility of the proposed error correction BCI will be evaluated in section 5.6.

5.2 Experimental setup

The P300-speller paradigm itself consists of a matrix of letters as shown in figure 4.1. For each stimulus presentation round each row and column of this matrix are highlighted once in random order for a short amount of time. The term *subtrial* will be used for the process of highlighting each row/column once. Whenever the row or column which contains the attended letter is highlighted, a P300 component can be measured. The symbol cell the user is focusing corresponds to the intersection of the row and column which elicited a P300. The predicted letter is usually displayed as a feedback stimulus after the end of a trial. Multiple subtrials are often used to improve the signal to noise ratio which results in a more accurate prediction of the target. To exploit activity related to incorrect feedback of the BCI, a classification method is required that is able to identify erroneous responses from the EEG within a single trial since feedback is only available once at the end of a trial.

Therefore it is necessary to investigate what type of ErrP occurs in a P300-Speller paradigm and what parameters might influence the elicitation. To answer these questions an experiment with 5 healthy mixed male and female subjects has been conducted. For this experiment, presentation parameters such as ISI, target-error probability and presentation duration have been chosen according to values which also produce high amplitude P300 components. A conservative ISI of 1s was chosen to eliminate overlapping epochs. A low target-error probability of 25% has been used for two reasons. First off, this corresponds to a realistic number of errors for an online BCI and secondly it is not clear whether an ErrP might vanish with increasing error probability. Since it has been shown by [Ferrez, 2008, Dal Seno et al., 2009] that ErrPs occur at 25% error probability, this value proved to be a reasonable choice. A parameter so far never mentioned in the literature in relation to ErrPs is the presentation duration of the predicted letter. This parameter could be linked to task difficulty since it is mentally more challenging to recognize letters quickly. This induces slight stress to the subjects which seemed to be an important parameter in earlier ErrP experiments [van Schie et al., 2004] involving reactive movements under strict time restrictions. To investiagte whether the stimulus presentation time, which is linked to task difficulty, has an effect on the ErrP components, two experimental blocks were conducted.

Condition 1 - Low Task Difficulty With respect to these requirements the experiment was designed to simulate a letter spelling task using the P300 Speller paradigm. Prior to each trial, a target letter that had to be focused was shown followed by a 1s period showing a black screen followed by the transition to the matrix presentation round.

After one complete round, the screen was set to black for 3s to prevent post stimulus effects like slow-waves to be included in the feedback evaluation. The predicted letter was shown afterwards for 1s followed by another 1s pause displaying a black screen. The timing structure of the paradigm is illustrated in figure 5.1.

For this experiment the accurate prediction of the target letter was not important as we were interested in the brain response to the feedback only. For each subject an experiment comprised of 5 blocks was conducted. Each experimental block used a fixed number of one subtrial per letter whereas the task for the subject was to spell the word "BRAINCOMPUTERINTERFACE12" containing 24 letters. Since the error-rate of the feedback could potentially have an influence on the error-related activity it had to be controlled. Therefore classification of the brain-response to predict the target was omitted and predetermined letters were used. The sets of predetermined outcomes were chosen to contain 25% wrong letters and were randomized with respect to their occurence in the sequence to account for habituation effects of the subjects. All five subjects were told to attend to the letter that was shown at the beginning of each trial and mentally count whenever the row or column containing this letter was highlighted. Further, they were informed that the BCI would predict which letter they attended after the end of the speller phase. None of the subjects was aware that the outcome was predetermined. After each completed block of 24 trials the subjects were given the option for a short pause to relax since the procedure can be very tiresome which in turn could have an effect on the measured signals.

Condition 2 - High Task Difficulty The second condition follows exactly the same scheme as condition 1. In contrast to the first condition though, the ISI and feedback stimulus presentation time as well as pause times had been adjusted to compress the time needed to complete a full run from pre- to post-trial. The time for the trial-to-trial pause was reduced to 500ms and the pre-feedback pause was reduced to 1s. Feedback presentation time was adjusted to 250ms to coincide with the stimulus presentation time of the speller round. The ISI was lowered to 500ms to speed up the speller round since these stimulations were not needed in the ErrP evaluation. A convenient side effect of this condition is the fact that less time was needed for each round. Therefore it was possible to collect more data with the same amount of time compared to the first condition. This is a crucial point for later classification and statistical analysis since each completed round of 24 letters contained only 6 epochs which are related to erroneous trials. While the experiments in the first condition could only collect data from 5 blocks in reasonable time, the second experimental condition could collect data from 7 blocks which equals 40% more error-related epochs.

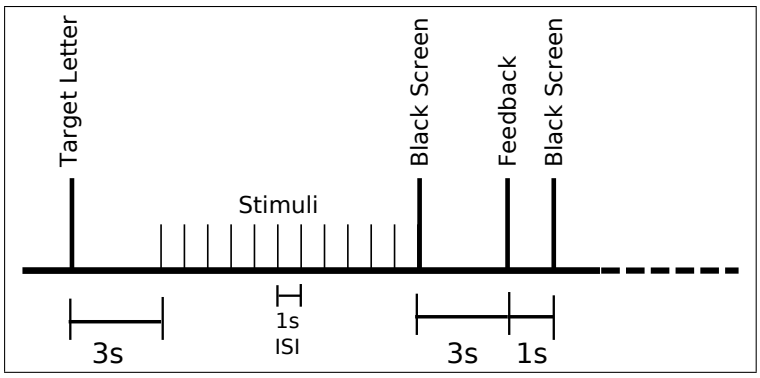


Figure 5.1. Structure and timing of the experiment (condition 1): Depicted are the three phases *target-*, *stimulus-* and *feedback presentation*. The transition delay between each phase was set to 3s. To maximize P300 amplitude 1s ISI was selected for all stimuli except the presentation of the target letter (3s).

5.2.1 Data acquisition

Ten subjects (7 male and 3 female) volunteered to participate in this experiment. The subjects were split into two groups where group 1 was assigned the low task difficulty (condition 1) and group 2 the high task difficulty condition (condition 2). As recording equipment a gUSBAmp EEG amplifier was used with 14 Ag/AgCl sintered ring electrodes attached to the positions Fp1, Fp2, AFz, Fz, F3, F4, Cz, C3, C4, Pz, P3, P4, O1 and O2. Reference electrodes were placed on the left and right ear in common reference montage. During recording, a highpass filter with cutoff frequency of 0.1Hz was applied. All data were sampled at 128Hz as a continuous stream and stored for the following offline analysis. Impedances ranged from $5-15k\Omega$ except for subject 2 with impedances of $20k\Omega$. As a consequence, below average signal-to-noise (SNR) ratios were expected for this subject. All subjects were placed about 1m in front of a video screen that displayed the stimuli which remained in the field of view during the whole experiment. Further, the subjects were advised to try to avoid eye blinks and body movements especially in the feedback interval of 3s. The timings for the speller task were set to 900ms for the stimulus intensification time and 100ms stimulus deactivation time during which the stimulus was set back to a neutral color. Adding these times resulted in a inter-stimulus interval of 1s between consecutive stimuli. After the speller phase, the screen was set to black for 3s followed by the feedback phase which displayed the BCI response for 1s. The completion of all 5 blocks of condition 1 resulted in EEG data of 30 erroneous feedback epochs (6 for each block) and 90 correct feedback epochs (18 for each block) per subject. For condition 2 42 erroneous and 126 correct epochs were collected. A typical experiment took around 2 hours to complete including preparation time.

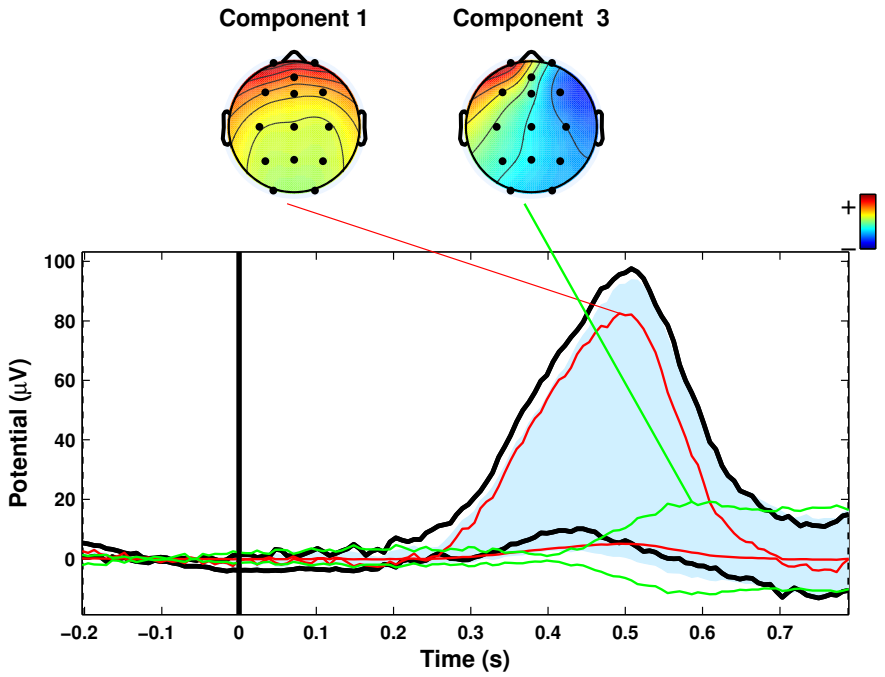


Figure 5.2. The two largest ERPs separated out by ICA are clearly localized towards the frontal part of the head which coincides strongly with eye blinks (component 1) and eye movements (component 3). The blue shaded area depicts the potential variance covered by those components.

5.3 ErrP offline analysis

As a first step, the data were analyzed offline in Matlab using the EEGLAB toolbox [Delorme and Makeig, 2004]. The data were analyzed with respect to artifacts, mean ERPs, single trial and cross subject ERP phase variability. A first inspection was devoted to identify and clean up artifacts present in the data. On average 23% of all epochs were contaminated with artifacts stemming from eye blinks. This ratio increased in the second experimental condition to 36%. In the case of one subject who participated in the second experiment, 80% of the trials were contaminated with blink artifacts. The eye blink artifacts were equally distributed throughout both conditions of error and correct trials. This increase might be related to the increased task difficulty coupled with the psychological stress to restrain from blinking while staying focused on the task.

5.3.1 Preprocessing

As an initial preprocessing step the data were highpass filtered to mitigate slow potential drifts which usually occur due to sweating and impedance changes of the skin-electrode connection. To reduce blink artifacts in the data an independent component analysis was

computed to split the raw EEG data recorded at different sensor locations into statistically independent signals. This is a widely used technique to reduce typical EEG artifacts like muscle activity or eye movements and blinks. It was shown by [Delorme et al., 2007] that ICA can be used to efficiently reduce those mentioned artifacts without significantly distorting the underlying EEG data which effectively raises the signal-to-noise ratio (SNR). The analysis of the raw 14-channel EEG data resulted in 14 independent components, with two of them exposing a very high amplitude and typical topographic localization towards the frontal part of the skull (see Figure 5.2). Since these components strongly correlate with artifactual eye blinks and movements they were rejected from the data by setting their corresponding weights in the unmixing matrix to zero. A backprojection resulted in a cleaned up dataset which was used for throughout the rest of the analysis. This preprocessing step was applied for both low- and high task difficulty conditions. Further, all epochs were bandpass filtered with corner frequencies of 3 and 20Hz.

5.3.2 Analysis of error vs. correct trials

The epochs of each experimental setting (high- and low task difficulty) were analyzed with respect to known ErrP characteristics like the Ne and Pe component, the spatial location of their maximum amplitudes, trial-to-trial variability and cross subject variability. For the low task difficulty condition it turned out to be fairly difficult to reproduce the results mentioned in [Ferrez, 2008]. Morphologically a strong negative peak around 250-350ms after the onset of an erroneous feedback stimulus was expected. For a correct feedback it was unclear what brain response was to be expected. Considering the the similarity to the given task to focus on the target letter, a correct feedback (i.e. the target letter) might elicit a P300 as well. To test whether there exists a significant statistical difference between the mean amplitudes of the correct and error condition a t-test was computed for all samples along the time axis. Data of all subjects were used as input and thus the results were expected to reveal time intervals and channel locations common to all subjects with significant difference between both conditions. For the experiment with low task difficulty, a rather weak statistical difference between erroneous and correct epochs could be detected. The significance values for each timestep and channel are drawn in the matrix of figure 5.3 **(B)**. The plots below and on the right side of the color coded matrix visualize the row and columns means of this matrix, i.e. the mean statistical significance per channel for each timestep. All colored cells in this matrix correspond to $space \times time$ pairs that passed a two-tailed paired t-test at a 0.01 significance level. Therefore, all these cells show at what time for each channel a significant difference in mean amplitudes between erroneous and correct trials exist. For the low task difficulty experiment **(B)** the most significant areas are found primarily in parietal and occipital locations with a latency of 250-300ms and 400-500ms after stimulus onset. Areas with weaker statistical significance are also found towards frontal and central sites with maximas occurring around Fz and Cz. These areas are activated at 350-400ms and 500ms after a stimulus has been presented. When taking the mean significance values along the channel axis into

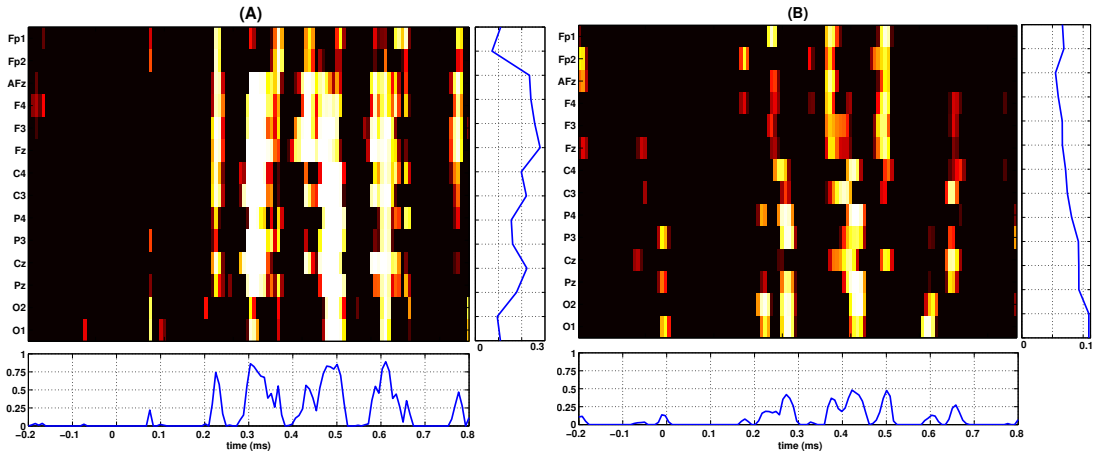


Figure 5.3. The plots show color coded significant differences between erroneous and correct trials. Brighter values correspond to higher significance. The side plots show the cross-sectional average along their respective axis. Differences between erroneous and correct trials in the high task difficulty condition (A) are much more pronounced than in the low task difficulty condition (B). For (B), most significant areas are around occipital and central areas while (A) shows highest significance over frontal areas.

account, it can be observed that the channels with highest statistical difference are the occipital followed by parietal and central areas. In general, highest significances were expected to occur at frontal parts of the head where the amplitude of error-related potentials is usually high and ErrPs were not expected to occur in correct trials. On the other hand, during correct trials which show the expected feedback which was also focused at the beginning of the speller matrix presentation, a P300 component was likely to occur due to the similarity to the prior speller task. Therefore additional significant differences around central to parietal/occipital sites could be expected. From the values shown in **(B)** it is rather difficult to prove a high correlation between the hypothesis and the actual measurement. Frontal parts did pass the t-test but most only slightly above significance level. Since the t-test only measures statistical significant deviations from the mean it does not answer the question how stable the onset latency of possible error-related components are To measure the degree of latency uniformity, i.e. the degree of time jitter from trial to trial, the inter-trial coherency **(ITC)**[1] can be computed which is solely based on phase angles of the signal and independent of the signal amplitude. A frequency 5Hz was used in the analysis since the associated ITC values were maximal compared to the remaining frequency band. Especially for the time intervals of interest around 200-400ms [Ferrez, 2008, Dal Seno et al., 2009] the phase coherency ranges from 0.15 to almost 0. The blue shaded area of these plots depict the 0.01 significance level. Since the ITC values of the erroneous trials never exceed this level it could be confirmed that there was no significantly latency-stable ERP component within the erroneous trial con-

[1] sometimes referred to as ***phase-locking factor***

dition. For the correct trial condition a slightly significant consistency could be detected at the Fz site at 200-250ms after stimulus onset. The result was equal for the lateralized sites around Fz which in turn coincides with the t-test seen in figure 5.3 **(B)**. It should be noted however, that due to the low number of erroneous trials in this setting the statistical evaluation should be taken with a grain of salt. It is possible that minor differences could not be recognized due to the low number of samples.

Exactly the same experiment has also been conducted with more stringent time constraints by lowering presentation intervals and thereby increasing the task difficulty. Afterwards the same analysis as explained above has been conducted. The results for this high task difficulty experiment are visualized in 5.3 **(A)** and figure 5.4 (bottom). Here the statistical difference between error and correct conditions are much more explicit with prominent peaks at latencies of 200-250ms (prefrontal to central), 300ms (frontal to occipital), 400ms (frontal), 500ms (frontal to occipital) and 600ms (frontal to occipital) after stimulus onset. For the erroneous condition at Fz location, ITC values do by far exceed the 0.01 significance level from 200ms onwards with a much larger spectral power increase after stimulus onset compared to the low task difficulty experiment. Even without statistical analysis it can be observed in the single-trial visualization (Figure 5.4 bottom) that a amplitude modulation with similar latencies in response to an erroneous stimulus is present. This signal is composed of 2 main peaks at 300ms (negative deflection) and 400ms (positive deflection) and 2 smaller peaks at 200ms and 500ms. This is exactly what was to be expected for ErrPs which occur at 200-300ms at fronto-central locations. This also coincides with differences found in the t-test results shown in figure 5.3 **(A)**. The analysis of the correct trial condition revealed that no time-stable 5Hz components are present for this condition at a 0.01 significance level whereas the erroneous condition shows highly significant phase correlations for the 5Hz frequency. Compared to the low task difficulty experiment, no significant difference in spectral power increase could be measured for the correct trial condition as both groups only show around 1dB increase over baseline level.

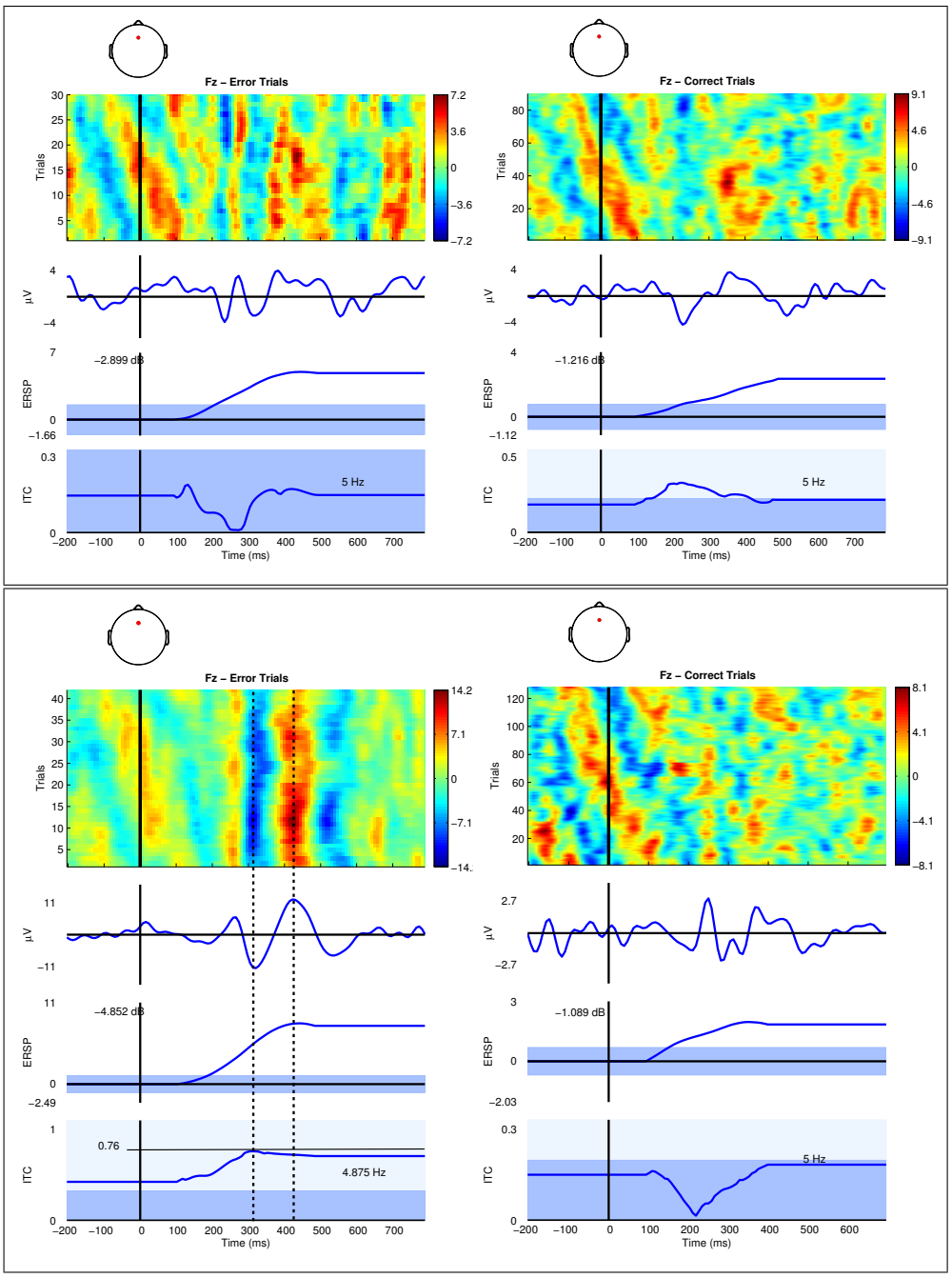


Figure 5.4. Single-trial visualization of frontal erroneous (left) and correct (right) trial ERPs for **low task difficulty (top)** and **high task difficulty (bottom)** experiments. The colored plot shows a 10-epoch smoothed visualization of single trial amplitudes. In addition, an event-related spectral perturbation (ERSP) and inter-trial coherence (ITC) plots are shown below. The blue shaded areas denote the 0.01 significance level which has to be exceeded in order to be regarded as a significantly stable time interval with respect to the stimulus onset.

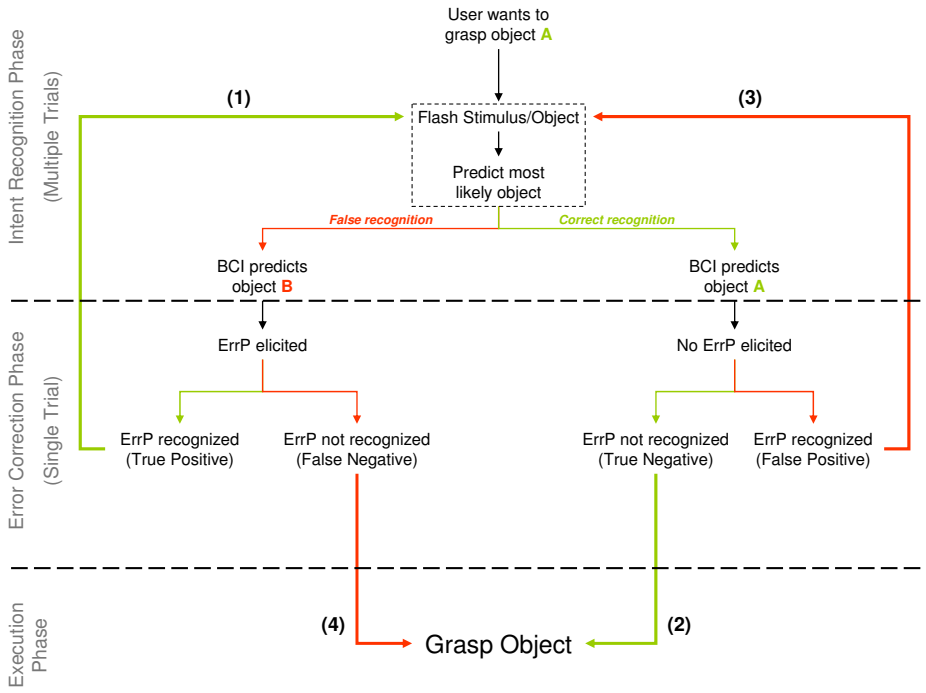


Figure 5.5. Schematic view of an error correction method using error-related potentials. Green paths correspond to increased or unchanged performance, whereas red paths denote performance loss.

5.4 Automatic correction of erroneous trials

In the case of the BCI developed in this thesis, an error correction method could be useful to prevent unintended grasping of objects and the need to correct for these mistakes. Manual correcting of such error would be require the user to drop the object at a certain location and instructing the system to grasp the object again just to put it to the correct location. Error related potentials could serve as a *last chance feature* to decide if the BCI prediction is really correct. Normally the BCI's prediction is visually displayed once before an action is carried out. If the EEG activity is significantly different between incorrect and correct predictions and if this activity can be recognized with only one sample of EEG data, then the action can either be carried out or stopped. A schematic overview of the proposed error correction system is shown in figure 5.5. Beneath the *Intent Recognition* layer, an additional *Error Correction* layer is introduced which receives EEG data as input that corresponds temporally to the point in time where the BCI prediction is presented to the user. On an incorrect prediction, an ErrP should be elicited. In order for ErrPs to be usable in an online BCI, a method to classify erroneous trials using only one single EEG epoch (single-trial) is required. Depending on the reliability of the classification, certain costs are involved with the outcome. In figure 5.5 these costs are visualized by red and green paths. Green paths correspond to no performance loss

or even improvement compared to the standard classification procedure without error correction, whereas red paths denote cases where a loss of performance is involved. Section 5.5 will investigate the performance costs inclined with the depicted paths (1)-(4) and propose an analytical method to compute the performance of such a BCI system.

The task of implementing such a system and automating the ErrP recognition can be split into three different sub-tasks:

Preprocessing A process that aims to convert the raw data into a suitable representation and improve the SNR of the data with methods that can also be applied in real-time in an online BCI.

Feature Extraction Based on the preprocessed data, specific features that discriminate well between classes are extracted.

Classification The classification problem involves assigning class labels to the input data consisting of the previously computed feature vectors.

Based on the results of the offline analysis, combinations of popular preprocessing, feature extraction and classification methods will be compared in this section.

5.4.1 Preprocessing

Common to all method combinations, the continuous EEG data were bandpass filtered to 1-10Hz and segmented into epochs from 100ms pre-stimulus to 800ms after stimulus onset. For each epoch a baseline correction was conducted with the data of the 100ms pre-stimulus interval. This was done in order to remove linear trends that might still be present after the filtering. As a first step the data were inspected with respect to obvious artifacts like eyeblinks or muscle activity. Even though the subjects were instructed not to blink especially not during the feedback phase, frequent eye blink artifacts were found which primarily occurred during the pre-stimulus time and near the end of the epoch. No muscle artifacts could be identified throughout the whole dataset. Every epoch with obvious artifacts exceeding $100\mu V$ was rejected from the dataset.

Downsampling In this step, the *downsampling* method was subject to evaluation. The idea is that the original data dimensionality can be reduced without any significant loss of information since the lowpass filtered data can be equally well represented by a lower number of samples per time interval. This reasoning is a direct result of Nyquist's theorem [Nyquist, 1928] which states that a signal with maximum frequency f has to be sampled at a frequency of at least $2 \cdot f$ to capture the full frequency content. Due to the lowpass filter which limited the maximum frequency to 10Hz from the original 256Hz, a dimensionality reduction factor of $\frac{10Hz}{256Hz} = 0.391$ can be achieved which equals almost 61% smaller input data. A lower dimensional dataset is highly preferable since any of the classification methods will benefit from lower dimensional input data given the limited number of input data for each class.

5.4.2 Feature extraction

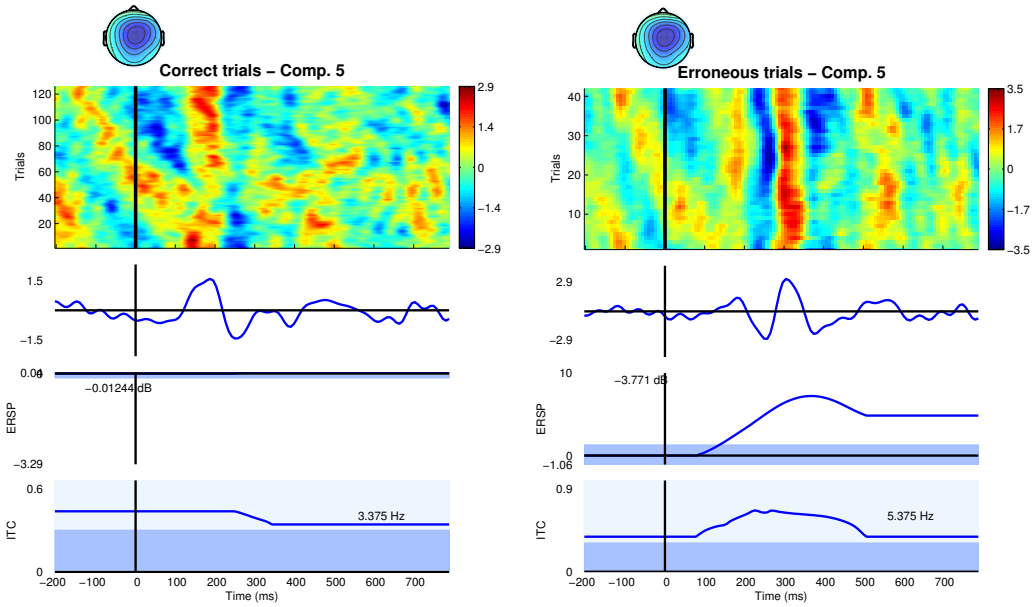


Figure 5.6. ERP image of subject 1 for condition *Correct* (left) *Error* (right) visualizing ICA component number 5.

