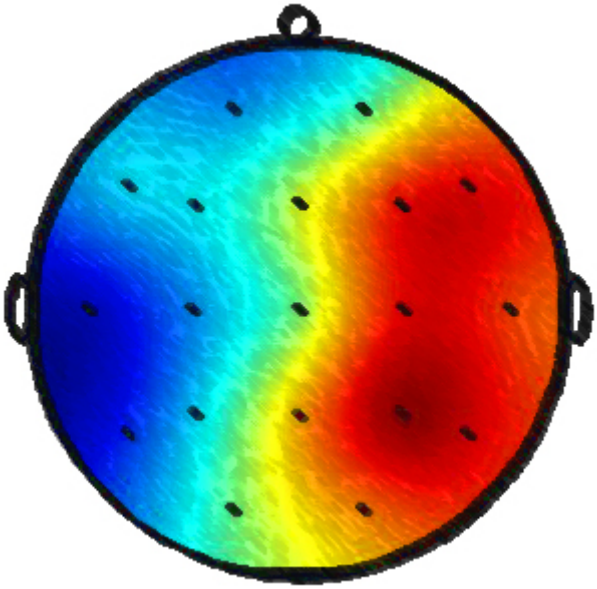
Pioneering research into Brain Computer Interfaces



Mark Wessel

28 March 2006

Delft University of Technology
Faculty of Electrical Engineering, Mathematics, and Computer Science
Mediamatics: Man-Machine Interaction

Graduation committee
Dr. drs. L.J.M. Rothkrantz
Dr. Ir. C.A.P.G. van der Mast
Dr. K. van der Meer

Mark Wessel
Pioneering research into Brain Computer Interfaces

Delft University of Technology
Faculty of Electrical Engineering, Mathematics, and Computer Science
Mediamatics: Man-Machine Interaction
Mekelweg 4
2628 CD Delft
The Netherlands
E-mail: markiewessel@gmail.com
Student number: 1015869

Keywords: brain computer interface, EEG, feature extraction

Abstract

A brain computer interface presents a direct communication channel from the brain. The BCI processes the brain activity and translates it into system commands using feature extraction and classification algorithms.

The overarching goal of this thesis is to make a start in the field of BCI. This thesis starts with an overview of the entire multi-disciplinary field of BCI, covering the basic fundamental ingredients, methods and modalities.

EEG-based BCI experiments have been designed and conducted. The experiments are designed to find distinctive brain patterns which are generated voluntary.

Next to the experiments an environment is created which offers a structured approach to the analysis of the EEG data from the experiments and allows for future continuation of BCI research.

The workbench contains, among others, the following models: ICA, FFT, AR, CSP and LD. The workbench performed well and produced quality results during testing. The quality of the experimental data after evaluation with the constructed workbench appeared to be mediocre at best, caused by the low spatial resolution of the EEG equipment, appliance errors and experimental design faults.

Recommendation for future work are to use different equipment, follow the line of synchronous BCI and construct a classifier to evaluate the data quality and construct an online BCI.

This thesis initiates BCI research at the TU Delft.

Contents

Definitions

ALS	Amyotrophic Lateral Sclerosis; a form of motor neuron disease
AR	Autoregressive model
BCI	Brain Computer Interface
CAR	Common Average Reference
CSP	Common Spatial Pattern
DFT	Discrete Fourier Transform
E-Brain	EEG-based BCI Research Analysis workbench
ECG	Electrocardiography; measurement of electrical heart activity
ECoG	Electrocorticography; invasive measurement of electrical brain activity
EEG	Electroencephalography; measurement of electrical brain activity
EMG	Electromyography; measurement of electrical muscle activity
EOG	Electrooculargraphy; measurement of electrical ocular activity
ERP	Event Related Potential
ERD	Event Related Desynchronization
ERS	Event Related Synchronization
ET	Eye Tracker
FFT	Fast Fourier Transform
FPE	Akaike's Final Prediction Error
GA	Genetic Algorithm
GUI	Graphical User Interface
IC	Independent Component
ICA	Independent Component Analysis
ITR	Information Transfer Rate
LD	Linear Discrimination
LVQ	Linear Vector Quantization
MEG	Magneto Encephalography; measurement of magnetic brain activity
MI	Motor Imagery
MRI	Magnetic Resonance Imaging; imaging using magnetic fields
MRP	Movement Related Potential
NIR	Near Infrared; imaging using near infrared light
NN	Neural Network
PET	Positron Emission Tomography; imaging using x-rays
ROC	Receiver Operator Characteristic
SBC	Schwarz's Bayesian Criterion
SCP	Slow Cortical Potential
SNR	Signal to Noise Ratio
SQL	Structured Query Language
SVM	Support Vector Machine
VEP	Visual Evoked Potential

Preface

Imagine sitting in a room and suddenly you would like to know every thing about the crusades in the Middle Ages, without moving a single muscle but by simply thinking about this idea triggers your brain interface system in searching the required data and transfers it directly to your brain. And you can start discussing it with your friends.

Imagine the old blind guy being able to see again by connecting a camera to his brain, although he lost his sight in a car accident years ago.

Imaging take a well deserved vacation to a tropical island by simply inserting the coordinates in your computer and your mind is off…

Many similar ideas have been uttered over the years and countless movies have been made concerning a link to and from the brain. Movies like *the Matrix* have always fascinated man. For years these stories belonged solely in the realm of science fiction; however the last couple of years a shift has been made in to the realm of reality.

While some proposed ideas may seem to be too futuristic to be possible, it is not unlikely that the principle idea behind it will become reality in the future. In fact it is already taking shape.

Science has always pursued the goal of aiding the human being in an ever increasing effort to increase the quality of life, whether this is to cure and aid the disabled, assist in high workload environment or simply for our pleasure and entertainment. It is apparent that the pursuit of technological advances and increasing science never stops.

Off course these techniques do not come falling from the sky, but require great effort from dedicated researchers all over the world. This is where we start our journey…

But before we do so I would like to thank Leon Rothkrantz for offering the research assignment and providing support and advice in constructing this thesis. Next I would like to thank Pavel Hrubeš in sharing his experience and aiding in the EEG experiments. Furthermore my gratitude goes out to Professor M. Novak and his group *Department of Control and Telematics* of the *faculty of transportation sciences* at the *Czech Technical University* in offering me the ability to complete part of this thesis in the wonderful city of Prague.

Last but certainly not least I am grateful for the support, motivation and advice from my parents, Paul and Dini, and my girlfriend Annelies.

Chapter 1
1. Introduction

1.1. *Brain Computer Interface*

What is a Brain Computer Interface? As mentioned in the preface a BCI represents a direct interface between the brain and a computer or any other system. BCI is a broad concept and comprehends any communication between the brain and a machine in both directions: effectively opening a completely new communication channel without the use of any peripheral nervous system or muscles.

In principle this communication is thought to be two way. But present day BCI is mainly focusing on communication from the brain to the computer. To communicate in the other direction, inputting information in to the brain, more thorough knowledge is required concerning the functioning of the brain. Certain systems like implantable hearing-devices that convert sound waves to electrical signal which in turn directly stimulate the hearing organ already exist today. These are the first steps. The brain on the other hand is on a whole other complexity level compared to the workings of the inner ear.

From here on the focus is on communication directly from the brain to the computer. Most commonly the electrical activity (fields) generated by the neurons is measured, this measuring technique is known as EEG (Electroencephalography). An EEG-based BCI system measures specific features of the EEG-activity and uses these as control signals.

Over the past 15 years the field of BCI has seen a rapidly increasing development rate and obtained the interest of many research groups all over the world. Currently in BCI-research the main focus is on people with severe motor disabilities. This target group has little (other) means of communication and would be greatly assisted by a system that would allow control by merely thinking.

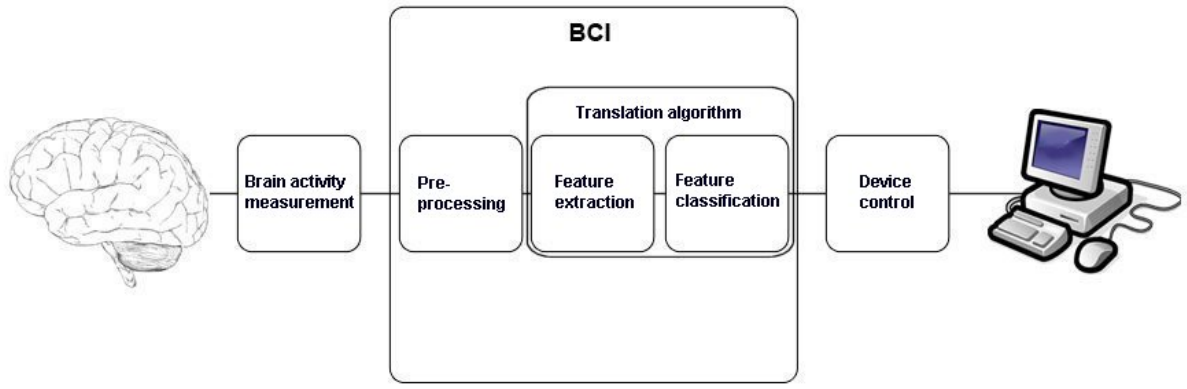


Figure 1.1 Basic BCI layout.

The concept of thinking is perhaps too broad a concept and can actually better be replaced by generating brain patterns. The general picture of a BCI thus becomes that the subject is actively involved with a task which can be measured and recognized by the BCI. This task consists of the following: evoked attention, spontaneous mental performance or mental imagination. The BCI then converts the 'command' into input control for a device (see figure 1.1).

This is the basic idea. With the continuously increasing knowledge of the brain and advances in BCI over time, perhaps BCI will be able to extract actual intentions and thoughts. This however does not appear to be on the cards for the very near future.

1.2. *The nature of EEG*

The reasons for selecting EEG as a measurement method of brain activity are based on the ease of appliance, portability, excellent time resolution and the financial picture. From all the options available, ranging from among others: MRI, PET, MEG to EEG, EEG is the cheapest variant and requires neither professional training nor the personnel to apply it. It consists of a cap of simple electrodes (10-20 system consisting of over 20 electrodes is used at the TUD, see figure 1.2) covering the cortex of the brain on the scalp.

One of the main advantages of EEG is that it gives an excellent temporal resolution (milliseconds range); any change in brain dynamics will be registered almost instantaneous. On the other hand the biggest disadvantage compared to other methods is the very poor spatial resolution (centimetre range), which makes it hard to locate the exact location of the activity.

Aside from the fact that the skull causes spatial smearing of the signal, two third of any activity generated by the neurons is lost due to misalignment of the firing neurons and the fact that any activity can only be measured on the surface of the cortex, which leaves out the majority of the neurons, since the voltages being measured are extremely low.

The EEG and therefore the combined activity from the neurons are characterized by high variability in the signal and large amounts of noise and artifacts. This does not only impose the need for heavy data processing but also makes it more difficult to predict and model the signal.

Figure 1.2 A typical day at the office.

1.3. *Problem domain*

Currently BCI has numerous problems to overcome: the gap in the knowledge concerning the functioning of the brain, signal noise sensitivity, the enormous variability in the brain dynamics, the complexity of the process of capturing intentions and conversion into action. On top of this there are questions concerning the measurement techniques, analysis & classification methods and the manner of interpreting the results. All these issues lead to a system which currently has a low transfer rate of

information. And is thus unable to compete with other methods of communication like for instance: sight, hearing or touching: our basic communication tools.
Another reason for the low bitrate is the fact that the subject must generate reproducible and distinctive brain patterns which at the very least will require more than a second. As mentioned previously the high levels of noise combined with the highly variable EEG signal lead to difficult extractable features.

But from an experimental point of view, it's the only trustful source of information. Brain processes are ever present. The brain is unable *not* to communicate.

1.4. *Relevance*

BCI is a very young field and from the larger perspective it is just starting up, however if significant gains can be made in both performance and practical appliance than a bright future lies ahead. Currently for every practical problem BCI proposes a solution, superior alternatives are present. This is off course not a valid reason to stop pursuing the improvement of brain computer interfaces.

Numerous examples can be given where BCI can be used ranging from helping disabled people to releasing people in high stress/workload environment to even entertainment. But for now the focus is on motor-disabled (paralyzed) people, this group can profit the most from this new technique. Mainly because they have little other means of communication and interaction.

Where BCI is today, advantages can be made in every aspect of this multidisciplinary field from the very start (increasing knowledge of the human body, brain and mind), the measuring methods, the actual BCI recognition system, adopting the system to be used and development of new paradigms.

1.5. *Scope*

BCI covers an extensive area of research over multiple fields. Setting the scope for this master thesis was one of the more difficult initial problems to tackle. Since no previous experience was available within the group of MMI at the TU Delft, it was hard to precisely define and confine the scope of the thesis. Since the possibilities, level of complexity and how much time it would consume were unknown quantities. Therefore the scope was set, by actually performing tasks and gaining knowledge, reviewing further possibilities to continue.

This thesis is the start of a new research line concerning BCI at the TU Delft and in part continues with a collaboration with work of colleagues from the technical university of Prague on microsleeps [36, 37]; part of this master thesis is performed there. In Prague experience with EEG-based experiments was available and their knowledge aided in performing similar experiments here in Delft.

The project is confined to performing a BCI-experiment and EEG signal analysis of brain to computer communication with the intention to find distinguishable brain patterns. Future steps are to construct a working online BCI and validation of the results of the initial experiment by other experiments. This would be too time consuming and far reaching for this master thesis.

1.6. *Research questions*

The overarching goal of this master thesis is to make a start with the research in the field of Brain Computer Interfaces; gain insight in the requirements of this particular field. This section will cover the questions which define this thesis. First of all knowledge must be gained concerning BCI.

Research question 1.

What is the current state of BCI?

The next step is designing and performing experiments and acquiring the resulting data.

Research question 2.
How to construct and perform an EEG-based BCI experiment?

The experimental data contains a lot of ambiguous information contaminated with noise and artifacts which must be analyzed. What is the best way to approach this problem, which algorithms and techniques to use? In short:

Research question 3.
How to process and analyze the data?

Analyzing the data is an important step in the BCI process, the quality of this analysis must be assessed. What is to be the best way to proceed; what pitfalls to avoid and what to keep in mind?

Research question 4.
How good is the performance of the data analysis and how useful are the results?

Furthermore it would be extremely useful to not make the same errors in constructing and performing the experiments and gaining increasing knowledge from the experiments in building a working BCI.

Research question 5.
How to proceed in future experiments and BCI research?

1.7. *Objectives*

The goal of this master thesis is to start initial research on EEG-based Brain Computer Interfaces. This contains a twofold proposition: to explore the possibilities of a BCI and to construct an EEG signal analysis workbench for further BCI research at the TU Delft.
Four objectives have been stated for this master thesis:
The master thesis starts with setting up the BCI environment and doing initial tests and experiments; obtaining data.

Objective 1.
Obtaining necessary and sufficient theoretical knowledge concerning Brain Computer Interfaces.

Objective 2.
Initial measurements and setup of EEG and consequently designing and performing EEG-based BCI experiments.

To analyze the experimental data, a structured analysis approach is required. The data obtained from objective 2 has a twofold goal. The first is analysis for scientific purposes and second to provide data for constructing, implementing and testing the analysis tool.

Objective 3.
Design and implement an EEG data analysis tool for a structured analysis approach of EEG-based BCI data.

After construction of the tool, the data can be analyzed. And the tool itself must be tested properly.

Objective 4.

Evaluation of BCI experiments and EEG workbench.

1.8. *Outline*

For the remaining part of this thesis it is chosen to follow the actual chronological construction of the thesis. Since this will be the clearest and most understandable way to present this story.

Therefore first the theory and the current state of BCI is explained in chapter two. Chapter three will describe in detail which tools and equipment have been used during this master thesis to conduct the experiments.

Chapter four will elaborate on the design of the performed experiments. After which chapter five covers all tools created for this research aside from the workbench in order to create a suitable research environment.

Chapter six will cover the design of the workbench and chapter seven continues with its implementation, chapter eight finishes with the testing and evaluation of the workbench.

After it is made clear what the capabilities of the workbench are the results from the experiments are analyzed and covered in chapter nine. This chapter is covered separately from chapter four, where it perhaps intuitively should be, but to understand the analysis of the results, first the workbench must be explained and understood.

Chapter ten will end with the discussion, conclusions and recommendations.

Chapter 2
2. Theory

Since BCI is a multidisciplinary field, a lot of theory exists on every aspect. This chapter will not delve into every facet, but the focus is to draw a global picture of the entire field of BCI concerning the brain, brain potentials and measuring techniques. Followed by a more in depth look at the analysis methods required for feature selection. Concluded by an overview of classification methods and used methods in the field. In the pursuit to answer the first research question: *What is the current state of BCI?*

2.1. *BCI overview*

2.1.1. BCI definition

The definition of BCI as quoted from the first international meeting devoted to BCI research in 1999:

"A brain–computer interface is a communication system that does not depend on the brain's normal output pathways of peripheral nerves and muscles." [64].

2.1.2. The history of BCI

In 1929 Hans Berger was the first to describe the electroencephalograph (EEG) [1], since that day people have speculated the direct use of brain activity for communication and control. This means communication aside from the normal intermediaries of peripheral nerves and muscles [64].

It took until 1970 before the first initiatives were started to develop simple communication systems based on brain activity. The American government (the Advanced Research Project Agency (ARPA)) immediately became interested in this kind of 'bionic' device. They saw the possibility to improve human performance in tasks with a high mental load. This first research made but minimal progress towards its stated goals.

The first fruits of the early research were the visual evoked potentials (VEP) generated by the brain, which could provide a communication channel. This and other early research led to the distinction between control by EEG and control by EMG (activity from scalp or facial muscles). EEG can be divided furthermore in the fundamental distinction between EEG dependent on muscle control (e.g. brain recognition of arm movement) and EEG control that is independent on muscle control (e.g. brain recognition of intention of arm movement). The latter being the focus of BCI: essentially opening a totally new communication channel for humans.

Since its first steps, BCI has gained from the growing recognition of the needs and potentials of people with disabilities, better understanding of brain function, the advent of powerful low-cost computers and the ever increasing interest of researchers for this field [60].

BCI is an interdisciplinary field which involves: neuroscience, psychology, mathematics, computer science, engineering and clinical rehabilitation [60, 64].

2.1.3. The target group

Currently the focus of BCI is mainly on persons that have severe motor disabilities. The focus on this specific group of people is due to the fact that the level of communication that BCI offers cannot (yet) compete with other methods of communications. For healthy subjects BCI is not a reasonable option, this could change in the future as BCI performance progresses, but will certainly take time.

BCI tries to improve the quality of life of people with disabilities like:

- Late-stage amyotrophic lateral sclerosis (ALS).

ALS is a progressive neurodegenerative disease that affects nerve cells in the brain and spinal cord. The progressive degeneration of the motor neurons will eventually lead to their death, as the neurons degenerate they lose the ability to transmit signals. The ability of the brain to initiate and control muscle movement is thereby gradually lost. Patients in a later stage can become totally paralyzed, however their mind remains unaffected. That is why these patients are called 'locked-in'. 'A-myo-trophic' means no-muscle-nourishment, in the spinal cord (lateral), the tissue in that area hardens and gets scarred (sclerosis) [35]. It is reported that about 2 of every 100.000 people have ALS. The rate between male and female is 2 to 1. This chance of getting ALS also increases with age. Once ALS is diagnosed, the patient has on average about two-to-five years to live. Although about 10% live for another ten years. There is no cure [67].

- Brainstem stroke (Locked-in syndrome).
  This is a condition resulting from interruption of the blood flow to areas in the brain; a blockage (blood clot) or rupture of an artery (hemorrhagic stroke) impairs the blood flow. The resulting lack of oxygen will lead to the death of the brain cells in that area. The disconnection of the communication pathway between the brain and the rest of the nerve system leaves the patient totally paralyzed, except for the head. Communication is possible via eye or jaw movements. The term 'stroke' comes from the ancient Greek belief that someone suffering a stroke, was struck down by the Gods. The prevalence of brainstem stroke overall is about 14 per 10.000, but it is most common with people over 65 years (about 70% of the total cases, 1 out of every 100) [68].
- Severe cerebral palsy.
  Cerebral palsy is a condition affecting body movement and muscle coordination. It is caused by damage to areas of the brain. This damage is usually obtained before, during or after birth and during infancy. The faulty development of the motor areas results in the disability of the brain to adequately control movement and posture. Cerebral refers to brain and palsy to muscle weakness. Cerebral palsy is not progressive. The disorder cannot be cured [69]. The prevalence of cerebral palsy is about 2 out of every 1000 births.
- High spinal cord accident resulting in quadriplegia.
  Quadriplegia is the inability to move arm and legs. This kind of disorder mostly occurs if the spinal cord is damaged between the 5^{th} to 7^{th} cervical vertebrae (the neck). This condition mostly occurs due to an accident. Quadriplegia means "four stroke".

The disabilities listed above have in common that the patient cannot use his or her body (especially arms and legs) anymore, but still has a good working mind. These patients are all trapped in their own body. In most cases the patient has a full understanding of what is going on the world around them, but has reduced or no means to communicate with this world (although communication is sometimes possible using a language based on eye- or jaw-movement).

For these cases BCI can offer a possibility to improve the quality of life by giving the ability to communicate through a totally new channel. Probably everybody has the ability to control a BCI device: it is in essence a new skill that must be attained [64]. Some people will perform better (generate more distinctive brain activity) than others. The performance can increase during training. Training is an intrinsically part of BCI, since generating specific brain signals is not something humans do throughout their pre-BCI lives; it can require long training periods to master the ability to communicate via a BCI.

2.2. *Basic BCI elements*

BCI consists of several distinct elements (see figure 1.1). All these elements combined make up the BCI. Basically the system consists of two adaptive controllers; on the one hand there is the user that

generates the commands via electrophysiological input to the BCI. And on the other hand the BCI that recognizes the commands and expresses these as output device controls.

In this BCI overview each element will be explained to such a level that the reader knows what its functions are and why it is required. As well as views on different approaches to problems that arise in designing and implementing a BCI.

The *translation algorithm* will be covered only briefly in this section and will be handled more elaborately in section 2.10.

BCI can be decomposed into four basic elements:

- The input, measuring the activity from the human brain.
- The signal pre-processing, the first step of acquiring usable signals by amplification, applying filters and reducing noise and artifacts of the input signals.
- The translation algorithm, this step compromises *feature extraction* which extracts the most valuable signals from the processed input. And *feature classification* which tries to classify the features into usable output for the next step.
- The output, from the classification is used as a control signal for various applications.

2.3. *The input*

The goal is to acquire knowledge of the intentions of the user either consciously or unconsciously by means of measurement of brain activity. This goal can be achieved in various ways, but it all starts with the brain and thus with the most basic element of the brain: the neuron.

2.3.1. The neuron

A neuron is a cell that uses biochemical reactions to receive, process and transmit information. It consists of the cell body (Soma) in which the cell core (Nucleus) resides (see figure 2.1). Each neuron has one axon; this is a long 'cable'-like part of the neuron which is used to reach other neurons. The soma of a neuron is branched out into dendrites to which axon-ends from other neurons connect.

The dendrites are not in actual physical contact with the axons of other neurons; a small cleft exists between them: the synaptic gap. This is the location where the impulse is transferred.

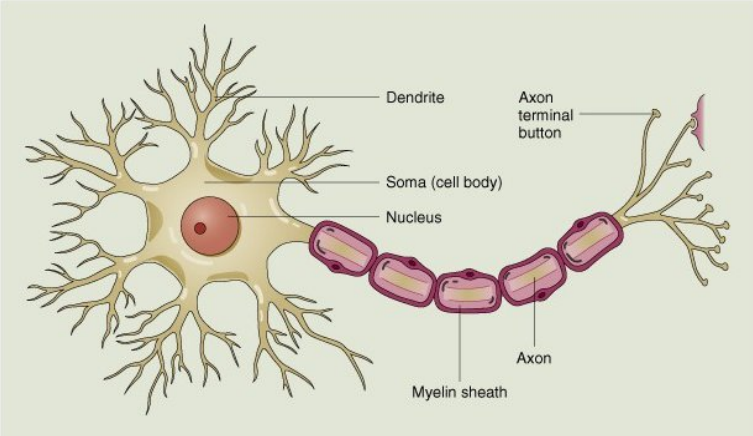


Figure 2.1 Overview of the neuron, image from "John Wiley & Sons".

When a neuron fires, it sends signals to all the neurons that are connected to its axon via the dendrites. The dendrites can be connected to thousands of axons; all incoming signals combined are added through spatial and temporal summation. If the aggregate input reaches a certain threshold, the

neuron will fire and send a signal along its own axon. The strength of this output signal is always the same, no matter the magnitude of the input.

This single signal of a neuron is very weak. The numerous neurons in the brain are constantly active. The generated activity can be measured. It appears to be impossible to measure the individual activity of every neuron. Moreover it is questionable whether it would be a real gain, since neurons work in groups to achieve a certain goal. The activity from a group of neurons however can be measured. For the signals of neurons to be visible using EEG in particular, a couple of conditions need to be met, which are summarized schematically in figure 2.2:

- The electrical activity of the neuron must be perpendicular to the scalp in order for the EEG to fully pick up the field.
- A large number of neurons must fire parallel to each other.
- The neurons must fire in synchrony with the same polarity, in order not to cancel each other out.

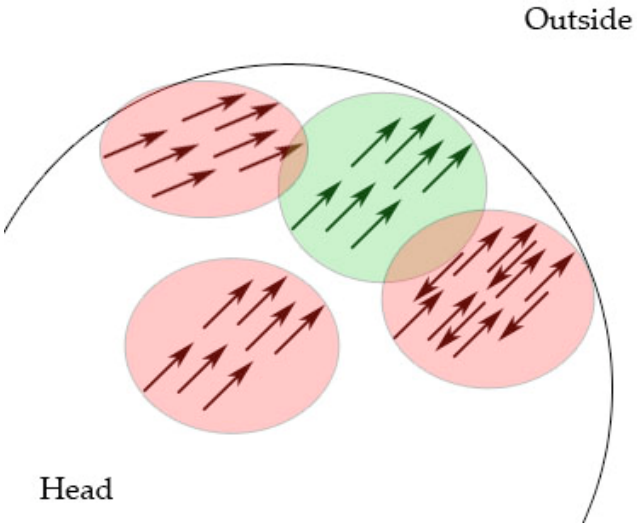


Figure 2-2 Cross-cut of the head: only the green neuronal activity can be measured using EEG.

Because of these requirements the larger part of the total neuronal activity remains invisible for EEG measurement.

2.3.2. The brain

Combining about 100 billion neurons results in what is called the human brain.
The brain consists of the following elements (figure 2.3):

- *The brainstem* is an important relay station. It controls the reflexes and automatic functions, like heart rate and blood pressure and also sleep control.
- *The Cerebellum* integrates information about position and movement from the vestibular system to coordinate limb movement and maintaining equilibrium.
- *Mid-brain: amongst others the Hypothalamus and pituitary gland* control visceral functions, body temperature and behavioural functions like, the body's appetite, sleep patterns, the sexual drive and response to anxiety, aggression and pleasure.
- *The Cerebrum (or cerebral cortex)* receives and integrates information from all of the sense organs and controls the motor functions. Furthermore it contains the higher cerebral functions like: language, cognitive functions and memories. Emotions are also processed in the cerebrum.

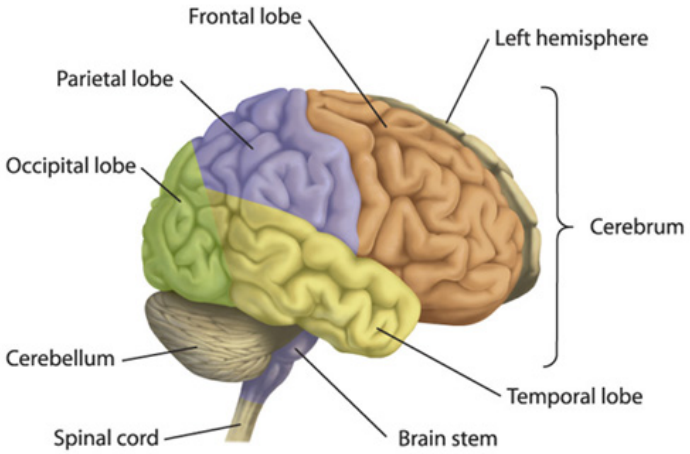


Figure 2.3 Brain overview, the Hypothalamus is localized in the center of the brain and not depicted here; image from "Heart and Stroke Foundation".

The cortex of the cerebrum is part of the brain which is of the most interest for BCI. It is responsible for the higher order cognitive tasks and is near the surface of the scalp. In addition that functionality in the brain appears to be highly local.

The cerebrum is divided into two hemispheres, left and right. The left halve senses and controls the right half of the body and vice versa. Each hemisphere can be divided into four lobes, the frontal, the parietal, the occipital and the temporal (see figure 2.3). The cortex can also by divided in certain areas each of which is specialized for a different function. Especially the sensorimotor cortex is important for BCI. Over this part the entire human body is represented. The size of area corresponds with the importance and complexity of movement of that particular body part (see figure 2.4).

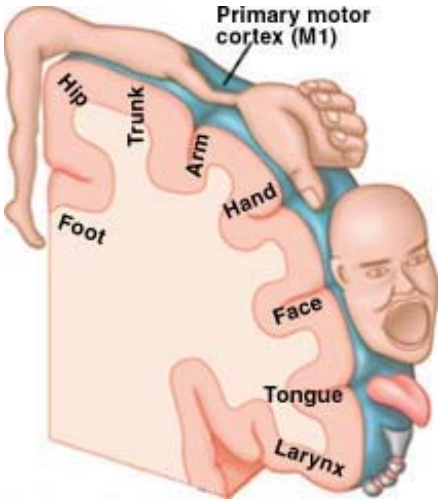


Figure 2.4 Homunculus: body drawn over sensorimotor cortex; image from "brainconnection.com".

In the light of BCI it is important to know in advance in which area to search for activity both spatially and in the frequency domain.

2.4. *Brain activity measurement*

To measure activity in the brain, several different approaches can be applied. Because different phenomena can be measured in different ways: ranging from direct measures like detecting the

electrical currents or magnetic fields to indirect measures like measuring metabolism or blood flow.

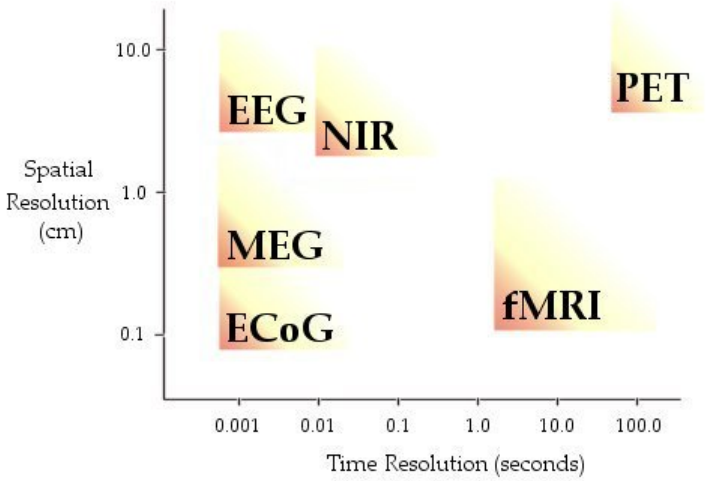


Figure 2.5 Overview of the spatial/time resolution of various measurement techniques.

Here follows a list of the most commonly used methods (see figure 2.5 for an overview of the spatial and time resolution of the mentioned methods).

- *EEG*, Electroencephalography involves recording the (very weak) electrical field generated by action potentials of neurons in the brain using small metal electrodes. The advantages of EEG are the high temporal resolution and (the possibility of) non-invasive measurement. Low spatial resolution, caused by spatial smearing of the skull and high variability in the EEG signal are disadvantages.
- *ECoG*, Electrocorticography involves the electrophysiology of extra-cellular currents. Has both high temporal as good spatial resolution. It is a form of invasive EEG where electrodes are placed directly on the brain. This technique is invasive and therefore requires expensive surgery and comes with significant safety risks for the patient.
- *PET*, Positron Emission Tomography indirectly measures metabolism on a cellular level by tracking injected radioactive isotopes. It is based on the principle that in areas of increased activity the metabolism is on a higher level and more isotopes are supplied by the blood flow. This knowledge can be used to determine which areas are generating activity. Good spatial resolution is an advantage of PET. The really bad temporal resolution (about 2 minutes) is a distinct disadvantage. This is due to the fact that metabolism is a relatively slow process. Moreover ionizing radiation makes this method harmful for the human body and thus unusable for applications like BCI.
- *MEG*, magneto encephalography directly measures the cortical magnetic fields produced by electrical currents. This method is non-invasive and has good spatial and temporal resolution. However the equipment is extremely expensive and due to the very weak magnetic fields requires a very impractical isolation/shielding room (cage of Faraday). The real-time properties for analysis are poor.
- *fMRI*, functional Magnetic Resonance Imaging provides information on brain metabolism using BOLD (Blood Oxygen Level Dependent). fMRI uses strong fluctuating magnetic field to measure the whereabouts of dipoles [25]. The advantages are good spatial resolution and the non-invasiveness. But the temporal resolution is poor (about one second) and this method requires very expensive equipment and procedures.
- *NIR*, Near-infrared light penetrates the human head to sufficient depths to allow functional mapping of the cerebral cortex. Changes in tissue oxygenation cause modulation of absorption and scattering of photons, which can be measured [12]. NIR can measure two responses.

- The slow response corresponds with the attenuation changes due to cerebral heamodynamics.
- The fast response could be used for BCI and is due to changes in scattering properties. This response has a good temporal resolution, but is not yet feasible.

To date the only experimental setup that has been used uses the slow response, which has a poor temporal resolution. Advantages of optical techniques: no interference, user is electrically isolated, non-invasive, non-ionizing and no gel is required. Overall this looks like a promising technique.

2.4.1. EEG

2.4.1.1. Selecting a measurement method – Why EEG?

The best brain measurement method would have a high spatial and temporal resolution, be very cheap, portable and easy to apply non-invasively. This method does not (yet) exist.

Of all methods listed in the previous section, EEG is by far the most commonly used in BCI. The prime reason for this is the excellent temporal resolution which is a necessity for real-time BCI. And although the spatial data resulting from EEG is often distorted and far from perfect, EEG offers direct functional correlation of brain activity.

Another major plus is the ease of applying this method. With a cap containing only a few electrodes measurements can start (see section 2.4.1.3). For practical uses and applications it is small and relatively portable, which improves prospects of future applications (figure 2.6).

Aside from the ease of appliance, this is also a relatively low-cost method, certainly compared to methods like PET, MEG or MRI, which require expensive equipment and skilled professionals to operate.

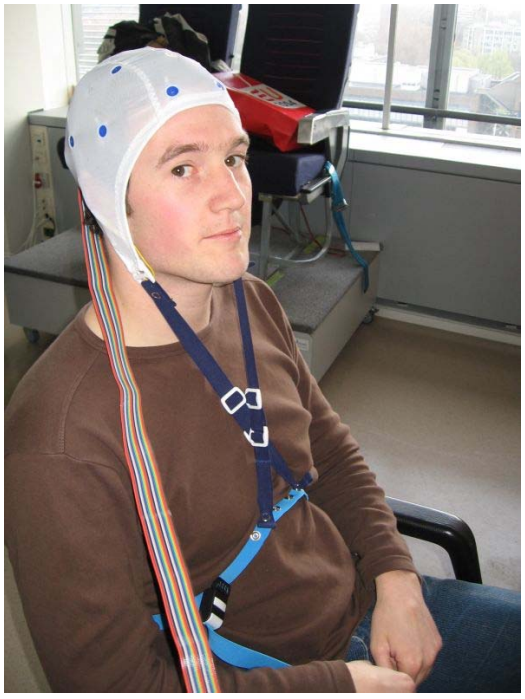


Figure 2.6 Subject one with EEG cap.

Although EEG is the most commonly used, this does not mean that others methods are not feasible. With the continuous improvement of the techniques involved, they can become a viable option in the future; like for instance Near Infrared measurement.

2.4.1.2. Invasiveness

EEG comes in two flavours; the most commonly used in BCI is the non-invasive variant. The electrode is placed on the scalp. The obvious advantage is that it can be safely applied to anyone at any moment without a lot of preparation.

The second variant is the invasive EEG. Instead of attaching the electrode on the skull, it is placed inside. The advantage of this variant is the higher spatial resolution obtained by it. With non-invasive EEG, the skull causes significant spatial smearing of the measured activity: leading to more difficult localization of the original signal, which degrades the quality of the signal.

Invasive electrodes offer the possibility to locate activity far more precise. The obvious drawback is that surgery is required to implant the electrodes. This comes with safety risks and high costs compared to non-invasive EEG. For application there must be large gain from the increased accuracy to validate invasive EEG on human subjects.

2.4.1.3. The 10-20 system

A cap with a number of electrodes is placed on the user's head. At the TU Delft the 10-20 system of electrode placement is used. This is an international standard used for comparing results among different research. The system is based on the relationship of the electrode placement and the underlying area of the cerebral cortex. Each location on the scalp has a letter to identify the hemisphere location (Frontal, Temporal, Central, Parietal and Occipital Lobe) and a number to define the hemisphere. Ranging from 1 to 8, with the even number referring to the right hemisphere and the odd numbers to the left hemisphere (see figure 2.7). The 10-20 refers to the distance (in percentage) between the different electrodes. Reference is needed to measure voltage. Reference electrodes are usually attached to relative stable points where the potential remains constant. Points like the earlobes or mastoid bones behind the ear.

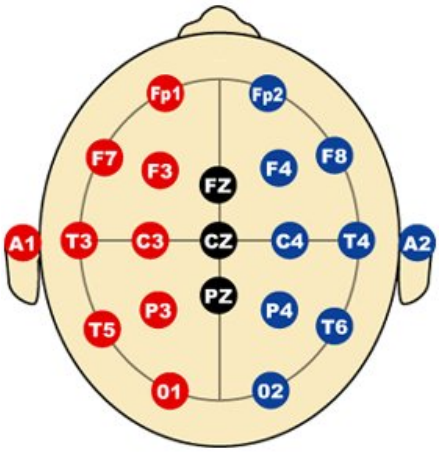


Figure 2.7 The international 10-20 system; image from "Immrama institute".

2.4.1.4. Channels

The electrodes are often referred to as channels. The 10-20 system contains 19 channels for recording. For most BCI application not all of these channels are actually used. Often, for the sake of reducing the calculation complexity, just a few are selected. Discarding channels can still lead to comparable accuracy [9]. Another possibility is selecting channels based on the user's capacity: individual tuning [56]. Or by selecting only certain data of the entire data pool generated by the electrodes.

Experiments indicate that with just a few channels about the same level of accuracy can be achieved. And therefore often the rule of thumb "simpler is better" is used.

There are several ways to select the locations; this can be done manually, if the sites of activity are known in advance. Another approach is using some sort of evaluation algorithm to determine those channels which contribute most to the signal. The brute force approach or a more sophisticated approach like the genetic algorithm can be applied to select the most favourable channels [56].

2.5. *The potential classes*

The signals measured by the electrodes are called potentials and are measured at multiple locations, and differ in frequency during time. This is what constitutes the raw data generated by EEG.
A range of different potential classes are used in BCI. These classes vary, among other properties, in their amplitude, duration, frequency and location. But also by which process they are activated in the brain. The following potential classes are used in BCI research.

2.5.1. Event related potentials

Event Related Potentials (ERP) are potentials that change due to a stimulus or event. There are two types: the exogenous ERP which occurs 100ms after the stimulus is applied and the endogenous ERP which occurs from 100ms and onwards. The idea behind ERP is that because it is known when the event happens, it is also known when the potential change happens in time. So the signal can be filtered out of the random background EEG signal and hereby improving the signal-to-noise ratio. The BCI system has 'locked' the time parameter. This makes the classification process simpler, because only spatial parameters have to be searched for to find the intentions of the user. ERP signals used in BCI are:

- VEP, Visual Evoked Potential is dependent BCI. The user must look at a certain scene. Changes and events in this scene will evoke potentials in the brain. The Steady-State VEPs (SSVEP) are natural responses to stimuli at specific frequencies (or multiples).
- P300 evoked potential is based on endogenous ERP response that occurs after 300ms. Usually the subject has been told to respond to a rare stimulus, which occurs at random moment, the so-called "odd-ball-paradigm".

Other ERPs are the readiness potential (RP) and the Movement-Related Potential (MRP). These are a response to a desire to perform and imagine movement respectively.

2.5.2. Rhythmic activity

This a-periodic and unpredictable activity is constantly present and is a result of the total activity generated by all the neurons in the brain. The frequency range is divided into different band: The Delta (0.1-3-5Hz), Theta (4-7.5Hz), Alpha (8-13Hz), Beta (14-30Hz) and Gamma (>30Hz), see figure 2.8. The mu-rhythm is a specific part of the Alpha rhythm (10-12Hz) and is located over the sensorimotor cortex. The main advantage of the mu-rhythm over the Alpha rhythm is that is does not appear to be influenced by eye-blinking therefore it is mainly used in BCI. Users can learn to voluntary control the rhythms after training to some extent. This concerns the (de-) synchronization of the rhythm.

Motor Imagery (MI) is a commonly used method in BCI. To obtain MI, the user should imagine moving a hand, finger or leg but not actually moving it. Thereby generating the pattern in the brain that goes with this movement, but not disturbing the EEG measurement by the actual activity of muscles. Measurement of muscle activity is called EMG (electromyography) and this activity overwhelms the EEG. The MI is used in combination with ERD/ERS (see next section).

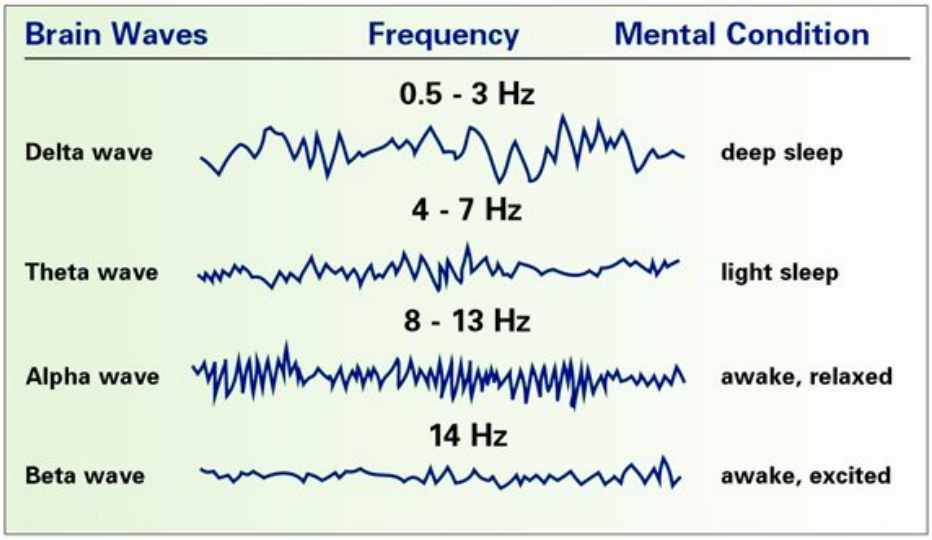


Figure 2.8 Overview of the categorization of brain waves; image from "C. Perrini, CLS, CNC".

2.5.3. Event related (de)synchronizations (ERD/ERS)

ERD/ERS are amplitude attenuation and enhancement respectively. The activity is monitored prior, during and after an event over a number of trials and the potential can be calculated. The ERD/ERS is the power decrease/increase relative to the starting reference.

The Slow Cortical Potential (SCP) is a potential shift of the electrical activity in the brain lasting from several hundred milliseconds to several seconds. The user can learn to voluntary influence the cortical positivity/negativity.

The SCP can be generated even if the motor periphery is completely disconnected from the central nervous system. It can be used as a binary signal, using a certain amplitude threshold as selection criterion [24], but can be artificially extended to offer more choices like in figure 2.15.

2.5.4. BCI approaches

Various different approaches regarding BCI exist. In this section different perspectives on BCI are compared.

2.5.4.1. Natural intent versus active control

The ideal approach would be to directly know from the user what he or she wants, in such a way that the subject merely needs to think it. Or perhaps even more automatic like the way humans move their hand and fingers. It is goal oriented; there is no need to really think of each separate finger movement. It *just* happens. This approach is called natural intent [60] (the simpler, the better). The reality however is different. The current knowledge of brain function is still far to shallow to say anything useful about the higher cognitive processes. Therefore this first approach seems far beyond the current grasp.

The second approach, active control, is more down to earth. It is build around the idea that the subject thinks about a particular issue (for example right hand movement) or performs a specific task (for example solving a mathematical equation), which generates a particular pattern in the brain. If a certain pattern can be recognized, it can be coupled to a certain command. The number of these commands is limited from a practical point of view. In a typical BCI system this is binary choice, which with some tricks can be extended. But it is not exactly mind blowing.

Because of the limited amount of command options provided by these approaches, the emphasis should be on goal communication (also called intelligent control, communication of intentions) over command communication (trying to communicate every detail) [60]. The main

problem with goal communication (versus command) is the possibility of incorrectly recognizing the goals of the user concerning perceived activity. This should be avoided, because it is of the utmost importance to let the user feel in control. Incorrect assumptions are off course also possible with simple command communication, but then their impact is far less.

2.5.4.2. Pattern recognition versus operant conditioning approach

The currently used BCI can be divided into the pattern recognition approach and the operant conditioning approach.

The pattern recognition approach is based on cognitive mental tasks. The principle behind performing different cognitive tasks is that they produce different signals in different areas of the brain. The BCI can locate these signals using EEG and a classifier can be trained on this data. These tasks consist of for instance motor imagery, visual counting, mental figure rotation, arithmetic and baseline tasks [27]. The selection of mental tasks can and should be determined individually and based on the experiment. The need for user training is not so high, since the machine tries to learn about the users mental states which are already there.

The operant conditioning approach is based on the self-regulation of the EEG response, like the SCP or mu/beta-rhythm. The following elements are important for a successful approach:

- Real-time feedback of the specific EEG activity (see section 2.8.1.2).
- Progressively demanding tasks including rewards.
- Positive reinforcement of correct behaviour.

Using this approach the user learns to control his or her brain patterns. In contrast to the pattern recognition approach the bulk of the training load is on the user's shoulders.

2.5.4.3. Synchronous vs. asynchronous control

BCI has two distinct approaches to control. One being synchronous control which means that the BCI continuously interprets the input as commands. The other approach is asynchronous control; meaning that the user determines when interaction takes place: the BCI still continuously interpreting the input however acts only after pre-defined commands.

Obviously asynchronous control is preferred over synchronous. In most other communication mediums which are used nowadays asynchronous control is applied: achieving optimal control for the user. It is logical that the user determines when he or she wants to use a device.

Figure 2.9 Synchronous control of a wheelchair does not seem very handy.

The same goes for BCI; however there is a major obstacle that must be overcome when using asynchronous control. It must be determined from the brain signals what the user wants and when.

The BCI must constantly analyze the brain state of the user and not mistake it for trying to use the BCI, when the user could be doing something completely different (false positive). The system has to remain in a stable off state when there is no control intended [2].

Intermediate systems have been suggested that turn on after a certain command or chain of commands. This precise command can only be used as an on/off-switch. An intermediate form is also possible where after the BCI is turned on, it continuous with synchronous control.

The advantage of synchronous control is the fact that it is less complex. The BCI can interpret everything as a command and in some cases the time does not have to be tracked, which loses one dimension to search for, since the system determines when the user should act. The EEG only needs to be analyzed in a predefined window. Whereas in asynchronous control the EEG needs to be continuously analyzed and classified.

Because of the defined time period used by synchronous control, the speed of commands is also less; however the accuracy appears to be better due to the higher simplicity and lower requirements for classification.

2.5.4.4. Universal vs. individual

When constructing a BCI system, two philosophies exist. A BCI can be build using the data of a couple of individuals to construct a universal classifier, which can be used by virtually anyone.

This is quite the opposite of the belief that no two brains or persons are the same: neither physical nor mental. Consequently there is a need for an individual BCI, which is fine-tuned for every individual.

2.5.4.5. Online vs. offline

The usable version of BCI is the online version, with which the user can interact. Every process is computed in real-time: from the signal processing, device control and feedback and every step in between. Offline BCI stores the data for later analysis and typically uses data from more electrodes. The goal of offline BCI is to analyze what is the best signal processing, feature extraction; classifier etc. and adaptation of the parameters for use when the BCI is in online mode, to maximize performance.

2.6. *Signal pre-processing*

The signals coming from electrodes connected to the brain range from 0Hz and upwards. And can contain every possible variation and distortion. Therefore the quality of the signal must be improved. This quality is defined by the Signal-to-Noise Ratio (SNR). This ratio defines that a higher quality signal has a higher SNR. The SNR can be improved by applying filters that reduce the noise and amplify the desired signal, as well as by removing undesirable artifacts or data.

2.6.1. Amplification & A/D-converter

Brain signals are very weak, therefore to do any processing at all, they need to be amplified. The amplified signal should not be distorted more than before the process. After amplification the analogue signals from the brain are converted to digital using an A/D-converter [57].

2.6.2. Filters

2.6.2.1. Reference filters

A reference filters improves the quality of the input by referencing the value of interest to its neighbouring values.

Different methods exist to perform this operation, the most commonly used filters in BCI follow here, however the optimal use of a certain method depends on the circumstances:

- Laplacian filter (small/large), the Laplacian spatial filter enhances local activity and reduces diffuse activity [36, 50, 63]. The small Laplacian filter subtracts the average of the four surrounding channels from the channel of interest. The large Laplacian filter subtracts the average of the next-neighbours from the channel of interest [51].
- Common Average filter (CAR), works as a high pass spatial filter. It reduces a large part of the total population of channels. It subtracts the average value of all electrodes from the electrode of interest.

$$v_i^{CAR} = v_i^{ER} - \frac{1}{n}\sum_{j=1}^{n} v_j^{ER} \qquad (2\text{-}1)$$

where v_i is the channel of interest [7, 36, 51, 63].

2.6.2.2. Bandpass filter

After amplification the signals are passed through a bandpass filter. The band pass filter virtually lays a band over the incoming signal. Every frequency outside this band is removed from the signal. Using this filter for instance the mu-rhythm can be extracted from the input by discarding every frequency less than 10Hz and over 12Hz, leaving the desired band.
The bandpass filter can be implemented using a FIR (Finite Impulse Response) filter algorithm. FIR does not distort the signal.

2.6.3. Artifacts

The EEG signals are always imperfect and always contaminated with artifacts. Artifacts are undesirable disturbances in the signal. These artifacts range from bioelectrical potentials produced by movement of body parts like, eyes (figure 2.10), tongue, arms or hart or fluctuation in skin resistance (sweating). And can also have a source out side the body like interference of electrical equipment nearby or varying impedance of the electrodes [55].

All these issues influence the EEG data and should *ideally* be completely removed. A possible way to remove them is detection and recognition. This is not a trivial task.

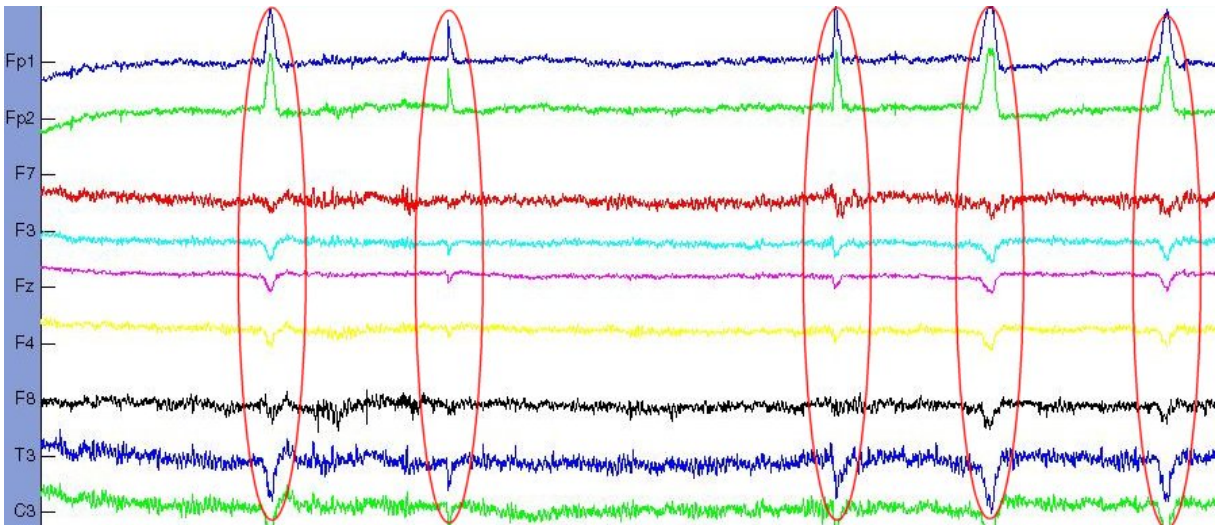


Figure 2.10 Example of eye-blink artifacts in the raw EEG data.

Recognition of for instance limb movement can be facilitated by using EMG. Whenever EMG activity is recorded, the EEG readings should be discarded (artifact rejection). The other artifact sources like

eye-movement (measured by Electrooculargraphy (EOG)) and heart activity (measured by Electrocardiography (ECG)) can also be removed. However most of the artifacts are continuously present and typically more present than EEG. If it is known when and where they occur, they can be compensated for.

2.6.4. Artifact removal

Whenever artifacts are detected the affected portion of the signal can be rejected. This can be a valid pre-processing step and does not have to be a problem. However the problem with deleting a specific piece of data is that it can result in strange anomalies where the two pieces are connected. Secondly, EEG data in general is relatively scarce. For that reason a better approach is to remove the artifact from the EEG data. This goes one step further than artifact rejection.

For practical purposes in an online system, it is undesirable to throw away every signal that is affected with an artifact. Retrieving the signal with 100% correctness is impossible; it is simply unknown what the data would have looked like without for instance the eye blink. For offline systems this is less critical, since it does not matter if some commands are lost. In the online case however, the user demands that every command that is issued to the system is recognized and executed. The user doesn't want to keep trying endlessly for a good trial.