The feature extraction step aims at finding and extracting patterns in the data that discriminate better between classes than the raw data alone. As methods, independent component analysis (ICA), principal component analysis (PCA) and t-test are considered in the evaluation.

ICA features ICA was chosen since it decomposes the signal into a set of statistically independent sources. The resulting signals can be interpreted as a decomposed linear mixture of EEG components that are characteristic for the measured time interval. Experimental results (e.g. [Jung et al., 1998], [Callan et al., 2001], [Jung et al., 1999]) showed that event-related activity is usually assigned to one specific independent component while the remaining signal activity like artifacts and noise are assigned to different components. As a result, the SNR of the relevant signal is improved.

To receive a transformation that decomposes the mixed signal, a projection matrix $\mathbf{W}$ is computed that minimizes the gaussianity of the data and transforms the sensor space data $\mathbf{X}$ into independent component space $\mathbf{S}$ or more formally

$$\mathbf{S} = \mathbf{W}\mathbf{X}. \tag{5.1}$$

As ICA algorithm, extended *Infomax ICA* [Bell and Sejnowski, 1995] was used which computes the unmixing matrix by minimizing the mutual information of the data projected on all axes. The resulting component activations show that it should be possible to discriminate between both conditions. One example is shown in figure 5.4.2 which visualizes the properties of component 5 of subject 1 who participated in the high task difficulty experiment. Especially the epochs of the error condition show a stable signal phase angle over consecutive trials which can be seen as straight vertical color gradients whereas the trial-to-trial variability of the correct condition is much higher. Further, the middle plot shows no significant increase in spectral power for the *correct condition*. This fact indicates that after a correct stimulus there is no significant change in the signal amplitude of any frequency compared to the baseline time interval. The highest phase coherence could be measured for the 3Hz frequency which is highest during baseline and extends to half the post-stimulus time interval. This finding corresponds to the analysis of the original sensor-space data in section 5.3. Therefore it was known that highly discriminable time intervals can be found from 200-500ms after stimulus onset. But since the former analysis was based on the combined dataset of all subjects, an automatic method to find the most discriminable intervals is likely to find better intervals since slight subject dependent latency variations can be expected.

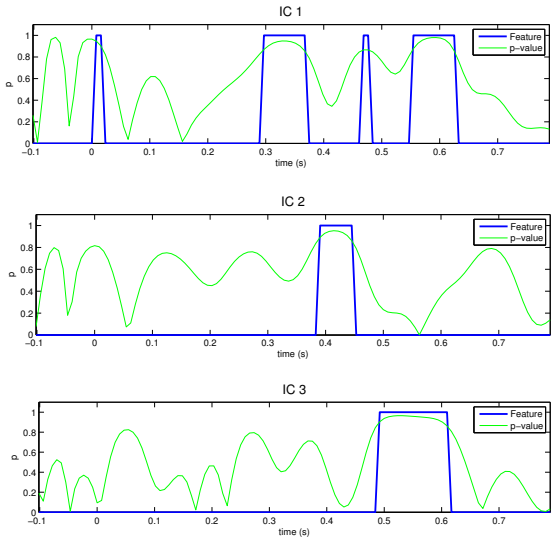


Figure 5.7. First 3 ICA non-artifactual components of subject 2. A timepoint is selected as feature when the p-value exceeds 0.99, i.e. the 0.01 significance level. The selected regions correspond to those significant regions found in the offline analysis.

t-test features The encouraging results of [Visconti et al., 2008] motivated for a similar procedure to identify regions of interest that could be used to reduce the dimensionality of the data while retaining the discriminable dimensions. For each epoch, channels were concate-

nated into one single feature vector. A data point was marked as a possible feature candidate if the null hypothesis of equal means using a two-tailed t-test at significance level α was rejected. The remaining data point indices were retained and were used to cut out the significant regions of the epochs in the dataset. The features found by this procedure partly matched what could be expected from visual inspection of the independent components and thus lends itself to be combined with ICA.

PCA features To compare the performance of the proposed t-test method a second feature extractor was added based on *principal component analysis (PCA)*. We have been using this method in our previous P300 experiments on time domain sensor space data with good success [Kaper, 2006, Lenhardt, 2006, Lenhardt et al., 2008, Finke et al., 2009]. Every 14×115 epoch in the training set was concatenated to form a 1×1610 vector. Applied to the full dataset, a $n \times 1610$ matrix , with n being the number of epochs in the training set, was obtained. PCA was computed on this set resulting in a $n \times n$ matrix since all principal components higher than n are essentially zero. The number of PCs were chosen based on the amount of variance they account for. The cutoff was set to $1 - \alpha$. Multiple values for α were tested.

5.4.3 Classification

The primary classification method used was *Linear Discriminant Analysis (LDA)* (c.f. 4.2.3). This method and its variants have been used in many studies (e.g. [Kaper, 2006, Visconti et al., 2008, Lotte and Cuntai, 2009] and in a comparison of classification methods [Krusienski et al., 2006]), LDA was among the best methods . Usually an LDA classifier is trained using many epochs of P300 and non-P300 epochs (e.g. 100+). In the special case of ErrP training it is not possible to collect such an amount of data for erroneous epochs since they occur at a very low rate during the experiment. Additionally, even more erroneous epochs would be required than P300 epochs to get a better estimate of their statistical properties. The reason for this is that for P300-speller tasks it is usually acceptable to accumulate epochs over multiple trials and classify on their mean, i.e. they don't rely on single-trial classification. In the case of error-detection using error potentials, the input data consists of only one epoch since the classified result is shown to the user only once. Due to the small number of training samples (usually in the order of 20-50) compared to the dimensionality of more than 1500, normal LDA cannot be used since the covariance estimate needed for the calculation of the weight vector becomes singular. To improve the covariance estimates, shrinkage of the covariance matrices has been selected as regularization method. To regularize the sample covariance matrix defined by

$$\mathbf{S} = \mathbf{X}\mathbf{X}^T / n \tag{5.2}$$

a linear combination of the type

$$\Sigma^* = p_1 \mathbf{I} + p_2 \mathbf{S} \tag{5.3}$$

has to be found which minimizes the expected quadratic loss and thus is optimal with respect to

$$\min_{p_1,p_2}\left[E\left[\left\|\Sigma^* - \Sigma\right\|^2\right]\right] \tag{5.4}$$

Finding these two parameters is a sensitive task. It is possible to search the parameter space for a good solution. This however requires extensive computation and it is impossible to know whether the found parameter set is optimal with respect to the real covariance matrix. The work of Ledoit and Wolf [Ledoit and Wolf, 2004] proposed an analytical method to compute optimal shrinkage parameters for the given sample covariance matrix. Ledoit and Wolf proved that the shrunken covariance matrix is always invertible, well-conditioned and more precise than the sample covariance estimator. The estimator for a $(p \times n)$ dataset is defined by

$$\mathbf{S}^* = \frac{b}{d} m_n \mathbf{I} + \frac{a}{d}\mathbf{S} \tag{5.5}$$

with

$$m_n = tr\left(\left\|\mathbf{S}\mathbf{I}^T\right\|^2\right)/p \tag{5.6}$$

$$d = \left\|\mathbf{S} - m_n\mathbf{I}\right\|^2 \tag{5.7}$$

$$b = \min\left(\frac{1}{n^2}\sum_{k=1}^{n}\left\|x_k x_k^T - \mathbf{S}\right\|^2, d\right) \tag{5.8}$$

$$a = b - d \tag{5.9}$$

and $\|\cdot\|^2$ being the squared Frobenius norm with its associated inner product

$$\mathbf{A_1} \circ \mathbf{A_2} = tr(\mathbf{A_1}\mathbf{A_2^T})/p. \tag{5.10}$$

5.5 Utility metric for ErrP error correction

In the recent literature (e.g. [Visconti et al., 2008], [Dal Seno et al., 2009] or [Takahashi et al., 2010]) it is common to assume that the correction of erroneous trials using ErrP reduces the communication time between the user and the BCI. This assumption is based on the fact that upon a successful recognition of an erroneous trial, a correction command (e.g. *backspace* for a spelling device) can be issued automatically without the need to explicitly communicate it to the BCI via brain signals. However, when analyzing the general structure of such an error correcting BCI as depicted in figure 5.5, it becomes obvious that the initial assumption is dependent on many factors. The figure describes the control flow of a BCI for robot grasping tasks in three phases. Only the top and bottom phase consume time which correspond to presentation of stimuli (*Intent Recognition Phase*) and the time it takes to execute the grasp

action (*Execution Phase*). Green paths denote the optimal flow of information while the red paths correspond to communication overhead which equals increased time to complete an action.

In the case of path **(1)**, an incorrect command was predicted by the intent recognition layer that would lead to a wrong object being grasped. But since the error correction method correctly classified the wrong prediction, the system returns to the stimulus presentation state to gather more data for a more accurate prediction. Thus, the accuracy is increased and the time spent to communicate the intended object is decreased since no manual correction (i.e. dropping the grasped object, reselecting it and dropping it at another location) is required. Path **(2)** is equivalent with the method without error correction since both the object and the ErrP have been correctly classified. No performance gains or losses are involved. The third path **(3)** however can be devastating to the performance and presumably to the motivation of the subject. If the user's selection was classified correctly but an ErrP was incorrectly detected, the system returns to the stimulus presentation state which will double the time needed to communicate the symbol even though it was correct in the first place. A similar issue arises for path **(4)**. If an ErrP is present but is not recognized, the action will be carried out which requires manual correction afterwards by the user. However, since this additional cost is also present when no error correction is applied, no additional performance loss compared to the former method is entailed. As a consequence, only path **(3)** can cause a significant performance loss compared to a method without error correction. An error correction classifier with 100% false negative rate and 0% false positive rate would be equivalent with no error correction at all since path (3) will never occur and path (4) will be entered on every occurence of an ErrP.

The former described structure can be interpreted in a more analytical and general way. In fact, the whole process can be modeled as a stochastic process with BCI actions as states and classifier performance values as transition probabilities between those states. Such a process is depicted in figure 5.8 which resembles mainly to the structure shown in figure 5.5. State **A** corresponds to the stimulus presentation step which obviously takes time to present all stimuli. Depending on the outcome of the P300 classifier, the system makes a transition to state **B** upon an incorrect P300 classification or state **C** in case the P300 classification is correct. From both states **B** and **C** a path is leading back to the stimulus presentation. In case of state **B**, an ErrP will be elicited since a wrong command was recognized by the previous intent recognition step. Thus, a true positive result of the ErrP classifier can be used to automatically return to the stimulus presentation instead of conducting a wrong action which has to be corrected afterwards. As a result, the time until a sequence of correct commands is communicated is reduced compared to the standard paradigm. On the other hand, if the system's state is **C** which corresponds to a correctly classified command, an ErrP classifier that erroneously finds an ErrP (i.e. false positive) will lead to an increased communication time by going back to state **A**. The same rule applies for the next state transitions. Compared to illustration 5.5, the action state here has been split into two different states even though both of them conduct the action and consume the same time. The reason will be explained in

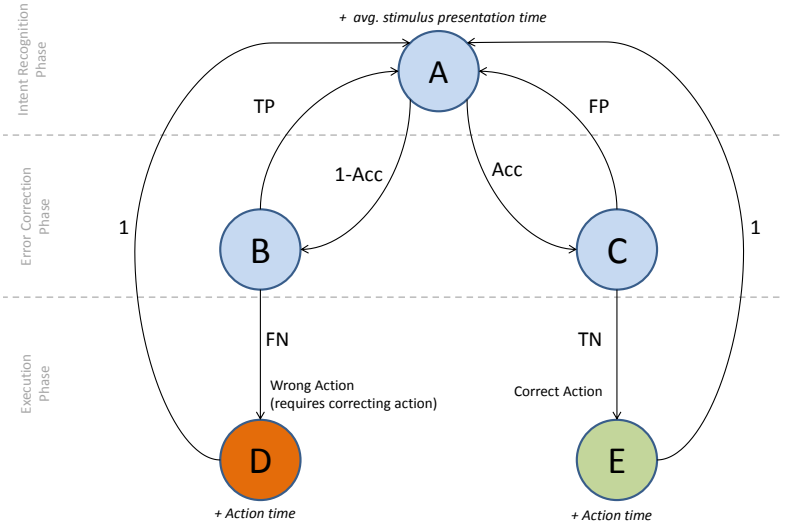


Figure 5.8. An error correcting BCI modeled as a stochastic process. State A, D and E consume time when entered while states B and C are intermediate states that immediately transit to either A, D or E. Transition probabilities are acquired from the P300 classifier (Acc) and the ErrP classifier's confusion matrix (TP, FP, TN and FN).

the following. Using this structure, an error correcting BCI can be simulated by issuing many trials and iteratively adding action times and stimulus presentation times. To simulate this process, certain parameters have to be known:

- P300 classification accuracy (Acc)
- ErrP confusion matrix (TP, FP, TN, FN)
- Action time (T_{act})
- Average stimulus presentation time (T_{stim})

The simulation will end once the given number of correct actions has been reached, i.e. whenever the emitted sequence of states contains a certain number of state **E**. The average time it takes to conduct a correct action can be easily calculated by dividing the total time by the number of simulated trials. For realistic parameters, it is possible to find a solution for the total time per trial. However, since the structure contains multiple feedback loops, certain parameter constellations might converge very slowly or not converge at all if stuck in a feedback loop. An extreme example would be an ErrP classifier with 100% false positive rate. In this case, the system would return to the stimulus presentation on every correctly predicted action and hence would never converge.

Instead of simulating the process with an iterative method, the whole structure can be solved analytically. The graph in figure 5.8 represents a *discrete-time random process* with Markov properties, which means that future states are independent of the the previous states and thus can be solved using the *forward Chapman-Kolmogorov equations* [Papoulis et al., 2002]. The state probability vector of a Markov chain can be defined as

$$p(n) = \begin{pmatrix} [p(n)]_1 \\ [p(n)]_2 \\ \vdots \\ [p(n)]_m \end{pmatrix} \tag{5.11}$$

where each $[p(n)]_i$ denotes the probability that the system is in state i after the $n-th$ transition. Further, the single-step probability matrix is defined by

$$P = \begin{pmatrix} p_{11} & \cdots & p_{15} \\ \vdots & \ddots & \vdots \\ p_{51} & \cdots & p_{55} \end{pmatrix} \tag{5.12}$$

with p_{jk} being the probability that the next state will be k given the current state is j. If the state probability vector at time n is known, the probabilities of every state i can be computed by the *forward Chapman-Kolmogorov equation*

$$[p(n+1)]_i = \sum_{k=1}^{5} [p(n)]_k p_{ki} \tag{5.13}$$

which leads to the recurrent computation of all states with

$$p(0) = \begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}^T, \text{ arbitrary stochastic vector} \tag{5.14}$$

$$p(n+1) = P^T p(n) \tag{5.15}$$

The goal with this formulation is to find the *steady-state vector*

$$\pi = \lim_{n\to\infty} p(n)^T \tag{5.16}$$

which corresponds to the occurrence probability for each state given an infinite number of trials. The *ergodicity theorem* claims that for any ergodic Markov chain, a limiting state distribution or *steady-state vector* π exists which is unique and does not depend on the initial state $p(0)$. Thus, the steady-state vector can be approximated by simply raising the single-step probability matrix P to a high power n with

$$\pi \approx \prod_{i=1}^{n} p(i)^T = p(0)^T P^n \tag{5.17}$$

Depending on the transition probabilities, π might converge after 20 steps or it might take 2000 steps. Raising P to very high powers π might converge in most cases but it is not ensured. A more elegant way to compute π is to treat the problem as one of solving the set of linear equations represented by

$$\pi P = \pi \tag{5.18}$$

This means that π is a left eigenvector of P with eigenvalue 1. Hence, the vector π can be found by

$$\pi = eig(P^T) \tag{5.19}$$

In the above equation, P is transposed to get the left eigenvector assuming that the eig function usually corresponds to the right eigenvector of a matrix. However, the left eigenvector of a matrix is the same as the right eigenvector of the same transposed matrix. One last thing to consider is the scaling of π. Since π as calculated above is an eigenvector, $\mu\pi$ is also an eigenvector. The scaling depends on the implementation of the solver and thus cannot be predicted. But since we know that the steady-state vector must sum to 1, π can be easily rescaled by

$$\pi' = \frac{\pi}{\sum_i \pi_i} \tag{5.20}$$

The resulting vector contains the probabilities for each state occurrence for an infinite number of simulation runs. Thus, it can be interpreted as percentages for each state of the whole set of entered states. The usefulness of this vector becomes obvious when it comes to the question, how long it will take for a BCI with given parameters to correctly recognize n commands without errors including correcting actions. As mentioned in the beginning, time is only consumed when state **A**, **D** or **E** are entered since these correspond to stimulus presentation and action execution. The steady state vector however contains all five states which means that the time consuming states have to be extracted into a new vector which has to be rescaled. Thus, the final vector which contains the relative occurrence frequencies of the time consuming states will be defined as

$$\Pi = \frac{1}{\sum_{i=\{1,4,5\}} \pi'_i} \begin{pmatrix} \pi'_1 \\ \pi'_4 \\ \pi'_5 \end{pmatrix} \tag{5.21}$$

The average total time T_{total} for n correct trials can be computed by rescaling Π such that $\Pi_3 = n$ and computing the dot product between Π and a vector containing the trial and action times as shown in 5.22.

$$T_{total} = \frac{n}{\Pi_3} \Pi^T \begin{pmatrix} T_{stim} \\ T_{act} \\ T_{act} \end{pmatrix} \tag{5.22}$$

At this point it becomes clear why the structure depicted in figure 5.8 deviates from figure 5.5 by splitting the *action state* into two different states. If the probability Π_3 corresponding

to state **E**, i.e. a correct action, equals 0 then the scaling factor $\frac{n}{\Pi_3} = \frac{n}{0} = \infty$. As a result, the average time to execute a single correct action T_{total} goes towards infinity and thus a correct action will never be achieved. Hence, the reason for splitting the action state is to allow to keep track of correct and wrong actions.

The proposed analytical solution of the convergence time is useful for formulating a performance metric that compares communication speed differences between two BCIs with known parameters.

A metric Θ to measure the percental difference is shown in 5.23, with T^A_{total} being the total communication time per correct symbol for one BCI and T^B_{total} the time for another BCI with different parameters.

$$\Theta = \frac{T^B_{total}}{T^A_{total}} \tag{5.23}$$

A value of Θ greater than 1 corresponds to an improvement in communication speed of A over B, while values smaller than one imply speed loss. The percental speed gain/loss of A over B can be quantified by simply setting

$$\Theta_{100} = \left(\frac{T^B_{total}}{T^A_{total}} - 1 \right) \cdot 100$$

with positive values corresponding to the percental speed increase of BCI A over BCI B.

5.5.1 Accounting for manual corrections

The above formulation of the problem considers only the time between two successive correct actions. For many real applications this solution might be sufficient. On the other hand, there may be tasks which require a correction of erroneously recognized commands. An example would be a spelling device with the aim to enter a 4 digit pincode to get money from an ATM. This BCI also offers a special `Backspace` command, to delete the last spelled letter. If the entered pincode contains a wrong character or has a different length, the ATM will not give out any money. Given the correct pincode is `ABCD`, entering those 4 characters would require 5 correct commands in succession. Realistically though, the BCI will produce some errors during the spelling of the sequence which might result in the sequence `ABCX`. In this case, the user will try to correct the last letter by entering the `Backspace` command and try to spell the correct letter once again. The sequence of commands to spell the pincode correctly might consist of `[A B C X Backspace D]`. It is obvious that this sequence is 2 commands longer than in the optimal case. As a general rule, each wrong command results in at least 2 additional commands which have to be correctly recognized (deletion and reentering the correct command). Of course the above example sequence is not the only possible one. In fact, a sequence like `[A Z Backspace B C X K O Backspace P Backspace Backspace Backspace D]` would also produce the output `ABCD` but is significantly longer

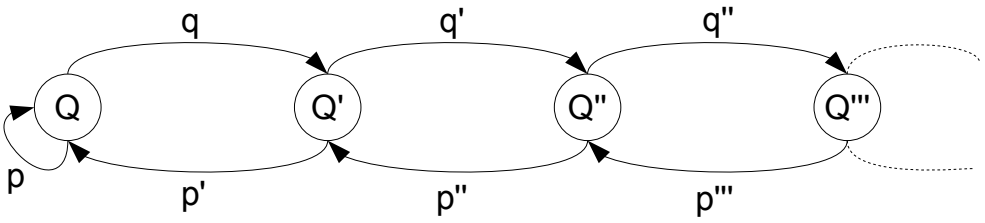


Figure 5.9. To model manual corrections, an infinite Markov chain can be constructed with each state **Q** denoting the chain as depicted figure 5.8. Transition from one state to the previous one via p happen whenever a correct classification occurs. Contrary, on every wrong command the next **Q** is entered.

than in the previous example. The question here is, how long such a input sequence is in the mean for a given output sequence of length n .

The proposed Markov chain can be extended to account for these correction actions. For this, the model depicted in figure 5.8 will be denoted by the state **Q**. When translated into a state graph, the correction process can be modeled as an infinite chain of **Q**'s, where each **Q** will transit to the next state **Q'** whenever a wrong command occurs, i.e. state **D** is entered. On every correct command, i.e. state **E**, the model will transit from the current state $\mathbf{Q}^{(n)}$ back to the previous state $\mathbf{Q}^{(n-1)}$ and thus jump up in the nesting level. A graphical representation of this extended model is shown in figure 5.9. This infinitely deep structure generally equals a *bounded 1D random-walk*. The corresponding transition matrix for this chain is defined by

$$M = \begin{pmatrix} p & p & 0 & 0 & \cdots \\ q & 0 & p & 0 & \cdots \\ 0 & q & 0 & p & \cdots \\ 0 & 0 & q & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} \tag{5.24}$$

with

$$\begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} \Pi_3 \\ \Pi_2 \end{pmatrix} \text{ and } p+q=1 \tag{5.25}$$

An analytical solution of the problem can be obtained in the following way.
From 5.24 at $n=1$ follows

$$\pi_1 = p\pi_1 + p\pi_2 \tag{5.26}$$

$$\pi_2 = \frac{1-p}{p}\pi_1 = \frac{q}{p}\pi_1 \tag{5.27}$$

Assuming

$$\pi_k = \frac{q}{p}\pi_{k-1} \quad \text{for } k = 2,3,\ldots,n$$

equation 5.24 yields

$$\pi_n = q\pi_{n-1} + p\pi_{n+1} \tag{5.28}$$

$$\Rightarrow \pi_{n+1} = \frac{1}{p}\left(\pi_n - q\pi_{n+1}\right) \tag{5.29}$$

$$= \frac{1}{p}\left(\pi_n - q\frac{q}{p}\pi_n\right) \tag{5.30}$$

$$= \frac{q}{p}\pi_n \tag{5.31}$$

$$\Rightarrow \pi_k = \frac{q}{p}\pi_{k-1} \quad \forall k > 1 \qquad \square \tag{5.32}$$

from which follows that all state probabilities π_n can be expressed as

$$\pi_n = \frac{q_{n-1} \cdot q_{n-2} \dots q_0}{p_n \cdot p_{n-1} \dots p_1}\pi_0 \tag{5.33}$$

$$= \prod_{i=1}^{n-1} \frac{q_i}{p_{i+1}}\pi_1 \tag{5.34}$$

$$= \left(\frac{q}{p}\right)^{n-1}\pi_1 \tag{5.35}$$

Further assuming that the system is currently in state $\mathbf{Q}^0$, a successful trial will consist of any sequence of the form

$$\mathbf{Q}^0, \dots, \mathbf{Q}^{n_i}, \dots, \mathbf{Q}^0 \qquad \text{with each } n_i \in 1, ..., \infty$$

The expected length i of this sequence, i.e. the average number of visited $\mathbf{Q}$ states until the system returns to the initial state $\mathbf{Q}^0$, can be calculated by

$$\pi_{corr} \equiv 1 + \frac{q_1}{p_1} + \frac{q_1 q_2}{p_1 p_2} + \dots \tag{5.36}$$

$$= 1 + \sum_{k=1}^{\infty} \prod_{i=1}^{n-1} \frac{q_i}{p_{i+1}} \tag{5.37}$$

$$= 1 + \frac{q}{p} + \left(\frac{q}{p}\right)^2 + \dots + \left(\frac{q}{p}\right)^n \qquad \text{since } p_i = p_{i+1} \forall i \quad \text{and} \quad q_i = q_i + 1 \forall i \tag{5.38}$$

Equation 5.38 corresponds to the geometric series

$$1 + r + r^2 + \dots = \sum_{k=0}^{n-1} ar^k \qquad \text{with} \quad r = \frac{q}{p} \tag{5.39}$$

and thus the final solution converges to

$$\pi_{corr} = \frac{1}{1 - \left(\frac{q}{p}\right)} \tag{5.40}$$

Intuitively, the error correction process will only converge if the probability for a correct classification p is higher than for an erroneous classification q which is also a constraint for the geometric series that convergence is only achieved for $\frac{q}{p} < 1$.

Informally speaking, the first term of equation 5.38 corresponds to the first state of the sequence $\mathbf{Q}^0$ which will be entered in any case. Each subsequent term $\frac{q}{p}$ denotes how many q transitions occur for each p transition. Thus, equation 5.40 returns the average number of **Q** states which are entered until the process returns to the initial state Q, i.e. the average sequence length between two states **Q** including the initial state and excluding the final **Q**. This means that the total time required to enter a sequence of n commands without errors is

$$T_{corr} = n\pi_{corr} T_{total} = n\frac{T_{total}}{1 - \frac{\Pi_2}{\Pi_3}}. \tag{5.41}$$

5.6 Experimental results

In this section results for the error-potential detection are presented based on data of 5 subjects for each of the 2 conducted experiments (i.e. low- and high task difficulty). Each dataset was tested using RLDA, SVM and k-Nearest Neighbor and for each of these classification methods a set of feature extraction and preprocessing methods was tested. Performance of the respective combination of methods was assessed in a 5-fold crossvalidation where any step that required training to some extend (i.e. ICA or t-test features) was carried out only on the fold containing the training data. Prior to each crossvalidation run, the data were balanced to contain the same amount of epochs from both classes. The balancing of classes is necessary to prevent overfitting of the classifiers since the error probability in the experiments was only 25% which means that the whole data set contains 75% epochs of the P^- class (containing no ErrP) and only 25% epochs of the P^+ class (containing an ErrP). This step however causes the performance results to differ for each crossvalidation run due to the random selection of epochs from the P^- class. To mitigate this effect and assess a more reliable result the balancing followed by a crossavlidation was carried out 10 times and classification accuracies were collected for each fold. Using a bootstrap sampling approach, the 0.01 confidence interval for the mean accuracy was determined after all 10 crossvalidation runs were completed. In addition, standard deviation and classification accuracies for each class were computed as an additional measure. The complete listing of all assessed measures can be found in the appendix (tables .1 - .5). In this table, the mean accuracies reflect correctly classified ErrP epochs vs. non-ErrP epochs. The classification threshold was chosen to minimize false positives and false negatives. For reference, the preprocessing combinations enumerated in the former table are given below:

The accuracy differences among the top methods are minor but RLDA slightly outperforms the SVM classifier as can be seen in figure 5.10. As for the preprocessing, table 5.1 summarizes the mean accuracies across subjects for the RLDA classifier. Methods (1)-(4) from the ICA category, method (7) from the Raw and method (11) from the PCA category are at the

(1)	ICA+Raw+DS
(2)	ICA+Raw+NoDS
(3)	ICA+TTest+DS
(4)	ICA+TTest+NoDS

(5)	Raw+Raw+DS
(6)	Raw+Raw+NoDS
(7)	Raw+TTest+DS
(8)	Raw+TTest+NoDS

(9)	PCA+Raw+DS
(10)	PCA+Raw+NoDS
(11)	PCA+TTest+DS
(12)	PCA+TTest+NoDS

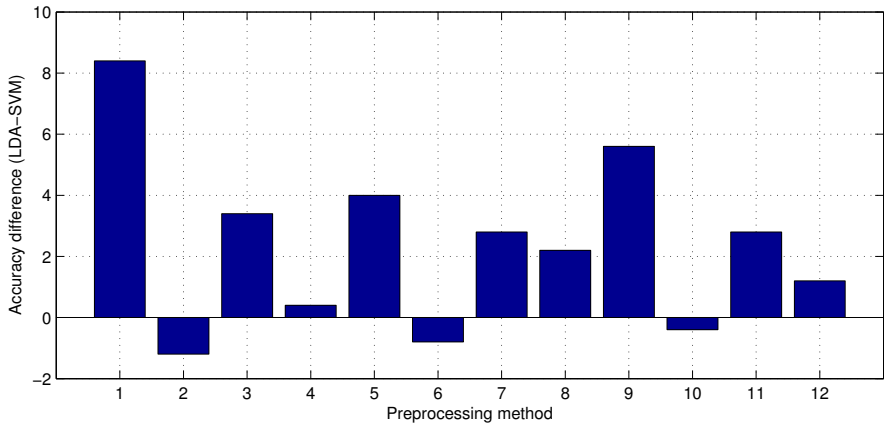


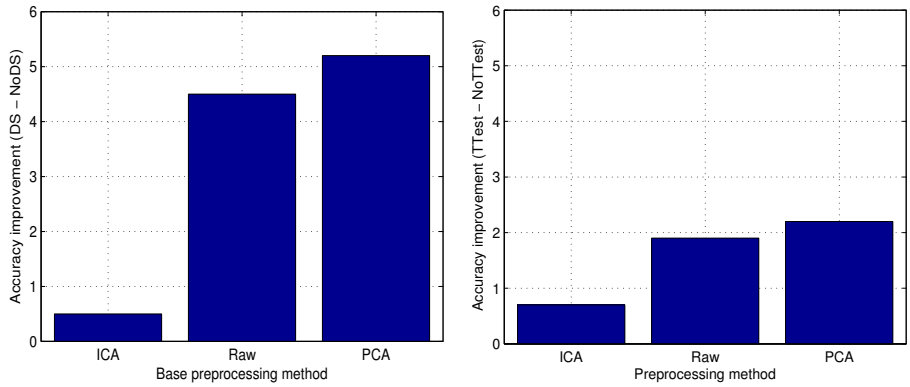
Figure 5.10. Mean accuracy (across subjects) difference between LDA and SVM classifier. Higher values correspond to LDA advantages, whereas negative values denote better performance of the SVM.

same level of accuracy. The worst performing ones are method (6) which is equivalent with no preprocessing except the initial low-pass filtering and method (10) which corresponds to the method mentioned in [Kaper, 2006]. A pairwise comparison of the methods with and without *downsampling* showed (figure 5.11(a)), that for the ICA based approaches, no significant improvement could be achieved. Both, Raw and PCA based methods showed a 4-5% improvement in accuracy. The results in this plot were computed by taking the average of all ICA based methods with downsampling and subtracting the average accuracies of the same methods without downsampling. Performance differences for Raw and PCA methods were computed in the same way.