Rejection therefore is not desirable. The goal is to keep a continuous signal. Ideally the artifact must be removed and the desired signal preserved. This can be done using filters or higher-order statistical elimination and separation, like for instance independent component analysis.

2.6.5. Independent Component Analysis

Higher-order statistical methods simultaneously use the information of all the electrodes available. This offers the possibility to locate a certain component and remove it from the data. One method often applied is Independent Component Analysis (ICA) also known as blind source separation.

ICA is a statistical computational spatial filtering method that decomposes the multi-electrode data into underlying independent components (or as independent as possible) [50]. The goal is to reveal hidden factors which underlie a certain dataset. ICA assumes linear independence of the sources and that the sources are a linear combination of the witnessed output. ICA does not take into account any 'ground-truth'-labels, which makes it an unsupervised method [28, 34].

ICA is related to principal component analysis; however ICA is much more powerful and capable of finding underlying factors when the classic method fails.

ICA & EEG

EEG data consists of recordings of electrical potentials in many different locations. These recorded potentials are presumably a mix of underlying components of brain activity and therefore ICA can be applied [66].

ICA Example

The easiest way to understand ICA is with an example (see figure 2.11). The classic example to explain ICA is to picture a room with talking people (or different sound sources). In this room two microphones records all the sounds; this represents the output of the room. When ICA is performed on the output it will return the independent sources (different sound sources in the room). So the idea behind ICA is that the original components can be obtained as long as they are independent. However in practice this requirement does not have to be entirely true.

$$x_1(t) = a_{11}s_1 + a_{12}s_2 \quad (2\text{-}2)$$

$$x_2(t) = a_{21}s_1 + a_{22}s_2 \quad (2\text{-}3)$$

The recorded signal (x) which is a combination (*a*) of the independent sources (*s*). ICA can predict *s* from x without knowing *a* and *s* [62]. This example can off course be expanded to any arbitrary number.

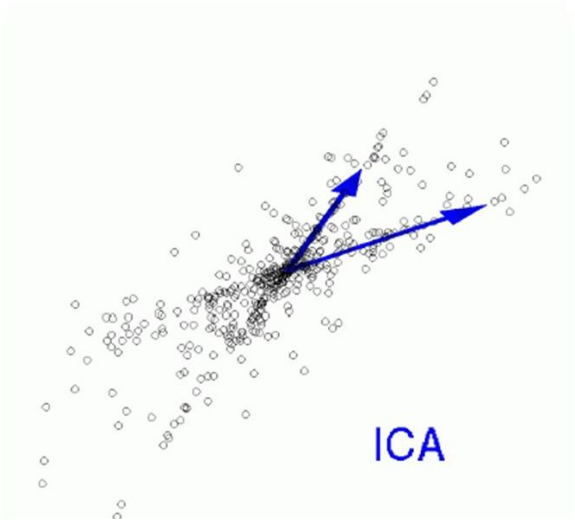


Figure 2.11 Example of ICA solution to a distribution. Two directions are found within the data.

2.6.5.1. Properties of ICA

ICA has the following properties [28]:

- ICA can only separate linearly mixed independent sources.
- The time-point order in the EEG has virtually no impact on the outcome.
- The same holds for any order of the electrodes presented to the ICA, the order has virtually no impact on the outcome.
- Although it is required for the initial sources to be independent; ICA will still find a space that is maximally independent for not completely independent sources.
- Perfect Gaussian-sources cannot be separated. It is easy to see this as they will have exactly the same distribution (for ICA the sources are of unit-variance). Although a maximum of one of the sources is allowed to be Gaussian.

2.6.5.2. Maximizing non-Gaussianity

ICA is based on the principle that the sum of two independent random variables has a distribution that is closer to Gaussian than the distribution of the original variables. The general situation is formulated in (2-4):

$$x = As \tag{2-4}$$

where x is the output vector of the mixing matrix A and the independent components s. The goal is to estimate both A and s knowing only x. Let $\tilde{y}_i$ be an estimated component ($\tilde{y}_i \approx s_i$) and w_i a row in the estimated inverse of the mixing matrix A ($W \approx A^{-1}$). The reconstruction of the component then becomes:

$$\tilde{y}_i = w_i^T x \tag{2-5}$$

Now back to the Gaussianity: the distributions of *x* are more Gaussian than the distributions of *s*, because *x* is a weighed sum of the components in *s*. The problem is now finding w_i so that the nongaussianity of $\tilde{y}_i$ is maximized.

A measure of nongaussianity is required; one way to define nongaussianity is by following approximation of negentropy

$$J(y) \text{ is proportional to } \left(E\{G(y)\} - E\{G(v)\}\right)^2 \tag{2-6}$$

v is a zero mean, unit variance Gaussian variable, for *G(y)* a non-quadratic function must be chosen. Negentropy is always non-negative, and it is zero only if *y* has a Gaussian distribution. Negentropy has the additional interesting property that it is invariant for invertible linear transformations.

2.6.5.3. The algorithm of ICA

The implementation of the FastICA is used which tries to solve the following optimization problem using (2-6):

$$\text{Find the } w_i \text{ that maximizes } J(w_i^T \tilde{x}) \text{ under the constraint that } \|w_i\| = 1 \tag{2-7}$$

Constraining w_i to unity is based on the fact that the variance of the independent components cannot be determined. Maximizing (2-6) is done by finding where the gradient of the Lagrange is zero. The Lagrange gradient with respect to *w* is

$$L_w'(\mathbf{w}, \lambda) = E\{\mathbf{x} g(\mathbf{w}^T \mathbf{x})\} - 2\lambda \mathbf{w} \tag{2-8}$$

Where *g(y)* is the gradient of *G(y)* mentioned in (2-6). The solution based on (2-8) is

$$E\{\mathbf{x} g(\mathbf{w}^T \mathbf{x})\} - 2\lambda \mathbf{w} = 0 \tag{2-9}$$

For the gradients one of the following options can be chosen:

$$g_1(y) = \tanh(a_1 y) \quad g_2(y) = y \exp(-½ y^2) \quad g_3(y) = 4y^3 \tag{2-10}$$

The equation $L_w'(\mathbf{w}, \lambda) = 0$ is solved iteratively. The estimation of the independent components is available and W returns the topographical distribution of every component [29, 31, 61, 62].

2.6.5.4. Limitations of ICA

The ICA model has some limitations [28, 31]:

- The variances (energies) of the independent components cannot be determined. Since both *s* and *A* are unknown any scalar multiplication in *s* can be undone by a scalar division of the corresponding row in *A*. The same goes for the sign.
- The order of the independent components is undetermined. For the same reason as the previous point: both *s* and *A* are unknown. And therefore any shift of rows and numbers can be made.
- ICA can decompose at most *N* independent components from *N* electrodes or sources. It is very well possible (and likely) that there exist more independent processes in the brain than those *N* since the effective number of processes is unknown.

2.6.5.5.Purpose of ICA

ICA is used for two purposes concerning analysis of EEG data [11, 12, 23]:

- Identifying artifacts, like eye blinks, ocular movement, breathing, pulse, interference of (outside) systems. The data can be reconstructed after removing unwanted artifacts. Thus the artifacts are removed and not rejected from the data. It must be stated that artifact removal using ICA is a computational demanding task, especially if it is required for online-systems and calculation need to done in real-time.
- The second purpose is to identify the real sources of activity (those which actually correspond to generated brain patterns). The problem with this approach is that it is mathematically undetermined. This means that ICA will tell *which* independent components compose the collected scalp recordings (what they look like) but it fails to tell *where* these components come from in the brain. So any topographical information is lost.

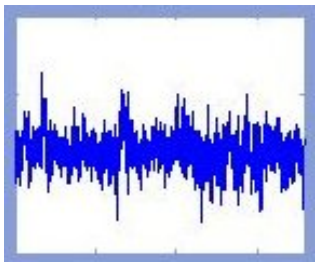


Figure 2.12 Example of IC.

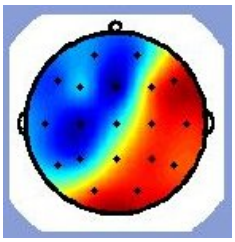


Figure 2.13 Weight distribution on the scalp belonging to figure 2.12.

2.6.5.6. ICA assumptions versus EEG

In order to correctly apply ICA on EEG data, the assumptions made by ICA must be valid. The following assumptions are made by ICA regarding EEG [31]:

- Signal conduction times are equal (no time delay from sources to sensor) and summation of currents at the scalps is linear. Both these assumptions are reasonable for currents carried to the electrodes by volume conduction at EEG frequencies.
- Spatial projections of components are fixed across time and conditions. Effectively meaning that the same processes must occur at the same sites in the brain, again a reasonable assumption following many (MRI) studies showing that the same processes always originate from the same space. Although this may not completely hold at millimetre scales or at macro-levels (depression, exhaustion, sleep).
- Source activations are temporally independent of one another across data. However there can be temporally overlap between activations, but the activation of the relevant data components is independent. Following the intuition that people can actively control their brain processes.
- The data should be non-Gaussian distributed (2.6.5.1). In theory ICA cannot not separate Gaussian sources, however in practice even small deviations appear to yield good results. In general EEG data is Gaussian. Sub-Gaussian independence has been demonstrated in EEG.

ICA is used for the purpose of data pre-processing: removing eye blinks and other muscle movement. This makes the constraints put in place by ICA somewhat looser. For instance the temporal independence of the artifacts versus the real brain processes is clear. And the constraint of ICA only capable of finding N sources (2.6.5.4) is not that important either, because it turns out that ICA finds the relatively large and independent sources, hence among others: the artifacts [30].
Searching for artifacts using ICA is enhanced by the fact that artifacts are in general less Gaussian than normal brain sources.

2.6.6. Channel selection

Not only should the artifacts be removed, but when using a number of channels there will be redundant and irrelevant (or less relevant) information in the produced data. It helps to select only the

best channels, those channels that contribute most to the desired information and keeping the resulting feature vector as small as possible whilst retaining the most valuable data.

A small feature vector is desirable, because the complexity of the calculations rises quickly with evermore dimensional data. Certainly for online application computational complexity should be minimized. This is mainly a concern in the classification phase.

2.6.7. Segmentation

Segmentation is a method to deal with non-stationarity. Stationary signals are preferred for analysis. Segmentation is the process of cutting the signal in to small stationary pieces (with possible overlap).

2.7. Translation algorithm

The translation algorithm is the main element of any BCI system. The function of the translation algorithm is to convert the EEG input to device control output. It can be split in to feature extraction and feature classification [64].

2.7.1. Feature extraction

Feature extraction is the process of selecting appropriate features from the input data. These features should characterize the data as good as possible. The goal of feature extraction is to select those features which have similar values for objects of one category and differing values for other categories. Reducing overlapping features will aid in the quest against poor generalization and computational complexity. The goal is to select of few features from the total feature vector, which still describe the data.
Feature extraction can be performed using:

- Time-frequency analysis, using Fast Fourier Transform (FFT)
- Autoregressive modelling (AR)
- Common spatial patterns (CSP)
- Linear Discrimination (LD)
- Genetic algorithm (GA)

More about feature extraction methods in section 2.10.1.

2.7.2. Feature classification

The next step in the translation algorithm is the classification of the acquired features. When presented with new data, the BCI should be able to tell which brain state it represents. Therefore it must recognize patterns in the data provided by the features. Classification will result in device control: acquiring the commands from the user. It can be achieved in various ways:

- Linear Vector Quantization (LVQ)
- Neural network (NN)
- Support Vector Machines (SVM)

The fundamental choice in feature classification is the choice between machine based learning and system analysis. Machine based approach has no need for a predefined structure, but will develop one along the way. System analysis on the other hand seeks an underlying model. Therefore the machine approach may work better when there is no or little information available on the underlying system.
More about feature classification methods in the section 2.10.2.

2.8. *Application*

This is the last BCI step, the actual performance. Everything comes down to the fact that the application should do what the user wants it to do. The list of possible applications of BCI is practically endless. In theory any application which exists today could be controlled using BCI. However with the current state of BCI in mind the focus of BCI applications is to aid disabled people with simple programs.
The applications range from simple decision programs to manipulation of the environment, from spelling programs (see figure 2.14) to controlling prosthesis and wheelchairs. These examples can be generalized in communication, environment control and movement restoration [60].

In the application domain the issue of self- versus evoked control is important. Good thought must be given to how this control is realized. Special care should be given to real-time control of devices that interact with the physical environment. Synchronous control of your wheelchair does not seem very handy. Although the intermediate approach using an on/off switch offers possibilities.
Optimizing the application so that it requires minimal effort from the user is another area where improvements can be made. For instance control-commands often used together should be in 'proximity' to each other or most easily accessible [59].

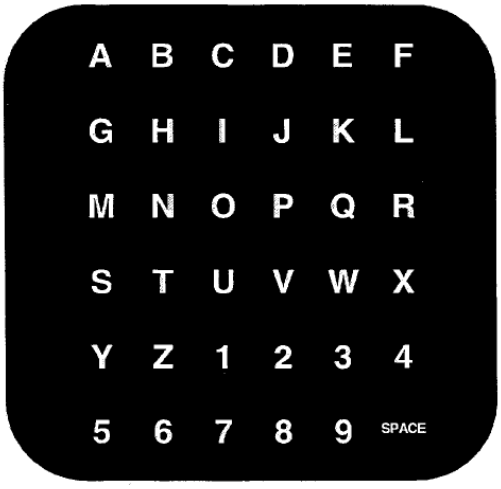


Figure 2.14 Example of BCI spelling program. The subject must focus on a single object, with short intervals a row or column is intensified (high lighted), which triggers a brain response.

The current maximum information transfer rate of about 25 bits per minute, strongly limits the application range and its applicability for mass society [47]. Moreover the dimensionality of the commands is today mostly one dimensional. This depends on the number of different brain states that can be recognized per time unit [14]. For good mouse control for instance, two dimensional control is a necessity.

2.8.1. Training

The concept of training has been mentioned before. And actually the entire working of BCI relies on the subject's ability to generate brain patterns. To acquire that particular skill, the subject needs to be trained. Most of the current BCI systems require the user to generate specific brain patterns; however there is a tendency towards BCI taking over this task. But no matter how it will turn out, the user must use his or her brain.

2.8.1.1. Training principle

Typically training is build up out of sessions. And each session out of trials. Working with a BCI system is a mentally demanding job and requires much of the concentration of the subject, therefore training is usually limited to a maximum of one hour per day.

Learning is most effective when the following conditions are met:

- Active engagement.
- Frequent interaction.
- Feedback.
- Connections to real world content.

Stimulus-rich and motivationally engaging environments lead to more efficient training [48]. The amount of training required depends on the sort of brain patterns required by the BCI. For instance the VEP and P300 require less training. It also depends on the load between the user and the BCI. Both systems are adaptive controllers. The focus can be on the subject trying to create the most suitable brain patterns, but also on the classifier in the BCI which could for instance do a good job at trying to classify even mediocre patterns (the two approaches mentioned in 2.5.4.2). Off course they can also meet somewhere in the middle, a mutual learning process.

Basically the operant conditioning approach requires more training from the user compared to the pattern recognition approach. This is a major drawback of the first approach. It will take users at least several weeks to get their skill up to an acceptable level [20]. However attempts have been made to try and teach a large group to control a BCI in a very short time (less than one hour). The resulting accuracy however cannot compete (yet) with the normal lengthy training procedures. These results are general and certainly do not mean that a single individual cannot achieve high accuracy in a short time span.

One issue must be kept in mind regarding training for instance ALS patients. There is no information on the state of consciousness, abilities or level of motivation of the patient in comparison with 'normal' individuals [4]. However it can be said that in general these persons are far more motivated than the average user.

2.8.1.2. Feedback

Training using operant condition approach requires feedback from the BCI. Feedback is the response of the system after an action from the user. This can be visual or auditory. And is used to show the subject the results of his or her action and can also be used to 'reward' good actions. Feedback speeds up the training process and improves overall performance. A well known example of feedback in BCI is cursor control (figure 2.15). The user tries to move the cursor to a target and can see the results of his or her actions directly on screen. If the target is reached it can for instance flash to give positive reinforcement. Feedback can motivate, hold the subject's attention and improve performance by allowing for corrections.

However there also exist some possible drawbacks on feedback. The feedback can affect the brain rhythms itself caused by anticipation of the results (correct or wrong classification), visual changes in the environment and finally it can distract the user from concentrating on producing the brain patterns.

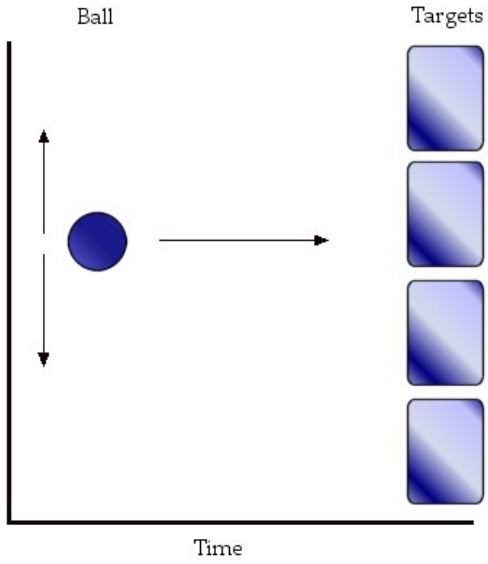


Figure 2.15 Example of feedback. The subject must steer the ball to one the targets, this is visually fed back by the moving ball. This figure also shows how to extend a binary input to a quadruple output.

Feedback is not essential for BCI using the pattern recognition approach, but it can be very beneficial. Since it gives the subject an understanding of what is going on and helps to focus.

2.9. *Comparing BCI*

Comparing different BCI systems is almost a science in its own right and one of the major obstacles to overcome in the general BCI research field. The difficulty comes from the facts that there exist many variables in every BCI system and no general framework to compare them. Objective methods for evaluation of algorithms and training protocols are required [60].

The BCI community lacks a common vocabulary; terms are diverse and inconsistently used. This can lead to poor inter-group communication [35].

2.9.1. BCI performance

The performance of different BCI systems is difficult to measure and compare. However there exist several performance criteria by which BCI systems can be judged.
First the criteria that can be quantitatively described:

- Speed, the speed at which commands can be given.
- Accuracy, the number of correctly interpreted commandos as a ratio of the total given commands.

Speed and accuracy are in twined. If the speed is high, but the accuracy is very low, the overall performance will not be very high and vice versa. Both values should be maximized, but the maximization of one value will degrade the performance of the other. So somewhere there is an optimum value for these parameters, which will results in the highest bit rate. This optimum value can vary across users. Moreover these values should be evaluated constantly during training [37].

- Information transfer rate (ITR), is a ratio for the amount of information transferred.

ITR is speed and accuracy combined in a single entity: the speed at which commands from the user actually gets transferred correctly.

The following issues are harder to quantify, but not less important for the overall performance, quality and value of a BCI system.

- Extent of training required by the user; the length and intensity of training required to get some basic skill at operating a specific BCI.
- Ease of use, this criterion compromises the complete usability of the BCI. This can be summarized in how effective the BCI can be used.
- Minimal constraint on use and preparation, how easily can be started with using the BCI (This especially reflects on invasive techniques or techniques requiring large and expensive equipment).
- Comfort, this is actually a more long-term criterion which becomes really important when the BCI is used in practical situations outside the laboratory. How does the user feel both physically and mentally about using the device?
- Costs, the cost-effectiveness is probably the most important condition for any system to be built. And this single condition has impacts on all the previous goals mentioned.

2.9.2. Comparing criteria

Here follows a thorough but not complete list of the diversity of elements BCI systems. Every BCI system consists of or has to do with each of these elements. The possible combinations of characteristics of each of the elements are huge.

- The characteristics of the users
- Method of training of the users and the duration
- How the actual experiment is performed
- Method of measuring activity
- Measured activity (Potentials)/Signal type
- Degrees of freedom
- Resolution of the data
- Single/average trial
- Method of signal pre-processing (Filters)
- Method of feature extraction
- Method of feature classification
- Type of application

These are all variables and the more variables there are the more complex a problem becomes. The first part (up to method of measuring activity) and the last factor (type of application) have the highest impact on the possibility of comparison. Therefore suggestions have been made to compare just the inner working of the BCI system: the translation algorithm (feature extraction and classification). This leads to a drastic decrease of possible variations and thus complexity.

And because it is possible to use the same (standard) input for each BCI system, this clears the way for comparing similar systems. It is possible to compare on the basis of speed, accuracy, response time and information transfer rate.

One of the most commonly used methods is the correct classification rate, although it does not take into account the rejected EEG and the fact that different classes can have different classification rates. To counter these problems a confusion matrix can be used. This matrix also lists the false classifications and therefore draws a better image of the situation [49]. However the confusion matrix is not often used in BCI, researchers rather use the hit-rate, which is the percentage of hits versus misses.

2.9.3. Combining approaches

It is off course not the case that it is completely unknown which BCI outperforms another.

And this information can be used to improve BCI. Other ways researchers are looking for improvements are that of combining certain advantageous features of a specific BCI approach. For instance taking an EEG for the temporal resolution and a fMRI for the spatial resolution, getting the best of both worlds: the good temporal resolution of EEG and the excellent spatial resolution of fMRI [25].

The same can be done in the translation algorithm to get better classification results. Using different EEG features which complement each other. Combination of single complementary features should give a performance gain [15]. The same goes for potentials, for instance the MRP and the ERD, using both potentials can lead to improved robustness.

But the problem still remains that it is hard to compare complete system which interact with users for specific applications. Therefore there is a need for standardization of hardware, recording methods, ethics and platforms [60]. And a functional model and taxonomy of a BCI system [35], a general framework.

2.10. *In depth look at the translation algorithm*

The translation algorithm is the heart of a BCI. It consists of the feature extraction and feature classification as mentioned in the previous section. Here both processes will be examined in more detail.

The most frequently used algorithms in BCI are described here. However this distribution of the algorithm over the different elements of BCI is not exactly black and white. Some algorithms naturally evolve into the next BCI step, which makes it harder to precisely define to which category they belong. This ambiguity also comes from the fact that BCI still lacks a good general framework and standard vocabulary, so that in different research projects these terms are differently used.

2.10.1. Feature extraction

The (pre-processed) input from EEG contains enormous amounts of information. What actually is needed from this data are features: distinctive characteristics. A feature is that part of the data that gives the best interpretation and most valuable information relating to it. Therefore these characteristics (the best features) must first be found by analyzing the data.

Two issues that need to be resolved: Which features and how many. The global goal is to find a small subset of all features present that suffices to represent information on the whole feature vector [29]. This depends on the particular BCI in question. Below follows a list of analysis methods frequently used in BCI. This list is not exhaustive, but gives a good indication of the possibilities. Although these methods essentially provide the features needed in the next step of classification, defining the features is a dynamic process in combination with the pre-processing process of reducing channels and filtering frequencies and feature classification. The methods covered here apart from the Genetic Algorithm are implemented in the EEG Workbench (chapter 6, 7 and 8).

2.10.1.1. Fast Fourier Transform

The Fast Fourier Transform (FFT is a faster version of the Discrete Fourier Transform (DFT) utilizing a clever algorithm to perform the same task as DFT. DFT takes a discrete signal in the time domain and transforms it into a discrete frequency domain representation (figure 2.16).

2.10.1.1.1. DFT algorithm

Let $x_0, ..., x_{n-1}$ be the sampled input values from the original signal and *f* the output frequency. The DFT is defined by the formula

$$f_j = \sum_{k=0}^{n-1} x_k e^{-\frac{2\pi i}{n} jk} \quad \text{for j = 0, 1, …n-1} \tag{2-11}$$

x_k represents sample $\frac{k}{n}$ of the complete input range. And the inverse Fourier Transform:

$$x_k = \frac{1}{n} \sum_{j=0}^{n-1} f_j e^{\frac{2\pi i}{n} kj} \quad \text{for j = 0, 1, …n-1} \tag{2-12}$$

Evaluating these sums directly would take O(n^2) arithmetical operations [54].

2.10.1.1.2. FFT

The FFT algorithm computes the same result in only O(n log n) operations and is based on the factorization of n. n is the number of pieces in which the signal will be divided. The FFT (and DFT) shows out of which basic frequencies the signal is build up. These frequencies represent the intrinsic properties of the signal [57].

Functionally the FFT decomposes the data into smaller segments (a couple of times) and performs DFT on these small segments and builds up the data afterwards. This goes at the cost of accuracy; however the speed gain is significant. For a sensible transform the sampling frequency of the signal should at least be twice as high as the highest frequency in the signal.

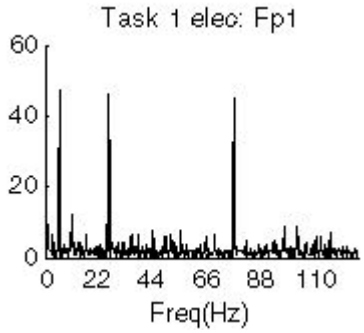


Figure 2.16 Example of FFT spectrum, vertically the magnitude and horizontally the frequency.

A piece of the signal used as input for the FFT is assumed to be representative for the rest of the signal. In principle any signal can be build up by adding a number of correct sinus waves. The FFT does just this; it calculates which waves are necessary to build up the perceived signal.

Every signal has a spectrum and every spectrum has a signal, it is possible to move back and forth between the two without losing information using (2-11) and (2-12). This is mathematically true, however in practice this is quite incorrect because there is one basic error in the FFT, which is that it presumes the signal to be build up out of an infinite number of sinus waves. In practice the signal is a short period of time, still it turns out that the FFT gives a good representation of the signal in the frequency domain.

2.10.1.2. Common Spatial Patterns

Common Spatial Patterns (CSP) have some overlap with the pre-processing phase. This method performs a weighting of the electrodes (channels) according to their importance for the classification task. It can be utilized for brains states that are characterized by a decrease or increase of a cortical rhythm with a characteristic topographic pattern (see figure 2.17).

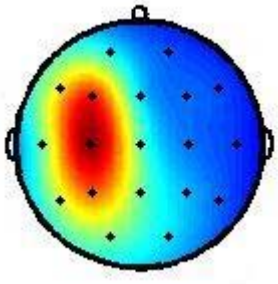


Figure 2.17 Example of CSP.

It creates spatial filters to construct new few time-series with variances that have the most discriminative power. Decomposition is performed on the raw EEG data into these spatial patterns, extracted from two populations of single-trial EEG and maximizes the difference between them [13, 14, 51]. The CSP projections provide the most discriminative scalp patterns [3].
CSP is based on simultaneous diagonalization of the covariance matrices of the two populations. The covariance matrices reflect the spatial distribution of the potentials in the brain.

CSP is a supervised method giving weights according to the power captured for two classes [34].