In practice a high true positive ErrP detection, while keeping a low false positive rate, is more important than overall accuracy for reasons which are explained in section 5.5. Since actual classification is nothing more than assigning class labels depending on a scalar threshold, the true positive rate (TPR) and false positive rate (FPR) can be tuned by modifying this threshold. To highlight the classifiers performances over the full spectrum of classification thresholds, ROC curves of the two best performing methods (red and green curves) and the

Method	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Acc %	85.6	85.2	86.4	85.8	83.6	78.8	85.2	81.0	83.8	77.6	85.0	80.8

Table 5.1. Mean LDA accuracy across all subjects for the 12 preprocessing combinations.



(a) Accuracy improvement due to downsampling.

(b) Accuracy improvement due to t-test.

original method from [Kaper et al., 2004] (blue curve) which was also used in [Lenhardt, 2006, Finke et al., 2009] are depicted in figure 5.11. Method (2) and (7) clearly outperform the original method proposed by Kaper for all subjects. The differences between the two best performing methods is minor. For subject 2, t-test feature selection and downsampling achieved better results than the ICA based method, while for subject 5 this result is reversed. For the remaining subject, both methods are equivalent in terms of accuracy.

To investigate the theoretical impact of the proposed error correction method, a simulation of an experiment has been conducted consisting of multiple runs of 10000 trials. P300 and ErrP classifiers were simulated by setting their confusion matrix (i.e. TPR, FPR, TNR, FNR)to fixed values for each run. The time needed to correctly classify all 10000 trials was assessed. This measure was assessed with error correction (EC^+) and without error correction (EC^-). The important parameters according to section 5.5 were set to values which can be expected for a real experiment:

- ISI: 200ms
- Number of stimuli: 8×8 matrix, i.e. 16 stimuli
- Average subtrials until classification: 6
- ErrP classifier FPR rate: 0.1
- Average stimulus presentation time $T_{stim} = ISI \cdot Stimuli \cdot Subtrials = 19.2s$
- Time to execute action $T_{act} = 15s$

The stimulus presentation time was set to $19.2s$, a value that corresponds to an average of 6 subtrials until a prediction is made by the BCI, an ISI of $200ms$ and a number of 16 stimuli which resembles an 8×8 stimulus matrix. The FPR rate of the simulated ErrP classifier was

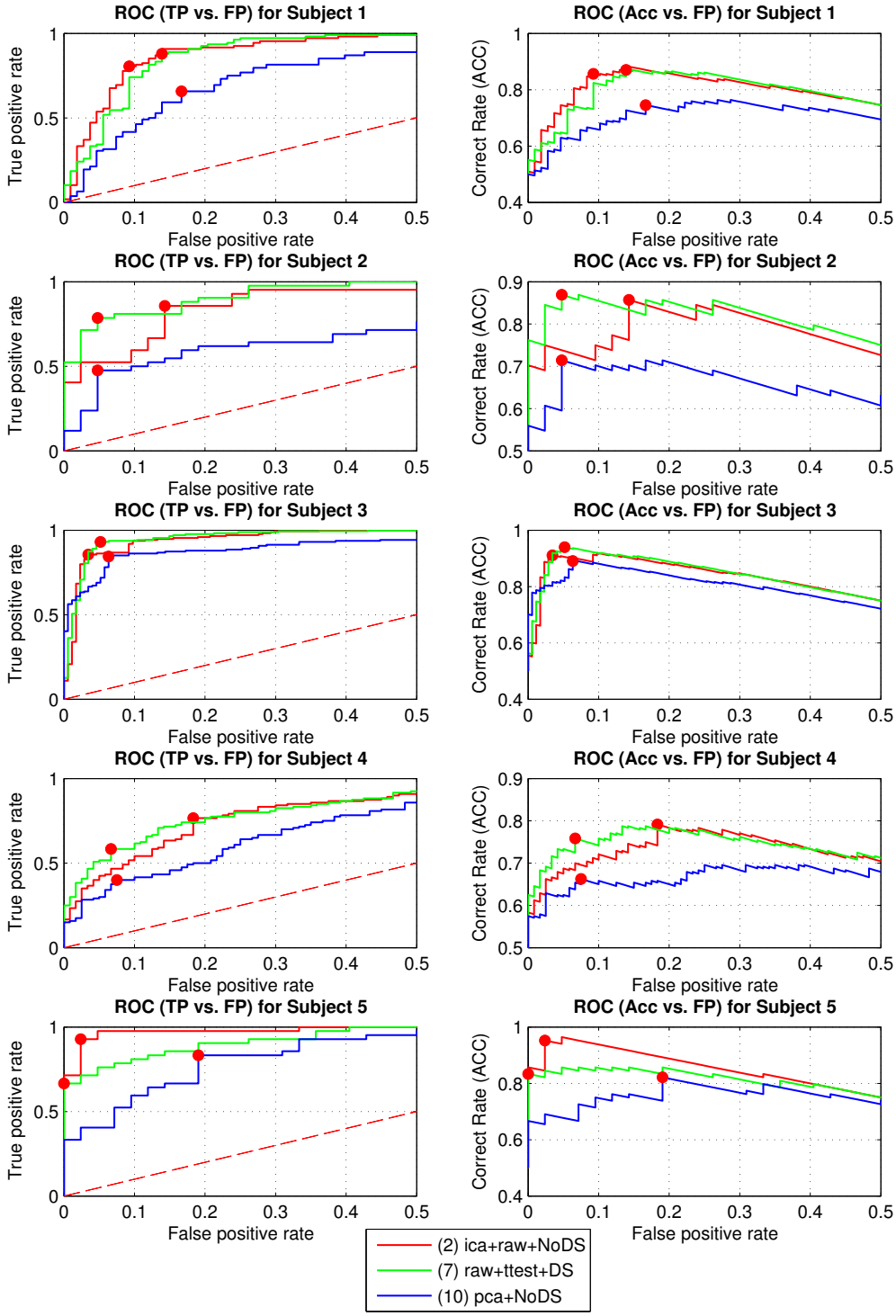
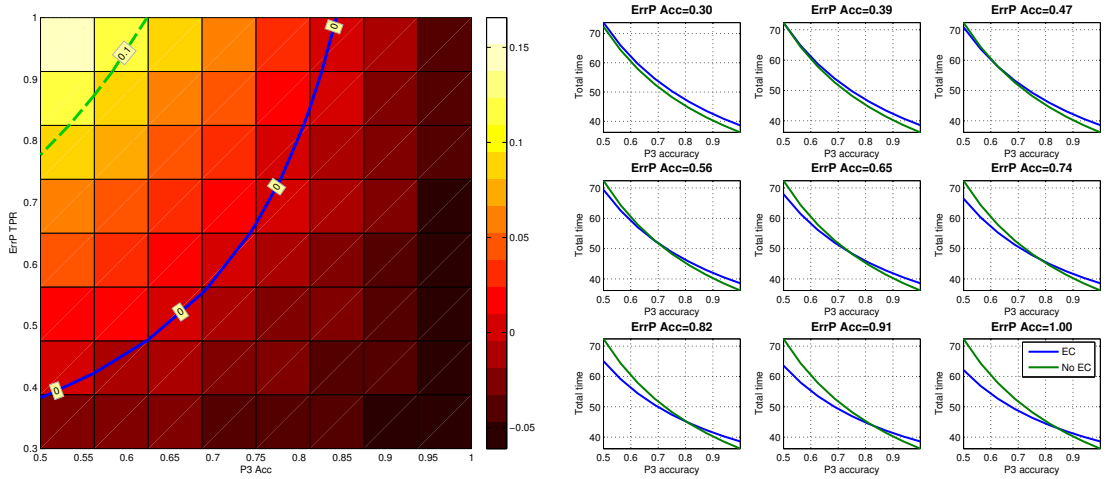


Figure 5.11. ROC curves for the *high task difficulty condition* for all 5 subjects.



(a) Visualization of Θ_{100} over different P3 accuracies and ErrP true positive rates. Bright values correspond to a communication speed advantage of the error correction method over the standard method without error correction. The blue line marks the break-even point where both methods perform equally well. In addition, 10% and 20% speed difference boundaries are shown which can be considered as significant improvement or degradation areas.

(b) Communication speed comparison of total time versus P3 accuracy for both methods given 9 different ErrP true positive rates. Lower values correspond to faster communication speed. The curve intersections correspond to the blue line in plot (a).

Figure 5.12.

constraint to 10% throught the whole simulation. The TPR parameter of the ErrP classifier and the overall accuracy of the P300 classifier (assuming equal amount of false negatives and false positives) were varied from 50-100% (P300 Classifier) and 30-100% (ErrP classifier) resulting in a matrix of time measurements for the respective parameter combinations. The false positive rate of the ErrP classifier was fixed at 10% which means that according to the ROC curves presented in figure 5.11, true positive rates of more than 70% can be expected in a realistic setting.

Figure 5.12(a) shows the percental communication speed improvement of the error correction method over the standard method without error correction. The values were computed using the utility metric presented in 5.5 and hence are a visualization which depicts Θ_{100} over multiple parameter values. The blue line in the figure marks the break-even point where neither performance improvement nor performance loss is evident between both methods. For all points above this line, a communication speed improvement can be expected in the long run. The dashed green line denotes the start of regions which result in > 10% difference in communication speed.

According to this, the proposed error correction method offers a slight improvement of up to 5% if the P300 classification accuracy drops below 65%, the ErrP TPR is greater than 70%

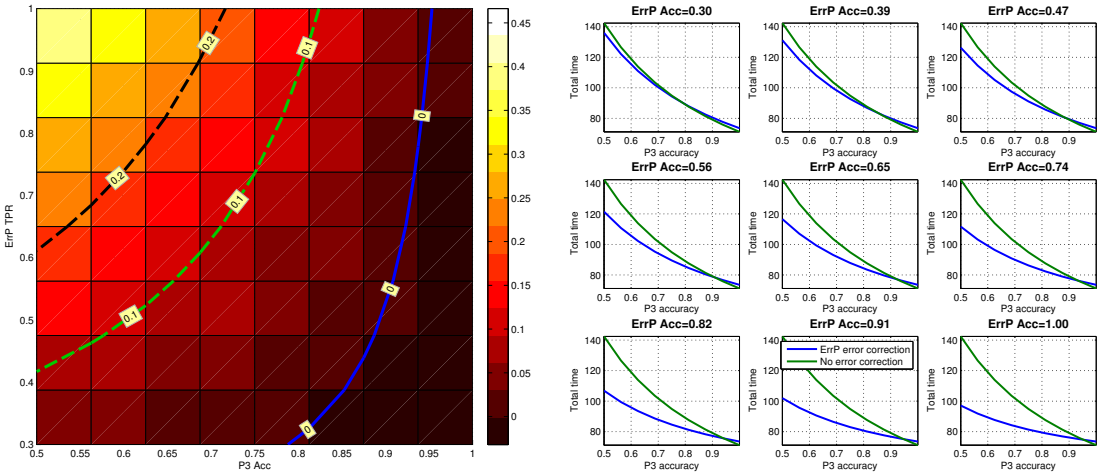


Figure 5.13. Action time increased to 50s. The same remaining parameters as in figure 5.5 were used. Time factors for variable P3 accuracy and ErrP true positive rates. Lower values (bright values) correspond to fast communication speed. The blue line marks the break-even point where both methods perform equally well. In addition, 10% and 20% boundaries are shown which can be considered as significant improvemnt or degradation areas.

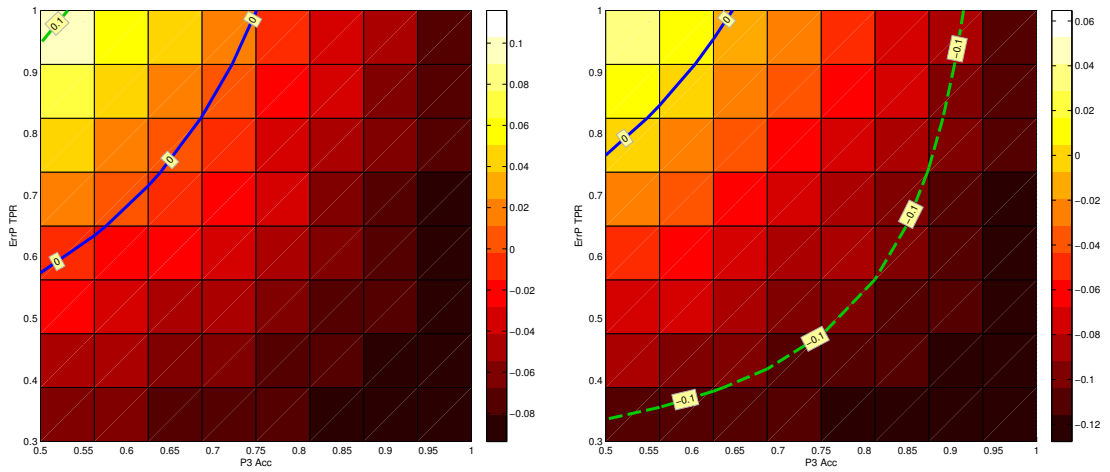
and the FPR is does not exceed 10%.

With regard to the previously reported single-trial accuracies for the ErrP detection in figure 5.11, the method can be considered very useful in certain cases. An example would be the BCI controlled wheelchair (e.g. [Iturrate et al., 2009] or [Rebsamen et al., 2007]). Both authors were using the P300 signal to commit commands to a wheelchair that drives to a specific location. Naturally the action time, i.e. driving to the location, can take a significant amount of time compared to simple grasping of an object with a robot arm. When the action time is increased to 50s, a different picture emerges as seen in figure 5.13.

One last thing left to prove is the hypothesis, that the FPR of the ErrP classifier is the parameter that can lead to significant performance degradation. In the beginning of this section it was shown, that the FPR controls path (2) (Figure 5.5), a feedback connection which causes the system to use more subtrial presentations than necessary and is not present in a BCI without error correction. And indeed, increasing this parameter to 15% (Figure 5.14(a)) and 20% (Figure 5.14(b)) of the initial 10% and computing Θ_{100} over a range of parameter values results in a serious performance drop as can be seen in figure 5.14(a) and 5.14(b).

5.7 Conclusion

In this chapter, an experiment to evoke error-related potentials has been conducted and the differences of the EEG patterns occurring after incorrect BCI predictions to EEG patterns evoked in response to correct prediction have been analyzed. Based on these findings, an



(a) ErrP FPR set to 15%.

(b) ErrP FPR set to 20%.

Figure 5.14. Increasing the false positive rate of the ErrP classifier leads to significant communication speed loss of the BCI.

error correction method has been proposed that could potentially help to improve the communication speed of BCIs. A classification method has been developed that is able detect the presence of an ErrP using only one EEG epoch as input data. Based on this method, 12 combinations of preprocessing and feature extraction methods have been compared which revealed that ICA based preprocessing and downsampling combined with a t-test for feature selection outperformed the remaining methods with mean accuracies of 85% across all 5 subjects. Given that P300 classification accuracy using an average over multiple subtrials usually resides in the range of 80-90% accuracy, this can be considered as a surprisingly good result. The data obtained from the ROC curves in figure 5.11 showed that accuracies of $85-90\%$ are possible to achieve while even keeping the false positive rates under 10%.

In addition, the feasibility of error correction has been studied using a probabilistic model of a generic BCI with automatic error correction. For BCI's without error correction, the average time to communicate a command can be calculated very easily since the BCI's control flow can be described as a simple feed-forward structure which depends only on the classifier's correct rate. This task becomes more complicated for BCI's with automatic error correction which introduces internal feedback loops. Using the proposed model it is now possible to predict the time between consecutive correct commands and thus compute the average time the BCI needs to correctly communicate an arbitrary number of commands. The model accounts for the internal feedback loops and is able to predict in which cases the BCI will get stuck in the error correction feedback loops. It was also shown, that the average time to spell an error free sequence of commands, which requires deleting wrong commands and communicating the correct one afterwards, can be described as a discrete-time infinite Markov chain. This advanced model accounts for manual user corrections of

erroneous BCI commands. As an example, spelling the word [HELLO] might be the result of the command sequence [H,E,L,L,Z,Backspace,I,Backspace,O]. From this sequence of 9 commands, only 2 are erroneously recognized commands (Z and I). For every wrong command, a Backspace has to be entered to correct for the mistake. In this sense, the infinite Markov chain can predict the time required to enter the the word HELLO while also considering the time required for the correction of erroneous commands. Using this theoretic framework it is now possible to directly compare the communication time for different BCI's. In contrast to the usual measure of *bits/sec*, the outcome has more practical relevance since knowing the raw bits/sec carries no information about the practical communication speed in terms of time to communicate a sequence of commands correctly.

In a simulation study, the performance difference between a BCI with error correction and without error correction has been investigated. P300 detection accuracy and ErrP true positive rates served as free parameters to highlight their effect on the overall communication time. The results showed, that ErrP true positive recognition rates must rise with P3 classification accuracy if a performance improvement should be achieved. For tasks where the action time is long, e.g. for wheelchair navigation similar to [Rebsamen et al., 2007] where the action ends when the wheelchair arrived at the selected destination, error correction is highly feasible since it mainly avoids excessive execution of wrong actions. Further, the experiment validated that the *false positive rate (FPR)* of the ErrP classifier is the parameter that can cause a BCI with error correction to perform significantly worse than without error correction. The usage of the proposed method extends even into the performance optimization domain. Since the global system performance can be predicted for any classifier threshold (i.e. all possible TPR/FPR combinations), a parameter set can be found that minimizes Θ and thus minimizes mean communication times. The search space for the TPR/FPR combinations consists of the ROC curve of the respective ErrP classifier which can be obtained from the training set.

A limitation of this model is the assumption that the input data to for the BCI do not change over time, or more specifically that classification accuracy remains constant over time. Fatigue of the user or an increasing lack of concentration might lead to degraded classification performance in the intent recognition and error correction phase which might invalidate the model parameters.

To summarize, the goal of implementing an ErrP based error correction system for BCIs has been achieved. Important aspects of the feasibility of the error correction system have been studied which resulted in the development of a probabilistic model to compute the BCI's communication time per correct symbol. The use of this theoretic framework might become more evident in the future for optimizing classifier thresholds in a complex BCI and as a comparison measure whose accuracy does not suffer from low population sizes as it is common in real experiments. The effect of the involved parameters on the overall performance has been investigated which will prove to be useful for other researchers when it comes to optimize their own BCI's performance. A single-trial classification method has been proposed which performed reasonably well with more than 80% correct rate in an offline setting. In conclu-

sion it can be ascertained, that tasks which involve long action execution times, automatic error correction offers a great communication speed improvement. Thus, the proposed system might be of great value for other researchers who apply ErrP based error correction with P300 BCIs in a setting which involves long command execution times. When optimizing the system with respect to communication time for correct commands, the probabilistic framework presented in this chapter will be useful for adapting the classifier's threshold. This problem which was usually solved by choosing a point on the ROC curve with heuristic optimality constraints as in [Takahashi et al., 2010] while the presented method is able to analytically determine the optimal operating point and even accounts for manual corrections.

6 Augmented Reality-based BCI

This thesis was motivated by the question how a P300-based BCI can be used to efficiently control reaching and grasping movements of robotic actuators. A solution for this problem requires that the term *efficiently* has to be clearly defined. When talking about an efficient brain controlled robotic actuator it should imply that

1. Operation of the system should add as few additional cognitive load as possible
2. Operation should be intuitive and effortless
3. No subject training should be required
4. Translation of mental commands should be fast and reliable

In the following, the design of a novel system to control robotic devices in a natural way by brain signals will be presented. Initial design decisions and principles regarding the employed paradigm will be discussed as well as the subcomponents involved in the whole process of detecting the users intentions down to the low-level control of the robotic endeffectors.

6.1 System design

When evaluating the existing approaches as presented in section 3.3.2 with respect to these claims, it turns out that none of them is able to fully comply with these propositions. A reasonable assumption is that controlling many degrees of freedom also requires increased effort from the user to be able to command all of them in a coordinated manner. However, this would contradict the first claim since control over many degrees of freedom implicitly allocates higher cognitive resources to control all DOF with respect to a goal. This raises the question how the process can be simplified for the user without sacrificing functionality of the system.

Two popular strategies are employed in modern BCIs which control artificial actuators. As reviewed by [Wolpaw, 2007], control strategies can be distinguished by *Process Control* and *Goal Selection* (see figure 6.1). Thereby, process control strategies aim for direct control of the available motors or muscles while goal selection approaches focus on recognizing the intent of a user and delegating low-level control to an autonomous subcomponent. The intent

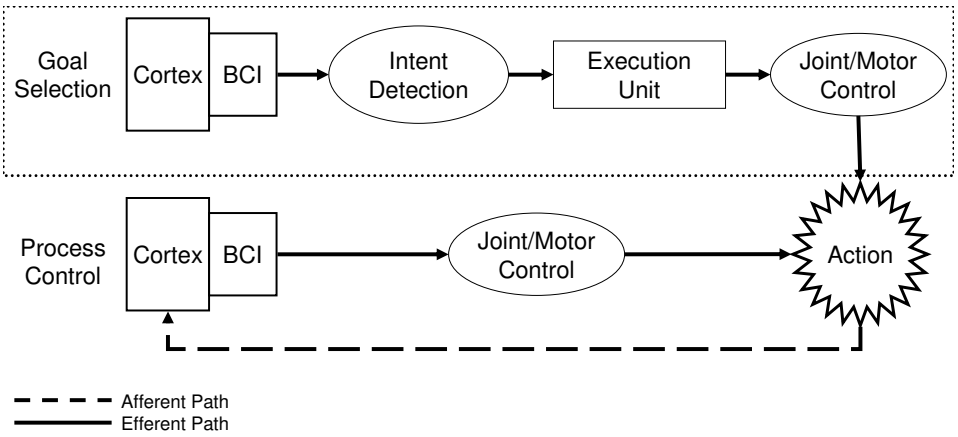


Figure 6.1. Robotic control via brain signals can follow two different control strategies. Process control offers the most flexibility since a closed feedback loop between action and the brain is established with full control over the robots movements. Goal selection approaches offer limited flexibility in that only *intent* is extracted from brain signals while motor control is delegated to soft- and hardware subcomponents.

serves as input for an execution unit which translates it to a sequence of motor actions. Obviously, the goal selection approach does not require the user to achieve full control over the output device via brain signals. A rather simple *intent detection* scheme would be sufficient which requires only a one dimensional selection of a target symbol corresponding to the desired action while the process control approach would require the user to be able to control at least 3 DOF to navigate freely in space. In goal selection approaches, actual control is delegated to subcomponents which are not bound to higher cognitive functions. This reflects the normal output pathways of the brain and its underlying systems for motor control. Thus, it can be stated that goal selection approaches are currently the most natural way to control robotic output devices by brain signals. A statement that Wolpaw seems confirm in his review of BCI technologies [Wolpaw, 2007]:

> Thus, with further development, including refinement and elaboration of feedback from the downstream apparatus to the CNS, the goal-selection strategy has the potential to emulate with increasing fidelity the brain's normal output pathways. In contrast, process-control BCI methods that vest control entirely in the cortex are likely to remain an artificial and fundamentally unnatural approach.

P300 event-related BCI approaches perfectly fit for the goal selection approach since intents (or actions) can be represented as single stimuli as is the case with the P300-Speller paradigm. Therefore, the decision to use a P300-based approach was the most natural choice for this thesis. A drawback of the goal selection approach is however, that the symbols which represent intents are static (i.e. they have to be pre-programmed) and thus only a limited set of intents can be represented. Further, a system that aims to allow the user to grasp objects

must implement a way to transform these objects into computer controlled stimuli and associate a reasonable set of actions with them. A possible setting could be a scene consisting of two objects, a cell phone and an apple, laying on a desk and a brain controlled robot arm with a gripper. The first problem to solve is how the intention of the user to grasp the apple can be extracted from the EEG signals. As stated above, in P300-based paradigms intents or goals can be represented as graphical stimuli that need to be selected according to the known oddball principle. In this setting, the graphical stimuli consist of real objects. Thus, computer-vision techniques are required to segment objects from background. Once the objects are recognized, they can be highlighted on a computer screen which now corresponds to the P300-Speller paradigm. Grasping the apple would additionally require the spatial position of the object relative to the robots coordinate system. This is a non-trivial task since either stereo-vision algorithms would be involved or a marker-based approach which relies on predefined markers attached to the objects. Once the apple and its position are known it is possible to use these information for a P300-Speller-like paradigm and select the apple by mere thought. The action to take on selecting the apple would most likely be grasping. An issue with context dependent actions could arise with the cellphone. It is not entirely clear what the most probable action would be when selecting the cellphone. It could either be `USE CELLPHONE` or `GRASP CELLPHONE`. A problem that might be possible to solve using AI techniques. This is, however, beyond the scope of this work and was chosen to be omitted.

To solve this task, a novel system called *Brain-Link* has been developed within this thesis that can operate in a way as described in the aforementioned example. It is structured into three different components: the vision, EEG and robot component. A schematic view of the system is shown in figure 6.2. The whole system is built in a modular way to allow for exchanging and adding subcomponents at any point of the pipeline. All subcomponents are able to communicate with each other over a network connection. This allows to distribute the components over many computers and even allows for a teleoperation scenario where the user is able to control a remote robot arm over the internet. As a general framework, BCI2000 [Schalk et al., 2004] was chosen which already offers a rich set of functionalities required for the module communication and data logging. The methology of BCI2000 can be best described as a distributed client-server architecture with a central component that takes care of all housekeeping functions like data logging, configuration propagation to connected modules, network access and module communication capabilities via global state variables. A module of this framework can be either one of three different types:

- Data Acquisition Module
- Signal Processing Module
- Application Module

Each of these modules is running as a separate process while inter-process communication is established via UDP network messages in form of state variables which can be defined by

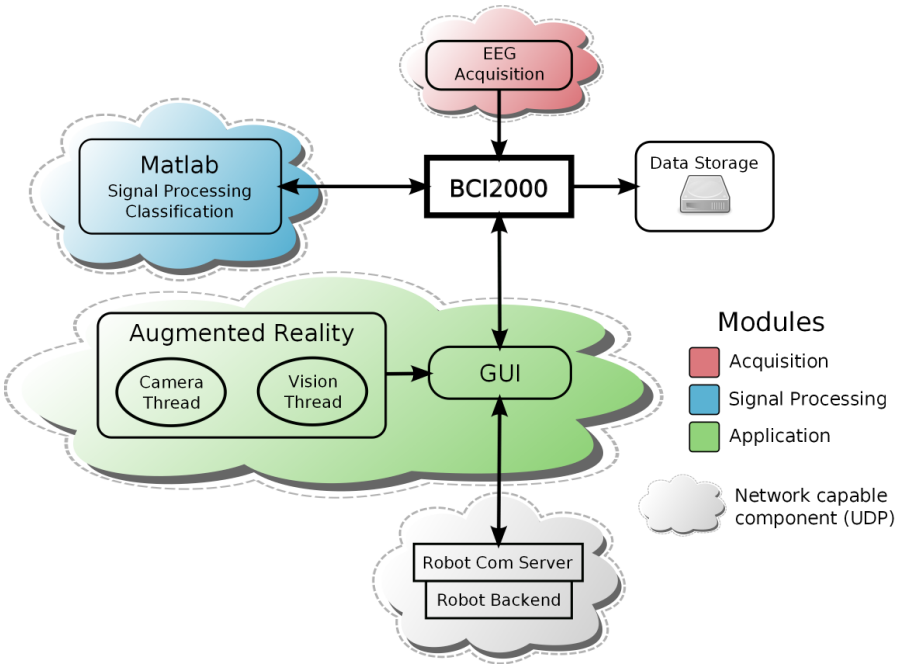


Figure 6.2. Distributed system architecture of the BrainLink system. Each cloud denotes a system component that can run on a separate machine and is able to communicate via network messages with other modules. Three main modules exist which integrate into the BCI2000 framework [Schalk et al., 2004]: Acquisition, Signal Processing and the Application (i.e. graphical interface). A fourth module, operating outside the standard BCI2000 framework is the robot control part which is implemented with the OpenKC framework [Schöpfer et al., 2010] and a UDP client-server interface.