2.10.1.2.1. The CSP algorithm

X is a N x K, a [channels x trial-concatenated time points] matrix. K denotes the samples, let X = $[\overline{x}_1,\ldots,\overline{x}_k]$ then the covariance matrix is defined by:

$$R^{(i)} = \sum_{k=1}^{K}\left(x_k^{(i)} - \frac{1}{K}\sum_{k=1}^{K}x_k^{(i)}\right)\left(x_k^{(i)} - \frac{1}{K}\sum_{k=1}^{K}x_k^{(i)}\right)^T \tag{2-13}$$

where $x_k^{(i)}$ is the N-dimensional vector at time *k*. This way the covariance matrices for the left and right hand data can be estimated. Next the normalized covariance matrices, where *l*, *r* correspond to the number of trials of left or right (the two populations) data respectively.

$$R_L = \frac{1}{l}\sum_{i=1}^{l}\frac{R_L^{(i)}}{trace(R_L^{(i)})} \tag{2-14}$$

and

$$R_R = \frac{1}{r}\sum_{i=1}^{r}\frac{R_R^{(i)}}{trace(R_R^{(i)})} \tag{2-15}$$

Trace (x) is the sum of the diagonal elements of x. Combing the R_L and R_R; the two covariance matrices will be simultaneously decomposed.

$$R = R_L + R_R = U\lambda U^T \tag{2-16}$$

The columns of U are the eigenvectors and λ is a matrix with the eigenvalues on the diagonal. Using these matrices the transformation matrix *P* can be found to calculate the CSP features.

$$P = \lambda_c^{-1/2}U_c^T \tag{2-17}$$

Whitening transformation to *P* equalizes the variances in the space by U_c. A whitened matrix has zero mean and its autocorrelation matrix is a multiple of the identity matrix. S_l and S_r are the transformed covariance matrices: $S_l = PR_lP^T$ and $S_r = PR_rP^T$, where the eigenvalues of PR_xP^T are equal to one.

$$\begin{aligned} S_L + S_R &= PRP^T \\ &= \lambda^{-1/2} U^T U \lambda U U^T \lambda^{-1/2} \\ &= I \end{aligned} \tag{2-18}$$

B is the matrix that contains the common eigenvector of S_L and S_R such that if $S_l = B\lambda_l B^T$, then $S_r = B\lambda_r B^T$ and $\lambda_l + \lambda_r = I$, which implies that if l is large, than r is small and vice versa. This gives the discriminative power.

$$Y = (B^T P)^T X \tag{2-19}$$

The decomposition of a trial amounts to Y=WX, where W is the projection matrix and the inverse of this matrix contains the common spatial patterns.

Classification is done with the feature vectors with the largest λ_L and λ_R and can be performed linearly. These feature vectors can be obtained using:

$$f_p = \log\left(\frac{\mathrm{var}(Y_p)}{\sum_{i=1}^{2m}\mathrm{var}(Y_i)}\right) \tag{2-20}$$

Where p= (1,…2m) are the m first and last rows of Y due to the calculation of P. The log-transformation is performed to normalize the distribution of the elements of f_p. These features are used in the classifier [13, 19, 44, 51].

Artifact removal is very important, because artifacts have a large influence on the construction of the spatial filters. When this action is performed (almost) completely, reference methods have little more influence on the end results by CSP.

CSP is normally used for two differing brain states; however it can be extended to more than two classes. This is a viable option if the total information transfer rate increases by this alteration of CSP. The possibility of using more than two classes can even achieve increased performance even though accuracy decreases.

2.10.1.2.2. Evaluation CSP

The high recognition rates and computational simplicity make CSP a good and fast method for an EEG-based brain–computer interface [13, 14, 19, 50].

An advantage of the CSP method is that it does not require subject-specific frequency bands as necessary for band power or frequency estimation methods [19]. A drawback of CSP is the fact, that it detects spatial patterns in the EEG, this perhaps sounds a bit strange, since that is the goal of SCP. However whenever the positions of the electrodes change, this may render improvements in the classification useless. Thus this method requires identical electrode positions for every trial [19]. Another disadvantage of using CSP is the large number of electrodes required [2]. And special care must be put into the removal of artifacts.

2.10.1.3. Parametric modelling

Parametric modelling involves fitting a mathematical model to a time-series. The fundamental assumption of time-series modelling is that the value of the time-series depends on the combination of the previous value and a random disturbance (stochastic process).

Autoregressive modelling (AR) is a form that is often used in BCI to create a smooth and continues frequency spectrum representing a small segment of EEG data.

2.10.1.3.1. Autoregressive modelling

Autoregressive Modelling (AR) transforms the actual voltages coming from the EEG into coefficients. The coefficients reflect the oscillatory properties of the EEG signals, but not the overall amplitude (figure 2.18). The coefficients allow the prediction of the values of the EEG time-series based on *p* previous values using a linear combination. *p* is called the order of the AR [15]. The feature vector of one trial is the concatenation of the AR coefficients with their variances of each channel.

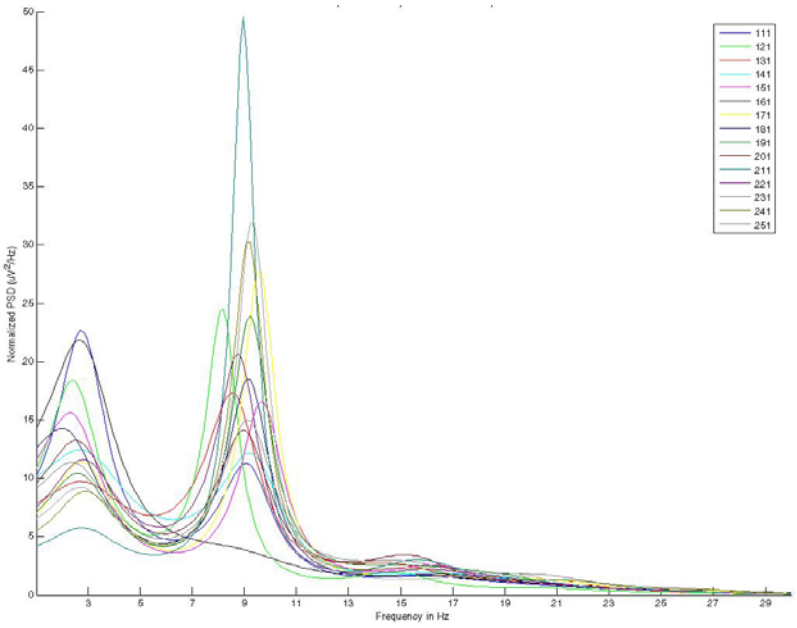


Figure 2.18 Example of AR spectrum.

Input data for AR, can both be Gaussian or non-Gaussian distributed [64]. The p-order autoregressive process, where the process *x (n)* is regressed upon '*p*' previous samples of itself:

$$x(t) = \sum_{k=1}^{p} a_k x(t-k) + e(t) \tag{2-21}$$

x (t) is the signal at the sampled point *n*, *e (t)* is a random white noise component and a_k are the AR coefficients to be estimated.

Using the (*p*-order) AR, the parametric modelling parameters (a_k) can be estimated using for instance the Yule-Walker algorithm, Burg's algorithm or Least Squares (LS) [27].

Choice of the estimator is not that important anymore, as they all produce similar good results and the computations can be performed easily on today's computers. More important are the length of the data segments and the order *p*. The AR coefficients are used as features for pattern recognition.

2.10.1.3.2. Adaptive Autoregressive modelling

The difference between Adaptive Autoregressive modelling (AAR) and normal AR is that the parameters slowly vary over time to better resemble the actual process [27]. This is mainly an issue in online usage of BCI. The *p*-order Adaptive Autoregressive method has time varying coefficients of order *p*:

$$Y_t = a_{1,t} Y_{t-1} + a_{2,t} Y_{t-2} + \ldots + a_{p,t} Y_{t-p} + E_t \tag{2-22}$$

Where E_t is a purely random white noise source with a zero mean and finite variance.
The parameters can be estimated using Least Mean Square (LMS) or Recursive Least Squares (RLS) [20]:

$$E_t = Y_t - a_1(t)Y(t-1) - \ldots - a_p(t)Y(t-p) \quad (2\text{-}23)$$

$$a_k(t+1) = a_k(t) + cE(t)Y(t-k), \text{ where } c = f / \text{var}(Y) \quad (2\text{-}24)$$

where k = 1,…,p and f is an update coefficient to set the ratio between speed of adaptation and accuracy [27, 53]. The dimensionality of AAR is greater than that of AR. So the adaptability of the model comes with a significant computational increase.

2.10.1.3.3. AR Model order estimation

The model order is a very important factor in the eventual level performance. If the order is too small (underestimation), no complete description of the signals is reached. If the order is too big (overestimation) the AR causes false peaks in the spectrum that have no relation to the signal. Selecting the order is not straight forward. It is to a large extent a matter of experience, application and trial-and-error.

2.10.1.3.4. Evaluation Parametric Model

The parametric model is one of the most commonly used for feature extraction in BCI. This model has been extensively studied for years in multiple other disciplines and appears to work in the field of BCI. AR considers the spectral distribution of the most prominent brain rhythms [15].

Where the various FFT algorithms or sensitive to slight shift of input data, the AR methods is sensitive for outliers.

AR is a method commonly used for evaluation of spectra of short data segments. If the segments decrease in size this speeds up the process, but does not decrease accuracy; this is ideal for EEG data analysis since it in general concerns relatively short trials. AR can be more accurate than FFT for very short segments of data.

2.10.1.4. Discriminant analysis

2.10.1.4.1. Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is actually in the middle of feature extraction and classification. Since it has the raw data as input and returns the most discriminative scalp pattern between two tasks as output [43]. To make a separation between the analysis, the goal of this thesis, and the classification it is listed here. However it could be listed in the classification section as well.

LDA classifies each EEG amplitude (both mean and variance) according to a preliminary decision value [24]. The LDA creates a hyper plane in the m-dimensional feature space, which forms the optimal spatial integration [27, 43]. The goal is to choose the decision function such that it maximizes the separation of the projected classes.
The degrees of freedom determine the number of classes that can be distinguished. More degrees do not necessarily mean better performance. The discriminant function:

$$L = w_1x_1 + w_2x_2 + \ldots + w_nx_n + c = \sum_{i=1}^{n} w_ix_i + c \quad (2\text{-}25)$$

With x as the input vector and w as the discriminant coefficients and c is a constant.

A data point x_i is assigned to a class according to its distance to this class. This can be determined by the weighing vector w. LDA is optimal if the distance between the classes is maximal. The criterion to be maximized:

$$J(w) = \left|\tilde{\mu}_1 - \tilde{\mu}_2\right| = \left|w^T(\mu_1 - \mu_2)\right| \tag{2-26}$$

μ is the average of a class, w represents the weight vector. In LDA each class has equal covariance matrix and equal prior probability. If the covariance matrices of the classes are not equal it becomes a non-linear discriminant, like for instance the quadratic discriminant analysis.

2.10.1.4.2. Fisher linear discriminant function

The problem with the criteria (2-26) is that is does not take into account the standard deviation within the class. The Fisher solution to this problem is that it normalizes the means by a measure of the scatter within the class. The Fisher linear discriminant is defined as the function that maximizes the criterion:

$$J(w) = \frac{\left|\tilde{\mu}_1 - \tilde{\mu}_2\right|^2}{\tilde{s}_1^2 + \tilde{s}_2^2} \tag{2-27}$$

$\tilde{s}_1^2 + \tilde{s}_2^2$ represents the within-class scattering, where each individual term is the equivalent of the variance. Fisher linear discriminant projects high dimensional data on a line and performs classification in one dimension.

2.10.1.4.3. Non-linear discriminant analysis (NLDA)

An example of a non-linear classifier is the well-known Bayes quadratic classifier (see also figure 2.19).

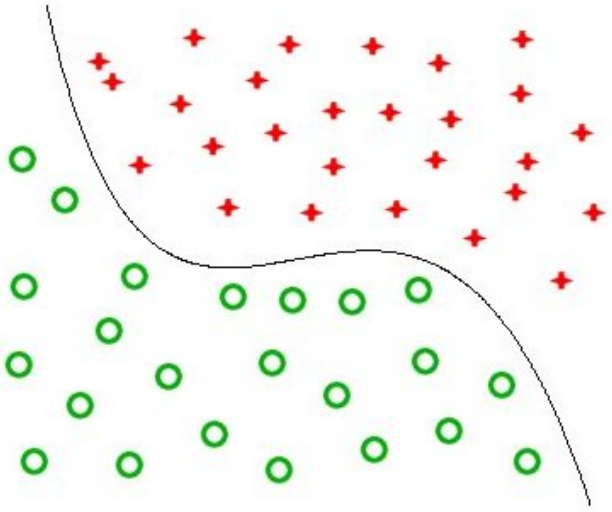


Figure 2.19 Example of classification problem that is best solved non-linearly.

This classifier uses the discriminant function $g_i(x) = P(\omega_i \mid x)$. Determining which class ω to select for the value x. $P(x \mid \omega_i)$ has the Gaussian likelihood density. Applying the Bayes rule to this function:

$$g_i(x) = P(\omega_i \mid x) = \frac{P(x \mid \omega_i)P(\omega_i)}{P(x)} \tag{2-28}$$

Now the $g_i(x)$, using the Gaussian distribution and (2-28), can be rewritten into:

$$g_i(x) = -\frac{1}{2}(x-\mu_i)^T \sum_i^{-1} \frac{1}{2}(x-\mu_i) - \frac{1}{2}\log\left(\left|\sum_i\right|\right) + \log(P(\omega_i)) \tag{2-29}$$

$P(\omega_i)$ represents the prior probability associated with the i^{th} class. μ_i and Σ_i are the mean and covariance of the i^{th} class respectively [58].

This is the quadratic discriminant function, which should be minimized. The quadratic term is known as the Mahalanobis distance (MD). The MD is used to determine the similarity of a set of values from an unknown sample to a set of values measured from a collection of known samples. MD takes into account the distribution of the samples (correlation).

2.10.1.4.4. ROC

The Receiver Operating Characteristics (ROC) shows the performance of a classifier on the basis of sensitivity versus specificity according to a varying parameter. ROC is often used in situations where a hypothesis is tested over two populations, which explains the terms 'positive' and 'negative'.

- Sensitivity: is the True Positive fraction. The number of correctly classified instances of class A, relative to all instances of class A.
- Specificity: is the True Negative fraction. The number of correctly classified instances of class B, relative to all instances of class B.

Typically a ROC-curve shows the false positive rate versus the true positive rate (see figure 2.20). The area under the curve (Az) gives an indication of the classifying performance. Optimally the area equals 1: true positives and true negatives only. If the value is 0.5 this indicates that the two populations are probably very similar [18, 22].

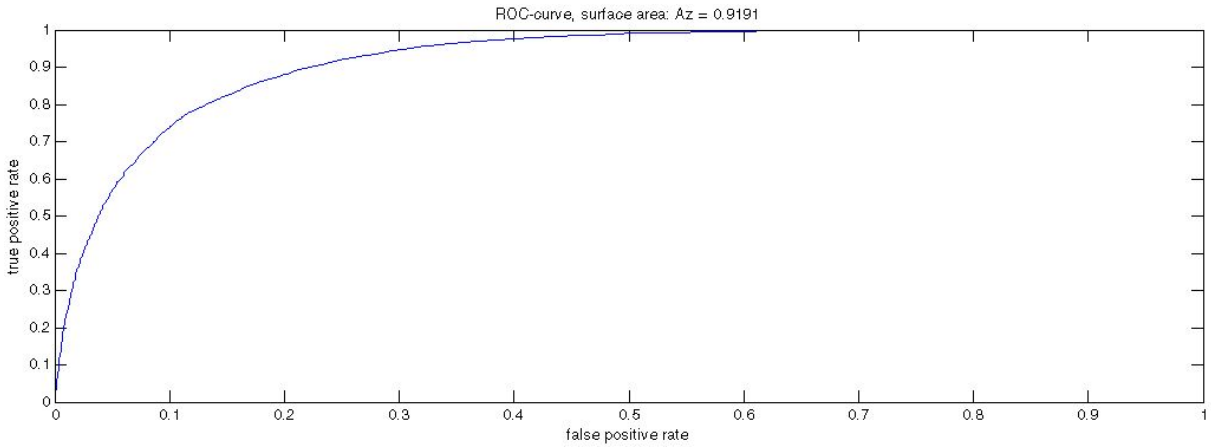


Figure 2.20 Example of a ROC-curve; the higher the surface *Az*, the better the discrimination.

2.10.1.4.5. Evaluation of Discriminant Analysis

In BCI LDA is the most commonly used and preferred method for feature classification. LDA is relatively simple and easy to implement and most importantly very effective at low computational

costs. This is an advantage for real-time applications, although LDA can miss higher relations which could be present within the data. These can be discovered using non-linear classifiers, although this comes at a significant computational cost [44]. Another promising technique is Support Vector Machines (SVM) which is briefly covered in section 2.10.2.4.

The use of a non-linear classifier (vs. a linear classifier) depends on the characteristics of the data (size, nature etc.). Again in general the saying goes: the simpler, the better. Linear appears to be more robust, although not with non-robust data [60]. The LDA requires less computation than Neural Networks (section 2.10.2.3); however it requires more discriminatory feature vectors to classify correctly.

2.10.1.5. Combining paradigms

As mentioned previously, methods can be combined to improve performance. Most will be gained if the different features provide complementary information.

AR considers spectral distribution of the most prominent brain rhythms [16]. And CSP considers the spatial distribution. These methods can support each other to increase the overall performance. This can be done in different styles like: concatenation, winner-takes-all. However these two approaches can never increase performance above the best single feature vector analysis.
More sophisticated methods mentioned in [15]:

- PROB, the idea of PROB is that the features are combined in one feature vector under the assumption that all the individual features are mutually independent. A decision function that forces the elements of the estimated covariance matrix that belong to different features to zero is applied. Thus fewer parameters have to be estimated and distortions by accidental correlations of independent variables are avoided.
- META, which applies each individual classifier on each feature vector and after that step applies a meta-level classifier to their output. This also gives the possibility of taking high level relationships into account.

2.10.1.6. Feature extraction conclusion

Now that the features are found, a selection must be made on which features to use. This can be done in various ways; for instance taking the top *'k'* scores. This could leave some important features out of the equation. Therefore obtaining the top level of features which amount to a total sum of scores can also be a valid way of selection.

Now how can the actual quality of the features be determined? The best way is to use a classifier [39]. This will show the performance which is desired. Classifying features is covered in the next section.

2.10.2. Feature classification

To recognize patterns from the brain activity, the features that are extracted from the data must be classified into several classes which represent the output. This output can be used as device control. Several methods exist to classify the features into usable output.

A problem in classification that should be avoided is overfitting to the data. Overfitting is the process that results in really good classification for the training data, but since the classifier is completely focused/fitted to that data, other (even slightly differing) data will not perform good, due to this overfitting [4]. Classification in BCI is typically about high dimensional vectors with a low number of samples [15]. Classification of high dimensional data for which few samples are available may lead to overfitting. To reduce the possibility of overfitting two measures can be taken:

- Strong data pre-processing can reduce the dimensionality of the data. The drawback of strong pre-processing is that strong assumptions about the data distributions have to be made, which is especially hard for the variable EEG signals.
- Leaving the data high dimensional, but carefully regularizing the data in such a way that high dimensional features can be handled. This method circumvents the need for strong data assumptions.

The other problem of the BCI data is that there are generally relatively small numbers of samples available to train classifiers; this problem is more difficult to overcome. Collaboration between research groups is undertaken to create bigger data-pools.

Verification of the training is performed using *n*-fold cross-validation [51]. This method divides the available data into *n* folds test data and *n-1* training data. *N* classifiers are trained and tested with a different fold of this data to judge the performance. This type of verification is pessimistically biased, because not all the data is used for training.

In the following sections the methods that are frequently used in BCI are covered. Again this list is not exhaustive, but tries to give in indication of the possibilities. Moreover it tries to give notion rather than to explain in detail about the methods, since the focus of this thesis is not on classification, but the analysis part.

2.10.2.1. Genetic algorithm

In BCI mostly the standard 10-20 system is used which contains 19 electrodes. The fact is that some of these channels provide data which is irrelevant or redundant (for a specific task). The most appropriate channels must be selected from the total population.

The Genetic Algorithm (GA) can also be used to select the best features from a total list of features provided by the EEG. Therefore it is listed here under the classifiers, but this is not exactly black and white.

Instead of using brute force to calculate all possibilities, which is a possibility but not in all cases, especially in real-time applications dealing with high-dimensional data this can be a problem, a genetic algorithm can be applied to circumvent this problem.

GA starts with an initial population, where each chromosome corresponds with a set of EEG channels (genes) [10, 56].

2.10.2.1.1. Genetic Algorithm in detail

The GA applied to BCI contains the following steps:

- *Initial population*: limited number of chromosomes (a combination of electrodes or features) which are randomly chosen from the search space.
- *Fitness assignment*: a fitness value is computed and assigned to each chromosome of the initial population. To achieve this goal each of the feature channels are applied to the input of a Neural Network, which performs the classification task a number of times. These results are averaged and considered as the fitness value for the chromosome. Chromosomes with high mean and low variance are considered fitter.
- *Selection*: Now the fittest chromosomes must be selected by any of the following methods:
    - Natural selection: Two chromosomes are drawn at random from the population from which the fittest of the two is placed in a subset. Both chromosomes are then returned to the population. This process is repeated until the subset is full.
    - Tournament selection: this approach resembles natural selection only usually a couple of chromosomes compete to win the tournament.
    - Roulette-wheel selection: each chromosome has a change according to its fitness of being selected. With higher fitness comes higher chance of selection (from which comes the roulette analogy). It seems that weaker chromosome are not eliminated as

thoroughly as possible, however this way of selection has the advantage that, although the chromosome may be weak, some genes can still prove useful in the recombination process.

- *Recombination*: Two chromosomes (parents) of the subset (with the fittest chromosomes) are selected to mate (with a very high probability 0.9 or more). If they don't mate, they are propagated to next generation. The uniform crossover method is used to produce the children. The method applies recombination to the individual genes as a result of interpolating the parents. The probability of crossover between genes is normally set to a low 0.1.
- *Mutation*: mutation simply changes the value of a particular gene, with a very low probability of 0.01 or lower.
- *New generation*: the population is now full with new chromosomes and step 1-5 is repeated for a number of generations.

The mutation parameter can be lowered in the final generation to help convergence and avoid corruption of the better genes [56]. GA can also be used to simultaneously optimize the feature extraction and the classification, by optimizing the coefficients to acquire a suitable classifier [10].

2.10.2.1.2. Evaluation GA

The solution of the GA is an approximation of the optimal solution; however it offers a suitable approach of finding acceptable performing parameters for the BCI. One of the main strengths of the GA is that it can calculate entire populations in parallel. Due to this parallel nature, the GA is less likely to get stuck in local optima. And the fact that the GA only produces an estimate is not such a big issue, because of the variability of the brain signal, there are few absolute truths.

2.10.2.2. Linear Vector Quantization

The goal of Linear Vector Quantization (LVQ) is to reveal how the data is clustered by creating a network of vectors. An assumption is made about the number of clusters/classes in the data and each cluster has a labelled dataset. A data point belongs to the cluster for which its index is closest to the prototype for that cluster.

LVQ Algorithm

Choose number of clusters M and initialize prototypes vectors $w_1, w_2, ..., w_m$ with random weights. Randomly pick an input x from the total input space. Determine the winning prototype for which the distance to input x is the smallest (using the Euclidean distance). Update only the winning prototype vector according to

$$w_{new} = w_{old} + \eta(x - w_{old}) \tag{2-30}$$

where η is the learning rate. This adaptation is performed to reinforce correct classifications. The goal is to minimize the overall distance to the prototypes.

2.10.2.3. Neural Networks

The idea behind neural networks (NN) is based on the simulation of the neurons in a human brain. It is a network of neurons, which processes incoming information and outputs information based on this input, very much like our own neurons do.

In BCI a neural network assigns each feature to the class with the highest probability for that particular feature. This decision is based on the value of the features and the values of the weights

within the neural network. These weights are continuously updated during training to optimally fit to the data [8]. And can also be updated during actual use, which is called an adaptive neural network.

By feeding the training-input to the NN with the desired output, the network can be trained. If a NN is properly trained, than new (unknown) input from the same system will result in proper classification.

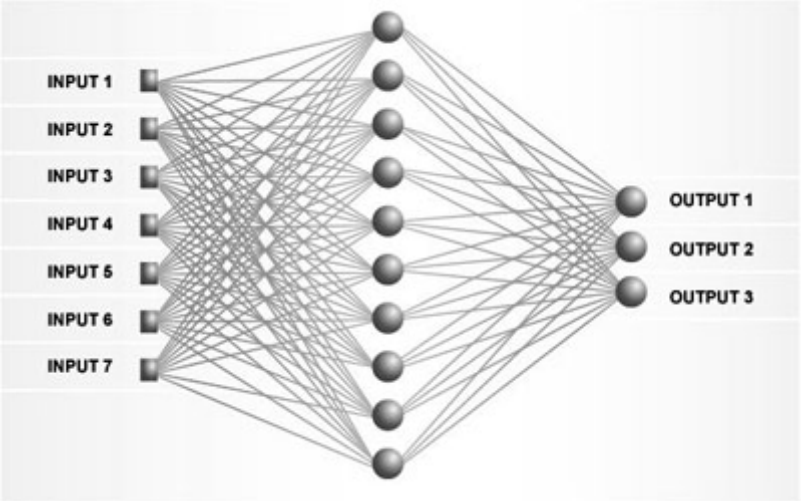


Figure 2.21 Example of NN architecture; image from "Yale Scientific".

A neural network's topology exists of layers of neurons. Typically one input layer, one or more hidden layers and one output layer (figure 2.21). A neuron is referred to as the processing element (PE) of the NN. The PE performs an operation on its combined input. This combination of the input is usually linear summation according to weights. These weights can be adapted during training, to reflect the importance of each of the inputs. The operation performed on the input is an evaluation function, which results in a certain output that is transferred to the next layer. Typically it will look like $g(x) = f(h(x))$, where *h(x)* is the linear combined input:

$$h(x) = \sum_i w_i x_i + b \tag{2-31}$$

where *b* is the bias. The adaptation of the weight in this equation generally follows this rule:

$$w(n+1) = w(n) + \eta(d(n) - y(n))x(n) \tag{2-32}$$

This equation reflects that the new weight is the old weight plus a certain difference between the desired output *d* and the actual output *y* (the error) times the input and a certain learning rate η. This equation shows that if the correct response is given for a certain output, the learning stops. In fact if the desired output is exactly the same is the actual output (the error is zero), no weight change occurs at all.

These networks are called feed forward back propagation network. The input is fed forward and after which the error is back propagated through the network to update all the weight in the PEs. The goal is to minimize the error.

The number of input PEs must be equal to the number of feature values. The number of output PE must be equal to the number of classes in which the features must be classified. Setting the number of hidden PEs (the layer(s)) in between is not a trivial task. If the NN does not have enough degrees of freedoms (facilitated by the PEs) it will have a high error in classifying since it cannot solve the presented problem. Having to much PEs may results in overfitting to the training data [49].

Evaluation Neural Networks

The major advantage of the NN is that there is no need for assuming an underlying data distribution; a good approximation can be obtained without knowledge of the underlying model. A NN is robust for noise data. Sensory data like EEG is in general very noisy. NN can be implemented and computed in parallel. NN can give real-time performance, once properly trained.

The fact that the relation between the input and output is not defined is a major disadvantage of the NN-approach; this is known as a black box. This means that once the NN works it is not clear how it works or what is going on 'inside'. Another disadvantage is that for training a NN, large amounts of data samples are required which in BCI are typically not available.

2.10.2.4. Support Vector Machine

The Support Vector Machine (SVM) is a different kind of classifier. It can handle high dimensional data. The degrees of freedom are decoupled from the size of the input. And it can use higher dimensions to separate data linearly, that otherwise would be impossible.