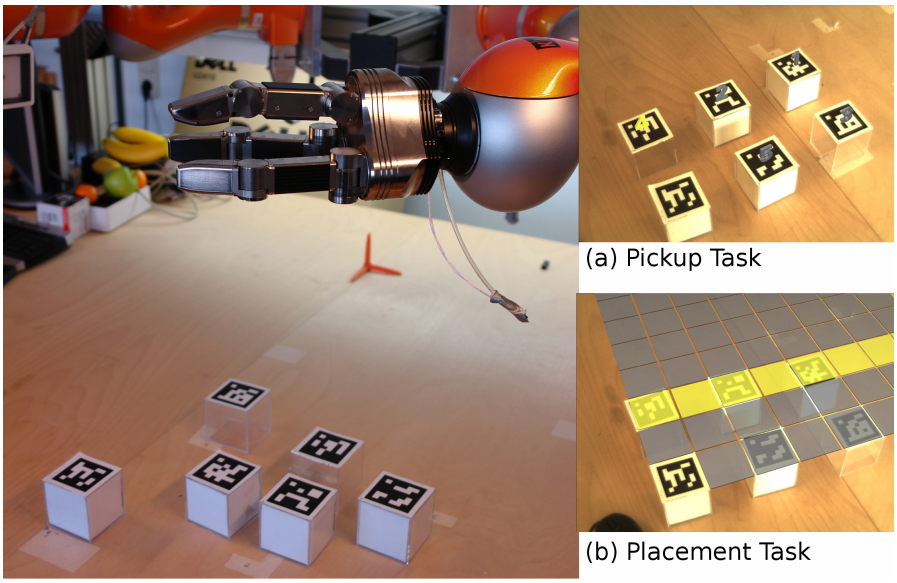


Figure 6.3. Subject view of the scene. Computer generated 3D models are augmented onto the cube objects which represent the stimuli in the augmented-reality paradigm. When a stimulus is intensified, the color is changed to yellow (top left). Neutral stimuli appear as gray clored numbers.

the modules. A further advantage of using BCI2000 is the ability to easily insert Matlab code into the module chain which speeds up development times greatly.

6.2 Augmented Reality paradigm

The most natural approach to select real objects would be to implement a way to highlight these objects. This allows for dynamic stimuli in the scene without the need to pre-define selections as in [Rebsamen et al., 2007] or to use static stimuli that serve as waypoints as in [Iturrate et al., 2009]. The novel paradigm presented in this section employs Augmented Reality techniques to augment the natural vision of the user with computer generated graphical stimuli, whic are aligned with the spatial position and orientation of the real objects. A *head-mounted display (HMD)* with two firewire cameras is used to stereoscopically display the scene in front of the user. Objects can be of any shape and form as long as they have a special marker attached that is being recognized by the vision component of the framework. The system works in two phases. The first phase is devoted to recognize a target object which has to be grasped. Figure 6.3 shows a typical scene with multiple objects distributed on a table. As stimuli, 3-dimensional numbers are augmented on top of the cubes. During a running selection task, these numbers will flash on a single-object basis. In this special case the single-stimulus highlighting scheme was favored over the grid based design since the number of stimuli is too low to gain any benefit from a grid-based design (c.f. 3.2.6). Once an

object has been identified as the intended target, the robot arm will grasp and lift the object to a specified height and the second phase is started. During this next round a 3D grid model is overlayed with the scene (Figure 6.3). This stimulus presentation shares a strong similarity to the known P300-Speller matrix but selection target (i.e. cell) can only be distinguished by their spatial location, whereas the cells of the speller matrix can be distinguished by their content (i.e. different letters) and spatial location. Stimulation is done in the same way as with the speller matrix using a multiple stimulus approach where rows and columns are flashed in random order. A recognized target cell serves as a dropoff coordinate for the robot arm which puts the already grasped object at the location coinciding with the augmented grid cell. A sequence of the involved processes is shown as a flow diagram in figure 6.4. The system is running in a synchronous fashion. Each module is waiting for input from its preceding module in the pipeline. Since the EEG data acquisition has to query the internal buffers on a regular basis to prevent an overflow, a real-time constraint is implied with the system. One full cycle of processing including EEG acquisition, signal processing, marker recognition and stimulus augmentation have to be completed within the timeframe defined by the EEG data buffer size and samplingrate. During the experiments, a block size of 15 samples at 256Hz was chosen resulting in a timewindow of roughly 60ms in which one full pipeline cycle had to be completed. On a coarser level, the sequence of processes consists of two main components driving the functionalities of the BCI. The central part of the system, the AR Paradigm, controls the behavior of the graphical frontend with respect to the classification results and also triggers robot actions by sending appropriate commands and parameters to the second main component, the robot backend. This backend module is not a designated component of the BCI2000 framework which only knows acquisition, signal processing and application modules. It consists merely of a remote server accepting incoming network messages from the AR Paradigm and performs the necessary low-level commands to control the robot according to the received command.

6.3 Augmented-Reality component

For the sake of simplicity, sophisticated stereo-vision and object recognition methods were rejected in favor of a marker based vision approach. ARToolkitPlus (ARTK+) which is based on ARToolkit (ARTK) [Kato and Billinghurst, 1999] was chosen as Augmented-Reality toolkit which offers marker position and pose estimation via monocular camera images. The system is able to identify predefined square black-white marker patterns printed out on white paper. The resulting pose matrices can be used to orient a virtual 3D object in virtual space which can be aligned with the camera image. This process of combining the camera image and the computer generated (CG) 3D object is commonly called *augmentation*. Augmenting camera images with CG objects requires a series of processing methods. With ARToolkit, these steps can be separated into 5 groups as shown in figure 6.5:

- Pixel Format Selection

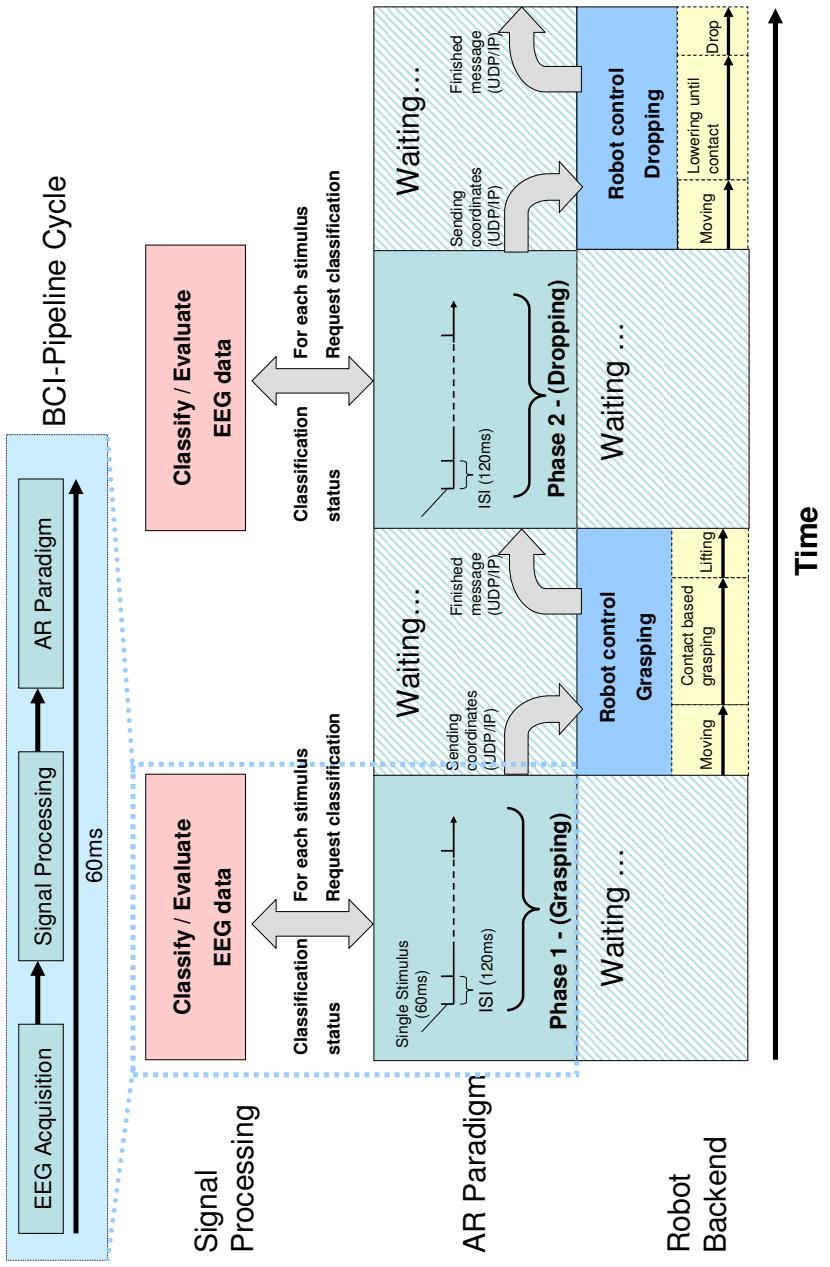


Figure 6.4. Diagram of the module interactions and timing contraints for the *Grasping Task*.

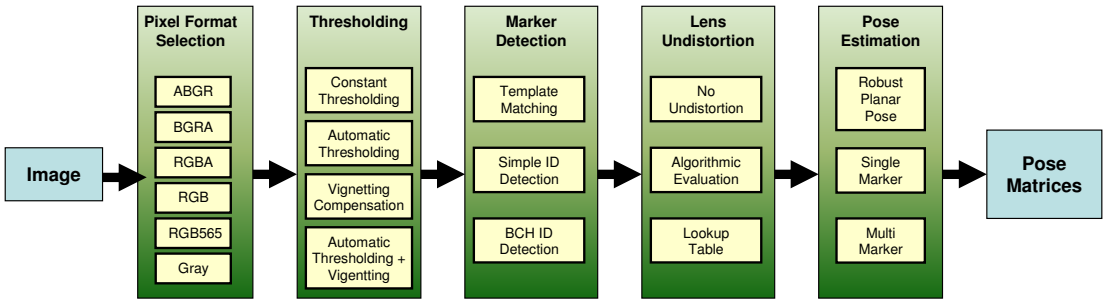


Figure 6.5. Pose calculation pipeline of ARToolkit. After converting the raw image into the required image format, thresholding is applied to get a binarized image. This image serves as input for the marker detection algorithm. In this step, the binary coded IDs of the marker are extracted using the BCH pixel patterns on the marker. Next, the image is undistorted using the camera parameters which have to be acquired in a calibration procedure. The final step is the calculation of the marker pose which is achieved by the *Robust Planar Pose* algorithm [Schweighofer and Pinz, 2006].

- Thresholding
- Marker Detection
- Lens Undistortion
- Pose Estimation

Pixel Format Selection

The process of pixel format selection is straightforward since it only requires to convert the native camera image format into one of the available formats that can be used with ARToolkitPlus. For tracking purposes luminance images are well suited since they conserve memory and computation time. Ultimately all images are converted into grayscale images inside ARToolkit but since the display on the HMD required color, the acquired images were converted to the RGB format.

Thresholding

In a next step, the acquired and converted image has to be binarized, i.e. converted into a black-and-white image. This is achieved by a thresholding algorithm which sets every pixel above a certain luminance threshold to white while all other remain black. Stationary lighting setups can be controlled in such a way as to provide very equal lighting throughout the whole scene. Such a setup usually requires special lighting equipment like diffusers and specific light angles to suppress glaring and reflections on the markers. Therefore the standard thresholding algorithm of ARTK might be sufficient which requires the programmer to set a constant threshold value for the mapping. However, as soon as daylight hits a room, the

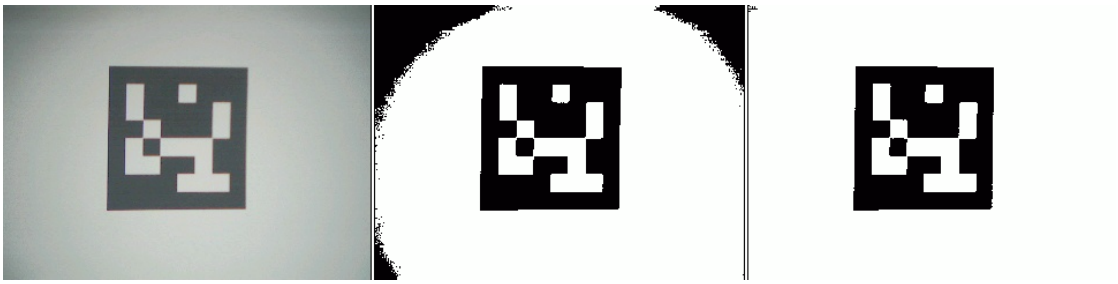


Figure 6.6. Thresholding of camera images. Left: Original image with radially imbalanced lighting. Middle: Constant global thresholding results in a radial artifact. Right: Thresholded image after vignetting compensation.

quality of the constant thresholding can suffer. Further, some cameras are less light sensitive at the outer diameters of the lens and produce a seemingly unequal lighting condition on the scene. The problem of changing lighting condition can be tackled by the use of the automatic threshold method provided by ARTK+. To improve performance, only the last seen marker is considered. The median of all marker pixels is computed which serves as a threshold for the next frame to process. If no marker is found in the next frame, the threshold is randomized for every new frame until a marker is detected. Using this heuristic it is possible to reacquire a marker within a few frames in case it got lost. A second optimization called *vignetting compensation* can be used for images showing a radial light intensity falloff. This effect can be very noticeable on certain camera types. But since the cameras did not suffer from such an effect, no vignetting compensation was used.

Marker Detection

To detect markers in the binarized image, a *connected component labeling* algorithm finds areas that are connected by adjacent pixels of the same color. This step is required to find the thick marker border which serves as input for the subsequent pattern normalization and pose estimation. Once the all connected components have been labeled the pattern is normalized and edge contours are traced. Normalization is the process of reverting the perspective distortion resulting from the patterns pose angle to the camera.

All regions with edge contours that can be fitted by four line segments are regarded as a detected marker. The four vertices resulting from the intersection of the four line segments are stored for the subsequent processing algorithms. Further, the inner section of the rectangle is extracted which serves as an identifier for the different markers. Two methods can be used for marker identification

1. Template matching (requires training)

2. Binary BCH code patterns (no training required)

Figure 6.7. Pattern normalization: a) Original pattern with perspective distortion. b) perspective corrected pattern. c) Downsampling of resolution to speedup recognition times.

In the first case of template matching, the inner pattern has to be trained by capturing a camera image perpendicular to the marker. This step has to be repeated for every new pattern that should be recognized.

A more practical solution was introduced in ARTK+ with the use of BCH coded markers which allows automatic identification of the patterns using a *syndrome decoding*[1] method. BCI codes represent a class of parameterized error-correcting codes. They require very little computation time for decoding which makes them suitable for this application. They are made up of a 2-dimensional grid of black or white pixels (Figure 6.6) which code a certain number which serves as an ID. Since this method is more flexible in terms of adding new markers in addition to the trainingless nature, it was chosen as the preferred method for this work.

Lens Undistortion

Accurate pose estimation entails the requirement for near-optimal screen coordinates which approximate a perfectly flat projection surface of the virtual camera. With physical cameras however, lens distortion effects can be observed (see figure 6.8). This is especially visible with *fish-eye* lenses. But also standard lenses show distortion effects on the edges of the image. To mitigate these effects, an initial camera calibration procedure is applied to calculate the intrinsic camera parameters which will be used to undistort the image. As seen in figure 6.8 the undistorted image will not fit the screen and therefore a scale adjustment is required to fit the whole rectified image on the screen.

Pose Estimation

Marker pose estimation is based on known sizes of the markers. A transformation $\mathbf{T_{cm}}$, as pointed out in eq. 6.1, has to be found which transforms the marker coordinates into camera

[1] An efficient method to decode a linear code over a noisy channel. An example is the well known *Hamming code*.

screen coordinates as shown in figure 6.9.

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} = \begin{pmatrix} V_{11} & V_{12} & V_{13} & W_x \\ V_{21} & V_{22} & V_{23} & W_y \\ V_{31} & V_{32} & V_{33} & W_z \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{pmatrix} = \mathbf{T_{cm}} \begin{pmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{pmatrix} \tag{6.1}$$

The transformation T_{cm} is found by an iterative process that aims to minimize

$$err = \frac{1}{4} \sum_{i=1,2,3,4} \{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2\} \tag{6.2}$$

where (x_i, y_i) represent the previously extracted vertices and $(\hat{x}_i, \hat{y}_i)$ the four vertices in ideal image coordinates (eq. 6.3).

$$\begin{pmatrix} h\hat{x}_i \\ h\hat{y}_i \\ h \end{pmatrix} = \mathbf{C} \cdot \mathbf{T_{cm}} \begin{pmatrix} X_{Mi} \\ Y_{Mi} \\ Z_{Mi} \\ 1 \end{pmatrix}, i = 1,2,3,4 \tag{6.3}$$

The transformation into ideal screen coordinates is implicitly shown in eq. 6.3 by the multiplcation of $\mathbf{T_{cm}}$ with a perspective transformation matrix $\mathbf{C}$. The latter matrix is found, along with the distortion parameters, during the camera calibration process.

As a result, the AR component calculates the position and orientation of the recognized markers with respect to the current camera coordinate frame. In the context of the BCI application however, marker positions that conform with the coordinate frame of the robotic arm are required. Since the subject is wearing the HMD and thus is mobile in general, a calculation of a transformation matrix from the user to the robot coordinate frame is not possible without further tracking equipment and calibration techniques. As a simple solution, a reference marker is placed at a known position in the robot coordinate frame and serves solely as a reference point. To determine all other marker positions with respect to the reference point, as depicted in figure 6.10, the transformation matrix which positioned the reference marker in the camera frame is inverted and multiplied by the transformation matrices of the other markers. The known position of the reference marker allows to calculate all positions of the other markers by simply multiplying their transforms with the inverse of the reference marker's transform. In practice however, placing the reference marker at the origin would not be an optimal place since the origin is too far away from the cameras and can be occluded by the robot arms. Therefore, a position closer to the cameras was chosen which is 90cm along the y-axis of the robot's coordinate frame. The resulting relative positions of the other markers have to account for this translation.

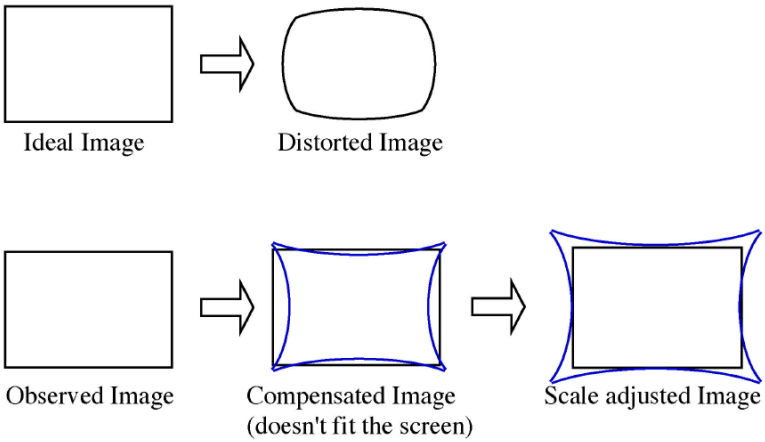


Figure 6.8. Images are compensated for lens distortions induced by the cameras to achieve accurate results of the pose calculations. Unwarping the image leaves areas with no image content. The actual image size is reduced to completely fit into the inner boundary of the unwarped image.

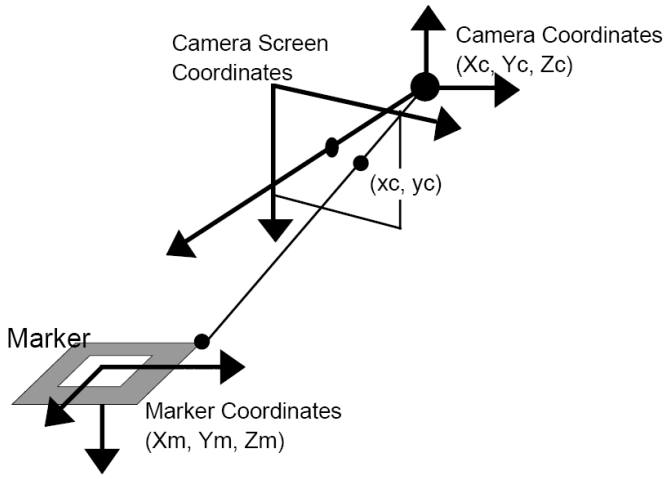


Figure 6.9. Transformation between different coordinate systems.

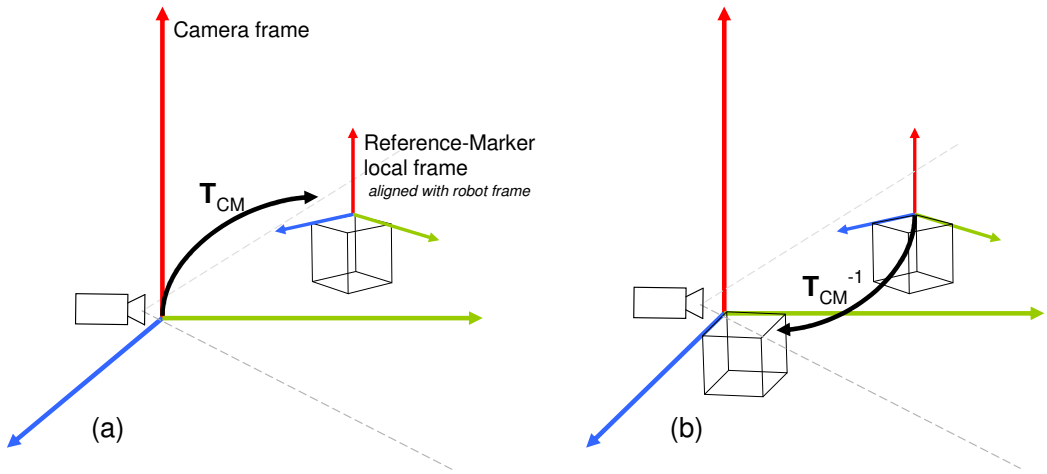


Figure 6.10. Due to the unknown head position position and orientation of the camera frame with respect to the robot frame are also unknown. **(a)** Stimuli are transformed into the robots fixed coordinate frame by using a reference marker placed at a known position in the robot frame. **(b)** The position of each marker relative to the reference marker can be derived by multiplying the markers transformation matrix $\mathbf{T}_{CM}$ with the inverse transform $\mathbf{T}_{CM}^{-1}$ of the reference marker.

6.4 EEG component

The EEG component consists of a pipeline of signal processing and classification modules which were all implemented in MATLAB. Insights from the previous experiments were incorporated and modified to yield an optimal result. The basic procedure shares strong similarities to the methods described in section 4.2. Optimizations were primarily applied to the *dynamic subtrial algorithm* and the LDA classifier to conform with the classification method described in chapter 5. An in-depth explanation of the above methods will be explained in chapter 7. Schematically, the pipeline can be divided into four components each with a distinctive role:

Acquisition The data acquisition component simply manages calls to the low-level driver functionalities of the EEG and keeps requesting data buffer from the device in cyclic time intervals. Further, it controls the built in hardware IIR digital filters which helps to reduce the required workload of the subsequent signal processing steps. The acquired data is passed to the BCI2000 operator module which handles data logging and storage facilities and passes the data to the preprocessing node in the pipeline.

Preprocessing When running in online mode (i.e. live classification of unknown data) the preprocessing extracts epochs from the continuous EEG data stream and applies feature extraction methods to each epoch. Further, it changes the internal data representation to conform with the format required by the subsequent classification method.

Classification Classification is carried out for each new epoch received from the preprocessing step. As classifier, LDA is used with analytically determined regularization parameters for the covariance estimator as described in 5.4.

Postprocessing The postprocessing step consists of the *dynamic subtrial algorithm* which collects classification scores for the respective epochs and is able to control the behavior of the visual paradigm by setting appropriate global state variables.

When running in *offline mode*, all steps after the data acquisition are skipped since only data are recorded without any classification.

6.5 Robot component

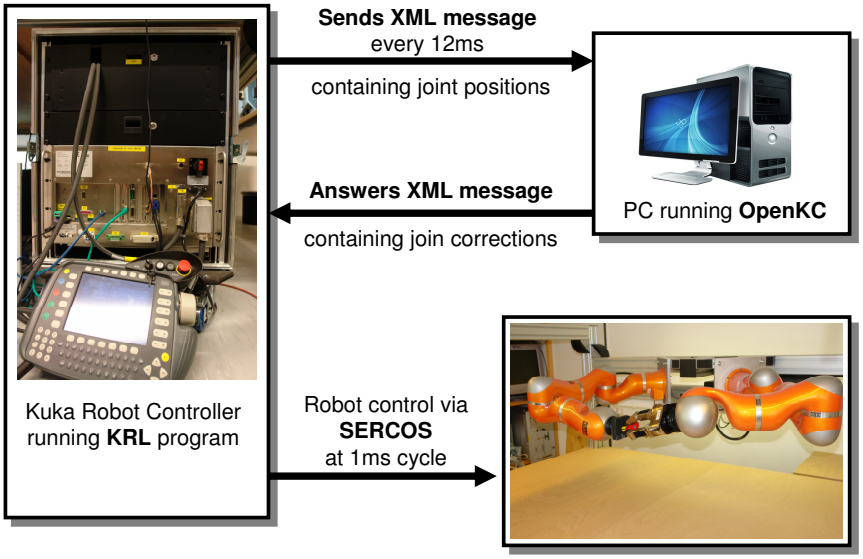


Figure 6.11. Schematic view of the low-level robot control infrastructure using OpenKC [Schöpfer et al., 2010] and *Kuka Robot Language (KRL)* to control the endeffectors.

All previous components were primarily concerned with the recognition of mental commands. This last component constitutes the interface from the mental to the physical world. Commands detected in the former steps are translated into control statements of a robotic actuator. Thereby, sensor data originating from the 3D position and pose estimation of the AR component serve as target coordinates for the robot controller. Using the AR paradigm scenario only two different types of actions are possible, grasping an object and releasing it at a desired location. The mechanism of the high-level interfacing part of the robot component is quite simple. An UDP network socket is bound to a specified port while listening for incoming packets. Packets will be sent out from the AR paradigm contained in the Augmented-

Reality component. Each packet contains a list of variables (i.e. a *state vector*) representing the next required state of the robot. The components of the state vector $\bar{S} = \left(t_x, t_y, t_z, a\right)$ consist of the target location $\left(t_x, t_y, t_z\right)$ with respect to the robots coordinate frame and an action type variable $a \in [1,2,3]$ where the numbers represent a certain type of action which can be either *GRASP*, *RELEASE* or *RESET*. Only the first two commands *GRASP* and *RELEASE* can be issued by mental commands. The last command *RESET* merely serves as a convenience command to bring the robot into a predefined position. Low-level control of the robotic actuator can not be achieved by mental commands. Instead, the robot backend completely controls arm trajectory and grasp parameters. The low-level control process consists of integrating arm position, hand posture, grip force and tactile feedback into a control loop to be able to grasp arbitrary objects by simply knowing their coordinates. Information about the objects shape, size and weight are important parameters to apply correct grip pressure and hand postures for grasping and releasing an object. This problem however is solved in the *Open Kuka Control (OpenKC)* framework [Schöpfer et al., 2010]. All low-level control is managed with the use of the OpenKC API. In the default setup of the Kuka arms, a robot program written in *KukaRobotLanguage (KRL)* running on an embedded realtime capable Windows using VxWorks is sending joint control commands to the endeffectors at a 1ms cycle. The current setup adds a remote realtime capable computer running the OpenKC framework. As seen in figure 6.11, the remote computer communicates with the Kuka controller by sending out XML network messages. These messages comply with the XML version of the *Robot Sensor Interface (RSI-XML)* protocol which is an open protocol provided by Kuka for real-time control of the manipulator. Once a network communication is established, the robot controller starts to send XML messages in a 12ms interval containing current joint positions to the remote computer which in turn is required to answer this message with appropriate joint correction values.

7 An Asynchronous BCI for Robot Control

7.1 Introduction

This final study aims to investigate the usability of the Augmented-Reality based Brain-Computer Interface. The focus is put on the usability evaluation in real-world situations and improvements related to issues arising from the trial-based nature of P300-based paradigms. Contrary to the majority of BCI studies, the subjects were allowed to talk, move to a certain extent and look around freely which entails a high grade of robustness of the classification methods. This setting should allow to simulate the behavior of people using such a system as a tool in an arbitrary environment. In addition, the previously proposed dynamic subtrial algorithm (c.f. chapter 4) has been reinvestigated and extended to incorporate and asynchronous operation. In this version, specific information is extracted which can be used to decide whether a user actually intents to communicate with the BCI or is just reacting to a stimulus which was not originating from the BCI. Knowledge about the user's intent to communicate with the system is a crucial point when implementing a BCI for robot control. Since the event responsible for the evocation of an event-related potential, which is used to derive the intended action of the user, can stem from the stimulation paradigm of the BCI or from some arbitrary uncontrollable outside stimulus it is difficult without additional sensors like microphones or cameras to identify the origin of the event. The idea behind the extended dynamic subtrial algorithm is that the temporal event onsets stemming from an event issued by the BCI should also temporally correspond to a P300 potential over multiple epochs. If the user is paying attention to an outside event which is not controlled by the BCI but occurs at the same time as a BCI related event, an ambiguity arises. This ambiguity however can be resolved since it is unlikely that the outside stimulus will be temporally correlated to the BCI event multiple times in a row. The extended version of the dynamic subtrial algorithm keeps track of the temporal correlation of consecutive P300 epochs for each stimulus and combines this information with the classification output. Using this method, the BCI becomes more robust to outside events, artifacts, misclassifications and allows to transform the trial-based nature of P300 based BCI's into a continuous one. The advantage of a continuously running BCI is that it can receive commands at an arbitrary time in contrast to the trial based approach which defines a point in time where the user has to start concentrating for a given number of subtrials.

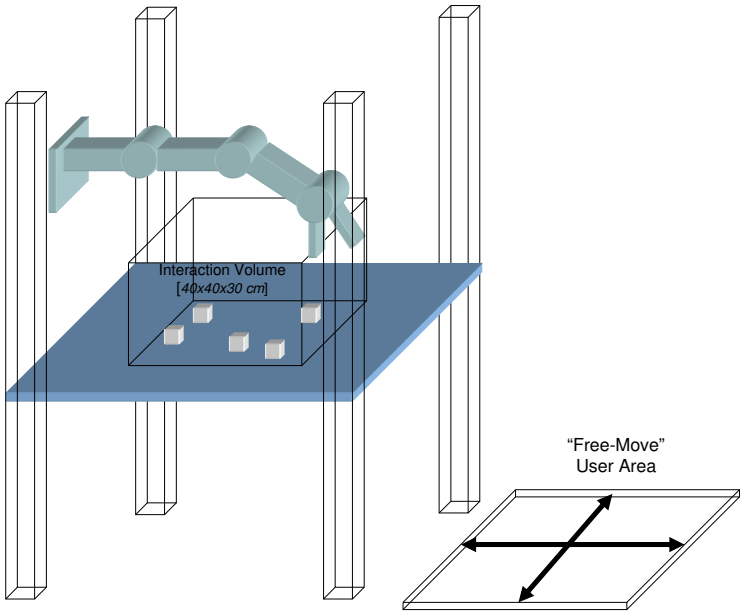


Figure 7.1. Illustration of the environment setup of the experiment. The volume in which actual object interaction can occur is 40x40x30 cm large. A "Free-Move" area is located in front of the cage where the user is allowed to move around freely while viewing the scene.

7.2 Experimental Setup

The physical environment which served as stimulus input was restricted to a $170cm \times 170cm$ robot cage. A table was located inside the cage at 130cm above the ground. The Kuka robot arm was mounted at the back of the cage and 50cm above the table surface. Due to robot joint constraints, the table area on which actual object manipulation could occur was limited to a $40cm \times 40cm$ square area aligned around the tables center. Five plastic cubes with attached AR markers on the top were placed on the table inside the 40×40 interaction volume with random initial positions. An illustration of the experimental setup is shown in figure 7.1. According to the AR paradigm as described in chapter 6, each subject was equipped with EEG electrodes and a stereoscopic Trivisio AR-Head Mounted Display (HMD). The user was viewing the physical scene solely through the LCD displays of the HMD at a 800×600 resolution and 60Hz refresh rate. Image acquisition from both cameras was synchronized via the firewire API of the drivers and displayed on the corresponding LCD screen. The narrow field of view of the oculars made it was necessary to manually adjust the ocular positions for each user. Since no blinders were used with the HMD it was possible to direct the gaze past the extents of the oculars. Thus, parts of the environment were visible in the peripheral view of the user but not in the field of view of the cameras. When viewed through the HMD, each marker was augmented with a 3-dimensional model of a number with the exception of one special

marker. This special marker was placed at a predefined position with known coordinates inside the robot coordinate frame and served as reference point for the required coordinate transformations from the camera to the robot frame. When trying to place a grasped object, a grid was augmented on top of this marker where each grid cell corresponded to a selectable target location.

Data acquisition

To evaluate the usability of the Brain-Link system, a study involving 4 healthy subjects was conducted. Based on the knowledge gained from previous experiments, electrodes were attached at the positions O1, O2, Pz, P3, P4, Cz, C3, C4, Fz, F3 and F4 to cover the primary areas related to the P300 ERP (central to occipital sites) as well as frontal areas which seemed to be prominent locations for the detection of error-related potentials. Reference and ground electrodes were attached to the left and right ear lobes. The scalp of the subject was prepared at the electrode locations with ethanol and an abrasive paste to improve skin conductance. Impedances greater than$10k\Omega$ were rejected and the site was prepared again until a proper impedance was achieved. Data recording was done with a 16 channel gUSBAmp EEG amplifier with corner frequencies of the internal bandpass filter set to 0.1 and 60Hz to account for slow signal drifts which can occur from changing skin conductance. Throughout the training session, data of the resulting continuous EEG data were stored to disk for later offline analysis.

Experimental protocol

Similar to the previous studies, the grasp experiment was broken down into a training and an online phase.

Training Phase During the training session, the subject was viewing the scene through the HMD and was asked to keep all marker cubes in his field of view at all times. A gray colored 3-dimensional model of unique numbers was displayed above the cubes. The procedure required the subjects to focus on a given cube (i.e. the number above it) and count the number of intensifications. Training started with marking the target cube by changing the color of its associated number to green for 3 seconds, followed by 10 intensification rounds. Each intensification round consisted of flashing all visible numbers on a one-by-one basis for 60ms. This corresponds to the *Single-Speller Paradigm* as described in section 3.2.6. For 5 available symbols a *subtrial* denotes a full round of 5 intensifications while 10 consecutive subtrials denote a *trial*. A complete training session covered 30 trials which resulted in 60 epochs from the P300 class and 240 epochs containing no P300. Once the data collection had been completed, the data were passed to a Matlab script which computed an LDA classifier.

Online Phase During the online phase, the subjects were also viewing the scene through

the HMD. The robot arm and thus movement of the cubes could be controlled in a two-staged process (c.f. figure 6.4). In a first step, the cubes were augmented with their corresponding number and selection took place by focusing on a certain number while mentally counting it's number of flashes. In contrast to the training procedure, no fixed number of subtrials was given. The optimal number of subtrials to predict the letter was determined by an extended version of the dynamic subtrial method. Once the target cube had been identified, a grasp command was sent to the robot backend along with the coordinates of the cube' relative position to the robot's coordinate frame. Pathplanning and grip force was handled autonomously by the backend and was not controllable in any way by the user. The first stage was completed once the robot had grasped and lifted the target cube 30cm above the table.

At the beginning of the second stage the system switched into the *grid-mode* which concealed all cube augmentations and displayed a grid plane which was anchored at the reference marker. Each grid cell corresponded to a target symbol and marked a spatial location for placing the grasped object. Selection of the cells followed the procedure of a normal P300-Speller paradigm by highlighting rows and columns in random order. Again, the user had to focus on a symbol (i.e. grid-cell) and count the number of corresponding flashes. As in the first stage, the extended dynamic subtrial method determined the number of subtrials. In response to an identified target grid-cell the object was placed at the desired location. Afterwards the arm was moving back to it's initial position, ready to receive a new command from stage one.

To evaluate the efficiency of the system, the subjects were required to control the robotic arm during the online phase and rearrange the five marker cubes at randomized initial positions. The task was to move 10 arbitrary markers to a different location. During the running experiment the subjects were standing in front of the robot cage and were allowed to walk freely from left to right in front of the cage. Further, talking and asking questions was allowed as well as looking around the room freely. Relaxing the experimental constraints usually involved in BCI studies (i.e. sitting still, not moving) allowed to evaluate the usability for healthy subjects. Since these are potential sources of EEG artifacts, misclassification on single-trial basis is an inevitable drawback. To detect and mitigate these effects was subject to the extended dynamic subtrial method (EDS) and thus served as a benchmark to test its robustness.

To get an objective measure of the performance of the extended dynamic subtrial method, the BCI remained active and was ready to receive commands after completing the actual goal of rearranging the cubes. During that time, the subjects were interviewed by the experiment supervisor. The questionnaire contained questions about the ease of use, subjective efficiency, identified problems and suggestions to improve the system. During that interview, the BCI remained active but was to required to stay idle and not conduct any robot actions. The study closed with a last verbally given order to the subject to move the first cube back to the lower left grid-cell.

In summary the experiment aimed to investigate two aspects of the BCI:

- Evaluation of the **Object Interaction** task, involving *pickup* and *placement* subtasks
- Evaluation of the asynchronous **Extended Dynamic Subtrial (EDS)** method

Prior to each experiment, an initial training session was conducted to gather information required by the classifier and EDS algorithm. In the following analysis of the results, the object interaction task will be referred to as *Block 1* and the evaluation the EDS method as *Block 2*.

7.3 Methods

The methods to determine the intended symbol selection combine the findings of the previous studies. In specific, these are P300 detection using a Fisher Linear Discriminant Analysis (LDA) using EEG data preprocessed with an *Infomax ICA*. As demonstrated in the first study, an automatic determination of an optimal number of subtrials can effectively reduce the time needed to communicate a symbol. Therefore, this technique has been incorporated into the final system. In addition, this technique lends itself to implement a technique that is capable to indirectly determine whether an EEG epoch, identified by the system as a P300 containing epoch, represents information that is directed to the BCI or is merely a response to outside events or even just a misclassification. Hence, an extension to the method is proposed which incorporates the desired characteristics.

Preprocessing and P300 Classification

During the training session, a stream of EEG data which contained 300 stimulus events was acquired. As a next step, the data were split into epochs with total lengths of 800ms ranging from 100ms pre-stimulus to 700ms post-stimulus. An *Independent Component Analysis (ICA)* was computed on these data to unmix statistically independent sources into separate channels. In specific, prior to ICA a spatial *Principal Component Analysis (PCA)* was performed to reduce the data to their intrinsic dimension and sphere the data. The sphering step is in fact not required but reduces the convergence time of the following *Extended Infomax ICA* algorithm [Lee et al., 1999]. Traditional ICA algorithms like *Infomax ICA* [Bell and Sejnowski, 1995] are unable to separate sub-gaussian [1] sources. Some types of ocular artifacts were shown to have a sub-gaussian distribution [Jung et al., 1998]. Therefore extended ICA was chosen to be able to decompose sources exposing a variety of distributions. According to the findings of the ErrP study (chapter 5), classification of EEG epochs benefit from downsampling of the signals. Downsampling resamples the signal at a lower rate. However, simple skipping of sample values will inevitably result in *aliasing effects* since frequencies

[1] The most prominent artifacts have been shown to have a sub-gaussian distribution. In [Lee et al., 1999] an extended version of Infomax ICA has been suggested to separate these artifacts from actual EEG.

higher than the Nyquist frequency (with respect to the new samplerate) will be mirrored into the lower frequency spectrum. To mitigate this problem, the data were processed with an 8th order IIR Chebyshev Type I lowpass filter before decimating the signal. The cutoff frequency of the filter was set to $C = 0.8 * (F_s/2)/R$ with F_s being the samplerate and R the decimation factor. Since the frequency content of a normal P300 potential does not exceed 20Hz, a decimation factor of $R = \lceil \frac{128Hz}{20Hz} \rceil = 7$ and a cutoff frequency of $C = 0.8 \cdot \frac{128Hz}{7} = 14.6Hz$. After decimation, the size of an epoch was 26 samples per channel. The final feature vector was formed by concatenating the channels into a single vector with $11 \times 26 = 286$ elements. Each concatenated epoch was added to the training set represented by a 300×286 matrix.

Next, a Fisher Linear Discriminant classifier with regularization parameters for the covariance matrices was trained based on the former data set. The computational steps are similar to the approach described in the first study (c.f. 4.2.3). Adapting an LDA weight vector $\mathbf{w}$ to the training data requires to calculate the pooled *within scatter matrix* $\mathbf{S_W}$ which is defined as

$$\mathbf{S_W} = \mathbf{S_{P^+}} + \mathbf{S_{P^-}}, \tag{7.1}$$

where $\mathbf{S_{P^+}}$ denotes the covariance matrix estimate for all P300 epochs and $\mathbf{S_{P^-}}$ the estimate for all non-P300 epochs. For small sample sizes, it is known that the sample covariance estimation is rather imprecise and for $N << p$ even ill-posed, with N being the number of observations and p the number of features. As a result, the smallest eigenvalues of the matrix are too small while the largest eigenvalues are too large compared to the real eigenvalues. For ill-posed matrices 0 eigenvalues exist and thus, its inverse cannot be computed. A popular method to improve the accuracy of the estimation is to use *shrinkage methods*. Thereby, the covariance ellipse is shrunken towards the center which effectively shrinks the eigenvalues of the matrix towards their mean. Maximum shrinkage in this context will modify the elliptic shape to a circle. Regularization parameters were computed using an analytic method as presented by [Ledoit and Wolf, 2004]. For both of these covariance matrix estimates, optimal shrinkage parameters have been calculated which modify equation 7.1 to

$$\mathbf{S^*_W} = \left[\frac{b_1}{c_1} m_n \mathbf{I} + \frac{a_1}{c_1} \mathbf{S_{P^+}} \right] + \left[\frac{b_2}{c_2} m_n \mathbf{I} + \frac{a_2}{c_2} \mathbf{S_{P^-}} \right] \tag{7.2}$$

The parameters a, b, c were calculated according to the equations already described in section 5.4.3. Using the modified $\mathbf{S^*_W}$ within scatter matrix, the LDA weight vector calculation changes to

$$\mathbf{w} = \mathbf{S^*_W}^{-1} (\mathbf{m^+} - \mathbf{m^-}), \tag{7.3}$$

with $\mathbf{m^+}$ being the mean vector of the P300 class and $\mathbf{m^-}$ the mean vector of the non-P300 class. As in the previous studies using LDA as classifier, projecting a new epoch onto the weight vector results in a scalar number

$$s(x) = \mathbf{w} \cdot x + b \tag{7.4}$$

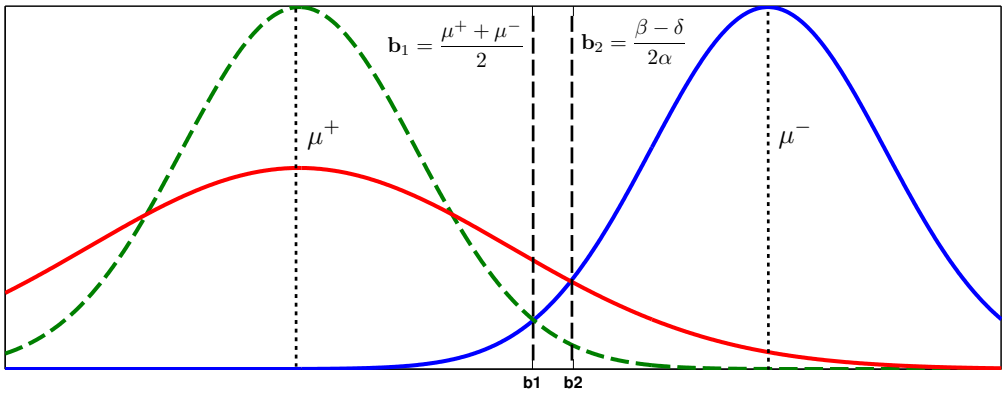


Figure 7.2. Bias calculation is done depending on the equality of covariances. In case of equal covariances, simply the center of the interval spanned by the mean values of the distributions is taken. For unequal covariances the gaussian intersection is used.

whereas the sign can be used to discriminate between both classes (i.e. positive numbers correspond to the P300 class while negative correspond to the non-P300 class). A last difference concerns the calculation of the bias parameter b which was found by cross-validation and a grid search in the first study. For a discrimination task of two classes with equal covariance b can be easily calculated by setting it to the center of the interval spanned by the P^+ and the P^- class means. In case of unequal covariances however, the gaussian intersection of those classes is calculated and taken as bias by

$$b = \frac{\beta - \delta^2}{2\alpha} \tag{7.5}$$

with

$$\sigma^+ = \mathbf{w}^T \mathbf{S}^*_{\mathbf{P}^+} \mathbf{w} \qquad \text{(the projected } P^+ \text{ class variances)} \tag{7.6}$$

$$\sigma^- = \mathbf{w}^T \mathbf{S}^*_{\mathbf{P}^-} \mathbf{w} \qquad \text{(the projected } P^- \text{ class variances)} \tag{7.7}$$

$$\alpha = \frac{1}{\sigma^+} - \frac{1}{\sigma^+} \tag{7.8}$$

$$\beta = 2 \cdot \left(\frac{s(\mathbf{m}^+)}{\sigma^+} - \frac{s(\mathbf{m}^-)}{\sigma^-}\right) \tag{7.9}$$

$$\gamma = \frac{s(\mathbf{m}^+)^2}{\sigma^+} - \frac{s(\mathbf{m}^-)^2}{\sigma^-} + \log(\sigma^+) - \log(\sigma^-) \tag{7.10}$$

$$\delta^2 = \beta^2 - 4\alpha\gamma \tag{7.11}$$

7.3.1 Extended Dynamic Subtrial (EDS) Method

An initial approach to adapt the number of presented subtrials has been presented in the first study. A more flexible version has been developed within this study that serves three main purposes:

- Online adaption of presented subtrials (communication speed increase)
- Provide a better degree of robustness with respect to classification outliers
- Enable the BCI to operate in an asynchronous mode

Both the Dynamic Subtrial (DS) and extended Dynamic Subtrial (EDS) method rely on the distribution of the projected class samples. Figure 7.3 visualizes a simplified example with two classes (red and blue) of 2D samples. First, the high dimensional samples are projected onto the LDA weight vector, which is orthogonal to the decision boundary of the classifier. The projection results in a dimension reduction by projecting all points into a 1D space. When using single-trial classification, all projected samples greater than the bias b are assigned to the blue class, while all projected values smaller than b are assigned to the red class. Both methods assume, that the mean score for each stimulus will converge towards their corresponding class mean.

Specifically, the DS method relies on the assumption that it is highly unlikely that the mean of a sequence of samples belonging to the non-P300 will be close to or greater than the projected mean of the P300 class. For a matrix-style paradigm, these scores are simply added to their corresponding cell. For the first subtrial, only single trial classification can take place which inevitably results in rather dispersed score distribution within the matrix. As more subtrials are added into the process, these values will converge to their expected means. The DS method estimates the point in time at which actual classification should happen, by defining a decision function based on the contrast (or the dispersion of values) of the score matrix. In essence, the DS method defines an abstract value based on a heuristic decision function which measures the dispersion of the score matrix. A drawback of this method is the parametrization. At the beginning of an experiment, a reasonable TP_{score} threshold can be found by using the training data to search for a value that equalizes false positives and false negatives. During a running experiment the score values for each symbol are determined by continuously averaging their single-trial scores. High variances in the underlying score distribution will cause the values of the score matrix to fluctuate randomly during the first few subtrials. Hence, with only few subtrials their explanatory power is poor. The method lacks the ability to account for the variance present in the score distribution.

The extended version in contrast uses a probabilistic approach which lets the user define a confidence level for detection of the best number of subtrials. This is achieved by interpreting classification scores as features and estimating the gaussian PDF parameters μ and σ^2 for both classes in the feature space. In its simplest form, each EEG epoch (corresponding temporally to a stimulus/symbol) is classified and the resulting classification score is collected for

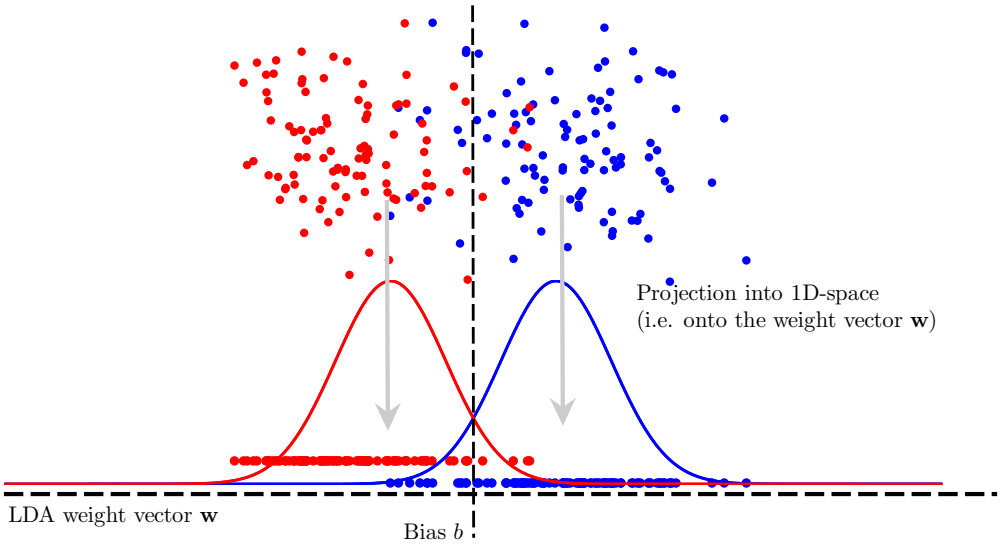


Figure 7.3. Projecting the high-dimensional EEG data onto the trained LDA weight vector **w** results in 2 classes of points which are normally distributed along an 1D space.

each symbol. Once a symbol has been presented n times, its associated sequence of collected scores form an observation vector $\mathbf{x}$ of size n. The observation vector is subject to a subsequent hypothesis test with regard to the question whether the mean of the observation vector significantly differs from the true mean of the P300 class. By transforming the principles of the DS method into a probabilistic framework, it is possible to base the decision function not only on the mean value but also account for variance and number of presentations. Instead of standardizing the scores by scaling them to a fixed interval, as it is done in the DS method, EDS utilizes the standardized z-statistic

$$x^{\phi} \equiv z(\mathbf{x}, \mu_+, \sigma_+^2) = \frac{\bar{\mathbf{x}} - \mu_+}{\sigma_+^2 / \sqrt{n}} \tag{7.12}$$

to estimate the observation mean value's deviation from the true mean μ_+; an approach that implicitly accounts for both, number of observed values and variance. Here, only the mean and standard deviation of the P^+ class are used. The reason is that we want to test how much an unknown sequence of scores deviates from the prototypical distribution of the P300 class which was obtained from the training set. This allows a more elegant formulation[1] of the decision function since not an abstract value like the TP_{score} (c.f. 4.2.6) has to be given but a *confidence level* can be defined. When choosing a suitable confidence level, the general aim is to minimize *type I errors*, i.e. labeling a sequence of classification scores for one symbol as

[1] the formulation is considered as more elegant since it is embedded in a theoretically sound framework of probability theory.

P^- when it is in fact belonging to the P^+ class. The new formulation of the decision function is defined as

$$\mathscr{D}(\mathbf{x},\alpha,\mu,\sigma^2) = \begin{cases} \text{true} & P(\Phi(z(\mathbf{x},\mu_+,\sigma_+^2)) \leq z_{p=\alpha}|H_0) \leq \alpha \\ \text{false} & otherwise \end{cases} \tag{7.13}$$

with z_p being the quantile function

$$\Phi(x)^{-1} \equiv z_p = \sqrt{2}\,\text{erf}^{-1}(2p-1), \quad p \in (0,1) \tag{7.14}$$

which can be derived by inversion of the standard normal CDF

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-t^2/2}\,dt = \frac{1}{2}\left[1 + \text{erf}\left(\frac{x}{\sqrt{2}}\right)\right]. \tag{7.15}$$

Thereby, the *erf(x)* function denotes the *error function* which is defined as

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\,dt. \tag{7.16}$$

In practice, during a running trial every classification score received from the LDA is added to the observation vector for the respective symbol. A standard score is calculated based on the classification score of the observation vector $\mathbf{x}$ after every presented stimulus using 7.12. The resulting scores are now represented in terms of standard deviations from the P300 class. In the following, these values will be referred to as *confidence values*. As a consequence, these confidences can now be used to calculated the probability of a *type I error* (false positive). The practical result of a type I error would be a wrong action conducted by the robot. Practically it is more efficient to present another round of stimuli than to end the trial too early which could potentially result in a wrong classification due to too few score observations. Correcting a wrong robot action would most likely require more time than presenting more subtrials before the action was conducted. Thus, in cases where accuracy is of greater importance than speed, the critical value used in the hypothesis test should be biased to towards the P^+ class. This ensures that false positives are minimized at the expense of false negatives. The latter type of error does not have any influence on the the issue of prematurely ending a trial. Thus, it only increases the amount of time needed to correctly classify a set of epochs which is a rather uncricitcal issue.

A question not addressed so far is how a suitable confidence level α as required by the decision function (eq. 7.13) can be found. Setting this value too high will most likely lead to an infinitely running trial as the required threshold is never reached. On the other hand, using a too low value will prematurely end a trial with a very high probability of a wrong classification. A convenient way to calculate α is to express it as a function of false positives

$$\alpha(FP) = z(z^{-1}(z_{p=1-FP},\mu_-,\sigma_-^2),\mu_+,\sigma_+^2). \tag{7.17}$$

This formulation requires to define an acceptable false positive probability denoted by FP. The quantile function z_p transforms this value into a z-score. To find the critical value based

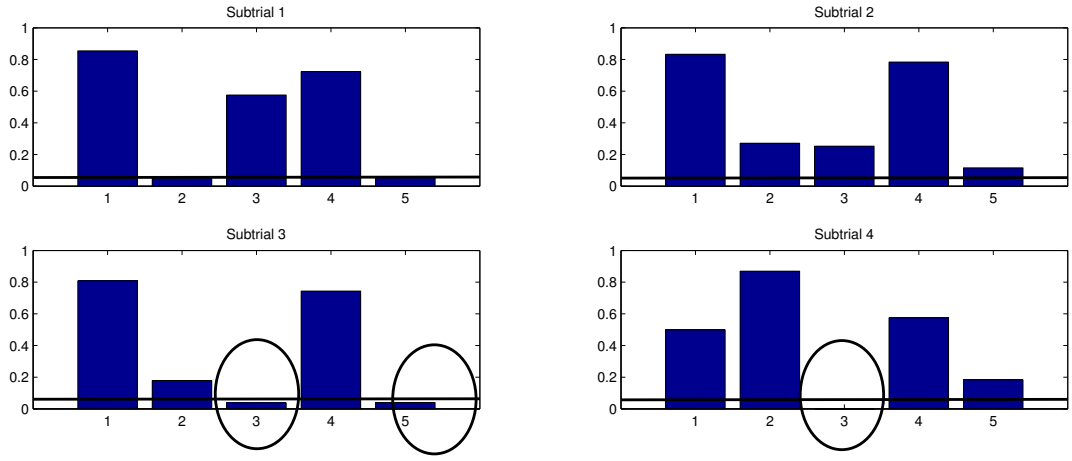


Figure 7.4. Evolution of the test statistic values for 5 symbols over 4 subtrials. The black line marks the critical value for rejection of H0. During subtrial 1-2 the hypothesis that the scores for the corresponding symbols belong to the P^+ class is rejected for symbols 1-5. For the third subtrial, two possible candidates for a target symbol show up since H0 can not be rejected for them, however, only one candidate is allowed so one more subtrial is necessary. Subtrial 4 finally delivers only one single candidate and EDS terminates the trial.

on the distribution parameters of the P^+ class, the z-score is destandardized by z^-1 using the gaussian parameters of the P^- class and subsequently restandardized using the mean μ_+ and standard deviation σ_+^2 of the P^+ class. Once this z-score is known it can serve as the critical value for the hypothesis test. For the single symbol paradigm, the iteration of the subtrial presentations finishes whenever the z-test rejects all but one symbol. In a similar way, during the matrix style paradigm, the presentation finishes whenever the z-test rejects all but two symbols.