The idea behind SVM is to separate the data from two classes by finding a weight vector w and an offset b of a hyperplane with the highest possible margin (figure 2.22). Because it is not always possible to separate two classes perfectly, the SVM will search for the hyperplane with the highest margin and lowest training error. The datapoints closest to the hyperplane are called the support vectors, these are the important points, and the rest can be ignored [33, 38].

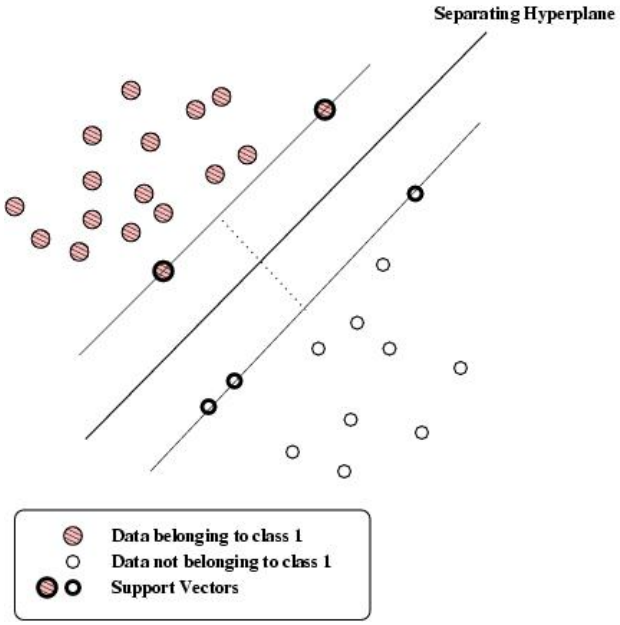


Figure 2.22 Example of SVM problem; image from "Monash University".

The SVM leads to the following optimization problem for the pairs (x, y) for finding w (weight vector) and b (offset) by:

$$\min_{w \in R^d} \frac{2}{w^T w} \text{ and } y_i (w^T x_i) + b \geq 1 \tag{2-33}$$

The solution to this problem involves constructing a dual problem using Lagrange multipliers which should be maximized. The solution has the form:

$$w = \sum \alpha_i y_i x_i \text{ and } b = y_k - w^T x_k \text{ for any } x_k \text{ such that } \alpha_k \neq 0 \tag{2-34}$$

Therefore each non-zero α corresponds with a $\boldsymbol{x}$ that is a support vector. If the data is not linearly separable slack variables can be entered into the equations to allow for misclassification.

The fact that SVM can handle high dimensionality is an obvious advantage for BCI. Furthermore it can offer good generalization and is computationally efficient, however the computation and memory requirements grow quickly with increasing size of the training data. The SVM can classify both linear and non-linear data. The advantage over NN is that its implementation can be represented geometrically in contrast to the NN's blackbox.

2.11. *Overview of current BCI methods*

In table 2.1 an overview is given of algorithms and methods frequently used in current BCI-research. The data for table 2.1 is taken from the "BCI competition III results", the "Brain-Computer Interface Technology: Third International Meeting" and from various scientific papers used in this thesis.

Table 2-1 Overview of the currently used methods and techniques in BCI.

Pre-processing methods	**Use**	**Researcher**
Segmentation	10	D. McFarland, L. Trejo, C. Anderson, D. Krusienski, B. Blankertz, P. Hammon, X. Gao, A. Buttfield, A. Schloegl, G. Gage
Laplacian filter	3	D. McFarland, B. He, D. Krusienski, G. Xiaorong
Band pass filter	13	B. Blankertz, X. Gao, B. He, P. Hammon, L. Yang, M. Krauledaut, M. Hieden, A. Rakotomamonjy, L. Yandong, Z. Zangton, L. Zhonglin, G. Gentiletti, S. Parini
Independent Component Analysis	4	P. Hammon, B. He, L. Yandong, J. Hill
Baseline correction	3	Z. Zongtan, J. Hill, M. Bensch
CAR	1	Y. Li
Analysis methods	**Use**	**Researcher**
(Adaptive) Autoregressive Model	7	C. Brunner, A. Schloegl, P. Hammon, D. McFarland, M. Hieden, S. Parini, Y. Wang
Common Spatial Patterns	10	B. Blankertz, C. Brunner, X. Gao, L. Yang, C. Guan, G. Xiaorong, Y. Wang, Y. Li, Z. Zongtan, L. Remy
Fast Fourier Transform	11	A. Buttfield, P. Hammon, R. Kaidar, L. Trejo, A. Schloegl, C. Brunner, X. Gao, B. Blankterz, E. Geissels, X. Pei, D. Coyle
Wavelet Functions	2	M. Fatourechi, P. Hammon
Principal Component Analysis	6	M. Fatourechi, G. Gage, P. Hammon, Z. Zangton, G. Gentiletti, J. Hill
(Fisher) Linear Discrimination Analysis	14	C. Anderson, B. Brunner, B. Blankertz, A. Schloegl, Q. Wei, L. Yang, A. Geissels, L. Zhonglin, G. Kunapuli, X. Pei, D. Coyle, Y. Wang, D. Zhang, I. Sturm
Genetic Algorithm	1	M. Fatourechi
Gaussian Mixture Model	1	M. Krauledaut
Common Spatial Subspace Decomposition	4	Q. Wei, A. Schloegel, L. Remy, D. Zhang

The table is far from exhaustive, but merely gives an indication of which methods are frequently used. Many other less well known methods are used as well, but are not listed here. More over researcher are in general not clear on which pre-processing steps they make in their research, usually skipping right ahead to the feature extraction phase.

2.12. *Current software architecture*

One of the objectives of this thesis is to construct an analysis tool for EEG data. This section gives an overview of already available software for the purpose of BCI research. The basic properties of the program are covered and their advantages and disadvantages are reviewed with the goal in mind for EEG analysis. The Truscan program provided with the EEG equipment is covered in section 3.3.

2.12.1. Software

- g.Bsanalyze
  - License: commercial.
  - Developer: *g.tec medical engineering*.
  - Purpose: offline analysis of biosignals.
  - Platform & software: Windows based, Matlab with Signal Processing Toolbox.

- BCI2000
  - Free of charge for educational and scientific purposes.
  - Developer: *G. Schalk et al.*
  - Purpose: general-purpose BCI research. From acquisition to stimulus presentation.
  - Platform & software: Windows based.

- EEGlab
  - Free of charge for educational and scientific purposes.
  - Developers: *A. Delorme and S. Makeig et al.*
  - Purpose: processing continuous and event-related EEG, MEG and other electrophysiological data using independent component analysis (ICA), time/frequency analysis, and other methods including artifact rejection.
  - Platform & software: Platform independent, Matlab with Signal Processing Toolbox recommended.

- Biosig
  - Free of charge for educational and scientific purposes.
  - Developer: *A. Schloegl.*
  - Purpose: a Matlab software library for data acquisition, artifact processing, quality control, feature extraction, classification, modelling, data presentation.
  - Platform & software: Platform independent, Matlab.

- Brainstorm
  - Free of charge for educational and scientific purposes.
  - Developers: *R. Leahy, S. Baillet, J. Mosher.*
  - Purpose: a tool to visualize EEG, MEG and (f-)MRI and combine these approaches.
  - Platform & software: Platform independent, Matlab.

- Letswave
  - Free of charge for educational and scientific purposes, does require a license.
  - Developer: *A. Mouraux.*
  - Purpose: a tool to apply FFT on EEG data.
  - Platform & software: Platform independent, Matlab.

2.12.2. Evaluation

First of all is noted that none of the programs support in anyway the Truscan Equipment or dataformat. Secondly almost all the analysis tools (except BCI2000 which is not exactly an analysis tool) are all Matlab based. And finally there is a basic difference between a tool that offers for instance AR functionality for further processing and a tool that offers to analyze data using AR.

In the evaluation of the current available software following here it is explained per tool why it is not suitable for the goal of objective three.

- *g.BSAnalyze* is not free and requires Signal Processing Toolbox from Matlab.
- *BCI2000* is not built for analysis. In short BCI2000 is build for online applications of BCI [52]. It can only be used if it is clear what features are searched for. The purpose of BCI2000 is to enable universal use of the components (measuring activity, pre-processing, feature extraction and classification and application), however this does mean that a component which does the acquisition is required (this is unavailable for Truscan software).
- *EEGLab*, although it is stated that it does not require the Signal Processing Toolbox, without it the most important functions will not work. Furthermore EEGlab is focused on analysis of ERP experiments and searching for rhythmic activity seems to be less obvious. Although EEGlab offers a GUI, it is not exactly user friendly and offers little guidance whatsoever to the user.
- *Biosig* is a software library and is not properly tested. Although the functionality might be interesting this is hard to verify. At halve way through 2005, a Signal Viewer (named Sigviewer) was added to the functionality which allows to view and inspect the raw data.
- *Brainstorm* is build for the purpose of data visualization and especially for combining different measurement methods.
- *Letswave* is focused on FFT transform. That is the only method it supports. This is not enough for a solid analysis of the data. Acquiring a license for the program is quite a hassle as well.

There are other programs available as well, but those listed in this section represent the main bulk of what is used in BCI research. Furthermore the 'analysis tool' which is incorporated with the EEG equipment is handled in section 3.3.

2.13. *Discussion and conclusion*

Reviewing the field of BCI it becomes clear that there exist many different systems, which all have their own distinct characteristics. The combinations of all the elements in a BCI system determine the performance of the entire system. For instance a certain brain signal can be pre-processed better with method X than method Y; however this other brain signal will go better with method Y over X. The same goes for the next steps of feature extraction and classification, which all have their influence on the eventual outcome. Certain algorithms work for certain data (dimensionality), signals (frequencies), potentials, brain activity and number of electrodes.

When reviewing the papers of the current BCI-researchers, it appears that the choice of methods that they use is mainly based on experience with these algorithms and their common knowledge. The actual choice is virtually never backed up with arguments. This conclusion focuses on the methods used in the translation algorithm.

The variety of possibilities for BCI does not mean that certain methods do not outperform others in general; however it must be stressed that the setup of the experiment and the variability of the measured signal have a major impact on this performance. And therefore conclusions about the

performance of algorithms from research in other fields cannot be blindly interpolated to BCI. Competitions have been used to draw a picture of what approaches give the best performance for given data sets [4]. And even such competitions show that for the same dataset different roads can lead to similar results. And also that different datasets require a completely different approach.

In general terms some comments can be given on performance and usage of methods in BCI. For pre-processing the Laplacian and CAR are used most frequently. Especially the Laplacian filter has been extensively studied. It is a straight forward method and it is clear how to implement it and what results to expect.

AR and AAR (adaptive AR) are the most commonly used methods for feature extraction used in BCI setup at present. CSP is on the increase. Good reasons must be given to use AAR over AR, since AAR comes with a dramatic increase of dimensionality and thus computational complexity.

LDA is preferred over its non-linear variant; mainly because of the fact that they produce almost similar results for the same datasets, so here the rule-of-thumb holds that: simpler is better. But the continuous increasing knowledge about the functioning of the human brain will perhaps require the complexity of non-linearity to model the higher relationships that could be (or probably are) involved in the cognitive brain processes.

The best performance is produced by the regularized LDA, which reduces the need for strong pre-processing and therefore the need for strong data distribution assumptions.

LVQ, NN and SVM are all used by several research groups and although it is mentioned that NN outperforms LVQ this is not backed up with facts. NN and SVM are used more frequently.

In general LDA is preferred over neural networks because it is computationally less complex and requires less training. When modelling more complex relationships, this advantage can become a disadvantage, because LDA will not notice these relations. And moreover LDA requires some prior knowledge of the underlying model.

Furthermore an increasing effort is made to combine algorithms to profit from their distinct advantages in how they acquire results from the brain. Looking from different angles at the same problem seems to give more robust performance. However it must be kept in mind not to over do it, for that tiny extra bit of accuracy.

Since BCI is such a young field, it is not more than logical that various options are tried to see how they perform. It can be said that overall the tendency is towards algorithms that are computationally efficient using estimations, rather than being extremely precise in their outcome.
This is due to two facts:

- The real-time nature of the application of BCI, which requires fast processing of the data.
- The noise and variability of the brain signals. The EEG is in it self not accurate and the brain process it measures are equally not accurate and therefore there is no reason to get extremely precise results. Estimation will also do perfectly.

This does not mean that accuracy is not so important and speed is everything, it merely implies where the emphasis lies.

The main problem with the current level of BCI is the low level of the information transfer rate. Other aspects that should be solved are the dimensionality of the commands and the ease of appliance and training.
For the field as a whole the BCI framework should be improved, to be able to more easily compare different setups. If these issues are properly addressed, this will greatly increase the applicability of BCI.

To conclude this chapter it can be said that the setup of the experiment is the most important part, in its own right. Giving an advice as to which algorithm to use depends almost entirely on the BCI in question. The various elements should be fitted to each other to give the best performance. If the ITR can increase significantly, than an interesting future lies ahead for BCI.

Chapter 3
3. Tools

This chapter contains descriptions of all the third party tools and software supplied with the hardware used in this master thesis. The experiment tools are the programs and equipment used during the EEG-based BCI experiment. Two of the three programs used are supplied by Deymed (Truscan acquisition and Truscan explorer) and the third program is provided by SMI and concerns the eyetracker software.

It must be mentioned before hand that all the Truscan programs and equipment are built for medical purposes and with the medical personnel in mind. This clearly has reflections on the abilities and functionality of the tools.

3.1. *Truscan EEG equipment*

3.1.1. General description

The Truscan EEG equipment is provided by Deymed. Its sole purpose is to perform EEG measurements of brain activity using Silver Chloride (AgCl) electrodes. It consists of the following items:

- EEG-cap (small, medium, large), see figure 3.1.
- EEG-amplifier, see figure 3.2
- EEG-adapter, see figure 3.3.
- Wiring (both electrical and optical).
- Extra electrodes: spare, pulse, breathing, clams.
- Accessories like belt, gel (see figure 3.4), injection.

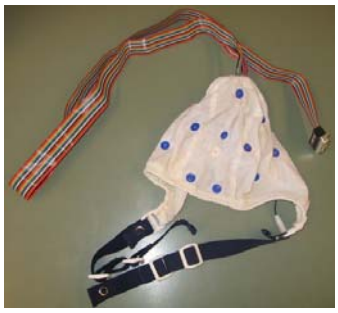


Figure 3.1 EEG-cap.

Figure 3.2 EEG-amplifier.

Figure 3.3 EEG-adapter.

Figure 3.4 EEG gel.

3.1.2. System configuration

The adapter is connected to the computer via the parallel-port and connected to the amplifier using an optical wire. This is done to prevent any electrical charge to be able to reach the subject. The cap is connected to the amplifier using a flat cable which contains a wire for each electrode.

The cap is placed with frontal two electrodes just under the hair line of the forehead and clipped to the belt surrounding the body to ensure that the cap will stay in the same position (see also figure 2.6).

Every electrode must be filled with a conducting gel using a syringe. The impedance of each electrode must drop under a certain level to make any sensible measurement.

3.1.3. Usage

Filling the cap with gel is an arduous task, but will get easier with some practise (the first time it will take anything up to 1.5 hours and with increasing experience and skill this can decrease to even less than halve an hour). It is imperative that the resistance in the electrodes is very low, since the voltages measured are already in the range of micro volts and the brain activity measured is very variable and noisy. Minimizing any extra disturbances is key to a valid good quality experiment.

During the experiment the subject must be in a comfortable and stable position, which requires no effort to maintain and will be least prone to any body movements. Not only will the act of movement severely disturb any recordings at that time. Body movement can also shift an electrode from its location resulting in ambiguous experimental data or even the gel inside the electrode can shift or change shape, which might alter the resistance leading to a flawed measurement.

All the equipment should be handled with care since the wires of the EEG-cap are fragile and irreplaceable; if a single wire is broken, the whole cap must be replaced.

While filling the cap the subject could be entertained be a game of any sort. For the detailed information see the appendix. Once the cap is in place and all the electrodes are filled, the measurement can commence (see section 3.2).

3.2. Truscan acquisition software

3.2.1. General description

The Truscan Acquisition (TA) program is provided by Deymed. The main function of the TA is to facilitate the recording of EEG data. It provides the environment to setup and record any EEG-based experiment (see figure 3.5).

3.2.2. Functionality

The TA offers the following functionality:

- Management of the patient records: creation, alteration and deletion.
- EEG-cap status: TA shows the resistance in every electrode on the cap in both colour and number.
- Experiment settings like for instance bitrate, EEG-cap configuration, port communication.
- EEG recording.
- EEG experiment markings; both from within the program and from an external source (experiment control program see section 5.2.1).
- Saving EEG records and marks to file.

Before any experiment is started several options must be set. First of all a 'patient' must be created to whom the entire EEG recording is dedicated. The next important item is the EEG-cap settings; which spatial distribution of the electrodes is selected and which electrodes are used in addition to those placed on the scalp. Electrodes can be used for instance for measuring heart rate and respiration speed. Other options can be set like the bitrate for the experiment to be measured in.

Once everything is properly set the experiment can be started. The program is very straightforward from here one. It simply consists of a start-pause-stop interface, with the addition of inserting marks in the data (see figure 3.5 top-middle).

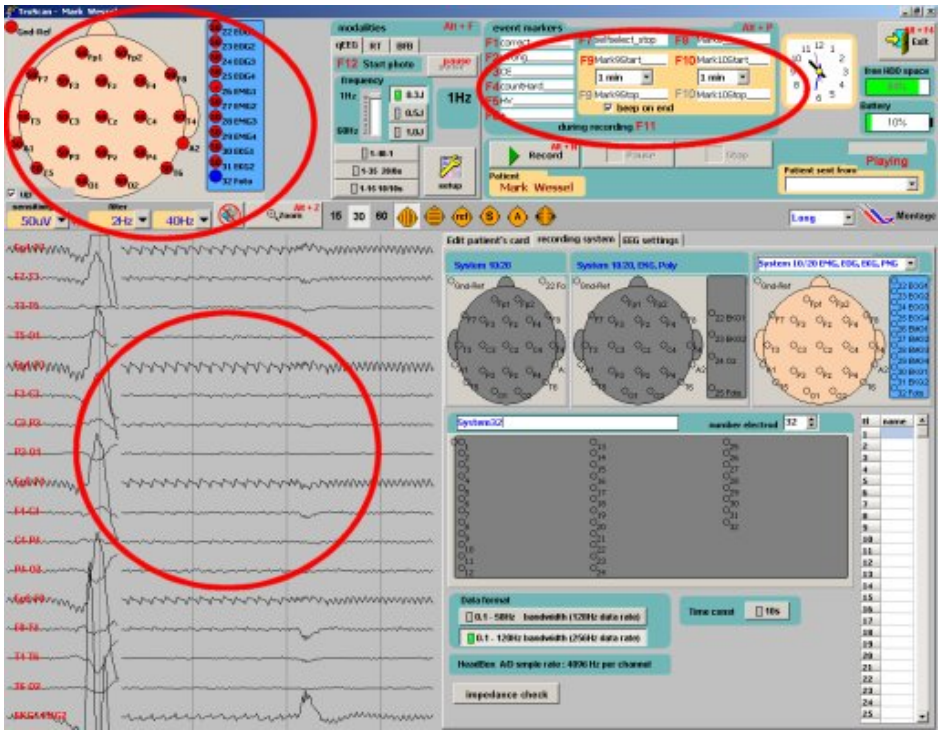


Figure 3.5 Screenshot of Truscan Acquisition software. Upper left corner: depiction of the head with resistance in the electrodes. Down left corner: depiction of the voltages measured in the electrodes. Right side: configuration and progress report.

The measured voltages during the experiment can be viewed for each electrode as well as any inserted marks. The marks can be inserted manually using the AT; however it is preferred to let the EEG-computer which performs the experiment take care of this. It is possible to communicate with the AT via TCP/IP. Several commands can be sent to the AT, later these numbers can be used to recognize the events during experiments. For each experiment TA creates a binary file with the measured voltages from the electrodes and a text file with all the marks inserted during the experiment.

3.2.3. Remarks and conclusion

When performed the first time configuring the TA can be quite a job, since some of the settings are some what hard to locate. And moreover at every start-up of the program they are reset to there default values and should be checked.

Although filters can be applied, the data is always stored in the most general and unspoiled form so that at a later stage all the options for processing are available. The data in the data-file is only determined by the electrode-setup, bitrate and resistance levels in the individual electrodes during the experiment.

After setup the program is very straightforward to work with. The program looks more complex than it is. The layout is not optimal and some terms are still in Czech.

3.3. *Truscan explorer software*

3.3.1. General description

The Truscan Explorer (TE) program is also provided by Deymed (figure 3.6). The main function of this program is to review the EEG data obtained by the TA. As mentioned in the previous section, the data stored in the data-files is as raw as possible. And this program offers the possibility to apply different filters and settings in order to inspect the data.

3.3.2. Functionality

The TE offers the following functionality:

- EEG data viewing, the raw voltages can be shown like in figure 3.6. The TE shows the data in the same way as the TA, only now it is possible to scroll through the data.
- Amplitude maps, shows the amplitude map of the voltages at a specific time in a colour distribution.
- Conversion of up to four selected sections of data to frequency spectra frequency.
- Conversion of up to two selections in to frequency intensity distributions.
- Conversion of up to two selections in to coherency maps. Coherence is a way to evaluate the capabilities of the brain to connect different part of the brain in communication. In other words to check if the brain is not afflicted with non-functional areas.
- Export to Matlab of selected segments of data.

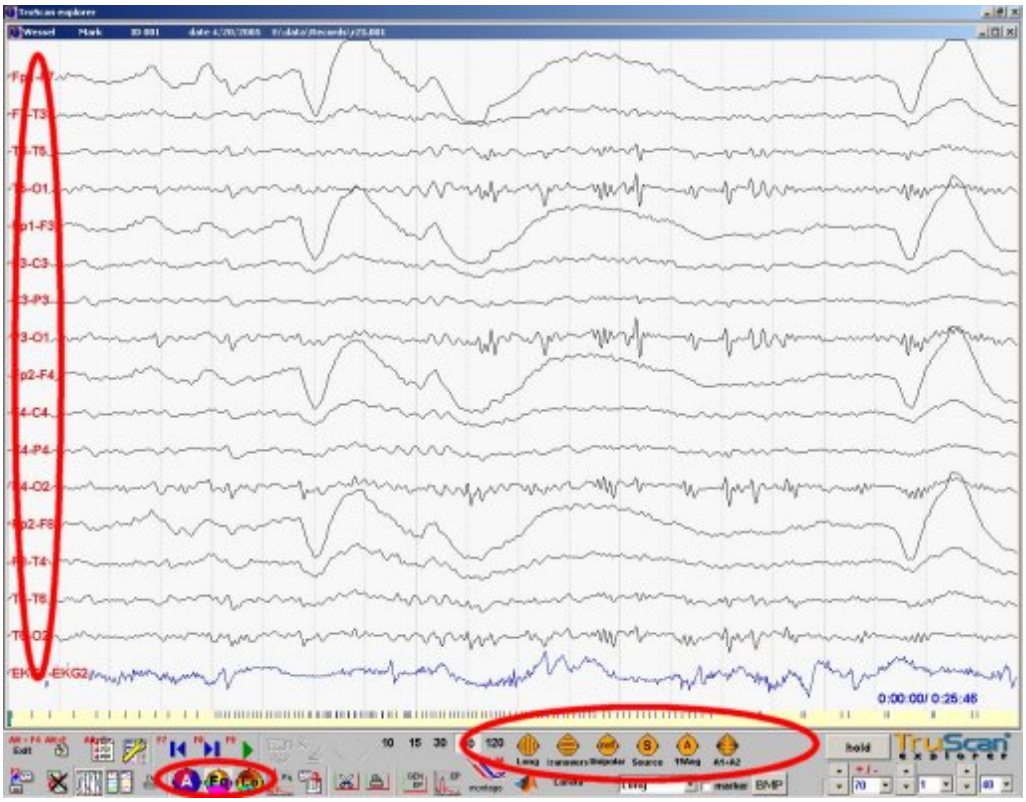


Figure 3.6 Screenshot of the Truscan explorer software. The measured voltages at the electrodes in time are visible, left the electrode definition, bottom right possible alternate viewing options and bottom middle selection of reference schemes.

All the functionality mentioned here offers the possibility to view any of the reference schemes shown in figure 3.6 at the bottom middle.

To view the data different action can be performed:
Apply different reference schemes

- Electrode-to-reference (default, this is the way it is stored in the data-file)
- Electrode-to-source
- Longitudinal
- Transversal
- Electrode to ear (A1 and A2)

Apply different filters

- Bandpass filters
- Sensitivity levels

For all the models used in the TE, it is unknown how they are calculated, since no documentation exists concerning the used algorithms.

3.3.3. *Remarks and conclusion*

The TE is a very basic program for viewing the data. TE offers no possibilities at all for a structure analysis approach; every aspect must be manually selected and storage of results is sparse. Furthermore the analysis options are limited to only frequency spectra, coherency maps and intensity distributions. It is however unknown how the acquired results are obtained. No documentation exists on the used methods and algorithms; therefore the usability of any produced result is questionable. The program is built to quickly find major brain defects like epilepsy, strokes and Alzheimer disease.

Furthermore it is impossible to compare different sessions (read different files) or more than four selected timeframes within one session. This program is clearly build for medical evaluations and clearly falls short of supplying an adequate environment for structured analysis of EEG data for the BCI research field.

3.4. *IView X tool*

3.4.1. *System description*

The IView X system is an eyetracker tool provided by SMI. The eyetracker has the ability to record eye movement in real-time. It consists of a separate camera which can move independently (see figure 3.7).

The IView X uses an infra-red beam to locate the pupil and calculates the centre. The system has the ability to track the head-movements to a certain extent.

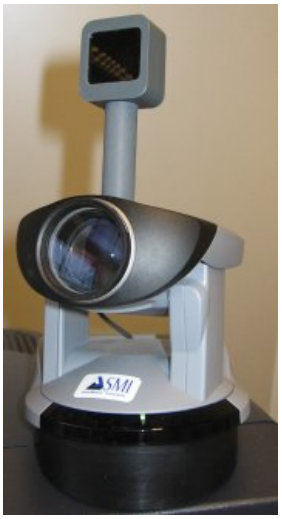


Figure 3.7 Close-up of the eye-tracker camera. On top is the infra-red generator.

3.4.2. Configuration and usage

The subject must sit in front of a computer screen and should be comfortable enough so that there is no need to move see figure 3.8. Although the camera can compensate slight movement, it is not advisable to do so, since it will go at the expensive of accuracy. The camera is placed either at the side of the monitor or under it, if possible. It is essential to match the screen with the settings in the software.

Figure 3.8 Subject one during an experiment in front of the experiment control computer. Centred under the monitor is the eye-tracker camera.

Before the measurement starts the system must be calibrated. The user will be presented with a dot in the middle of the screen. The experimenter must focus the camera exactly on the pupil. After which the calibration starts; the user must look at several points shown on the screen.
After calibration the correct functioning of the system should be verified. At a rate of 50 times per second the coordinates of the subject's gaze on the screen are broadcasted over the network and can be used by any application.

3.4.3. Remarks and conclusion

Getting the IView X to work for the first time is quite an undertaking. Once everything is set consecutive trials are easy to perform. Every trial however must be recalibrated.

The performance of the tracking is quite good, but the precision depends heavily on the quality of the calibration and the subject's ability to move as little as possible. The accuracy *can* be in the range of millimetres.

The IView X system is originally designed to be used in experiments in order to find out where the subject is looking (psychology, product and marketing research). And is not designed with the ability to interact and control a system; however it could be used for this purpose.

3.5. IView X software

3.5.1. General description

With the IView X system a complete software package is provided to calibrate and control the camera and record the user's gaze and also record the scene including gaze coordinates (see figure 3.5).

One program runs on the eyetracker PC which controls the camera and recordings. One small program is run on the subject PC for calibration purposes. Communication is performed over the net facilitated by UDP (TCP/IP).

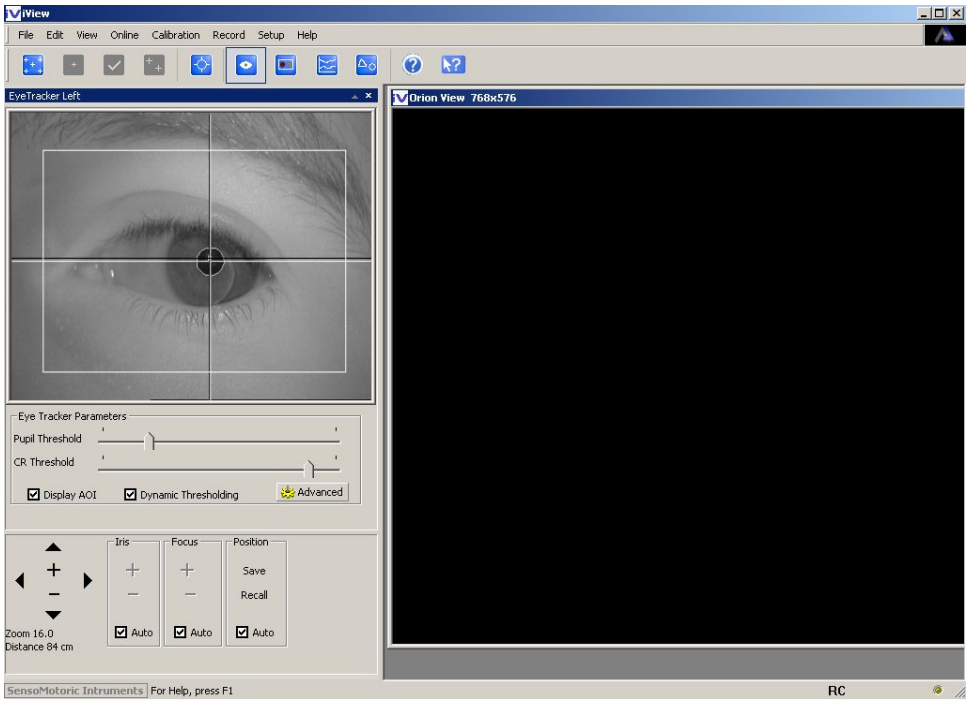


Figure 3.5 Screenshot of the IView control program. In the top left corner the camera has tracked the pupil (white) and the cornea reflex (black).

3.5.2. *Configuration and usage*

The first the calibration program on the subject PC is started, after which the control program is started on the eyetracker PC.

In order for the two programs to communicate, the network-settings have to be checked and set properly. For correct calibration the screensize of the subject PC must also be set in the control program. If everything is correct the subject screen is shown on the eyetracker PC. During the calibration the option data-streaming must be disabled, because the calibration program will not be able to respond to any command issued by the control software.

The calibration must be performed as mentioned in section 3.4. After calibration the data-streaming must be enabled if any interaction with the subject PC is required. The movement of the gaze can be viewed in the control program. The entire session is stored in a text-file for later offline analysis.

3.5.3. *Remarks and conclusion*

Finding the correct settings initially and getting the entire process in motion can be a difficult task. After this initial struggle phase the program works well and performs as expected. It is not possible to adjust the accuracy during a trial, if this is desired the entire calibration process must be performed again.

Chapter 4
4. Experiment

The MMI-group (Human Computer Interaction) has the intention to start research in the field of BCI. No previous research in the area of BCI has been conducted at the TU Delft, neither equipment nor experience was available at the start of this thesis. Therefore everything had to be built up from scratch. The EEG measuring equipment had to be ordered and knowledge and experience gained.

To answer the second research question: *How to construct and perform an EEG-based BCI experiment?* This chapter explains the actual undertaking of measuring brain activity and the consequently performed experiments with the reasoning behind the construction of the experiment. The results of the experiment and analysis using the workbench and consequent discussion and conclusion will be covered in chapter 9.

4.1. *Basic principle*

The idea behind this EEG experiment is to find out which brain patterns are most distinctive. In the EEG experiment the user basically issues commands to the system by performing different tasks. These tasks should in principle generate different brain patterns. If these patterns can be recognized separately they can be used as input to a device.

4.2. *Goals*

The EEG experiments are performed to achieve the following goals:

The first goal
The first goal is to obtain knowledge and experience about EEG measurement; how to perform the actual measurement on a person, how to obtain data from the tools, how to setup a workable environment and to pinpoint problem areas. The best way to find an answer for all these questions is simple: perform an experiment.

The second goal
The second goal is to generate data for the analysis and construction of the EEG workbench. Data is required to test the E-Brain and validate results. The requirements for the data are that they are representative of future experimental data and that it shows the different features of the EEG, so that the performance and returned results of the E-Brain are of good quality.

The third goal
Exploration of the possible brain states that can be used to instruct a BCI.

The fourth goal
The fourth and last goal is to see whether or not it is already possible to take this research to the next level and create a basic BCI. This goal can be too farfetched in light of this master thesis (especially the time constraints) and is therefore merely mentioned as a possible goal, but not a requirement for this thesis.

However analyzing the experimental data can shed light on the quality of the experiment and moreover can give recommendation for future experiments.

4.3. *Experimental design*

4.3.1. Experimental design approach

This EEG-experiment is performed with a two goal objective: to evaluate the possibilities to *voluntarily* generate *distinctive* brain patterns (see section 4.1). *Voluntary* meaning that a subject actively tries to generate a brain state independent of the outside world and *distinctive* meaning that two different tasks are on average significantly distinguishable.

The tasks that the subject must perform are chosen because of the way they differ from each other in how they are generated *and* where the activity in the brain is located. Intuitively the most straightforward way to find different patterns is to search for patterns which have known activity centres in different locations.

If the brain focuses on a specific task, not only does activity rise in a certain location it also decreases in others during the performance of a specific task, which can lead to a more expressive pattern.

In BCI there exist two basic ways of control: evoked and voluntary. These experiments try to apply the voluntary approach. Therefore the principle behind the selected tasks for the job is that they should offer the possibility to be performed on a voluntary basis by user activation.

Since this is an initial EEG experiment with one of the main goals set out to explore the possibilities, also task are performed which are not voluntary, but evoke a reaction from the subject. These tasks are included to see whether or not these evoked reactions can also provide some way to generate distinctive brain patterns. In the following sections an outset of the selected tasks is given.

In section 4.4 the practical experiment setup is given. Figure 4.1 shows a photo taken during an experiment. The computer on the far right controls the eye-tracker and is used in the calibration process of the eye-tracker (see section 3.5).

Figure 4.1 Picture taken during an EEG experiment. Left the EEG recording computer, middle the subject computer, right the eye tracker computer.

4.3.2. Baseline task

The baseline task constitutes the control task or the task from which all other tasks must be distinct. The subject should remain completely still and relaxed; do nothing and think about nothing in

particular. This task will be the reference for all other tasks: namely no activity. Well this is not exactly true for the brain; simply doing nothing will still result in enormous activity, however this is unfocused activity which can hopefully be separated from focused activity.

The task is performed with the eyes open and with the eyes closed, because these two states have distinctly different brain activity characteristics. During the experiment the subject is auditively instructed to perform the task. Immediately after hearing the command, the subject starts with the execution. This task can be performed voluntary.

4.3.3. Mental rotation

This task requires the subject to imagine rotating a geometrical object mentally, for instance a cube (see figure 4.2). The goal is to activate that part of the brain which is used for geometrical insight.

The first 5 seconds the cube is displayed on the screen for the subject to see and the subject is auditively instructed to perform the task. After this period it disappears and the subject starts the rotation (although the subject can start before this time). This task is performed both with the eyes open and eyes closed.

The only problem is that without disturbing the measurement it is not possible to validate that the user is actually performing the task. This task can be done voluntary.

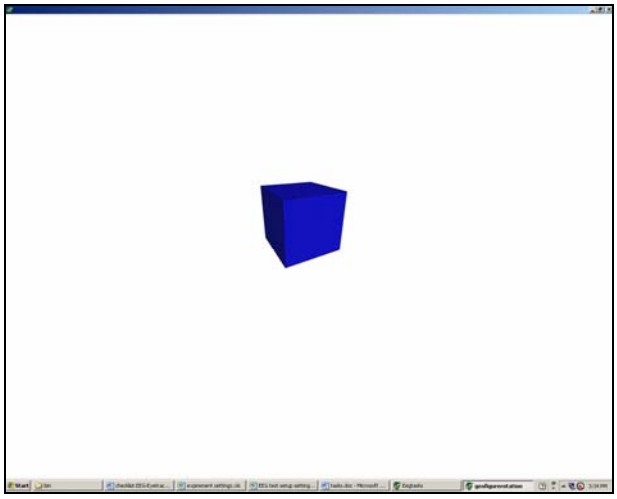


Figure 4.2 Screenshot of object shown before mental rotation.

4.3.4. Motor imagery

The sensorimotor-part of the brain is located in the middle of the cortex and spans across both hemispheres. In general the left part of the brain controls the right part of the body and vice versa. This task tries to activate a specific part of the sensorimotor-cortex by simulation of limb movement. The actual movement must not be performed, because this will result in such muscle activity that will overwhelm the brain activity and no valid signals can be measured. Therefore the movement must be imagined. The brain activity measured when actually performing a task and only imagining it, will be the same [21].

In order to make a first approach to evaluate the difference between movements, two different movements must be imagined for this task. The user will perform the task for both the left hand and the right hand. The first task is rotating the complete limb where the hand makes small circles in front of the subject. The second task is to imagine grabbing an object in front of the subject.
After hearing the 3 second command the action is executed. To be sure that the subject is concentrated on the imaginary movement a 4 second period must be skipped in the data. These tasks are performed both with the eyes open and closed. This task can be done voluntary.

4.3.5. Mathematical calculation

To solve a mathematical problem by calculation of numbers, a specific area of the brain is activated. This task concerns three different scenarios which are auditively presented to the subject:

- Summation, concerns adding circa ten numbers ranging from one to nine (figure 4.3).
- Subtraction, concerns subtracting circa ten numbers ranging from one to nine from the number one hundred.
- Multiplication, concerns multiplying two non-trivial two-digit numbers (figure 4.4).

For the summation and subtraction each number is uttered auditively and the subject adds or subtracts these numbers one at a time until the subject is asked to produce the answer. The main problem with these two tasks is the constant interference of the audio; the impact on the brain patterns is so far unknown and must be considered in the analysis.

For multiplication both numbers are uttered and shown on screen at the start, after which the subject starts calculating the result. When finished the subject calls the answer out loud. The result from the subject is checked by the experimenter for correctness and a mark is inserted in the experimental data. The usable data starts a couple of seconds after the information is uttered and somewhere over ten seconds before the answer is uttered, to be on the save side.

These tasks are performed once with eyes open and once with eyes closed. Although the summation and subtraction task require external input, they are considered to be voluntary by the definition used in the thesis, because the user could think of the number him- or herself. The same goes for the multiplication task, although it is far easier to keep performing this task for an extended period of time.

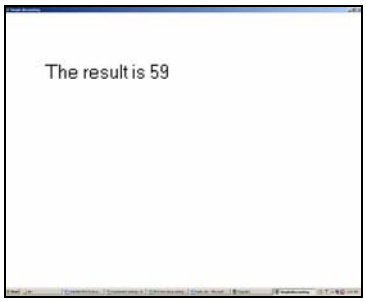


Figure 4.3 Screenshot of math task: answer to summation displayed.

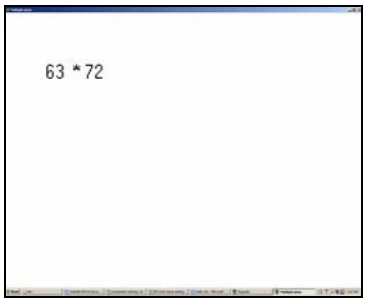


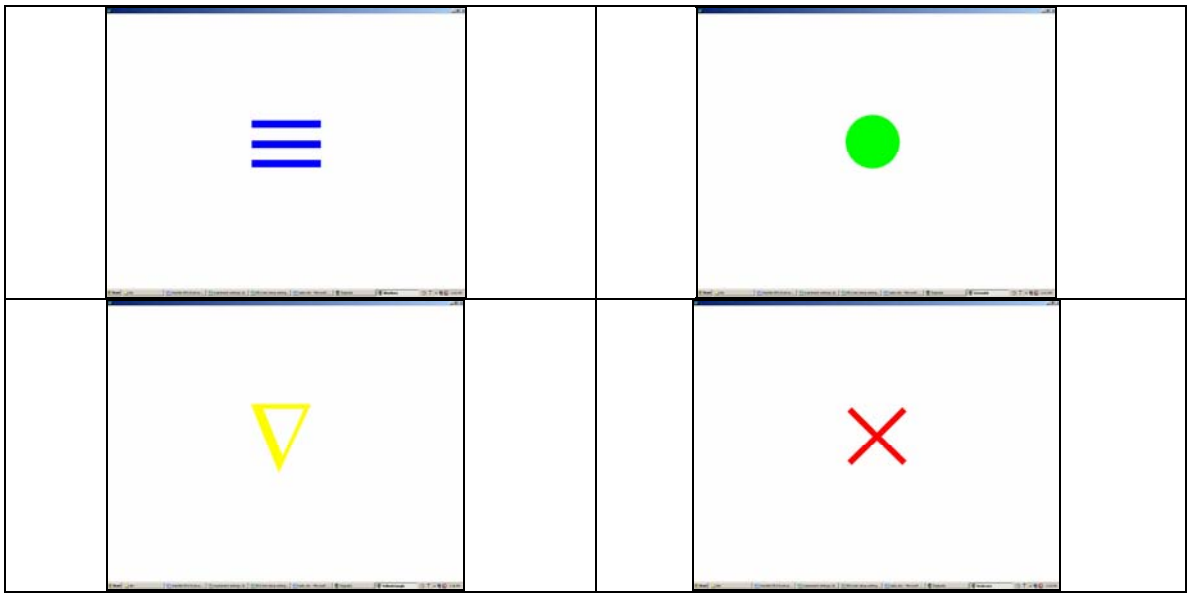
Figure 4.4 Screenshot of math task: multiplication assignment.

4.3.6. Visual presentation

During the visual presentation task four different objects (blue lines, yellow triangle, red cross, green dot, (see table 4.1) are presented to the subject consecutively and random with two second pause in between for eye blinks where the screen is empty. The subject concentrates on the objects for five seconds by actively thinking about the shape and colour of the object.

The idea is to activate the visualization centre of the brain. And moreover, however this must still be proven, if different objects can be recognized from the subject's brain patterns. The idea behind this task is that the subject can voluntary start visualizing a certain object, which can be recognized in the brain patterns, the second stage would be to find differences in what the subject is visualizing, although this latter step seems to be to far fetched at this time.
It must be stressed here that this task is fundamentally different from the geometrical figure rotation task, since that task depends on the subject's spatial understanding.

Table 4.1 Screenshots of objects shown during visual presentation. Clockwise top left: blue lines, green dot, red cross and yellow triangle.



4.3.7. *Visual self selection*

On the screen the four objects are presented in the corners (see figure 4.5) and the gaze position of the subject is tracked by eye-tracker. The subject voluntary selects one of the objects and then concentrates on the object in a similar fashion as the previous task (around five seconds, however this period should not be actively counted). When finished, the subject's gaze returns to the centre, where it is possible for the subject to blink. Then the process is repeated several times. The subject issues a command (hitting the spacebar) to indicate he or she is finished. The actual period of interest starts about one second after the eyetracker picks up a corner location and about one second before the centre location; this is due to eye-muscle movement.

Before this task is commenced, the recording of EEG is paused to calibrate the eye-tracker. The task is performed with open eyes and is considered voluntary.

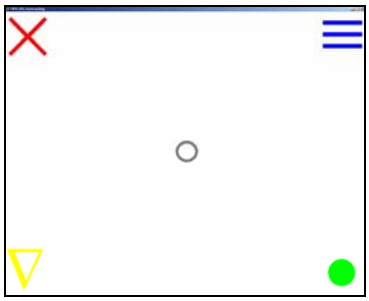


Figure 4.5 Screenshot of objects shown during visual self selection.

4.3.8. *Visual and auditive presentation*

This task is very similar to the visual only task; in addition now not only is the figure shown in the centre of the screen it is also auditively uttered. The subject must concentrate on the object for a duration of five seconds. Again there is a two second period between two consecutive objects for the subject to blink his or her eyes.

The actual recording of the data starts when the auditive uttering of the object is finished. This task is performed with open eyes and is considered voluntary.

4.3.9. *Auditive presentation*

During auditive presentation the objects are presented to the subject auditively only: nothing is presented visually, the screen is blank. The subject has to imagine the object in his mind and concentrate on its features. Immediately after one object is finished, the next follows in case of the eyes-closed condition. In the eyes-open condition there is a two second blink period, which is also auditively presented. This task is considered to be voluntary.

In the visual, visual-auditive and auditive tasks the subject has to focus and concentrate on a particular object. The goal is to compare this brain state of focusing on an object to the other (main) tasks. The focused state may differ from the baseline task. In addition, the secondary goal is to find differences between these three tasks since they are evoked in different fashions. The third goal would be to find differences in the brain patterns for different objects, however this is probably to farfetched in the scope of this thesis.

4.3.10. *Hyperventilation*

This task is not included with the goal to create distinctive patterns, but for calibration purposes. Simulation of the hyperventilated stated is reached by complete inhalation followed by complete exhalation in fast repetition. This will create a feeling of dizziness in the subject and heavy activity in the brain signals, in part created by the heavy muscle movement need to create this state. It is thought that using this brain pattern can assist in determining the brain characteristics of the subject.

4.4. *Experiment setup*

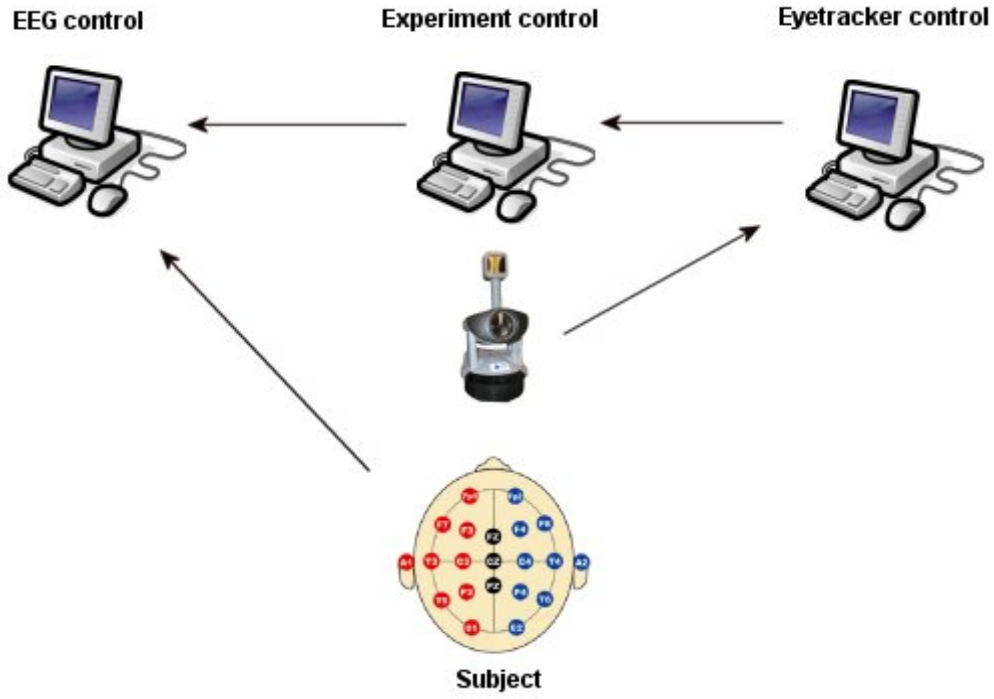


Figure 4.6 Schema of experiment setup. Arrow shows direction of communication. The EEG control computer records the signals coming from the EEG cap on the subject and inserts the marks coming from the Experiment control computer. The Eyetracker control computer communicates the gaze coordinates measured by the eyetracker camera and sends these to the Experiment control computer.

A schematic representation of the practical setup of the experiment is given in figure 4.6. It consists of:

- One computer (left in figure 4.6) to record EEG signals from the EEG cap, facilitated by the TA software (section 3.2). And insert the marks coming from the experiment control computer.

- One computer (middle in figure 4.6) to instruct the subject and communicate the tasks to the EEG control computer. This is controlled by the Experiment control program (section 5.2.1).
- One computer (right in figure 4.6) to control the eyetracking process and sent the gaze coordinates to the Experiment control computer.

The eye-tracker camera is located in front of the middle screen. And the subject is sitting comfortably and relaxed in a chair with the EEG cap on his head.

4.5. *EEG cap setup*

Measurement electrodes
The configuration used for the EEG experiment is the full 10-20 system, containing all the electrodes on the cap: Fp1, Fp2, F7, F3, Fz, F4, F8, T3, C3, Cz, C4, T4, P3, Pz, P4, T5, O1, O2 and T6.
As well as both ear-electrodes A1 and A2, which are attached using the clip-electrodes. Furthermore a pulse measurement is added to the scheme (as electrodes EKG1 and EKG2). Together these electrodes measure the pulse generated at both wrists.

Reference- and ground electrodes
The reference electrode is the electrode in front of the head, just in front of electrode Fz.
The ground electrode is the electrode at the back of the head, just below electrode Pz. These two electrodes are not shown in the data.

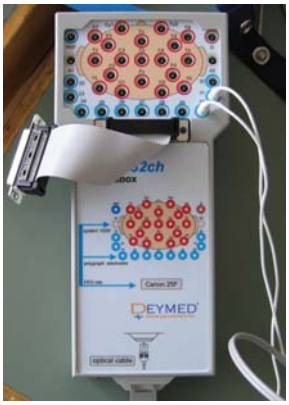


Figure 4.7 The EEG headbox. The flat cable (middle) is connected to the EEG cap. The holes on top can be used to place additional electrodes like pulse or ECG.

The EEG cap communicates with the acquisition software via the EEG Headbox (see figure 4.7). The EEG cap is connected with a flat cable to the box. Additional cables can be plugged in to measure other phenomena like pulse and muscle movement

4.6. *Artifacts*

The subjects are expected to be comfortable and stable in their chairs and therefore should generate almost no muscle movement aside from that generated by the eye muscles.
During tasks with eyes open there is always the possibility of eye-blinks. However during the following tasks this risk is reduced, because of the option to blink between trials:

- Visual presentation.
- Visual self selection.
- Visual and auditive presentation.
- Auditive presentation.

In all other tasks, and also in these tasks mentioned but to a lesser extend, it is possible that eye blinks are present in the recorded EEG data. These should be removed in the data pre-processing phase.

Eye-movement from using eye tracker might interfere with EEG measurement; therefore the data from these periods is not used at all.

4.7. *Task localization in brain*

A head start is made if it is known in advance in which area of the brain it is highest likely that the bulk of the activity will occur (as mentioned in 2.3.2). In table 4.2 the most likely location of activity for

each task is listed. For an overview concerning the brain: take a look at figure 4.8. Validation should be done with classification.

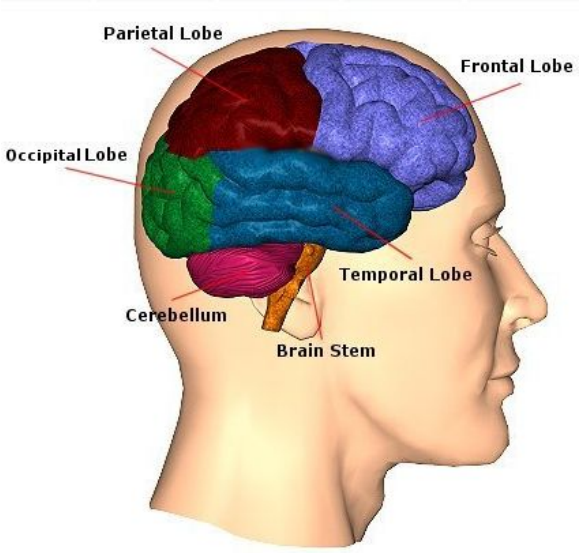


Figure 4-8 Overview of the brain lobe locations; image from "Centre for Neuro Skills".

Although this is a good place to start, for some tasks the areas are less strictly defined than for others. Additionally no two people are the same and differences (can) exist between two brains. For the location of the mentioned electrodes in table 4.2, refer to figure 2.7. The discussion, recommendations and conclusion concerning the experiments are covered at the end of chapter 9.

Table 4.2 Table of most likely brain locations, frequencies and most likely electrodes of specific tasks [51].

Task	Location	Frequencies	Electrodes
Baseline task	No specific region.	Increased Alpha rhythm	All
Imagery Movement task	Sensorimotor cortex (sensory resides in parietal lobe and motor resides in frontal lobe). Right hemisphere for left side of body and vice versa. Hand movement is located slightly up from the middle. Arm movement is located just above hand movement (see figure 2.4).	Mu & central beta rhythm	Focus on T3, C3, Cz, C4 and T4.
Mathematical Equation task	Mainly activity in the parietal lobe (with bias to the right side). Studies have shown that increase activity is also present in the middle of the frontal lobs [17].	No specific frequency	Focus on P3, Pz and P4.
Mental rotation task	Occipital cortex which is used for visualization. Rotation however is partly a mathematical task and there for also activity in the right (superior) parietal lobe can be expected [11, 23].	No specific frequency	Focus on T5, P3, Pz, P4, T6, O1 and O2.
Visualization task	Activity should arise in the occipital cortex.	No specific frequency	Focus on O1 and O2.
Auditive task	Activity should arise in the temporal lobe.	No specific frequency	Focus on T5 and T6.

Chapter 5
5. BCI Environment

Performing the EEG experiments covered in chapter 4 required certain programs. These programs are handled in this chapter, except for the analysis workbench which is handles in the next chapters. The programs concerned here to create a suitable BCI research environment are the d*ata conversion program* and the programs build to support the experiment (ET tracker and experiment control software).

5.1. *BCI research environment requirements*

The following functionality must be reached to create a suitable research environment for BCI research:

- *Experiment tools*
  A program which controls and monitors the experiment at a basic level; the top level control is off course in the hand of the experimenter.
- *Data conversion*
  Conversion of raw EEG data provided by the Truscan software to useable data format for further analysis.
- *Data pre-processing (E-Brain chapter 6, 7 and 8)*
  The ability to retrieve any portion of the data and to remove artifacts and other undesired properties from the data.
- *Data analysis (E-Brain chapter 6, 7 and 8)*
  An environment where the EEG data can be qualitatively examined in a structured way, so that valid conclusions can be drawn concerning the performed experiments. The results must also be re-producible.