The algorithm itself is simple to implement (see algorithm 2). For each symbol presented n times calculate $\mathscr{D}(\mathbf{x}^\phi, \alpha, \mu_+, \sigma_+^2)$ with $\mathbf{x}^\phi$ being the list of standardized classification scores for the symbol ϕ.

Asynchronous Mode

A side-effect of the EDS method is that the concept used for adapting the number of subtrials implicitly enables the BCI to be used in an asynchronous mode. While synchronous BCIs require the user to start focusing on a target symbol at a specific time for a defined time span, the EDS can automatically decide whether the acquired observations correlate to a P300 event or do not contain sustained correlations over a certain time interval at all. However, using the method as described in this section will inevitably lead to problems due to the continuously growing averaging window for the observations. Once a user stops focusing on the BCI while it is running, observations are still being accumulated for each symbol. Given

Algorithm 2 Pseudocode of the Extended Dynamic Subtrial (EDS) algorithm.

repeat
  $i \leftarrow random(1, n)$
  present stimulus i
  $\mathbf{x} \leftarrow$ extract EEG epoch
  $S[i] \Leftarrow$ LDA($\mathbf{x}$) {Add LDA score to the list $S[i]$}
  for all $S[i]$ **do**
    $x^{\phi} \leftarrow z(\mathbf{S}[i], \mu_+, \sigma_+^2)$
    $H_i \leftarrow \mathscr{D}(x^{\phi}, \alpha, \mu_+, \sigma_+^2)$
  end for
until $\sum_i (H_i) ==$ true

the user starts focusing on the BCI again after 50 subtrials have been presented (i.e. 50 observations per symbol), more than 50 subtrials are needed to mitigate the influence of the 50 non-P300 epochs to the averaging. In essence, the EDS will treat the whole time span until a classification occurs as one single trial and will also keep accumulating observations during that time. As a result, switching from a *no-control (NC)* state to *intentional-control (IC)* state can potentially take very long. The switching time until $\mathscr{D}$ returns a true value is bound to the time spent in the NC state. A small adaption to the the previously presented method is required which replaces the growing averaging buffer by a moving average filter with dynamically shrinking and extending window sizes. Instead of continuously gathering observations for the averaging, a window $\omega(t_{min}, t_{max})$ is defined using a minimum t_{min} and maximum length t_{max}. At the beginning of a new trial, the size of ω is initialized with t_{min}. The window is required to contain at least t_{min} observations before it is passed to the decision function for the first time. This allows to set a minimum number of subtrials to be presented to improve the reliability of the hypothesis test. A value of 1 would simply turn off this feature and classification can occur from subtrial 1 onwards. With each additional subtrial ω is extended by one sample until it reaches the maximum size t_{max}. From this point on, the window remains static in size until a classification is made. For each additional subtrial, the oldest observation is replaced by the most recent one. A suitable value for t_{max} was selected by taking the results of previous BCI experiments into account (c.f. 4.2.6). High recognition rates could be obtained by using only 4-5 subtrials. Thus, 4 subtrials have been chosen as maximum window size for the grasping experiment. The modified algorithm is outlined in 3.

Using this method, it is now possible to ignore the BCI stimuli for a long period and switch back into IC state within just a couple of subtrials. A benefit of this method is, that no additional training is required (as in e.g. [Zhang et al., 2008]) for approximating a probabilistic model of EEG data that do not correspond to intentional control commands. At the same time, no additional computation time and space is needed for the online adaption of subtrials since the method solves both tasks at the same time. The approach is graphically illus-

Algorithm 3 Pseudocode of the asynchronous EDS. The continuously growing score list S[i] has been replaced by a variable size window ω. In single-symbol mode, only one symbol should lead to a rejected H_0 whereas in grid-mode two symbols will cause H_0 to be rejected (N=1 or 2).

Require: t_{min}, t_{max}
  if Single-Symbol mode **then**
    $N \leftarrow 1$
  else
    $N \leftarrow 2$
  end if
  initialize ω with size t_{min}
  repeat
    $i \leftarrow random(1,n)$
    PresentStimulus(i)
    x $\leftarrow$ extract EEG epoch
    if $size(\omega[i]) > t_{max}$ **then**
      $\omega[i]_{oldest} \Leftrightarrow LDA(\mathbf{x})$
    else
      extend size of ω by 1
      $\omega[i] \Leftarrow$ LDA(**x**)
    end if
    for all $\omega[i]$ **do**
      $x^{\phi} \leftarrow z(\omega[i], \mu_+, \sigma_+^2)$
      $H_i \leftarrow \mathscr{D}(x^{\phi}, \alpha, \mu_+, \sigma_+^2)$
    end for
  until $\sum_i (H_i^{true})$ == N

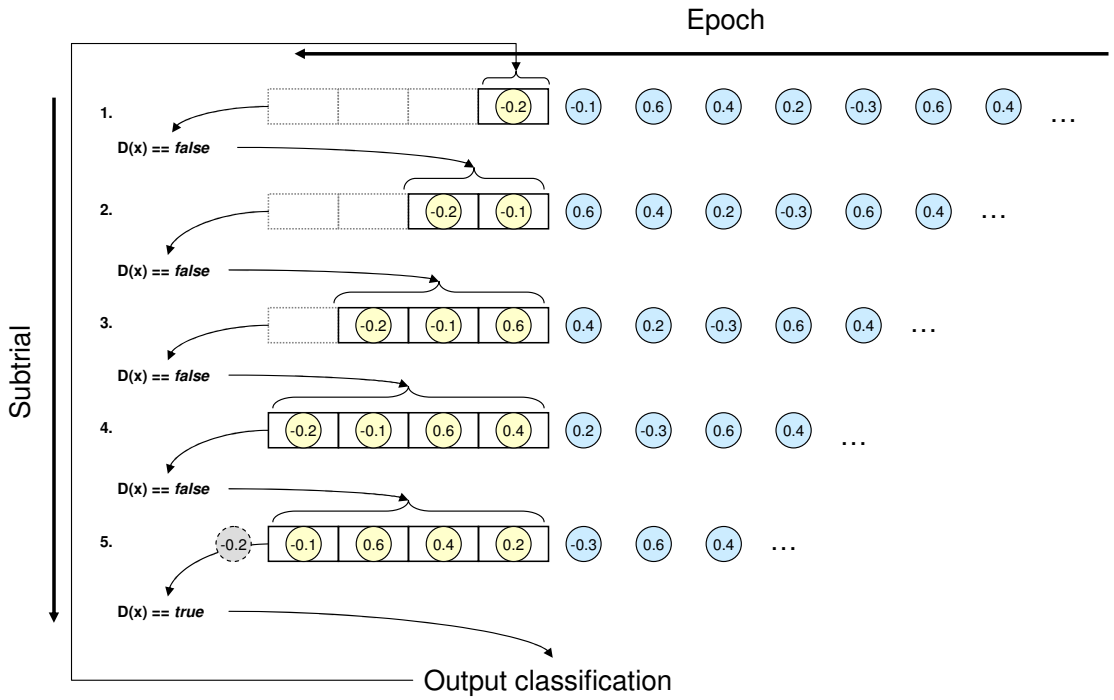


Figure 7.5. Illustration of the asynchronous EDS algorithm with dynamic window sizes. An averaging window grows with increasing number of subtrials until a maximum length is reached. For each subtrial, the current window serves as input for the decision function $\mathscr{D}$ which either start another subtrial if the contained values do not significantly correlate with the P300 score distribution, or outputs the result if the contained values are significantly similar to the P300 score distribution.

trated in figure 7.5 which shows how each consecutive subtrial adds one more classification score to the window. For each added score, the windows grows by one until it reaches its maximum size of 4. After this point, the oldest score observation gets replaced by the most recent one. The decision function will evaluate the current window for every subtrial and return *true* whenever the sequence of scores is likely to stem from the P300 score normal distribution with parameters μ_+ and σ_+.

7.4 Results

An experiment was conducted with 4 mixed male and female subjects with the aim to evaluate the overall usability of the system and the feasibility of the asynchronous protocol. The task for the subject was to move objects placed in front of them to a different location on the table. Both, the object and the target location were chosen by the subject. Whenever the robot picked up or placed an object, the subject was asked to report on the correctness of the robots action. The experiment ended when the subject successfully moved 10 objects to

	Pickup Task		Placement Task		No-Control Task		
Subject	Acc.	Sym./min	Acc.	Sym./min	FPR	FPR^*	TTA
S1	80%	3.3 (2:20)	70%	1.1 (6:30)	0.4	*0.2*	12s
S2	90%	2.3 (2:50)	70%	1.4 (5:00)	0.6	*0.71*	18s
S3	100%	5.9 (1:40)	90%	1.7 (5:50)	0.2	*0.39*	8s
S4	80%	2.3 (3:40)	60%	0.7 (8:20)	0.8	*1.54*	10s
Mean	87.5%	3.45	72.5%	1.23	0.5	*0.71*	12s

Table 7.1. BCI performance achieved in the study. The measures accuracy (Acc.), correct symbols per minute (Sym./min), false positive rate as wrong actions per minute during the *no-control* period and *time to active* (TTA) are shown. For comparison, FPR^* reflects the results from [Zhang et al., 2008] who also used 4 subjects in his study. These rates were assessed with a different set of subjects.

different locations which required 20 selection commands (i.e. 10 for *pickup* and 10 for the *placement* task). The task performance was measured by calculating the communication rate in correct symbols per minute, as well as the overall accuracy.

To evaluate the feasibility of the asynchronous protocol, the subject's focus had to be distracted from the BCI stimulus presentation. For this reason, they were asked to fill out a short questionnaire and answer to questions of the experiment supervisor after the experiment ended. During that time the BCI was still running and was ready to receive commands. The evaluation criterion here was the number of action conducted during this period. A perfect recognition of user inactivity could not be achieved but the number of 0.5 conducted actions per minute is slightly better than the method proposed by [Zhang et al., 2008] which was also used in P300 wheelchair control of [Rebsamen et al., 2007]. Further, to evaluate how long the system takes to recognize that the subject is now actively communicating with the system, one more object relocation had to be carried out. Table 7.1 summarizes the number of wrongly conducted actions per minute during the questionnaire period as well as the time it took the system to recognize a voluntary selection command of the user (*time to active (TTA)*).

For the best performing subject 3, a close to perfect result in both tasks could be achieved with accuracies of 100% for the *pickup task* and 90% for the *placement task*. As a general trend however, the accuracy in the placement task is inferior to the pickup task with about 10-20% less accurate predictions. The practical communication speed during the pickup task ranged from 2.3-5.9 symbols per minute seem to be comparable to other state of the art work (e.g. [Townsend et al., 2010]). In contrast, the communication speed during the placement task dropped by a factor of almost 3. Considering the much larger set of stimuli, this is not a surprising fact since for the placement, a set with 64 stimuli has to be presented whereas the pickup task only required 5 stimuli to be displayed. For this reason, it would be inaccurate to directly compare these results to other works which work with different number of stimuli and different time periods between consecutive stimulus presentations.

The results of the *no-control* state detection are comparable to the method proposed by

Number	Question
1	The system is intuitive
2	The system is exhausting
3	The system reacts as I expect it
4	The system is slow
5	The system usage is complicated
6	The system reacts different when I talk or move
7	Concentration declines after the training phase
8	I had to concentrate hard to make it work
9	My commands were recognized reliably

Table 7.2. Translated questions from the questionnaire. The subjects had to assign scores to each question between 1 (does not agree) to 10 (fully agree).

Zhang. Even though a slightly lower FPR was achieved by the EDS method, the rates per subject seem to suggest that there is no significant difference in detection performance between both methods. This fact is not surprising considering the similarity of the methods. Both exploit the scalar score distribution of a classifier and statistical significance testing to assign class labels. The difference consists mainly of the fact that the EDS requires only P300 and non-P300 class epochs for the training while Zhang's method requires epochs acquired during user inactivity as a third class.

To evaluate the practical usability of the system, relevant information from the questionnaires were considered. The appendix lists the original questionnaires in german language while table 7.2 shows a translated version of the questions.

The mean scores assigned by the subjects are shown in figure 7.6. For convenience, the mean values have been converted into *performance values*. The *performance value* will be referred to as a value that equals 1 if the given score corresponds 100% to the desired behaviour of the system. For instance, question 2 states that the system is exhausting. When a subject fully agrees to this question, a 10 will be assigned. Since the intention of a BCI developer would be to design a system that is not exhausting at all, the *performance value* will be 0. On the other hand, a system that is intuitive and easy to use (question 1 and 5) is highly desirable, hence a user score of 10 would correspond to a *performance value* of 1. With respect to the intended behavior, questions 2, 4, 5, 6, 7 and 8 are negative formulations which require negation (i.e. $11 - score$) of the assigned scores to receive unnormalized performance values. After negation, all scores were normalized to the [0..1] interval.

The results show that the BCI did not perform well in all parts concerning concentration (i.e. question 2, 7 and 8). None of these values could reach 50% performance. All subjects testified that the BCI usage is very exhausting after a while. It is not surprising that question 7 and 8 did not receive favorable scores as well since an exhausted user will have difficulties concentrating on a specific task. However, the standard deviation for question 7 and 8 are much larger than for question 2 which indicates that the subjects opinions differed to a

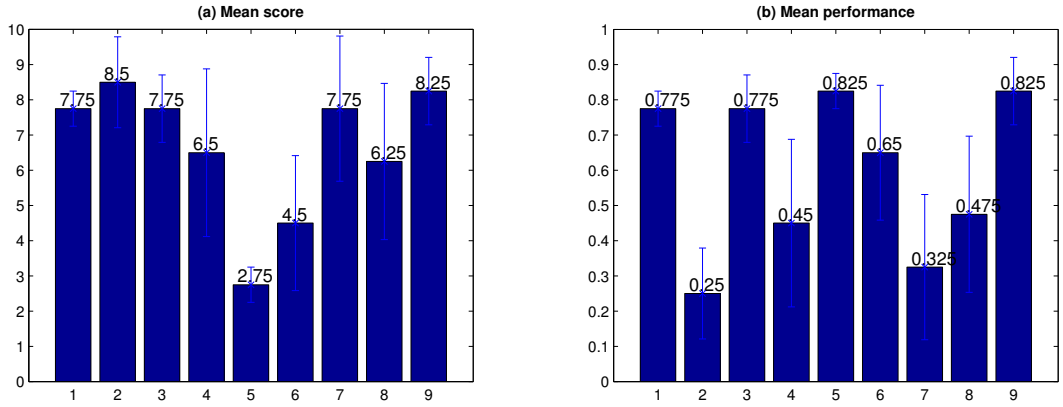


Figure 7.6. (a) Mean scores of the usability study with standard deviation bars. (b) Normalized performance values, with 1 corresponding to best and 0 to worst outcome for the specific question.

larger extend. Another important factor is the perceived communication speed (question 4). On average, speed was rated as mediocre. If the score assigned by subject 2 was not included, the result would be even worse. For subject 3 however, even though he achieved statistically the highest speed and most accurate results, only a performance of 0.3 (score of 7)was assigned whereas subject 2 rated the speed with 0.7 (score of 3). This phenomenon shows that the perceived communication speed is highly subject dependent. It remains unclear, which factors have an influence on the speed perception in this setting. On the more positive side, the remaining fields covered by the questionnaire performed much better. Questions 1 and 5 were aimed to assess the ease of use of the BCI with the expectation, that both question would be answered with the same score. The mean scores shown in figure 7.6 reveal that question 1 and 5 were consistently given similar performance rating of 0.77 and more with low standard deviations. These results indicate that the subjects had no difficulties using the BCI. A similar outcome could be ascertained for the questions 3 and 8, both aimed to assess the subjectively perceived BCI accuracy. Both questions were given similar scores of more than 77% with comparable standard deviations. As an interesting fact, the perceived accuracy even matches the real mean accuracy of 80% (averaged over all subjects and both tasks). Thus it can be asserted, that subjects of this study had good feeling for the overall accuracy and did not get biased by the long and exhausting experiment.

The last topic to be evaluated was the *no-control state detection* which was covered by question 6. Overall, the scores showed that most of the subjects did not notice a huge difference while talking or moving (performance values of 0.5-0.7), except subject 2 who assigned a performance value of 0.3. The mean value still exceeds 60%.

7.5 Conclusion

This final chapter demonstrated the use of a novel Augmented Reality based P300 paradigm for robot grasping tasks. The presented system features a novel asynchronous dynamic subtrial paradigm which allows the user to communicate with the system at arbitrary times and eliminates the need for a fixed number of subtrials. In an online study, covering 4 subjects, the performance of the system was evaluated with respect to overall accuracy, communication speed and reliability of the *no-control state* detection. The results confirmed that the proposed methods are suitable for use with an online BCI that controls a robot arm for object grasping tasks. Especially the *pickup task* revealed consistent accuracies of 80% and higher. An accuracy drop could be measured for the *placement task*. A possible reason for this effect might be the larger number of stimuli (i.e. 16 for the placement task and 5 for the pickup task). Additionally, due to the row-column paradigm, 2 out of the 16 epochs have to be correctly detected as P300 epochs while for the pickup task, only 1 out of 5 epochs have to be correctly labeled as P300 epoch. Thus, the second task can be considered as more difficult than the pickup scenario.

The reliability of *no-control* phases, i.e. periods of user inactivity where no communication with the BCI takes place, consistently achieved less than 1 wrong action per minute. Even though the achieved average of 0.5 wrong actions per minute is slightly better than the results reported by Zhang [Zhang et al., 2008], for critical tasks which result in a real action being conducted this value is still too high to be practical in an *out-of-lab* setting. Yet, the presented method achieves to solve two different goals, the dynamic adjustment of subtrials and the detection of no-control states at the same time. This process comes without the need to record additional training data for the no-control state detection and thus can be applied to every subject without prior training.

In the end, a usability study was conducted which emphasized some weak points of the BCI. Primarily the exhausting nature of the paradigm and monotone usage, which is common to all other BCIs, proved to be a crucial factor that was collectively rated with low scores. The perceived speed of the system was also only in the middle of the ranking. System ease of use and accuracy however, were given high scores. Further, the *no-control state detection* did prove to be useful since most subjects could not recognize a huge difference in the BCI behavior when they were talking or moving. In this sense, the presented techniques can be considered as a promising step towards a reliable asynchronous P300 BCI.

8 Conclusion

Brain-Computer Interface (BCI) systems provide a promising and new way to control robotic actuators for grasping tasks. Classical P300 based BCI were mostly limited to applications like spelling devices. Due to their dependency on external stimuli to evoke a P300 potential, the most straight forward way to utilize this type of BCI is to associate predefined commands with each stimulus. As demonstrated first by Farwell and Donchin [Farwell and Donchin, 1988], a spelling device can be constructed which is solely controlled by brain signals. In more recent years, researchers like Rebsamen [Rebsamen et al., 2007] adopted this approach to control physical devices like wheelchairs with the P300 paradigm. This task demands for more dynamical stimulus-action associations since positions and locations can be dynamic. Rebsamen resolved this issue by mounting additional sensors and control units on the wheelchair that realize obstacle avoidance and path planning autonomously. Stimuli consist of predefined locations like *Office, Toilets* or *Lift*. Coordinates are supposed to be available for these locations and are not acquired dynamically. A change in the environment would lead to invalid commands until the internal map has been rebuilt. A similar approach was presented by [Iturrate et al., 2009] but instead of using predefined locations, Iturrate mounted laser range finders on a wheelchair to reconstruct the surrounding in a virtual reality scene. A grid pattern was overlayed with the scene where each grid intersection served as a stimulus corresponding to the spatial location in the VR scene. This was, the system enabled the user to drive the wheelchair to arbitrary locations. A downside in this work was the missing possibility of asynchronous control and thus relied on the fact that the user was communicating with the BCI at all times.

In this thesis, a BCI system has been developed that solves both of the mentioned problems and even goes beyond by investigating a method to correct for errors caused by incorrect BCI classification. A novel paradigm based on Augmented Reality techniques was proposed which enables physical objects being used as stimuli. Instead of associating fixed locations with the stimuli, they can be extracted from camera images using computer vision methods. To highlight the practical use of the method, a BCI system was developed that enables a user to control a robotic arm and hand to grasp plastic cube objects. This system features a novel method that detects the optimal number stimulus presentation rounds in real-time. The initial method as published in [Lenhardt et al., 2008] was extended by a *no-control state* detection with promising results. To my best knowledge, the only other method that deals with this issue is the approach presented by Zhang [Zhang et al., 2008]. In constrast to the

method presented in this thesis, Zhang uses training samples from user inacticity periods to construct a statistical model. The results of both approaches seem to be comparable but both studies are limited in their significance due to the rather small number of subjects. A perfectly working method like this would finally transform the synchronous P300 paradigms into asynchronous ones which would allow BCI technology to be used in other settings than scientific laboratories.

A further achievment of this thesis is a new method that is rarely seen in current online BCI, which is the detection of erroneous classifications of the BCI based on error-related potentials (ErrP). A type of event related potential (ERP) that occurs in response to errors committed by the BCI. In the second study (chapter 5), possibilities to utilize this ERP to avoid wrong robot actions being carried out during a running experiment were examined. Compared to P300 classification, the ErrP detection is supposed to be more difficult due to the fact that only one trial is available for the decision. Obviously, single-trial classification does not allow for signal averaging and thus signal-to-noise ratio (SNR) is much worse than for the corresponding P300 classification method. In the presented study, a classification method was developed that could achieve an true positive rate of close to 90% with a false positive rate of less than 10% for two out of five subjects. A mean accuracy across all subject of more than 80% could be measured which can be considered as a very promising result compared to other recent studies like [Dal Seno et al., 2009] or [Chavarriaga et al., 2010], both reporting ErrP recognition rates of 60-70% at a comparable occurrence frequency. A simulation study also conquered the question whether the application of ErrP error correction is beneficial for a given BCI system. The study investigated how the BCI's communication throughput depends on P300 classification accuracy and ErrP true positive/false positive rates and investigated the role of static parameters like average trial times and action times which are different for every BCI and task. Since many parameters influence the performance of the BCI, a way to analytically calculate the expected communication speed was developed. A probabilistic model based on markov chains was presented that allows to compute the average time such a recurrent error correcting BCI takes to correctly predict an arbitrarily long sequence of commands without errors. This novel model allows to compare different parameter setups and find out which parameter constellation leads to performance improvement or loss. The model might serve as a performance metric, similar to the *bits/min* of Wolpaw [Wolpaw et al., 2002], which puts more focus on comparing different BCI approaches regarding their practical communication time instead of simple information throughput. Another application could be the usage as a function subject to optimization when searching for the best classifier thresholds. Researchers could optimize their BCI by tuning their classifier thresholds using ROC curves (i.e. TPR vs. FPR) whose optimal points can be calculated by the proposed model.

The last topic of this thesis covers novel augmented reality paradigm. Methods for marker recognition, signal processing and an advanced version of the dynamic subtrial algorithm were presented. With the introduction of the extended dynamic subtrial (EDS) algorithm, detection of user inactivity periods became possible. This new method does not require ad-

ditional training like other methods (e.g. [Zhang et al., 2008]) but performs equally well or better. Further it achieves to solve both problems of dynamic adaption of subtrial presentation and no-control state detection in one single step. The feasbility of the EDS was evaluated in a final online experiment (section 7.4). During the experiment, the four participating subjects had to grasp objects with the robot arm and place them at certain locations. Along with a statistical evaluation of accuracy, communication speed and no-control state detection time, a questionnaire was completed by all participants to assess subjective usability values.

The augmented reality-based paradigm was developed under the premise that:

1. Operation of the system should add as few additional cognitive load as possible
2. Operation should be intuitive and effortless
3. No subject training should be required
4. Translation of mental commands should be fast and reliable

The results of questionnaire indicate, that the current system fulfills most of these claims. Only the subjectively perceived communication speed did not achieve good user ratings. A problem which every BCI has to face, to either be accurate or fast. Even though much effort has been put into resolving this isseu partly, the presented methods could not convince the study participants in that sense. On the other hand, all participants collectively rated the system as intuitive to use and very reliable. Since P300 paradigms do not require specific user training sinc evocation of P300 happens subconsciously, claim 1 and 3 were also fulfilled.

8.1 Future directions

This thesis demonstrated the usefulness of augmented reality techniques when it comes to interacting with dynamical environments. The presented system uses marker based object recognition to incorporate real objects into the BCI pradigm. For practical application, this approach is probably not sufficient since everyday objects would have to be equipped with markers. A more convenient way would be to use markerless tracking and object recognition methods (e.g. [Comport et al., 2006], [Guo et al., 2009] or [Ta et al., 2009]). Further, semantic associations with the objects could control context sensitive actions which are possible for a certain object. The current BCI offers only grasping and placing object. For a real application, possible actions for a light switch would be pressing while actions for a banana could be *pick up* or *peel*. However, implementing all of these aspects demands cooperation from more than one scientific field. A joint work from artificial intelligence, computer vision, robotics and BCI researchers could potentially lead to a very sophisticated system. Moving away from the almost historic target audience of locked-in patients, applications to *hands-busy tasks* as encountered in underwater robot control, or crane steering, would also be possible.

Last but not least, the proposed utility metric based on markov chains (see 5.5) will provide a useful tool for other researchers to optimize their error-correcting BCI's. Further adaption

of the chains might be needed to account for different BCI structures. As a logical next step, the framework should be incorporated into the existing robot control BCI as presented in chapter 7. Online experiments are needed to confirm the practical correctness of the theoretical predictions. In addition, a method to automatically switch between the use of ErrP error correction and the standard paradigm is conceivable. If parameters like the average intent recognition time and action time are continuously evaluated, the model can be adapted to reflect the current BCI performance prior to the error correction phase. If the model based on the current parameters shows that error correction is not feasible anymore, the error correction phase could simply be skipped or the classifier threshold can be changed to yield best possible performance under the current circumstances. This way the model will also become more stable over time in case certain parameters do not remain constant during sustained use of the system.