5.2. *Experiment tools*

The tools developed for the purpose of the experiments are covered briefly here. Both programs are implemented in Delphi (see section 5.3.5.1).

5.2.1. Experiment control program

The experiment control program is used during the EEG experiments. The program is mainly written by Pavel Hrubeš. Various screenshots of the program can be found in section 4.3.

5.2.1.1. Goal

The program has the following goals:

- Control the flow of the experiment and present the tasks to the subject.
- Control the communication with the EEG system.
- Control the communication with the Eye-Tracker system.

5.2.1.2. System

The program allows loading tasks which should be executed during the experiment. Each task has a delay before execution, a duration, a task-specific code and a definition of the task-specific program to run.

These task-specific programs can consist of any executable program or sound file and are covered in section 4.3. The tasks are executed consecutively.

The 'visual self selection' task used in the experiment (see section 4.3.7) requires the eye-tracker to recognize the position of the subject's gaze on screen. The eye-tracker continuously sends these coordinates to the experiment-program which recognizes at which object the user is looking. This information is sent to the EEG-system.

5.2.2. *Mouse control program*

In the initial phase of exploring the possibilities of creating a new sort of device control, the first steps were taken in the direction of a multi-modal mouse *and* brain control. The using the eye-movement to define cursor position and BCI-action to present commands to the system. In light of this idea a program was constructed which allows cursor control using the eye-tracker (see figure 5.1).

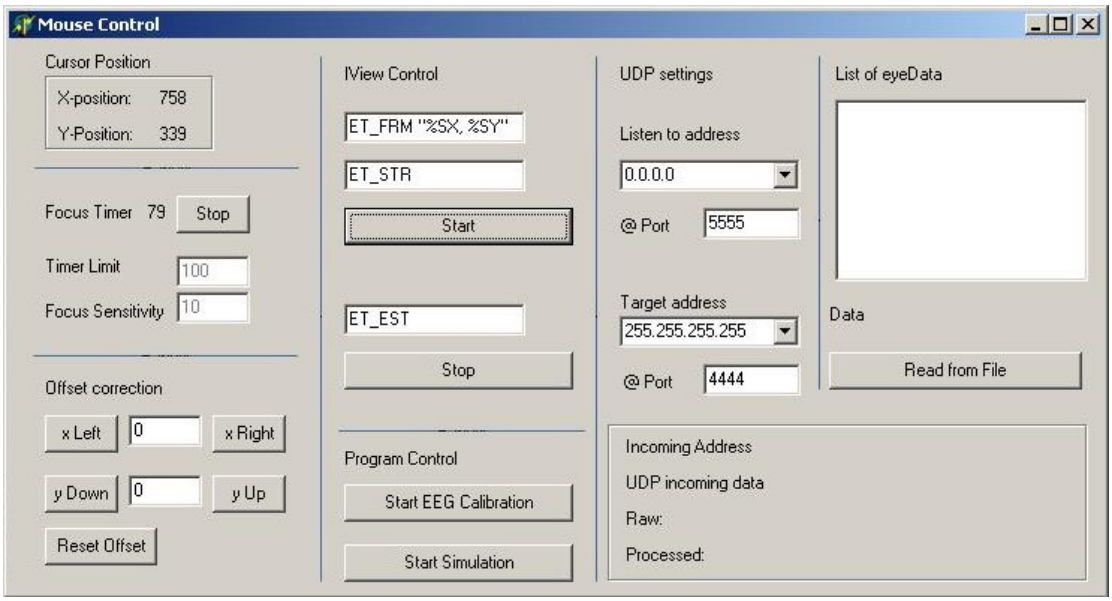


Figure 5.1 Screenshot of the mouse control program. Upper left corner displays current coordinates.

Later it became clear that for several reasons this was not a viable way to continue:

- BCI is a big enough challenge in its own right.
- Superior (commercial) systems exists, they are based solely on eye-tracking and blink recognition.
- Eye-movement control inherently inflicts heavy artifacts on the EEG data.
- The challenge of BCI is to create a system of control based solely on the brain (see section 2.1.1) and not directly offer a practical solution and thus is of a more fundamental nature.
- The precision offered by the eye-tracker at the TU Delft, which is not meant for control but for psychology research, is not on a level to actively control the cursor for a longer period of time.

5.2.2.1.Goal

Control the cursor movement on screen with direct visual feedback using the gaze position of the subject's eyes.

5.2.2.2.System

The mouse control program is designed to show the performance of the user and return feedback.
In combination with the eye-tracker system the following functionality is offered:

- Eye-movement recognition.

- Visual feedback using the mouse and display of gaze coordinates.
- Focus timer, which fixates the mouse position if the position of the mouse is within a certain sensitivity area after *x* seconds. It was thought that keeping the mouse steady *and* issuing commands to the BCI is too difficult. This option was provided to offer an intermediate for BCI commands input. The idea behind it is that the user can completely focus on the BCI and does not have to 'keep the mouse cursor steady'.

5.3. *Data conversion*

The Truscan Acquisition program (see section 3.2) generates two files. The first contains the measured voltages at the scalp during time. This format (binary stream) is not directly usable. Therefore a dataconversion program must be constructed which streamlines this process from raw data to a workable format which can be used for further data processing and analysis. The second file contains the marks/timestamps of the experiments which are merged into the database in the process.

5.3.1. Conversion steps

Steps in the conversion process (figure 5.2):

- Brain activity.
- Measured voltage in the electrodes.
- Conversion from analogue to digital.
- Storage of digital signal in binary stream file.
- *Conversion from binary file to database (or text file).*
- Conversion from database to workable Matlab format.

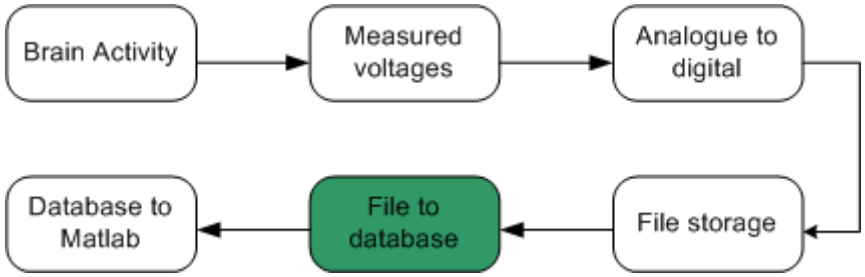


Figure 5.2 Data flow of conversion process.

The first four steps are covered by the EEG equipment provided by Deymed. The step from binary stream to database must be facilitated by the dataconversion program.

5.3.2. Functional requirement

The *data conversion program* must meet the following functional requirement:

- Convert data from the binary files (samples) and text files (timestamps) to a user-defined structure into a database.

5.3.3. Non functional requirements

The *data conversion program* must meet the following non functional requirements:

- No data loss must occur from the initial format to the end format.
- The data must be stored in a database.
- The conversion must be fast and reliable.
- The dataconversion program must be user friendly and intuitive.

5.3.4. Design

The *data conversion program* has been structured into several components. Each component has its own responsibilities and scope based on the requirements and the characteristics of Delphi. To define the basic functionality required by the program the next section covers the use case.

5.3.4.1. Use case

Use cases are used to identify the basic functionality to the user and define the system boundaries of the *data conversion program*. Use case diagrams represent this functionality from the user's point of view.

Actors:
User, represents the user who interacts with the program as the person wants to convert files to the database.
Storage, represent the storage entity either as file system or database.
This case (figure 5.3) represents the user interaction with the *data conversion program*.

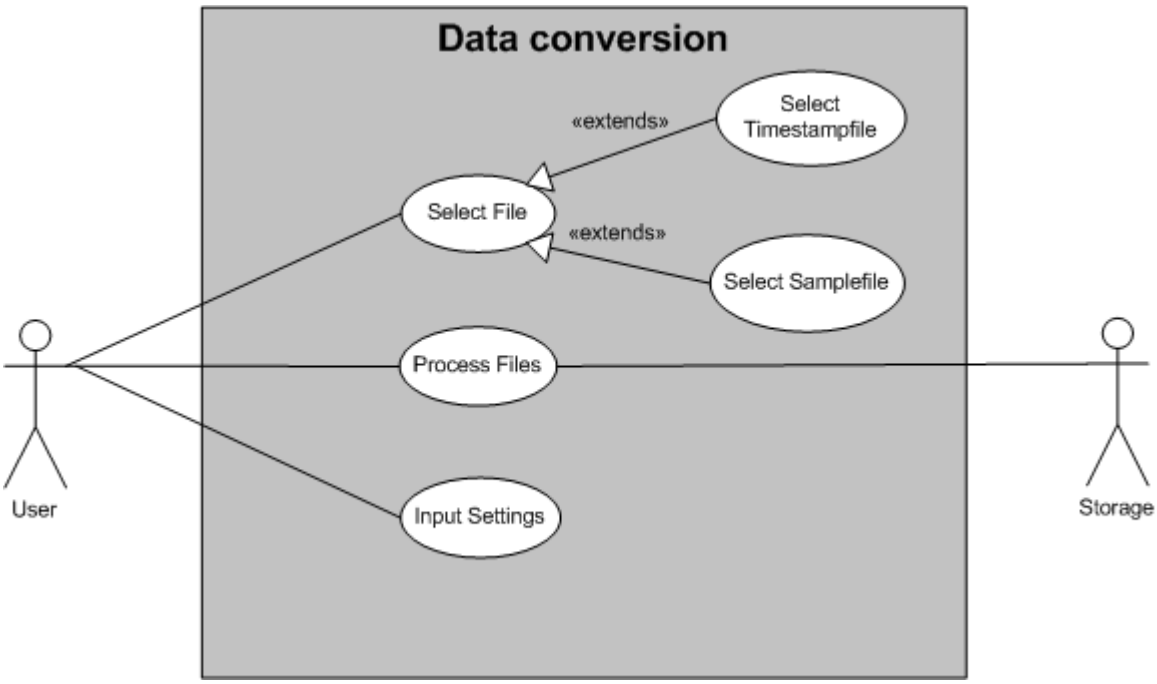


Figure 5.3 Use case diagram Data conversion.

Use cases:

- *Select File*
  - Description: The *User* selects a file for further processing.
  - Extensions:
    - *Select Timestampfile*, the *User* selects a file containing the timestamps from an experiment.
    - *Select Samplefile,* the *User* selects a binary file containing the samples from an experiment.
  - Actor: *User.*
- *Process Files*
  - Description: The *User* starts the conversion process of the selected files to the database
  - Actors: *User* and *Storage.*

- *Input settings*
  - Description: The *User* inputs the desired settings concerning the processing of the files: codes for defining timestamps, database connection settings, database structure definitions and performance settings.
  - Actor: *User*.

5.3.4.2. Architecture

The next step is to convert the use case into an architecture. The following class diagrams (figure 5.4) represent the structure of a system in terms of objects, attributes and relationships.

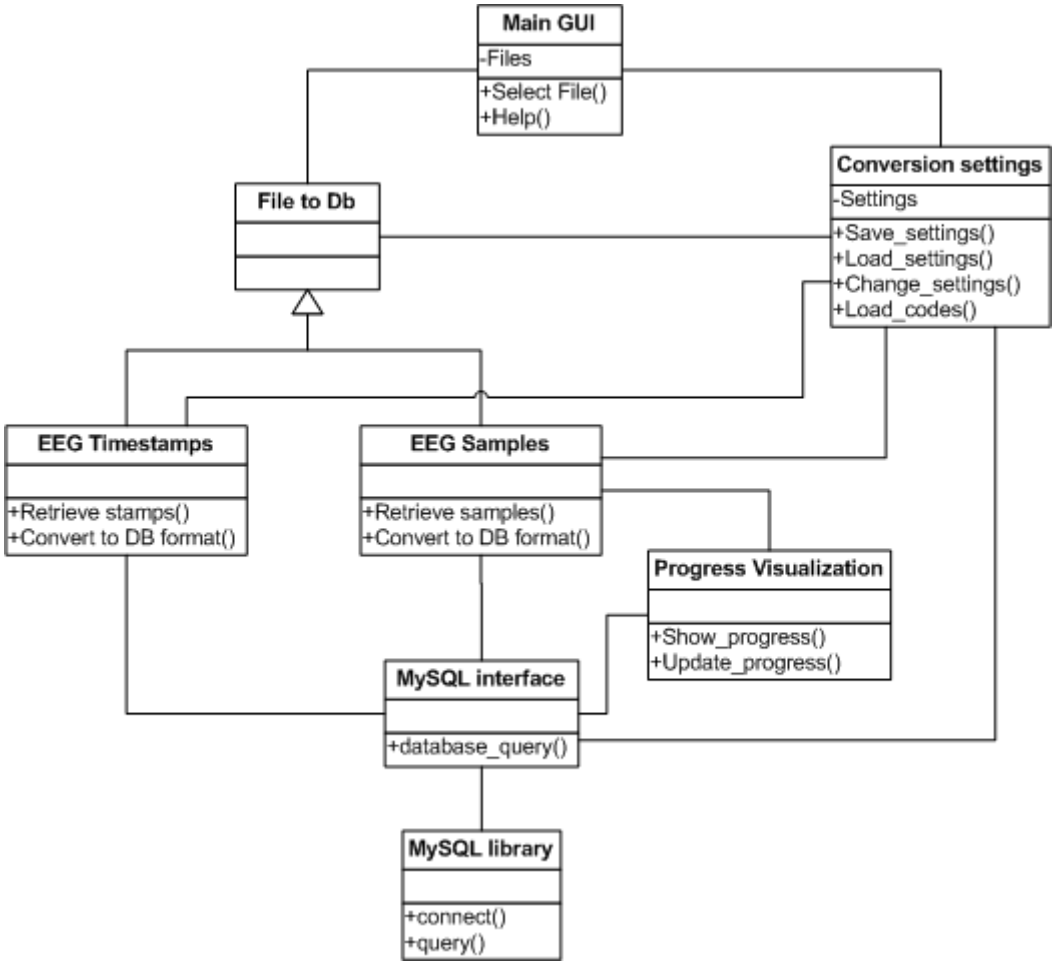


Figure 5.4 Class diagram of *Data conversion program*.

Description of the classes from figure 5.4:

- *Main GUI* covers the basic interaction with the user: file selection, program settings and help functionality. It also initiates the process of the data conversion.
- *File to Db* converts the selected files to database format and inserts that data into the database. It delegates this functionality to *EEG Timestamps* and *EEG Samples*.
- *Conversion settings* keep track of all the settings required by the program and data conversion. It enables the user to modify all the settings.
- *EEG Timestamps* converts a timestamp file to the appropriate database format and calls the *MySQL Interface* to transfer the data to the database.
- *EEG Samples* converts a sample file to the appropriate database format and calls the *MySQL Interface* to transfer the data to the database.

- *MySQL interface* handles all the database queries from the program and transfers them to the MySQL database using the MySQL library.
- *Progress Visualization* enables to represent the progress of any process in the *data conversion program*.
- *MySQL Library* contains the functionality to communicate with the MySQL database.

5.3.5. Implementation

5.3.5.1. Language

The *Data conversion program* is implemented using the language Delphi from Borland. The Borland Delphi offers a visual programming environment to construct Delphi programs fast and reliable. Delphi is well documented and offers the possibility of writing and acquiring stand alone programs quickly. The program is used in construction of the dataconversion program, mouse control program and the experiment program.

Borland Delphi 6.0 Personal Edition - www.borland.com/delphi

5.3.5.2. Function overview

The main functions used in the *data conversion program* (see figure 5.5) are listed here with a short description.

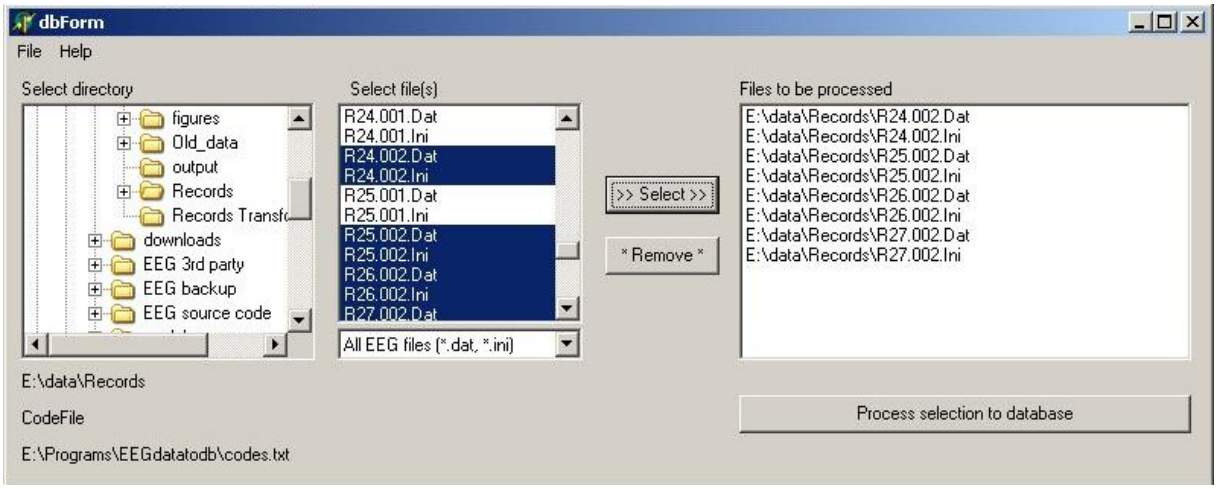


Figure 5.5 Screenshot of the data conversion program.

- Select File(), this is not a real function but more a combination of different functions which offer the user the possibility to select multiple different files from a location and manipulate the acquired list.
- ProcessFilestoDb(), first checks the database and code settings and then processes the file list based on the type.
- ConvertToTimestamp(), searches the timestampfile for the correct information in combination with acquiring the correct codes and sends this data to the timestampToDb() function.
- TimestampToDb(), transforms the timestamp information into a database structure and sends a query to writeToDb().
- ConvertToSamples(), processes and evaluates the header of a sample file to find the information on the used electrodes and samplerate. After which the database structure can be defined and the body of the header is sent in a query to writeToDb().

- WriteToDb(), processes the input query to the MySQL database and shows the progress to the user.

5.4. *Data storage design*

The generic structure in the database for storing the EEG data consists of four tables, from which the first two (*sample* and *timepnt*) are necessary for correct functioning of the EEG workbench. This structure is the default used, however any custom structure can be provided by the user. The underlined words represent the primary keys:

Table name: *sample*

smpl_id	sess_id	Fp1	Fp2	… all other electrodes

The *sample*-table contains a list of electrodes used in the experiment. The data in the sample table is a list of all the measured values from the binary stream. The *smpl_id* combined with the *sess_id* (from session) is the primary key.

Table name: *timepnt*

ts_id	tses_id	tscode	tstime	tsname

The *timepnt*-table contains a list of all the timepoints recorded during the experiment with the *tscode* (and *tsname*) indicating the event and the *tstime* corresponding with *smpl_id* from sample. The *ts_id* combined with the *tses_id* (from session) is the primary key.

Table name: *session*

session_id	date_time	subj_id	samplerate

The *session*-table contains information on the session. The *sessiond_id* combined with the *subj_id* (from subject) is the primary key.

Table name: *subject*

subject_id	name	birthdate

The *subject*-table contains information concerning the subject used in the experiment.

Implementation
At first the Firebird database was used, however at a later stage severe compatibility problems required a switch to MySQL.

MySQL
MySQL is the most popular and leading open-source database available. It is a high-performance and reliable database, which has an excellent documentation and good support. The MySQL database is used to store data from the EEG experiments. For administration purposes the MySQL Administrator 1.1 and the MySQL Query Browser 1.1.14 are used, both provided by MySQL.

MySQL 4.1.14 - www.mysql.com

Firebird
Firebird is a relational database offering many ANSI SQL-99 features that runs on Linux, Windows, and a variety of Unix platforms. Firebird offers excellent concurrency, high performance, and powerful language support for stored procedures and triggers. The Firebird database can be used to store data from the EEG experiments.

Firebird 1.5 - www.firebirdsql.org

Chapter 6
6. E-Brain Design

This chapter covers the construction of the workbench which is part of the goal to setup an environment for a structured approach of analyzing EEG experiments (see section 5.1). And thus tries to answer the third research question as stated in section 1.6: *How to process and analyze the data?*

Together with the experiments this is the main focus of this master thesis. In this chapter the E-Brain will be described in terms of goals, design decisions and requirements. Chapter 7 will continue with the implementation and chapter 8 with testing and evaluation of the E-Brain.

E-Brain is short for **E**EG**-**based **B**CI **R**esearch **A**nalysis **I**nterface **N**atural workbench.

6.1. *Why a workbench?*

Why create a workbench in the first place? There is a need for structured analysis approach of voluntary brain signals. Today already exist some alternative EEG research programs, they are covered in section 0; however the data provided by the Truscan equipment is not directly compatible with those programs. Moreover these programs will limit, if the possibility is offered in the first place, the way of analyzing. And additionally offer limited analysis capabilities for voluntary based BCI experiments, meaning that the methods for searching for features are limited. They focus on evoked control. BCI2000 is primarily focused on the classification and online performance of BCI.

Moreover if the workbench is 'home build', it is exactly known how the methods and techniques work and therefore sensible conclusions can be drawn about the analysis results and it can be modified and altered to fulfil specific needs when required.

6.2. *System requirements*

The E-Brain must provide a structured and qualitatively high environment to analyze EEG data coming from any EEG-based experiment. The E-Brain must meet the following requirements:

- Offer a structured approach for analysis.
- Produce qualitative results.
- Produce reproducible results.
- Offer stable environment.
- Be user friendly.
- Be effective.
- Be modifiable and extensible.

The effectiveness requirement is not only dependent on an intuitive and logical interface, but also on the skill of the user. The E-Brain will guide the user through the process of analysis; however the user is expected to have adequate knowledge concerning the EEG data and the analysis methods to produce any sensible results. The workbench is therefore labelled as for 'professionals'.

Defining a measure for the requirement of qualitative results is harder and can be re-stated as usable analytical performance: meaning that the results should be both meaningful and useful to continue in the next step of BCI. Although the eventual outcome of this criterion is heavily dependent on the quality of the EEG data and experiment.

The workbench must try to guide the user as much as possible, although not restricting any analysis decisions. Therefore the E-Brain should guide the user in the first part (pre-processing), where no fundamental deviations can be made in the process: to analyze one must have solid data which meets

certain requirements. Thus it should give the user some initial structure. After this initial process the user is free to choose any of the available analysis method and continue as he or she sees fit. It must be noted here that different methods require different data properties.

In order to create a usable tool for analysis, the E-Brain must proof to be stable and reliable. Since this thesis concerns pioneering research into BCI the E-Brain must be extensible and modifiable into the classification phase and offer appropriate conditions to do so.

6.3. *E-Brain functional requirements*

This section covers the functional requirements that are implemented in the workbench.

6.3.1. Overview functional requirements - pre-processing

The first step in the data analysis is making the now available data usable. As mentioned before typically EEG data is polluted with artifacts like eye blinks and muscle movements, but also an electrode drift during measurement. A lot of the recorded EEG data can be discarded, since no information is available about what was going on during these periods. The workbench must provide means to clear the data of these specific characteristic artifacts and parts; both automatically and manually/visually by the user. Furthermore the analysis methods require the data to be of a specific sample length.

Pre-processing functionality

- Retrieve data
- Basic pre-processing
    - Down sample data
    - Re-reference filter data
    - Bandpass filter data
    - Baseline correct data
- Inspect data, visual inspection
- Advanced pre-processing
    - Artifact rejection
    - Artifact removal: Independent Component Analysis (ICA)
- Segmentize data

6.3.2. Overview functional requirements - analysis

Literature shows (chapter 2) that there exist numerous analysis methods to inspect time series of the nature of EEG. Both from the view of sheer workload and from the view of the user analyzing the data it is not desired to implement all these methods. Therefore a selection of these methods must be made to conduct an explorative data analysis. The following analysis methods have been implemented in the E-Brain. The selection is based on the proven performance in the area of BCI and frequency of current use in the field of BCI (see section 2.11).

- Fast Fourier Transform (FFT)
- Autoregressive Model (AR)
- Common Spatial Patterns (CSP)
- Linear Discrimination (LD)

FFT and AR are the most commonly used methods for EEG analysis. And CSP is on the increase in the field of BCI. This section continues to describe every functional aspect of the workbench mentioned in this and the previous section.

6.3.3. Datastructure & data retrieval

Data retrieval concerns querying either the database or a file for the desired sections of EEG data. The data is stored in the database as explained in section 5.4. There are three ways to retrieve the data:

- Database; facilitated by the E-Brain database GUI.
- Structured file; previously saved data structure.
- Matrix file; to load a matrix (containing channels x samples). This function is added to load third party data (non-Truscan) in to the workbench.

A single experiment is called a session, which contains various tasks. Each task can consist of one or more trials. The retrieved data is stored in a datastructure in the E-Brain in the following way:

- Data; global name, can also be subject (e.g. exampledata)
  - Task; number of the task (e.g. exampledata.t1, exampledata.t5 etc.)
    - ID; identification field (e.g. exampledata.t1.id)
    - Header; (e.g. exampledata.t1.header)
    - Trial; number of the trial (e.g. exampledata.t1.tr1, example data.t1.tr23 etc.)

Every id-field contains a vector with the session id for each trial. Every header contains a vector with the electrode names. Every trial contains a matrix with all the electrodes and their respective data. In this structure it chosen to sort the data by task rather than by session, so that there will be easy access to similar data.

6.3.4. Data down sampling

During the experiments the samplerate was set to 256Hz it can be opted to down sample this to enable faster processing of the data in the next stages. Since it can be reasoned that a certain samplerate is too high to add precision to the calculations and thus will not add much more accuracy.

In practice 256Hz is a perfectly good samplerate, this option is added more to be able to down sample the samplerate of 'foreign' matrices loaded in to the workbench, which sometimes have a very high samplerate (e.g. 1000Hz).

6.3.5. Basic pre-processing

The workbench offers three transformations of the data:

- Reference filtering, improves the quality/Signal-to-Noise (SNR) ratio of the data (section 2.6.2.1). Two options are available:
  - Common Average Reference.
  - Small Laplacian Reference.
- Band pass filtering, allows for filtering frequencies from the data.
- Baseline correction, removes any trends in the data, like for instance drifting electrodes. It also leaves the data zero-mean. Between two sessions the electrodes are removed, washed and filled with gel, this inherently means that from the outset two sessions have different starting conditions.

Any combination of these options can be applied to the data.