Appendix

Preprocessing	Classifier	α	ErrP	non-ErrP	avg. Acc	Acc CI
(1) ica+raw+DS	***SVM***	0.10	0.72 ± 0.0	0.69 ± 0.0	0.71 ± 0.1	0.68 / 0.74
	LDA	0.10	0.85 ± 0.03	0.86 ± 0.03	0.85 ± 0.05	0.82 / 0.88
	KNN	0.10	0.72 ± 0.01	0.84 ± 0.03	0.78 ± 0.05	0.75 / 0.81
(2) ica+raw+NoDS	***SVM***	0.10	0.86 ± 0.0	0.87 ± 0.0	0.87 ± 0.1	0.84 / 0.90
	LDA	0.10	0.86 ± 0.05	0.89 ± 0.03	0.88 ± 0.05	0.85 / 0.91
	KNN	0.10	0.67 ± 0.03	0.84 ± 0.04	0.75 ± 0.07	0.71 / 0.79
(3) ica+ttest+DS	***SVM***	0.10	0.83 ± 0.0	0.84 ± 0.0	0.83 ± 0.1	0.81 / 0.86
	LDA	0.10	0.87 ± 0.03	0.87 ± 0.02	0.87 ± 0.05	0.84 / 0.90
	KNN	0.90	0.76 ± 0.05	0.80 ± 0.05	0.78 ± 0.07	0.75 / 0.82
(4) ica+ttest+NoDS	***SVM***	0.50	0.85 ± 0.0	0.90 ± 0.0	0.87 ± 0.1	0.82 / 0.89
	LDA	0.50	0.88 ± 0.04	0.88 ± 0.06	0.88 ± 0.05	0.85 / 0.91
	KNN	0.90	0.70 ± 0.04	0.81 ± 0.05	0.76 ± 0.07	0.71 / 0.79
(5) raw+raw+DS	***SVM***	0.10	0.80 ± 0.0	0.79 ± 0.0	0.80 ± 0.1	0.77 / 0.83
	LDA	0.10	0.84 ± 0.02	0.86 ± 0.01	0.85 ± 0.04	0.83 / 0.87
	KNN	0.10	0.71 ± 0.04	0.71 ± 0.06	0.71 ± 0.08	0.67 / 0.75
(6) raw+raw+NoDS	***SVM***	0.10	0.80 ± 0.0	0.81 ± 0.1	0.80 ± 0.1	0.77 / 0.83
	LDA	0.10	0.80 ± 0.03	0.81 ± 0.05	0.81 ± 0.05	0.77 / 0.83
	KNN	0.10	0.72 ± 0.03	0.71 ± 0.05	0.72 ± 0.05	0.69 / 0.74
(7) raw+ttest+DS	***SVM***	0.10	0.82 ± 0.0	0.87 ± 0.0	0.85 ± 0.1	0.80 / 0.88
	LDA	0.50	0.85 ± 0.02	0.89 ± 0.04	0.87 ± 0.04	0.85 / 0.89
	KNN	0.10	0.72 ± 0.02	0.77 ± 0.05	0.75 ± 0.05	0.72 / 0.78
(8) raw+ttest+NoDS	***SVM***	0.50	0.78 ± 0.0	0.79 ± 0.0	0.78 ± 0.0	0.76 / 0.81
	LDA	0.90	0.79 ± 0.02	0.82 ± 0.04	0.80 ± 0.06	0.77 / 0.84
	KNN	0.50	0.75 ± 0.06	0.77 ± 0.01	0.76 ± 0.06	0.73 / 0.80
(9) pca+raw+DS	***SVM***	0.10	0.78 ± 0.0	0.81 ± 0.0	0.80 ± 0.1	0.76 / 0.83
	LDA	0.10	0.85 ± 0.02	0.85 ± 0.02	0.85 ± 0.04	0.82 / 0.87
	KNN	0.10	0.71 ± 0.04	0.72 ± 0.02	0.71 ± 0.04	0.69 / 0.74
(10) pca+raw+NoDS	***SVM***	0.10	0.80 ± 0.0	0.78 ± 0.0	0.79 ± 0.1	0.75 / 0.82
	LDA	0.10	0.74 ± 0.06	0.83 ± 0.04	0.78 ± 0.06	0.75 / 0.82
	KNN	0.10	0.71 ± 0.02	0.70 ± 0.02	0.70 ± 0.06	0.67 / 0.74
(11) pca+ttest+DS	***SVM***	0.10	0.79 ± 0.0	0.85 ± 0.0	0.82 ± 0.0	0.79 / 0.84
	LDA	0.50	0.85 ± 0.03	0.87 ± 0.03	0.86 ± 0.05	0.83 / 0.88
	KNN	0.50	0.72 ± 0.04	0.74 ± 0.03	0.73 ± 0.07	0.69 / 0.78
(12) pca+ttest+NoDS	***SVM***	0.90	0.81 ± 0.0	0.81 ± 0.0	0.81 ± 0.0	0.78 / 0.83
	LDA	0.90	0.76 ± 0.04	0.84 ± 0.06	0.80 ± 0.04	0.78 / 0.82
	KNN	0.50	0.73 ± 0.02	0.72 ± 0.10	0.72 ± 0.07	0.68 / 0.75

Table .1. Performances of subject 1 for the tested preprocessing and classifier combinations. The optimal parameter α was obtained in an iterative search. (DS=Downsampling to 32Hz)

Preprocessing	Classifier	α	ErrP	non-ErrP	avg. Acc	Acc CI
(1) ica+raw+DS	***SVM***	0.10	0.79 ± 0.0	0.81 ± 0.0	0.80 ± 0.1	0.76 / 0.85
	LDA	0.10	0.85 ± 0.05	0.85 ± 0.03	0.84 ± 0.10	0.77 / 0.89
	KNN	0.10	0.66 ± 0.04	0.70 ± 0.08	0.68 ± 0.09	0.61 / 0.73
(2) ica+raw+NoDS	***SVM***	0.10	0.77 ± 0.0	0.81 ± 0.1	0.79 ± 0.1	0.74 / 0.83
	LDA	0.10	0.76 ± 0.03	0.80 ± 0.01	0.78 ± 0.10	0.73 / 0.84
	KNN	0.10	0.70 ± 0.05	0.70 ± 0.10	0.70 ± 0.10	0.65 / 0.77
(3) ica+ttest+DS	***SVM***	0.10	0.82 ± 0.0	0.77 ± 0.1	0.80 ± 0.1	0.74 / 0.86
	LDA	0.50	0.82 ± 0.03	0.85 ± 0.04	0.84 ± 0.08	0.80 / 0.88
	KNN	0.10	0.73 ± 0.06	0.74 ± 0.07	0.74 ± 0.12	0.66 / 0.80
(4) ica+ttest+NoDS	***SVM***	0.50	0.80 ± 0.1	0.85 ± 0.0	0.82 ± 0.1	0.78 / 0.86
	LDA	0.90	0.80 ± 0.01	0.83 ± 0.06	0.81 ± 0.10	0.75 / 0.86
	KNN	0.10	0.73 ± 0.07	0.76 ± 0.07	0.74 ± 0.08	0.70 / 0.79
(5) raw+raw+DS	***SVM***	0.10	0.79 ± 0.1	0.74 ± 0.0	0.77 ± 0.1	0.71 / 0.83
	LDA	0.10	0.81 ± 0.07	0.73 ± 0.07	0.77 ± 0.10	0.70 / 0.82
	KNN	0.10	0.70 ± 0.05	0.74 ± 0.03	0.72 ± 0.09	0.67 / 0.77
(6) raw+raw+NoDS	***SVM***	0.10	0.84 ± 0.0	0.76 ± 0.1	0.80 ± 0.1	0.73 / 0.85
	LDA	0.10	0.80 ± 0.05	0.70 ± 0.05	0.75 ± 0.05	0.72 / 0.78
	KNN	0.10	0.68 ± 0.09	0.73 ± 0.03	0.71 ± 0.09	0.66 / 0.76
(7) raw+ttest+DS	***SVM***	0.50	0.76 ± 0.0	0.79 ± 0.0	0.77 ± 0.1	0.71 / 0.82
	LDA	0.50	0.84 ± 0.06	0.80 ± 0.03	0.82 ± 0.08	0.78 / 0.87
	KNN	0.50	0.75 ± 0.05	0.73 ± 0.07	0.74 ± 0.11	0.67 / 0.80
(8) raw+ttest+NoDS	***SVM***	0.50	0.81 ± 0.1	0.70 ± 0.0	0.75 ± 0.1	0.70 / 0.81
	LDA	0.10	0.82 ± 0.03	0.80 ± 0.05	0.81 ± 0.08	0.77 / 0.86
	KNN	0.50	0.69 ± 0.09	0.79 ± 0.08	0.74 ± 0.12	0.69 / 0.80
(9) pca+raw+DS	***SVM***	0.10	0.71 ± 0.1	0.74 ± 0.1	0.72 ± 0.1	0.65 / 0.79
	LDA	0.10	0.84 ± 0.01	0.76 ± 0.00	0.80 ± 0.10	0.74 / 0.86
	KNN	0.10	0.66 ± 0.04	0.76 ± 0.06	0.71 ± 0.11	0.64 / 0.77
(10) pca+raw+NoDS	***SVM***	0.10	0.78 ± 0.0	0.73 ± 0.0	0.76 ± 0.1	0.72 / 0.80
	LDA	0.10	0.79 ± 0.05	0.67 ± 0.06	0.73 ± 0.11	0.68 / 0.80
	KNN	0.10	0.64 ± 0.04	0.76 ± 0.07	0.70 ± 0.12	0.62 / 0.75
(11) pca+ttest+DS	***SVM***	0.10	0.80 ± 0.0	0.77 ± 0.0	0.79 ± 0.1	0.73 / 0.84
	LDA	0.50	0.85 ± 0.06	0.82 ± 0.09	0.83 ± 0.07	0.79 / 0.86
	KNN	0.10	0.72 ± 0.12	0.81 ± 0.08	0.76 ± 0.11	0.69 / 0.82
(12) pca+ttest+NoDS	***SVM***	0.90	0.79 ± 0.0	0.73 ± 0.1	0.76 ± 0.1	0.71 / 0.82
	LDA	0.50	0.76 ± 0.04	0.74 ± 0.04	0.75 ± 0.08	0.70 / 0.80
	KNN	0.90	0.62 ± 0.09	0.74 ± 0.13	0.68 ± 0.13	0.61 / 0.77

Table .2. Performances of subject 2 for the tested preprocessing and classifier combinations. The optimal parameter α was obtained in an iterative search. (DS=Downsampling to 32Hz)

Preprocessing	Classifier	α	ErrP	non-ErrP	avg. Acc	Acc CI
(1) ica+raw+DS	***SVM***	0.10	0.86 ± 0.0	0.88 ± 0.0	0.87 ± 0.0	0.85 / 0.89
	LDA	0.10	0.93 ± 0.01	0.93 ± 0.02	0.93 ± 0.03	0.92 / 0.94
	KNN	0.10	0.79 ± 0.01	0.78 ± 0.03	0.79 ± 0.04	0.77 / 0.81
(2) ica+raw+NoDS	***SVM***	0.10	0.93 ± 0.0	0.93 ± 0.0	0.93 ± 0.0	0.91 / 0.95
	LDA	0.10	0.94 ± 0.01	0.93 ± 0.01	0.94 ± 0.02	0.92 / 0.95
	KNN	0.10	0.77 ± 0.04	0.74 ± 0.01	0.76 ± 0.05	0.73 / 0.78
(3) ica+ttest+DS	***SVM***	0.10	0.92 ± 0.0	0.92 ± 0.0	0.92 ± 0.0	0.90 / 0.94
	LDA	0.10	0.94 ± 0.01	0.94 ± 0.02	0.94 ± 0.02	0.93 / 0.95
	KNN	0.10	0.86 ± 0.02	0.83 ± 0.03	0.85 ± 0.05	0.81 / 0.87
(4) ica+ttest+NoDS	***SVM***	0.90	0.92 ± 0.0	0.93 ± 0.0	0.92 ± 0.0	0.90 / 0.95
	LDA	0.90	0.94 ± 0.01	0.92 ± 0.02	0.93 ± 0.03	0.92 / 0.95
	KNN	0.10	0.84 ± 0.01	0.81 ± 0.04	0.82 ± 0.04	0.81 / 0.85
(5) raw+raw+DS	***SVM***	0.10	0.91 ± 0.0	0.88 ± 0.0	0.90 ± 0.0	0.88 / 0.91
	LDA	0.10	0.93 ± 0.00	0.93 ± 0.01	0.93 ± 0.03	0.91 / 0.94
	KNN	0.10	0.85 ± 0.03	0.77 ± 0.03	0.81 ± 0.05	0.78 / 0.84
(6) raw+raw+NoDS	***SVM***	0.10	0.89 ± 0.0	0.86 ± 0.0	0.87 ± 0.0	0.85 / 0.89
	LDA	0.10	0.91 ± 0.02	0.88 ± 0.03	0.89 ± 0.03	0.88 / 0.91
	KNN	0.10	0.84 ± 0.04	0.70 ± 0.05	0.77 ± 0.06	0.74 / 0.80
(7) raw+ttest+DS	***SVM***	0.10	0.91 ± 0.0	0.91 ± 0.0	0.91 ± 0.0	0.90 / 0.92
	LDA	0.50	0.94 ± 0.01	0.94 ± 0.00	0.94 ± 0.03	0.93 / 0.96
	KNN	0.10	0.86 ± 0.02	0.77 ± 0.02	0.81 ± 0.04	0.79 / 0.84
(8) raw+ttest+NoDS	***SVM***	0.90	0.90 ± 0.0	0.86 ± 0.0	0.88 ± 0.0	0.86 / 0.90
	LDA	0.90	0.91 ± 0.02	0.88 ± 0.02	0.89 ± 0.04	0.87 / 0.91
	KNN	0.10	0.87 ± 0.04	0.76 ± 0.03	0.81 ± 0.05	0.79 / 0.84
(9) pca+raw+DS	***SVM***	0.10	0.90 ± 0.0	0.90 ± 0.0	0.90 ± 0.0	0.88 / 0.92
	LDA	0.10	0.93 ± 0.01	0.94 ± 0.03	0.93 ± 0.03	0.92 / 0.95
	KNN	0.10	0.87 ± 0.03	0.73 ± 0.05	0.80 ± 0.04	0.78 / 0.82
(10) pca+raw+NoDS	***SVM***	0.10	0.89 ± 0.0	0.85 ± 0.0	0.87 ± 0.0	0.84 / 0.89
	LDA	0.10	0.90 ± 0.02	0.89 ± 0.02	0.89 ± 0.03	0.87 / 0.91
	KNN	0.10	0.85 ± 0.01	0.72 ± 0.02	0.79 ± 0.05	0.75 / 0.81
(11) pca+ttest+DS	***SVM***	0.10	0.93 ± 0.0	0.93 ± 0.0	0.93 ± 0.0	0.91 / 0.95
	LDA	0.90	0.94 ± 0.01	0.94 ± 0.02	0.94 ± 0.03	0.91 / 0.95
	KNN	0.10	0.90 ± 0.02	0.79 ± 0.03	0.85 ± 0.04	0.82 / 0.87
(12) pca+ttest+NoDS	***SVM***	0.90	0.89 ± 0.0	0.86 ± 0.0	0.87 ± 0.0	0.84 / 0.89
	LDA	0.10	0.94 ± 0.01	0.89 ± 0.01	0.92 ± 0.03	0.90 / 0.93
	KNN	0.10	0.86 ± 0.01	0.75 ± 0.02	0.81 ± 0.03	0.79 / 0.82

Table .3. Performances of subject 3 for the tested preprocessing and classifier combinations. The optimal parameter α was obtained in an iterative search. (DS=Downsampling to 32Hz)

Preprocessing	Classifier	α	ErrP	non-ErrP	avg. Acc	Acc CI
(1) ica+raw+DS	***SVM***	0.10	0.67 ± 0.1	0.69 ± 0.0	0.68 ± 0.1	0.64 / 0.72
	LDA	0.10	0.79 ± 0.03	0.79 ± 0.01	0.79 ± 0.06	0.76 / 0.82
	KNN	0.10	0.53 ± 0.08	0.80 ± 0.04	0.67 ± 0.08	0.62 / 0.71
(2) ica+raw+NoDS	***SVM***	0.10	0.76 ± 0.0	0.76 ± 0.0	0.76 ± 0.0	0.73 / 0.79
	LDA	0.10	0.76 ± 0.03	0.75 ± 0.01	0.76 ± 0.04	0.74 / 0.79
	KNN	0.10	0.53 ± 0.04	0.76 ± 0.01	0.64 ± 0.07	0.60 / 0.67
(3) ica+ttest+DS	***SVM***	0.10	0.76 ± 0.0	0.76 ± 0.0	0.76 ± 0.1	0.72 / 0.79
	LDA	0.10	0.79 ± 0.03	0.80 ± 0.06	0.80 ± 0.06	0.76 / 0.83
	KNN	0.10	0.59 ± 0.02	0.77 ± 0.01	0.68 ± 0.05	0.65 / 0.71
(4) ica+ttest+NoDS	***SVM***	0.90	0.78 ± 0.0	0.77 ± 0.1	0.77 ± 0.1	0.74 / 0.81
	LDA	0.90	0.80 ± 0.04	0.74 ± 0.05	0.77 ± 0.05	0.74 / 0.80
	KNN	0.10	0.62 ± 0.04	0.77 ± 0.03	0.69 ± 0.06	0.66 / 0.73
(5) raw+raw+DS	***SVM***	0.10	0.73 ± 0.0	0.74 ± 0.0	0.73 ± 0.1	0.69 / 0.77
	LDA	0.10	0.76 ± 0.04	0.78 ± 0.02	0.77 ± 0.06	0.74 / 0.81
	KNN	0.10	0.58 ± 0.07	0.72 ± 0.03	0.65 ± 0.06	0.62 / 0.69
(6) raw+raw+NoDS	***SVM***	0.10	0.69 ± 0.0	0.68 ± 0.0	0.68 ± 0.1	0.64 / 0.72
	LDA	0.10	0.70 ± 0.04	0.68 ± 0.04	0.69 ± 0.04	0.67 / 0.71
	KNN	0.10	0.66 ± 0.05	0.66 ± 0.06	0.66 ± 0.06	0.62 / 0.69
(7) raw+ttest+DS	***SVM***	0.50	0.74 ± 0.0	0.79 ± 0.0	0.76 ± 0.1	0.73 / 0.80
	LDA	0.10	0.76 ± 0.01	0.78 ± 0.03	0.77 ± 0.06	0.74 / 0.80
	KNN	0.10	0.63 ± 0.04	0.74 ± 0.05	0.69 ± 0.08	0.63 / 0.73
(8) raw+ttest+NoDS	***SVM***	0.90	0.68 ± 0.0	0.73 ± 0.0	0.70 ± 0.1	0.67 / 0.73
	LDA	0.10	0.70 ± 0.04	0.69 ± 0.05	0.70 ± 0.07	0.65 / 0.74
	KNN	0.50	0.63 ± 0.05	0.71 ± 0.05	0.67 ± 0.08	0.63 / 0.71
(9) pca+raw+DS	***SVM***	0.10	0.73 ± 0.0	0.73 ± 0.0	0.73 ± 0.0	0.70 / 0.75
	LDA	0.10	0.77 ± 0.02	0.76 ± 0.05	0.77 ± 0.06	0.73 / 0.80
	KNN	0.10	0.57 ± 0.02	0.73 ± 0.03	0.65 ± 0.06	0.61 / 0.68
(10) pca+raw+NoDS	***SVM***	0.10	0.67 ± 0.1	0.69 ± 0.0	0.68 ± 0.1	0.65 / 0.72
	LDA	0.10	0.69 ± 0.04	0.70 ± 0.02	0.69 ± 0.05	0.66 / 0.72
	KNN	0.10	0.63 ± 0.07	0.66 ± 0.05	0.65 ± 0.07	0.60 / 0.68
(11) pca+ttest+DS	***SVM***	0.50	0.74 ± 0.0	0.75 ± 0.0	0.75 ± 0.1	0.71 / 0.78
	LDA	0.50	0.76 ± 0.04	0.79 ± 0.02	0.77 ± 0.04	0.75 / 0.80
	KNN	0.10	0.59 ± 0.06	0.75 ± 0.05	0.67 ± 0.09	0.62 / 0.71
(12) pca+ttest+NoDS	***SVM***	0.90	0.70 ± 0.0	0.73 ± 0.0	0.71 ± 0.1	0.69 / 0.75
	LDA	0.90	0.76 ± 0.05	0.71 ± 0.05	0.73 ± 0.07	0.70 / 0.78
	KNN	0.90	0.59 ± 0.06	0.72 ± 0.04	0.66 ± 0.07	0.61 / 0.70

Table .4. Performances of subject 4 for the tested preprocessing and classifier combinations. The optimal parameter α was obtained in an iterative search. (DS=Downsampling to 32Hz)

Preprocessing	Classifier	α	ErrP	non-ErrP	avg. Acc	Acc CI
(1) ica+raw+DS	***SVM***	0.10	0.80 ± 0.1	0.80 ± 0.1	0.80 ± 0.1	0.73 / 0.87
	LDA	0.10	0.86 ± 0.03	0.89 ± 0.04	0.87 ± 0.07	0.83 / 0.91
	KNN	0.10	0.79 ± 0.05	0.64 ± 0.08	0.72 ± 0.09	0.67 / 0.77
(2) ica+raw+NoDS	***SVM***	0.10	0.89 ± 0.0	0.93 ± 0.0	0.91 ± 0.1	0.87 / 0.95
	LDA	0.10	0.86 ± 0.02	0.93 ± 0.02	0.90 ± 0.07	0.85 / 0.93
	KNN	0.10	0.73 ± 0.04	0.65 ± 0.09	0.69 ± 0.08	0.65 / 0.74
(3) ica+ttest+DS	***SVM***	0.90	0.82 ± 0.0	0.86 ± 0.0	0.84 ± 0.1	0.78 / 0.88
	LDA	0.90	0.86 ± 0.08	0.89 ± 0.06	0.87 ± 0.09	0.82 / 0.92
	KNN	0.50	0.78 ± 0.04	0.80 ± 0.07	0.79 ± 0.10	0.72 / 0.84
(4) ica+ttest+NoDS	***SVM***	0.90	0.88 ± 0.0	0.91 ± 0.0	0.89 ± 0.1	0.85 / 0.93
	LDA	0.50	0.88 ± 0.03	0.92 ± 0.06	0.90 ± 0.07	0.85 / 0.93
	KNN	0.10	0.75 ± 0.06	0.83 ± 0.05	0.79 ± 0.11	0.73 / 0.85
(5) raw+raw+DS	***SVM***	0.10	0.74 ± 0.0	0.82 ± 0.0	0.78 ± 0.1	0.72 / 0.84
	LDA	0.10	0.86 ± 0.02	0.86 ± 0.03	0.86 ± 0.08	0.81 / 0.90
	KNN	0.10	0.67 ± 0.02	0.66 ± 0.03	0.66 ± 0.11	0.60 / 0.72
(6) raw+raw+NoDS	***SVM***	0.10	0.82 ± 0.0	0.85 ± 0.1	0.83 ± 0.1	0.78 / 0.88
	LDA	0.10	0.83 ± 0.05	0.82 ± 0.05	0.82 ± 0.07	0.78 / 0.86
	KNN	0.10	0.68 ± 0.06	0.70 ± 0.04	0.69 ± 0.07	0.65 / 0.73
(7) raw+ttest+DS	***SVM***	0.10	0.80 ± 0.0	0.85 ± 0.0	0.83 ± 0.1	0.77 / 0.86
	LDA	0.10	0.87 ± 0.03	0.86 ± 0.02	0.86 ± 0.08	0.82 / 0.91
	KNN	0.10	0.77 ± 0.08	0.68 ± 0.02	0.73 ± 0.11	0.66 / 0.79
(8) raw+ttest+NoDS	***SVM***	0.90	0.83 ± 0.0	0.83 ± 0.1	0.83 ± 0.1	0.78 / 0.89
	LDA	0.10	0.85 ± 0.03	0.86 ± 0.03	0.85 ± 0.07	0.80 / 0.89
	KNN	0.10	0.77 ± 0.05	0.76 ± 0.08	0.76 ± 0.12	0.69 / 0.83
(9) pca+raw+DS	***SVM***	0.10	0.73 ± 0.0	0.78 ± 0.1	0.76 ± 0.1	0.68 / 0.81
	LDA	0.10	0.83 ± 0.03	0.84 ± 0.02	0.84 ± 0.08	0.79 / 0.88
	KNN	0.10	0.68 ± 0.05	0.64 ± 0.04	0.66 ± 0.11	0.61 / 0.73
(10) pca+raw+NoDS	***SVM***	0.10	0.82 ± 0.1	0.79 ± 0.1	0.80 ± 0.1	0.73 / 0.87
	LDA	0.10	0.82 ± 0.01	0.77 ± 0.07	0.79 ± 0.08	0.74 / 0.84
	KNN	0.10	0.63 ± 0.06	0.66 ± 0.09	0.65 ± 0.13	0.58 / 0.73
(11) pca+ttest+DS	***SVM***	0.10	0.79 ± 0.1	0.85 ± 0.1	0.82 ± 0.1	0.75 / 0.88
	LDA	0.10	0.85 ± 0.05	0.86 ± 0.06	0.85 ± 0.08	0.81 / 0.90
	KNN	0.10	0.74 ± 0.05	0.69 ± 0.05	0.71 ± 0.09	0.66 / 0.76
(12) pca+ttest+NoDS	***SVM***	0.10	0.82 ± 0.0	0.83 ± 0.0	0.83 ± 0.1	0.79 / 0.86
	LDA	0.90	0.86 ± 0.05	0.82 ± 0.08	0.84 ± 0.10	0.79 / 0.89
	KNN	0.10	0.74 ± 0.11	0.71 ± 0.09	0.73 ± 0.11	0.66 / 0.78

Table .5. Performances of subject 5 for the tested preprocessing and classifier combinations. The optimal parameter α was obtained in an iterative search. (DS=Downsampling to 32Hz)

Benutzerstudie zum Brain-Computer-Interface

Im Folgenden werden ein paar Fragen aufgelistet. Bitte vergib Punkte von 1-10 je nach Zutreffen der Antwort (1: trifft nicht zu, ..., 10: trifft voll zu)

Frage	Note
Das System ist intuitiv bedienbar	8
Das System ist ermüdend.	10
Das System reagiert so wie ich es erwarte.	7
Das System ist sehr langsam.	8
Das System ist kompliziert.	2
Das System reagiert anders, wenn Ich mich bewege oder spreche.	3
Die Konzentration lässt nach der Trainingsphase nach.	10
Ich musste mich stark konzentrieren, damit überhaupt etwas passiert.	7
Meine Befehle wurden zuverlässig erkannt.	8

Kommentar:

Benutzerstudie zum Brain-Computer-Interface

Im Folgenden werden ein paar Fragen aufgelistet. Bitte vergib Punkte von 1-10 je nach Zutreffen der Antwort (1: trifft nicht zu, ..., 10: trifft voll zu)

Frage	Note
Das System ist intuitiv bedienbar	8
Das System ist ermüdend.	9
Das System reagiert so wie ich es erwarte.	8
Das System ist sehr langsam.	7
Das System ist kompliziert.	3
Das System reagiert anders, wenn Ich mich bewege oder spreche.	5
Die Konzentration lässt nach der Trainingsphase nach.	8
Ich musste mich stark konzentrieren, damit überhaupt etwas passiert.	7
Meine Befehle wurden zuverlässig erkannt.	9

Kommentar:

Benutzerstudie zum Brain-Computer-Interface

Im Folgenden werden ein paar Fragen aufgelistet. Bitte vergib Punkte von 1-10 je nach Zutreffen der Antwort (1: trifft nicht zu, ..., 10: trifft voll zu)

Frage	Note
Das System ist intuitiv bedienbar	8
Das System ist ermüdend.	8
Das System reagiert so wie ich es erwarte.	9
Das System ist sehr langsam.	3
Das System ist kompliziert.	3
Das System reagiert anders, wenn Ich mich bewege oder spreche.	7
Die Konzentration lässt nach der Trainingsphase nach.	8
Ich musste mich stark konzentrieren, damit überhaupt etwas passiert.	8
Meine Befehle wurden zuverlässig erkannt.	9

Kommentar:

System funktioniert besser, wenn das Zählen des Stimulus so schnell wie möglich nach dem Aufblinken erfolgt. Entspanntes zählen scheint schlechter zu funktionieren.

Benutzerstudie zum Brain-Computer-Interface

Im Folgenden werden ein paar Fragen aufgelistet. Bitte vergib Punkte von 1-10 je nach Zutreffen der Antwort (1: trifft nicht zu, ..., 10: trifft voll zu)

Frage	Note
Das System ist intuitiv bedienbar	7
Das System ist ermüdend.	7
Das System reagiert so wie ich es erwarte.	7
Das System ist sehr langsam.	8
Das System ist kompliziert.	3
Das System reagiert anders, wenn Ich mich bewege oder spreche.	3
Die Konzentration lässt nach der Trainingsphase nach.	5
Ich musste mich stark konzentrieren, damit überhaupt etwas passiert.	3
Meine Befehle wurden zuverlässig erkannt.	7

Kommentar:

Bibliography

[Adrian and Matthews, 1934] Adrian, E. and Matthews, B. (1934). The berger rhythm: potential changes from the occipital lobes in man. *Brain*, 57(4):355.

[Bayliss, 2003] Bayliss, J. (2003). Use of the evoked potential p3 component for control in a virtual apartment. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):113–116.

[Bell and Sejnowski, 1995] Bell, A. and Sejnowski, T. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural computation*, 7(6):1129–1159.

[Bin et al., 2009a] Bin, G., Gao, X., Wang, Y., Hong, B., and Gao, S. (2009a). Research frontier: Vep-based brain-computer interfaces: time, frequency, and code modulations. *IEEE Computational Intelligence Magazine*, 4(4):22–26.

[Bin et al., 2009b] Bin, G., Gao, X., Yan, Z., Hong, B., and Gao, S. (2009b). An online multi-channel SSVEP-based brain–computer interface using a canonical correlation analysis method. *Journal of Neural Engineering*, 6:046002.

[Birbaumer et al., 1990] Birbaumer, N., Elbert, T., Canavan, A., and Rockstroh, B. (1990). Slow potentials of the cerebral cortex and behavior. *Physiological Reviews*, 70(1):1.

[Birbaumer et al., 2003] Birbaumer, N., Hinterberger, T., Kübler, A., and Neumann, N. (2003). The thought-translation device (ttd): neurobehavioral mechanisms and clinical outcome. *IEEE Trans Neural Syst Rehabil Eng*, 11(2):120–123.

[Blankertz et al., 2006] Blankertz, B., Dornhege, G., Krauledat, M., Müller, K., Kunzmann, V., Losch, F., and Curio, G. (2006). The berlin brain-computer interface: Eeg-based communication without subject training. *IEEE Trans. Neural Sys. Rehab. Eng. Volume*, 14(2):147–152.

[Brocke, 2004] Brocke, B. (2004). The multilevel approach in sensation seeking: Potentials and findings of a four-level research program. *On the psychobiology of personality: essays in honor of Marvin Zuckerman*, page 267.

[Brooks and Eccles, 1947] Brooks, C. and Eccles, J. (1947). Electrical investigation of the monosynaptic pathway through the spinal cord. *J neurophysiol*, 10:251–274.

[Brualla et al., 1998] Brualla, J., Romero, M., Serrano, M., and Valdizán, J. (1998). Auditory event-related potentials to semantic priming during sleep. *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, 108(3):283–290.

[Buttfield et al., 2006] Buttfield, A., Ferrez, P., and del RMillan, J. (2006). Towards a robust bci: error potentials and online learning. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2):164.

[Buzsaki, 2004] Buzsaki, G. (2004). Large-scale recording of neuronal ensembles. *Nature Neuroscience*, 7(5):446–451.

[Callan et al., 2001] Callan, D., Callan, A., Kroos, C., and Vatikiotis-Bateson, E. (2001). Multimodal contribution to speech perception revealed by independent component analysis: a single-sweep eeg case study. *Cognitive Brain Research*, 10(3):349–353.

[Carmena et al., 2003] Carmena, J., Lebedev, M., Crist, R., ODoherty, J., Santucci, D., Dimitrov, D., Patil, P., Henriquez, C., and Nicolelis, M. (2003). Learning to control a brain-machine interface for reaching and grasping by primates. *PLoS Biol*, 1(2):193–208.

[Chapin et al., 1999] Chapin, J., Moxon, K., Markowitz, R., and Nicolelis, M. (1999). Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. *Nature neuroscience*, 2:664–670.

[Chavarriaga et al., 2010] Chavarriaga, R., Millan, J., et al. (2010). Learning from eeg error-related potentials in noninvasive brain-computer interfaces. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 18(4):381–388.

[Coles et al., 1990] Coles, M., Gratton, G., and Fabiani, M. (1990). Event-related brain potentials. *Principles of psychophysiology: Physical, social, and inferential elements*, pages 413–455.

[Comport et al., 2006] Comport, A., Marchand, E., Pressigout, M., et al. (2006). Real-time markerless tracking for augmented reality: the virtual visual servoing framework. *IEEE Transactions on Visualization and Computer Graphics*, pages 615–628.

[Coulombe et al., 2007] Coulombe, J., Sawan, M., and Gervais, J. (2007). A highly flexible system for microstimulation of the visual cortex: Design and implementation. *IEEE Transactions on Biomedical Circuits and Systems*, 1(4):258–269.

[Dal Seno et al., 2009] Dal Seno, B., Matteucci, M., and Mainardi, L. (2009). On-line detection of p300 and error potentials in a bci speller. In *7th International Conference on Bioelectromagnetism*.

[Delorme and Makeig, 2004] Delorme, A. and Makeig, S. (2004). Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis. *Journal of Neuroscience Methods*, 134(1):9–21.

[Delorme et al., 2007] Delorme, A., Sejnowski, T., and Makeig, S. (2007). Enhanced detection of artifacts in eeg data using higher-order statistics and independent component analysis. *Neuroimage*, 34(4):1443–1449.

[Donchin, 1981] Donchin, E. (1981). Surprise!... surprise. *Psychophysiology*, 18(5):493–513.

[Dornhege et al., 2003] Dornhege, G., Blankertz, B., and Curio, G. (2003). Speeding up classification of multi-channel brain-computer interfaces: Common spatial patterns for slow cortical potentials. In *Proceedings of the 1st International IEEE EMBS Conference on Neural Engineering. Capri 2003*, pages 591–594. Citeseer.

[Dornhege et al., 2004] Dornhege, G., Blankertz, B., Curio, G., and M "uller, K. (2004). Increase information transfer rates in bci by csp extension to multi-class. *Advances in Neural Information Processing Systems*, 16:733–40.

[Durka, 2003] Durka, P. (2003). From wavelets to adaptive approximations: time-frequency parametrization of eeg. *BioMedical Engineering OnLine*, 2(1):1.

[Dvorak, 1984] Dvorak, J. C. (1984). San francisco examiner. Newspaper.

[Eccles, 1964] Eccles, S. (1964). *The physiology of synapses*. Springer.

[Elbert et al., 1980] Elbert, T., Rockstroh, B., Lutzenberger, W., and Birbaumer, N. (1980). Biofeedback of slow cortical potentials. i. *Electroencephalography and clinical neurophysiology*, 48(3):293.

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

[Ferrez, 2008] Ferrez, P. (2008). Error-related eeg potentials generated during simulated brain-computer interaction. *IEEE Transactions on Biomedical Engineering*, 55(3):923–929.

[Ferrez and del Millan, 2005] Ferrez, P. and del Millan, J. (2005). You are wrong!-automatic detection of interaction errors from brain waves. In *INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE*, volume 19, page 1413. Citeseer.

[Ferrez et al., 2007] Ferrez, P., des Prés-Beudin, A., and Millán, J. (2007). Eeg-based brain-computer interaction: Improved accuracy by automatic single-trial error detection. In *21st Annual Conference on Neural Information Processing Systems (NIPS)*. Citeseer.

[Finke et al., 2009] Finke, A., Lenhardt, A., and Ritter, H. (2009). The mindgame: A p300-based brain-computer interface game. *Neural Networks*, 22(9):1329–1333.

[Friedman et al., 2001] Friedman, D., Cycowicz, Y., and Gaeta, H. (2001). The novelty p3: an event-related brain potential (erp) sign of the brain's evaluation of novelty. *Neuroscience and Biobehavioral Reviews*, 25(4):355–373.

[Frigo and Johnson, 2005] Frigo, M. and Johnson, S. G. (2005). The design and implementation of FFTW3. *Proceedings of the IEEE*, 93(2):216–231. special issue on "Program Generation, Optimization, and Platform Adaptation".

[Fuentes et al., 2009] Fuentes, R., Petersson, P., Siesser, W., Caron, M., and Nicolelis, M. (2009). Spinal cord stimulation restores locomotion in animal models of parkinson's disease. *Science*, 323(5921):1578.

[GALAN et al., 2008] GALAN, F., NUTTIN, M., LEW, E., FERREZ, P., VANACKER, G., PHILIPS, J., and MILLAN, J. (2008). A brain-actuated wheelchair: Asynchronous and non-invasive brain-computer interfaces for continuous control of robots. *Clinical neurophysiology*, 119(9):2159–2169.

[Gao et al., 2003] Gao, X., Xu, D., Cheng, M., and Gao, S. (2003). A bci-based environmental controller for the motion-disabled. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):137–140.

[Graimann et al., 2008] Graimann, B., Allison, B., Mandel, C., Lüth, T., Valbuena, D., and Gräser, A. (2008). Non-invasive brain-computer interfaces for semi-autonomous assistive devices. *Robust Intelligent Systems*, page 113.

[Green and Arduini, 1954] Green, J. and Arduini, A. (1954). Hippocampal electrical activity in arousal. *J Neurophysiol*, 17(6):533–557.

[Guger et al., 1999] Guger, C., Harkam, W., Hertnaes, C., and Pfurtscheller, G. (1999). Prosthetic control by an eeg-based brain-computer interface (bci). In *Proc. AAATE 5th European Conference for the Advancement of Assistive Technology*.

[Guo et al., 2008] Guo, F., Hong, B., Gao, X., and Gao, S. (2008). A brain–computer interface using motion-onset visual evoked potential. *Journal of neural engineering*, 5:477–485.

[Guo et al., 2009] Guo, J., Wang, Y., Chen, J., Lin, J., Wu, L., Xue, K., Liu, W., and Zhang, J. (2009). Markerless tracking for augmented reality applied in reconstruction of yuanmingyuan archaeological site. In *Computer-Aided Design and Computer Graphics, 2009. CAD/Graphics' 09. 11th IEEE International Conference on*, pages 324–329. IEEE.

[Harbin et al., 1984] Harbin, T., Marsh, G., and Harvey, M. (1984). Differences in the late components of the event-related potential due to age and to semantic and non-semantic tasks. *Electroencephalography and Clinical Neurophysiology*, 59:489–496.

[Hinterberger et al., 2001] Hinterberger, T., Kaiser, J., Kübler, A., Neumann, N., and Birbaumer, N. (2001). *Sciences of the Interfaces*, chapter The Thought Translation Device and its Applications to the Completely Paralyzed, pages 232–240. Genista-Verlag Tübingen.

[Holcomb, 1988] Holcomb, P. (1988). Automatic and attentional processing: An event-related brain potential analysis of semantic priming. *Brain and Language*, 35(1):66–85.

[Iturrate et al., 2009] Iturrate, I., Antelis, J., and Minguez, J. (2009). Synchronous eeg brain-actuated wheelchair with automated navigation. pages 2530–2537.

[Jasper, 1958a] Jasper, H. (1958a). The 10-20 electrode system of the international federation. *Electroencephalography and Clinical Neurophysiology*, 10:371–375.

[Jasper, 1958b] Jasper, H. H. (1958b). Report on the commitee on methods of clinical examination in electroencephalography. *Electroenceph. clin. Neurophysiol.*, 10(370).

[Jung et al., 1998] Jung, T., Humphries, C., Lee, T., Makeig, S., McKeown, M., Iragui, V., and Sejnowski, T. (1998). Extended ica removes artifacts from electroencephalographic recordings. *Advances in neural information processing systems*, pages 894–900.

[Jung et al., 1999] Jung, T., Makeig, S., Westerfield, M., Townsend, J., Courchesne, E., and Sejnowski, T. (1999). Independent component analysis of single-trial event-related potentials. In *Proceedings of the first international workshop on independent component analysis and blind signal separation*, pages 173–178. Citeseer.

[Kaper, 2006] Kaper, M. (2006). *P300-Based Brain-Computer Interfacing*. PhD thesis, Faculty of Technology, Bielefeld University.

[Kaper et al., 2004] Kaper, M., Meinicke, P., Grossekathoefer, U., Lingner, T., and Ritter, H. (2004). Bci competition 2003 - dataset iib: Support vector machines for the p300 speller paradigm. *IEEE Transactions Biomedical Engineering*, 51.

[Kaper and Ritter, 2004] Kaper, M. and Ritter, H. (2004). Generalizing to new subjects in brain-computer interfacing. In *Proceedings of the 26th IEEE EMBS Annual International Conference (EMBC)*, San Francisco, USA.

[Katayama and Polich, 1996] Katayama, J. and Polich, J. (1996). P300 from one-, two-, and three-stimulus auditory paradigms. *International Journal of Psychophysiology*, 23(1-2):33–40.

[Katayama and Polich, 2001] Katayama, J. and Polich, J. (2001). Stimulus context determines p3a and p3b. *Psychophysiology*, 35(01):23–33.

[Kato and Billinghurst, 1999] Kato, H. and Billinghurst, M. (1999). Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *iwar*, page 85. Published by the IEEE Computer Society.

[Keirn and Aunon, 1990] Keirn, Z. and Aunon, J. (1990). A new mode of communication between man and his surroundings. *IEEE Transactions on Biomedical Engineering*, 37(12):1209–1214.

[Krauledat et al., 2008] Krauledat, M., Tangermann, M., Blankertz, B., and Müller, K. (2008). Towards zero training for brain-computer interfacing. *PLoS One*, 3(8).

[Krusienski et al., 2006] Krusienski, D., Sellers, E., Cabestaing, F., Bayoudh, S., McFarland, D., Vaughan, T., and Wolpaw, J. (2006). A comparison of classification techniques for the p300 speller. *Journal of neural engineering*, 3:299–305.

[Kübler, 2000] Kübler, A. (2000). *Brain computer communication: development of a brain computer interface for locked-in patients on the basis of the psychophysiological self-regulation training of slow cortical potentials(SCP)*. PhD thesis, Universität, Tübingen.

[Kübler et al., 2001] Kübler, A., Kotchoubey, B., Kaiser, J., Wolpaw, J., and Birbaumer, N. (2001). Brain-computer communication: unlocking the locked in. *Psychological Bulletin*, 127(3):358–375.

[Kutas and Hillyard, 1980] Kutas, M. and Hillyard, S. (1980). Event-related brain potentials to semantically inappropriate and surprisingly large words. *Biological Psychology*, 11(2):99–116.

[Kutas and Hillyard, 1983] Kutas, M. and Hillyard, S. (1983). Event-related brain potentials to grammatical errors and semantic anomalies. *Memory & Cognition*, 11(5):539–550.

[Kutas and Hillyard, 1984] Kutas, M. and Hillyard, S. (1984). Brain potentials during reading reflect word expectancy and semantic association. *Nature*, 307(5947):161–163.

[Lalor et al., 2005] Lalor, E., Kelly, S., Finucane, C., Burke, R., Smith, R., Reilly, R., and Mcdarby, G. (2005). Steady-state vep-based brain-computer interface control in an immersive 3d gaming environment. *EURASIP journal on applied signal processing*, 19:3156.

[Ledoit and Wolf, 2004] Ledoit, O. and Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411.

[Lee et al., 2007] Lee, P., Hsieh, J., Wu, C., Shyu, K., and Wu, Y. (2007). Brain computer interface using flash onset and offset visual evoked potentials. *Clinical Neurophysiology*.

[Lee et al., 1999] Lee, T., Girolami, M., and Sejnowski, T. (1999). Independent component analysis using an extended infomax algorithm for mixed subgaussian and supergaussian sources. *Neural computation*, 11(2):417–441.

[Leeb et al., 2007] Leeb, R., Friedman, D., Muller-Putz, G., Scherer, R., Slater, M., and Pfurtscheller, G. (2007). Self-paced(asynchronous) bci control of a wheelchair in virtual

environments: A case study with a tetraplegic. *Computational Intelligence and Neuroscience*, 2007:79642.

[Leeb et al., 2004] Leeb, R., Scherer, R., Lee, F., Bischof, H., and Pfurtscheller, G. (2004). Navigation in virtual environments through motor imagery. In *9th Computer Vision Winter Workshop, CVWW*, volume 4, pages 99–108.

[Lenhardt, 2006] Lenhardt, A. (2006). Implementation of a P300-based online brain-computer interface. Diploma thesis, Faculty of Technology, Bielefeld University.

[Lenhardt et al., 2008] Lenhardt, A., Kaper, M., and Ritter, H. (2008). An adaptive p300-based online brain-computer interface. *IEEE transactions on neural systems and rehabilitation engineering: a publication of the IEEE Engineering in Medicine and Biology Society*, 16(2):121.

[Lindsley et al., 1974] Lindsley, D., Wicke, J., Thompson, R., and Patterson, M. (1974). Bioelectric recording techniques.

[Lotte and Cuntai, 2009] Lotte, F. and Cuntai, G. (2009). An efficient p300-based brain-computer interface with minimal calibration time.

[Martens et al., 2009] Martens, S., Hill, N., Farquhar, J., and Sch
"olkopf, B. (2009). Overlap and refractory effects in a bci speller based on the visual p300 erp. *Journal of Neural Engineering*, 6:026003.

[McFarland and Wolpaw, 2008] McFarland, D. and Wolpaw, J. (2008). Brain-computer interface operation of robotic and prosthetic devices. *Computer*, 41(10):52–56.

[Mellinger et al., 2003] Mellinger, J., Hinterberger, T., Bensch, M., Schr
"oder, M., and Birbaumer, N. (2003). Surfing the web with electrical brain signals: the brain web surfer (bws) for the completely paralysed. *Proc. 2nd World Congr. Int. Soc. Physical and Rehabilitation Medicine*, pages 731–738.

[Melzack and Wall, 1965] Melzack, R. and Wall, P. (1965). Pain mechanisms: a new theory. *Science (New York, NY)*, 150(699):971.

[Middendorf et al., 2000a] Middendorf, M., McMillan, G., Calhoun, G., Jones, K., et al. (2000a). Brain-computer interfaces based on the steady-state visual-evoked response. *IEEE Transactions on Rehabilitation Engineering*, 8(2):211–214.

[Middendorf et al., 2000b] Middendorf, M., McMillan, G., Calhoun, G., and Jones, K. S. (2000b). Brain-computer interfaces based on the steady-state visual-evoked response. *IEEE Trans Rehabil Eng*, 8(2):211–214.

[Millán et al., 2009] Millán, J., Galán, F., Vanhooydonck, D., Lew, E., Philips, J., and Nuttin, M. (2009). Asynchronous non-invasive brain-actuated control of an intelligent wheelchair. In *Conference proceedings:... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, volume 1, page 3361.

[Misra and Holcomb, 2003] Misra, M. and Holcomb, P. (2003). Event-related potential indices of masked repetition priming. *Psychophysiology*, 40(1):115–130.

[Momose, 2007] Momose, K. (2007). Evaluation of an eye gaze point detection method using VEP elicited by multi-pseudorandom stimulation for brain computer interface. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 5063–5066. IEEE.

[Müller-Gerking et al., 1999] Müller-Gerking, J., Pfurtscheller, G., and Flyvbjerg, H. (1999). Designing optimal spatial filters for single-trial eeg classification in a movement task. *Clinical neurophysiology*, 110(5):787–798.

[Muller-Putz and Pfurtscheller, 2008] Muller-Putz, G. and Pfurtscheller, G. (2008). Control of an electrical prosthesis with an SSVEP-based BCI. *Biomedical Engineering, IEEE Transactions on*, 55(1):361–364.

[Näätänen, 1992] Näätänen, R. (1992). *Attention and brain function*. Erlbaum Hillsdale, NJ:.

[Näätänen et al., 1978] Näätänen, R., Gaillard, A., and Mantysalo, S. (1978). Early selective-attention effect on evoked potential reinterpreted. *Acta psychologica*, 42(4):313–329.

[Neumann and Birbaumer, 2003] Neumann, N. and Birbaumer, N. (2003). Predictors of successful self control during brain-computer communication. *British Medical Journal*, 74(8):1117.

[Nicolelis, 2001] Nicolelis, M. (2001). Actions from thoughts. *Nature*, 409(6818):403–408.

[Niedermeyer and Da Silva, 2004] Niedermeyer, E. and Da Silva, F. (2004). *Electroencephalography: basic principles, clinical applications, and related fields*. Lippincott Williams & Wilkins.

[Nieuwenhuis et al., 2005] Nieuwenhuis, S., Aston-Jones, G., and Cohen, J. (2005). Decision making, the p3, and the locus coeruleus-norepinephrine system. *Psychological Bulletin*, 131(4):510.

[Nyquist, 1928] Nyquist, H. (1928). Certain factors affecting telegraph speed. at&t, 1924 and certain topics in telegraph transmission theory. *Trans. American Institute of Elect. Eng*, 47:617–644.

[Palankar et al., 2009] Palankar, M., De Laurentis, K., Alqasemi, R., Veras, E., Dubey, R., Arbel, Y., and Donchin, E. (2009). Control of a 9-dof wheelchair-mounted robotic arm system using a p300 brain computer interface: Initial experiments. In *Proceedings of the 2008 IEEE International Conference on Robotics and Biomimetics*, pages 348–353. IEEE Computer Society.

[Papoulis et al., 2002] Papoulis, A., Pillai, S., and Unnikrishna, S. (2002). *Probability, random variables, and stochastic processes*, volume 73660116. McGraw-Hill New York.

[Patel and Azzam, 2005] Patel, S. and Azzam, P. (2005). Characterization of n200 and p300: selected studies of the event-related potential. *International Journal of Medical Sciences*, 2(4):147.

[Patterson and Grabois, 1986] Patterson, J. and Grabois, M. (1986). Locked-in syndrome: a review of 139 cases. *Stroke*, 17(4):758.

[Pause et al., 1996] Pause, B., Sojka, B., Krauel, K., and Ferstl, R. (1996). The nature of the late positive complex within the olfactory event-related potential (oerp). *Psychophysiology*, 33(4):376–384.

[Pfurtscheller and Berghold, 1989] Pfurtscheller, G. and Berghold, A. (1989). Patterns of cortical activation during planning of voluntary movement. *Electroencephalogr Clin Neurophysiol*, 72(3):250–8.

[Pfurtscheller et al., 2006] Pfurtscheller, G., Brunner, C., Schl
"ogl, A., and Lopes da Silva, F. (2006). Mu rhythm (de) synchronization and eeg single-trial classification of different motor imagery tasks. *NeuroImage*, 31(1):153–159.

[Pfurtscheller and Lopes da Silva, 1999] Pfurtscheller, G. and Lopes da Silva, F. (1999). Event-related eeg/meg synchronization and desynchronization: basic principles. *Clinical Neurophysiology*, 110(11):1842–1857.

[Pfurtscheller et al., 1997] Pfurtscheller, G., Neuper, C., Flotzinger, D., and Pregenzer, M. (1997). Eeg-based discrimination between imagination of right and left hand movement. *Electroencephalography and clinical Neurophysiology*, 103(6):642–651.

[Pfurtscheller et al., 2003] Pfurtscheller, G., Neuper, C., Muller, G., Obermaier, B., Krausz, G., Schlogl, A., Scherer, R., Graimann, B., Keinrath, C., Skliris, D., et al. (2003). Graz-bci: state of the art and clinical applications. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):1–4.

[Pfurtscheller et al., 2010] Pfurtscheller, G., Solis-Escalante, T., Ortner, R., Linortner, P., and Muller-Putz, G. (2010). Self-Paced Operation of an SSVEP-Based Orthosis With and Without an Imagery-Based "Brain Switch:" A Feasibility Study Towards a Hybrid BCI. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 18(4):409–414.

[Polich, 2007] Polich, J. (2007). Updating p300: an integrative theory of p3a and p3b. *Clinical Neurophysiology*, 118(10):2128–2148.

[Polich and Criado, 2006] Polich, J. and Criado, J. (2006). Neuropsychology and neuropharmacology of p3a and p3b. *International Journal of Psychophysiology*, 60(2):172–185.

[Prueckl and Guger, 2009] Prueckl, R. and Guger, C. (2009). A brain-computer interface based on steady state visual evoked potentials for controlling a robot. page 697.

[Rebsamen et al., 2007] Rebsamen, B., Teo, C., Zeng, Q., Ang Jr, M., Burdet, E., Guan, C., Zhang, H., and Laugier, C. (2007). Controlling a wheelchair indoors using thought. *IEEE intelligent systems*, pages 18–24.

[Ritter et al., 2007] Ritter, H., Kaper, M., Lenhardt, A., and Ontrup, J. (2007). Making human–machine interfaces more brain-adequate. In *International Congress Series*, volume 1301, pages 15–21. Elsevier.

[Rockstroh, 1989] Rockstroh, B. (1989). *Slow cortical potentials and behaviour*. Urban & Schwarzenberg.

[Rockstroh et al., 1984] Rockstroh, B., Birbaumer, N., Elbert, T., and Lutzenberger, W. (1984). Operant control of eeg and event-related and slow brain potentials. *Applied Psychophysiology and Biofeedback*, 9(2):139–160.

[Rugg, 1985] Rugg, M. (1985). The effects of semantic priming and word repetition on event-related potentials. *Psychophysiology*, 22(6):642–647.

[Rugg and Coles, 1995] Rugg, M. and Coles, M. (1995). The erp and cognitive psychology: Conceptual issues. *Electrophysiology of mind: Event-related brain potentials and cognition*, 25:27–39.

[Schafer and Marcus, 1973] Schafer, E. and Marcus, M. (1973). Self-stimulation alters human sensory brain responses. *Science*, 181(4095):175.

[Schalk et al., 2004] Schalk, G., McFarland, D., Hinterberger, T., Birbaumer, N., and Wolpaw, J. (2004). Bci2000: a general-purpose brain-computer interface (bci) system. *IEEE Transactions on Biomedical Engineering*, 51(6):1034–1043.

[Schalk et al., 2000] Schalk, G., Wolpaw, J., McFarland, D., and Pfurtscheller, G. (2000). Eeg-based communication: presence of an error potential. *Clinical Neurophysiology*, 111(12):2138–2144.

[Scherer, 2008] Scherer (2008). *Towards Practical Brain-Computer Interfaces: Self-Paced Operation and Reduction of the Number of EEG Sensors*. PhD thesis, Computer Science Faculty, Graz Univ. Technology.

[Schöpfer et al., 2010] Schöpfer, M., Schmidt, F., Pardowitz, M., and Ritter, H. (2010). Open source real-time control software for the kuka light weight robot. In *8th World Congress on Intelligent Control and Automation (WCICA 2010)*.

[Schulz, 2007] Schulz, H. (2007). Phasic or transient? comment on the terminology of the aasm manual for the scoring of sleep and associated events. *Journal of Clinical Sleep Medicine: JCSM: official publication of the American Academy of Sleep Medicine*, 3(7):752.

[Schweighofer and Pinz, 2006] Schweighofer, G. and Pinz, A. (2006). Robust pose estimation from a planar target. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 28:12.

[Serby et al., 2005] Serby, H., Yom-Tov, E., and Inbar, G. (2005). An improved p300-based brain-computer interface. *IEEE Transactions on neural systems and rehabilitation engineering*, 13(1):89–98.

[Shealy et al., 1967] Shealy, C., Mortimer, J., and Reswick, J. (1967). Electrical inhibition of pain by stimulation of the dorsal columns. *Anesth Analg*, 46:489–491.

[Snyder and Hillyard, 1976] Snyder, E. and Hillyard, S. (1976). Long-latency evoked potentials to irrelevant, deviant stimuli. *Behavioral Biology*, 16(3):319–331.

[Stuss et al., 1988] Stuss, D., Picton, T., and Cerri, A. (1988). Electrophysiological manifestations of typicality judgment. *Brain Lang*, 33:260–272.

[Sutter, 1992] Sutter, E. (1992). The brain response interface: communication through visually-induced electrical brain responses. *Journal of Microcomputer Applications*, 15(1):31–45.

[Sutton et al., 1965] Sutton, S., Braren, M., Zubin, J., and John, E. R. (1965). Evoked-potential correlates of stimulus uncertainty. *Science*, 150(700):1187–1188.

[Ta et al., 2009] Ta, D., Chen, W., Gelfand, N., and Pulli, K. (2009). Surftrac: Efficient tracking and continuous object recognition using local feature descriptors. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2937–2944. IEEE.

[Takahashi et al., 2010] Takahashi, H., Yoshikawa, T., and Furuhashi, T. (2010). Reliability-based automatic repeat request with error potential-based error correction for improving p300 speller performance. *Neural Information Processing. Models and Applications*, pages 50–57.

[Thompson and Patterson, 1974] Thompson, R. and Patterson, M. (1974). *Bioelectric recording techniques*. Academic Press.

[Townsend et al., 2010] Townsend, G., LaPallo, B., Boulay, C., Krusienski, D., Frye, G., Hauser, C., Schwartz, N., Vaughan, T., Wolpaw, J., and Sellers, E. (2010). A novel p300-based brain-computer interface stimulus presentation paradigm: Moving beyond rows and columns. *Clinical Neurophysiology*, 121(7):1109–1120.

[van Schie et al., 2004] van Schie, H., Mars, R., Coles, M., and Bekkering, H. (2004). Modulation of activity in medial frontal and motor cortices during error observation. *Nature Neuroscience*, 7(5):549–554.

[Vanderwolf, 1969] Vanderwolf, C. (1969). Hippocampal electrical activity and voluntary movement in the rat. *Electroencephalogr Clin Neurophysiol*, 26(4):407–418.

[Vanhooydonck et al., 2003] Vanhooydonck, D., Demeester, E., Nuttin, M., and Van Brussel, H. (2003). Shared control for intelligent wheelchairs: an implicit estimation of the user intention. In *ASER*. Citeseer.

[Verleger, 1988] Verleger, R. (1988). Event-related potentials and cognition: a critique of the context updating hypothesis and an alternative interpretation of p 3. *Behavioral and brain sciences(Print)*, 11(3):343–427.

[Visconti et al., 2008] Visconti, G., Dal Seno, B., Matteucci, M., and Mainardi, L. (2008). Automatic recognition of error potentials in a p300-based brain-computer interface. pages 238–243.

[Wang et al., 2008] Wang, A., Mouraux, A., Liang, M., Iannetti, G., and Lauwereyns, J. (2008). The enhancement of the n1 wave elicited by sensory stimuli presented at very short interstimulus.

[Warnke et al., 1994] Warnke, A., Remschmidt, H., and Hennighausen, K. (1994). Verbal information processing in dyslexia–data from a follow-up experiment of neuropsychological aspects and eeg. *Acta paedopsychiatrica*, 56(3):203.

[Warwick et al., 2005] Warwick, K., Gasson, M., Hutt, B., Goodhew, I., and Kyberd, P. (2005). Invasive neural prosthesis for neural signal detection and nerve stimulation. *International Journal of Adaptive Control and Signal Processing*, 19(5):365–375.

[Webster, 2007] Webster, J. (2007). *Medical instrumentation: application and design*. Wiley India Pvt. Ltd.

[Welch, 1967] Welch, P. (1967). The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, 15(2):70–73.

[Wessberg et al., 2000] Wessberg, J., Stambaugh, C., Kralik, J., Beck, P., Laubach, M., Chapin, J., Kim, J., Biggs, S., Srinivasan, M., and Nicolelis, M. (2000). Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. *Nature*, 408(6810):361–365.

[Wolpaw, 2007] Wolpaw, J. (2007). Brain–computer interfaces as new brain output pathways. *The Journal of Physiology*, 579(3):613.

[Wolpaw and McFarland, 2004] Wolpaw, J. and McFarland, D. (2004). 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.

[Wolpaw et al., 1991] Wolpaw, J., McFarland, D., Neat, G., and Forneris, C. (1991). An eeg-based brain-computer interface for cursor control. *Electroencephalography & Clinical Neurophysiology*, 78(3):252–259.

[Wolpaw et al., 2003a] Wolpaw, J., McFarland, D., Vaughan, T., and Schalk, G. (2003a). The wadsworth center brain–computer interface (bci) research and development program. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):205.

[Wolpaw et al., 2002] Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., and Vaughan, T. M. (2002). Brain-computer interfaces for communication and control. *Clin Neurophysiol*, 113(6):767–791.

[Wolpaw et al., 2003b] Wolpaw, J. R., McFarland, D. J., Vaughan, T. M., and Schalk, G. (2003b). The wadsworth center brain-computer interface (bci) research and development program. *IEEE Trans Neural Syst Rehabil Eng*, 11(2):204–207.

[Zhang et al., 2008] Zhang, H., Guan, C., and Wang, C. (2008). Asynchronous p300-based brain-computer interfaces: A computational approach with statistical models. *IEEE Transactions on Biomedical Engineering*, 55(6):1754–1763.