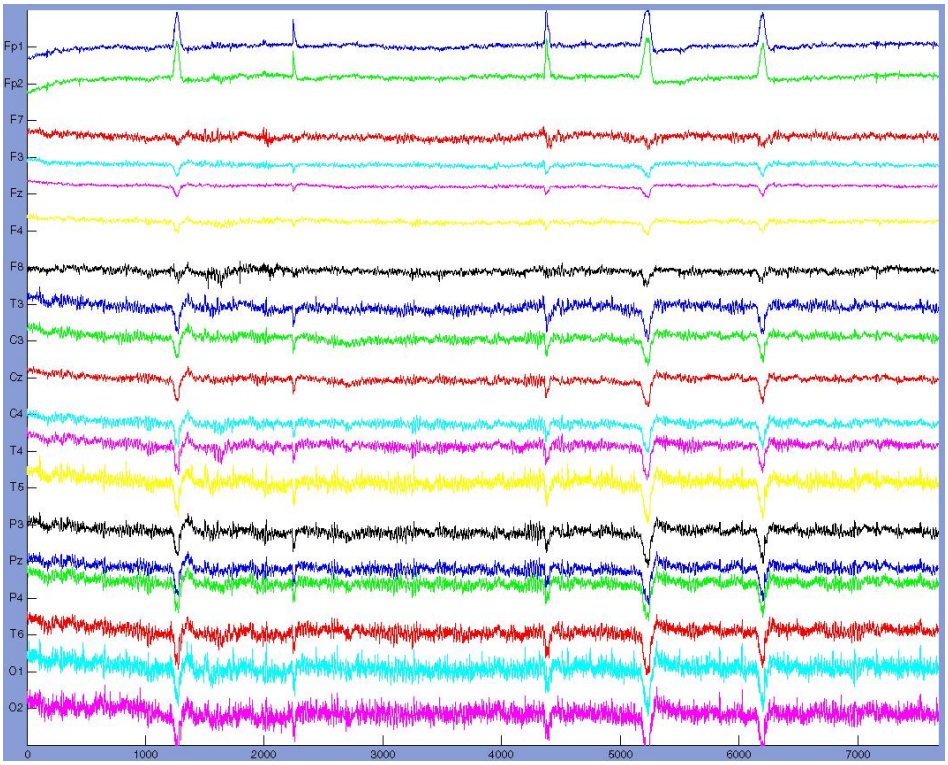


Figure 6.1 Raw EEG data. The y-axis (vertical) displays electrodes with the magnitude in Voltage. The x-axis (horizontal) displays time. The peaks shown in the data are eyeblinks.

In figure 6.1 the raw data is shown as it is stored in the database. Figure 6.2 shows the same data after applying a CAR filter, a bandpass filter between frequencies 1 and 30Hz and a baseline correction. The spikes that can be seen in the figures are eye-blinks.

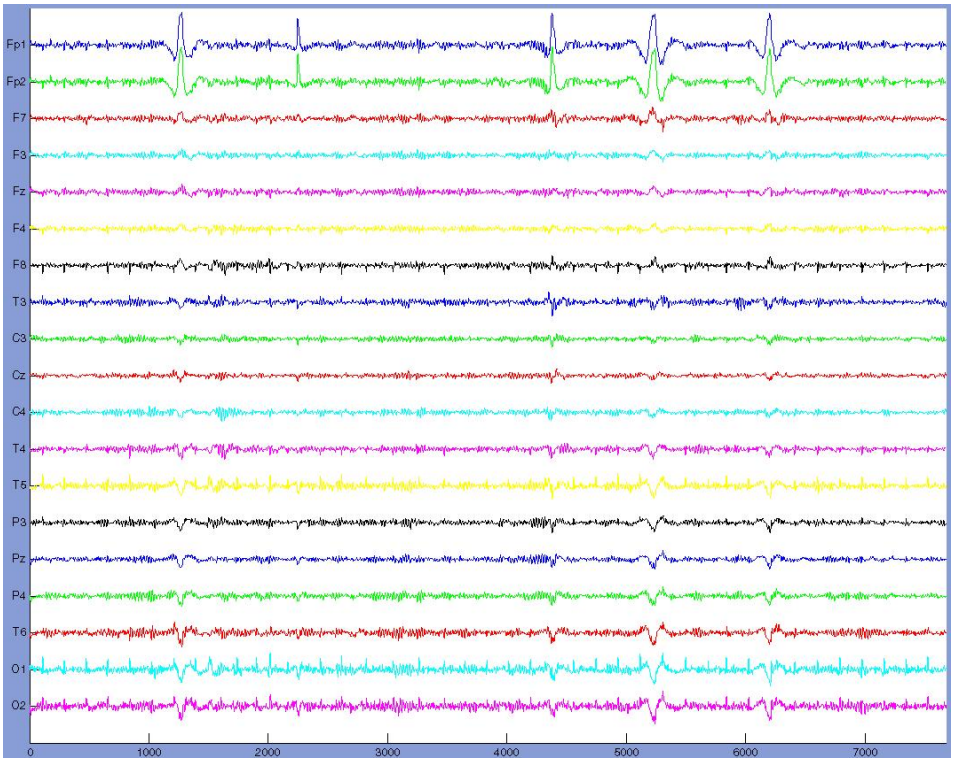


Figure 6.2 Raw EEG data after basic pre-processing. Clearly it can be seen that the variation in the lower electrodes has become smaller compared to figure 6.1. And the curve in the beginning of the signal is removed.

6.3.6. Advanced pre-processing

EEG data is very susceptible for artifacts as mentioned in section 2.6.4. There are two ways to get rid off artifacts, by rejection or ideally removal. The first option is the easiest way: simply cut the undesired piece from the data. This is a good possibility when there is an abundance of data or if a specific piece of data is not overly important to the analysis.

The second option tries to remove the artifact from the data and preserve the original signal. From a computational point of view it can be preferable to discard data instead of trying to remove an artifact; computation time is more an issue in online BCI.

6.3.6.1. Artifacts

First the artifacts must be recognized and categorized. The following artifacts are most prominent in EEG data (in order of importance: occurrence, avoid ability and impact) and visualised in figure 6.3.

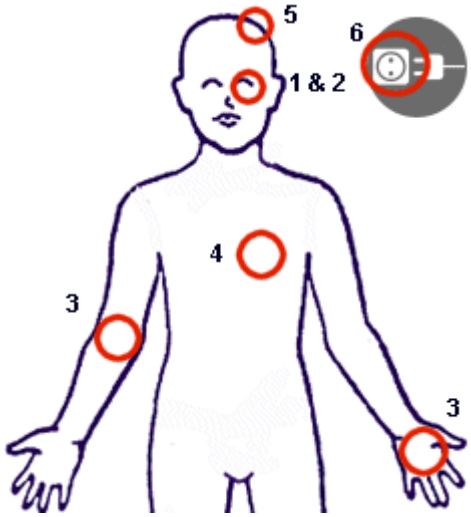


Figure 6.3 Artifacts sources in the human body & experiment environment.

1. Eye blinks, these artifacts are easily recognizable in the EEG data. Especially in the frontal electrodes. Eye blinks are very hard to prevent.
2. Eye movement, these artifacts are significantly harder to detect and are also most prominent in the frontal electrodes. Eye movement is impossible to prevent.
3. Body muscle movement, these artifacts are hard to detect in the EEG data and can show as some peak activity. However body movement can be minimized during the measurement.
4. Heart beat / pulse, these artifacts are ever present and cannot be prevented. It is also quite hard to recognize in the data.
5. Electrode drift, these artifacts are usually due to sweating and result in the signal's baseline to drift. These artifacts can be prevented to a certain extent by proper conditions during the experiment, although this is not a guarantee.
6. Electrical interference of the 50Hz net. Easy to remove using a bandpass filter.

6.3.6.2. Artifact rejection

For the purpose of artifact rejection, actual deletion of the data, two options are available. Next to automatic rejection, it is also possible to manually survey the EEG data. The possibility of zooming and cutting by the user provides a mean to remove any undesired section from the data.

Not all artifacts can be rejected, for instance rejecting heart beat inflicted EEG data will leave the experimenter with little data. Eye blinks and serious eye movement can be both recognized in the data and rejected. In case of heavy body movement it is better not to use that particular piece of data at all.

6.3.6.3. Artifact removal

Often it is desired to remove the artifact from the signal and preserve the EEG data. ICA offers this possibility (see section 2.6.5). Removal using ICA has another distinct advantage over rejection, which is that it can remove pulse artifacts from the data by identifying and singling out the heartbeat source.

In figure 6.4 the data from figure 6.2 is artifact cleaned using the ICA algorithm. It is visible that the eye-blinks have been *removed*.

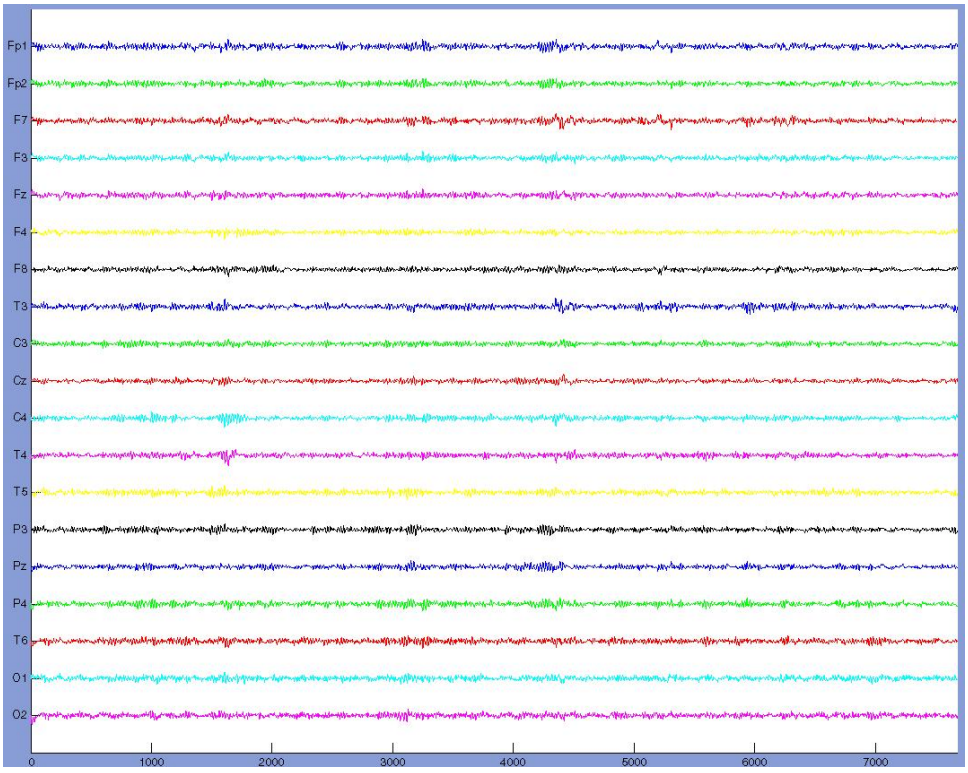


Figure 6.4 Raw EEG data after basic pre-processing and ICA. The eye blinks are removed from the data. The peaks from figure 6.2 have vanished.

6.3.7. Data inspection

Besides applying all kind of processes on the data the user can also view the effects of his or her actions on the data by viewing these plotted as the voltages measured against time (see figure 6.1, 6.2 and 6.4).

The user can specify a specific part of the selected data or interactively zoom with the mouse. It is also possible to manually remove selected portions from the data and save these to the datastructure.

6.3.8. Data segmentation

Once the data is selected, processed and all the artifacts are removed it must still be segmentized into smaller pieces. What size to select is not predefined but correlates with the next step of the data analysis: the analysis method. For instance FFT requires the size of a data piece to be carefully selected

and it will directly influence the result of the analysis. More on choosing the segmentation-size can be found in the respective section of each analysis method.

6.3.9. Data analysis

All the analysis functionality is covered next. Per method it is shortly summarized what functionality it offers. Furthermore all the analysis methods allow the images that are generated (see chapter 7) to be stored in a file for later reviewing.

6.3.10. FFT

The FFT is one of the most used and basic method to show the frequency spectrum of a time series such as EEG. As mentioned in section 2.10.1.1 it provides a way to make the frequencies in a specific part of the data visible.

The user should select a frequency range over which the FFT is calculated. It is advised to select the same range as used for the bandpass filter. Since any larger range will not provide more information and a smaller range might obscure important information.

Choosing a specific data length using FFT is not a trivial task. Picking the length to short and the lower frequencies will be represented incorrectly. Settings the length to long and the entire spectrum will flatten and the presented picture will be without any information. Once a data length has been chosen, this length should remain the same for the entire analysis. Or else it will be very hard to compare results.

FFT visualization options
The FFT spectra of the EEG time series can be shown in different ways:

- Per task over multiple sessions. A number of frequency spectra are drawn, in a single graph, depending on the number of trials used. One or multiple electrode at a time.
- Multiple tasks are shown over several sessions concerning a single electrode.
- A 3D overview of a specific time period calculated using a sliding window

6.3.11. AR

Autoregressive models are very often used in BCI to extract characteristic features from the EEG data. Background theory on AR models can be found in section 2.10.1.3.

The user will be able to select a range in which the optimal model order is searched, more on model order estimation in section 7.3.9.1. Although the AR is still effective for small segments, the model order estimation in general fails to return a reasonable order, when performed on small segments.

The user can add maximum, medians and average to the plots within a specific frequency range which will show up in all tasks to make comparisons. It is also possible to remove trials that appear to be outliers and generate an average picture from the tasks in a single view for easy comparison.

AR visualization
The AR spectra of the EEG time series can be shown in different ways:

- Per task over multiple sessions. A number of frequency spectra are drawn, in a single graph, depending on the number of trials used. One or multiple electrode at a time.
- Multiple tasks are shown over several sessions concerning a single electrode.
- An average of multiple tasks for a single electrode can be generated.
- Data lines with averages, medians and maximums can be added to existing graphs.

6.3.12. CSP

Common spatial patterns are an effective way to show where the activity arises between two tasks as described in section 2.10.1.2.

CSP will produce the most discriminative scalp patterns between two selected tasks. The common spatial patterns are produced in order of discriminating power: meaning that the first and last *m* patterns are the most important, the importance of the scalp patterns rapidly decreases.

CSP visualization
The CSP spectra of the EEG time series can be shown per two tasks for multiple electrodes, sessions and trials.

6.3.13. LD

Linear Discrimination offers an efficient way to determine the difference between two tasks. The ROC curve (see section 2.10.1.4.4) will show the discrimination performance between two trials, where the total surface corresponds with 1 and is the absolute perfect separation and 0.5 meaning no difference in the data distributions.

LD data visualization
Per comparison of two tasks (trials) one contour plot with the ROC-curve is shown.

6.3.14. Data management

The final functionality requirement consists of the ability to be able to manage the data structure. It offers the possibility to delete every single aspect of the data structure: datasets, tasks, trials and sessions. On top of this it is possible to merge datasets (useful to merge 'foreign' imported matrices). And rename any dataset.

6.4. *Non functional requirements*

The non functional requirements describe the aspects that are not directly related to the functionality of the system.

- *Reliability*
  Data is stored in a persistent and consistent way in the database and in files on the hard disk. MySQL has the ability to store and retrieve data in a reliable fashion (see chapter 7).
  E-Brain works with the data in such a way that data is always persistent through out a session. In case of a system crash all previously unsaved data will however be lost.
- *Robustness & Fault tolerance*
  E-Brain avoids any invalid user input. Mainly this is covered by representing the user with listboxes and pop-up menus (predefined input). All other input is checked for validity concerning the command at hand. The workbench will catch any error and report back to the user concerning the error and if possible the reasons for the occurrence of the error.
- *Security & Access control*
  No authorization or logon process of any kind is implemented in the E-Brain, anyone with access to the computer on which E-Brain is installed can use the program and all

its facets and the data is uses. No special protection is implemented against malicious attacks.

- *Portability & Adaptability*
  The E-Brain is designed using 'Matlab' from Mathworks and therefore is platform independent, although this requirement is not regarded as critical.
- *Modifiability & Extensibility*
  E-Brain is extensible to any new analysis method or in the direction of classification since the results from the analysis methods are available for further processing. It is also possible to modify any methods that are currently present. Furthermore all the code from the workbench is documented and commented in such a way that it is easily modifiable and extensible.
- *Performance*
  - *Response time*
    The goal is to respond instantaneously to user commands, however due to the nature of EEG data in some cases a significant amount of calculation time is required. The goal therefore is to signal the progress to the user in anyway possible so that the user is aware of the state of the system during calculation.
  - *Memory considerations*
    EEG is in general highly dimensional data which requires large amounts of memory therefore several issues are addressed to minimize the memory requirements.
    - Rounding data in pre-processing, this simple action inflicts a small but acceptable error, however results in a significant memory reduction (about 2.5 time memory reduction).
    - Deleting unnecessary data, once all the necessary data is pre-processed all the previous steps (which are stored in the datastructure) can be removed to speed up further processing. This is a user initiated and directed process.
- *Utility*
  The goal of the E-Brain is to completely facilitate the process of acquiring usable features from the EEG for the classification phase or to assess the quality of an experiment. Nothing outside this stated frame can be expected from the E-Brain.
- *Usability*
  As stated before the user is expected to have a significant knowledge concerning the algorithms, methods and techniques used in the E-Brain to obtain usable results. The E-Brain provides an environment which is user-friendly and intuitive to use for the professional user.

6.5. *Design*

6.5.1. E-Brain use cases

Use cases are used to identify the basic functionality to the user and define the system boundaries of the E-Brain. Use case diagrams represent this functionality from the user's point of view. To show the principle only one of the use cases is displayed here. For more use cases refer to the appendix.

Actors:
User, represents the user who interacts with the E-Brain as the data analyst.
Storage, represent the storage entity either as file system, database or datastructure.

This case (see figure 6.5) identifies all the interaction available for the actors concerning data analysis.

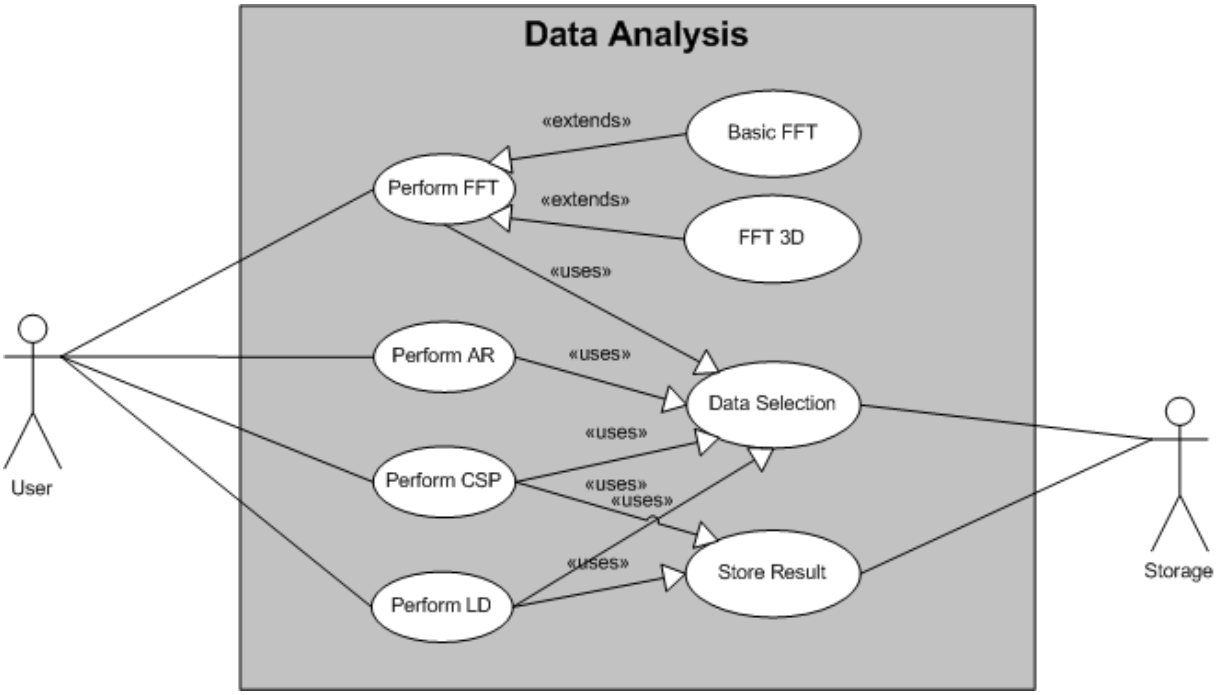


Figure 6.5 Use case diagram data analysis.

Use cases:

- *Perform FFT*
  - Description: the *User* analyzes the data using Fast Fourier Transform. Different data views are present.
  - Actor: *User.*
  - Extensions:
    - *Normal FFT*, the *User* analyzes the selected data using FFT with specified resolution.
    - *FFT 3D*, the *User* can view the FFT process in three dimensions.
  - Uses:
    - Data Selection.
- *Perform AR*
  - Description: the *User* analyzes selected data using the Autoregressive Model. Model properties can be specified, different data views are present.
  - Actor: *User.*
  - Uses:
    - Data Selection.
- *Perform CSP*
  - Description: the *User* analyzes selected data using Common Spatial Patterns between tasks and different parts of the data. Model properties can be specified.
  - Actor: *User.*
  - Uses:
    - Data Selection.
    - Store Result.
- *Perform LD*
  - Description: the *User* analyzes selected data using Linear Discrimination between different tasks and different parts of the data. Model properties can be specified.
  - Actor: *User.*
  - Uses:
    - Data Selection.

        - Store Result.
- *Data Selection*
    - Description: Select specific data for analysis from *Storage*.
    - Actor: *Storage.*
- *Store Result*
    - Description: Saves the results from a specific analysis method to the datastructure in *Storage*.
    - Actor: *Storage.*

6.5.2. Architecture

The use cases combined with the functional requirements lead to the following class diagram (figure 6.6) which represents an overview of architecture of the E-Brain. The class diagram shows the main functionality of the workbench in groups and reveals a global picture of the communication.

Description of the classes represented in figure 6.6:

- *Main* controls and initiates all the main functionality (Data management, Pre-processing and Data Analysis) of the program including the datastructure.
- *Data Handler* handles all data requests from and to the *storage* device.
- *Storage* represents either the file-system or database and physically stores all the data from the program.
- *Pre-processing* represents the complete pre-processing phase and handles all user interaction concerning this element. It contains the down sampling, referencing and bandpass filter, the baseline correction method, the ability to automatically remove eye-artifacts and the segmentation.
- *Inspect EEG* allows the user to inspect the EEG data for artifacts and review the desired response of applied pre-processing steps.
- *ICA* allows the user to remove artifacts using Independent Component Analysis.
- *Data Analysis* represents the complete data analysis phase and handles all user interaction concerning this functionality.
- *LD* allows the user to analyze the data using Linear Discrimination. This class represents all interaction concerned with this analysis method.
- *CSP* allows the user to analyze the data using Common Spatial Patterns. This class represents all interaction concerned with this analysis method.
- *FFT* allows the user to analyze the data using Fast Fourier Transform. This class represents all interaction concerned with this analysis method.
- *AR* allows the user to analyze the data using Autoregressive Models. This class represents all interaction concerned with this analysis method.
- *Visualization* governs all the possibilities offered to visually present any *data* to the user. All the results generated can be saved to file.

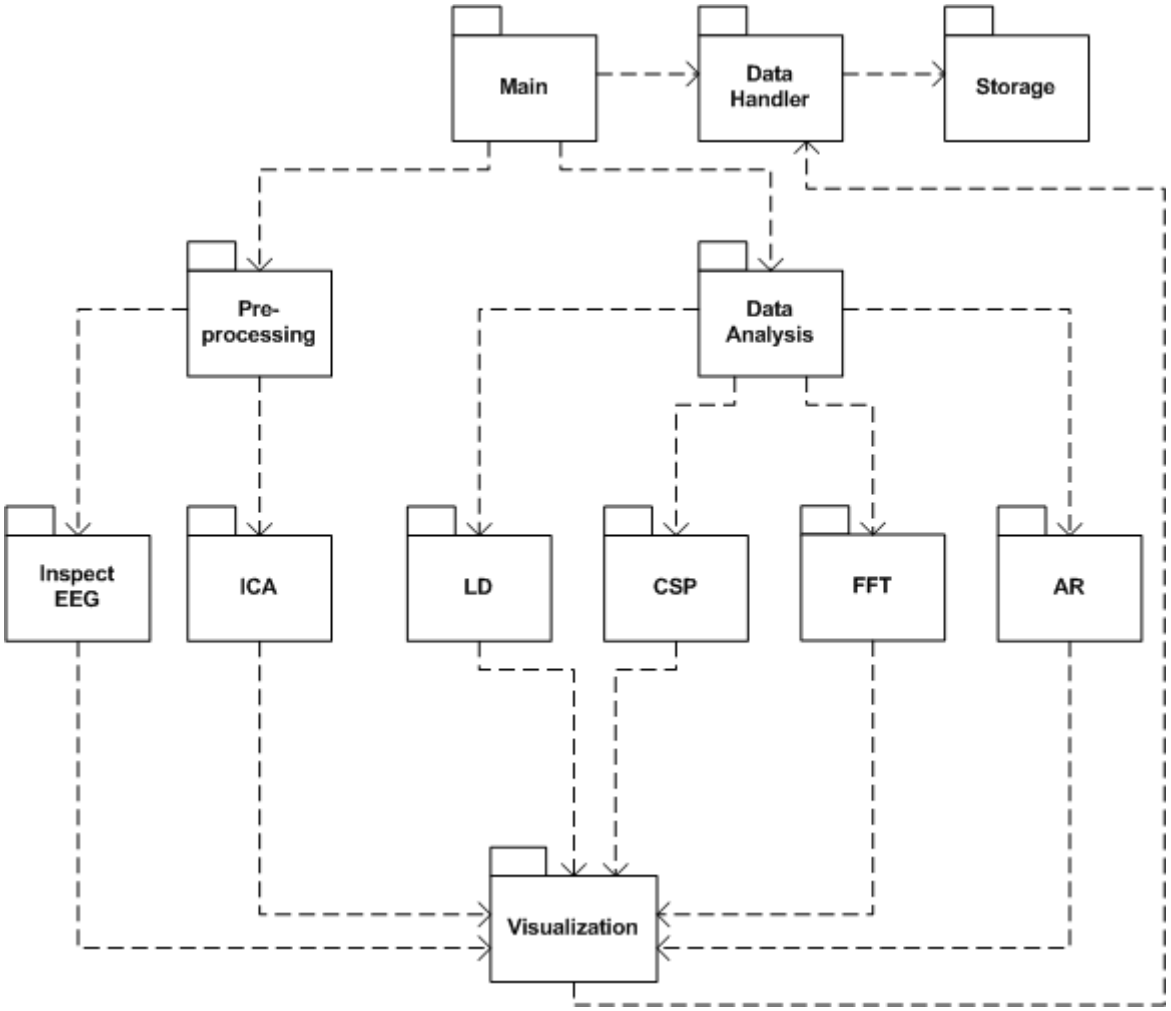


Figure 6.6 Overview class diagram of E-Brain.

Remark:
Matlab is not an object-oriented programming language; therefore speaking of classes in the direct sense of the word does not apply. However it is very useful to group the functionality in to separate classes to obtain a clear image of the structure within the workbench. This does not mean that these functionalities are really a coherent entity within the program.

6.5.3. *Dataflow*

The only step that is pre-defined is the loading of data; after this stage the user is free to proceed in any direction. The workbench does not place any restriction on the dataflow whatsoever: within the workbench there does not exist a single data flow.

However there is a 'likely' dataflow, which is represented in figure 6.7. This figure shows the 'most-likely' dataflow within the E-Brain visible; representing the stages the data goes through within the workbench. Although any of the stages can be skipped or returned to, this is the global manner in which the data is transformed in the pre-processing phase. So at the beginning of the chain the data is loaded and at the end of the chain it is possible to continue with the various analysis methods.

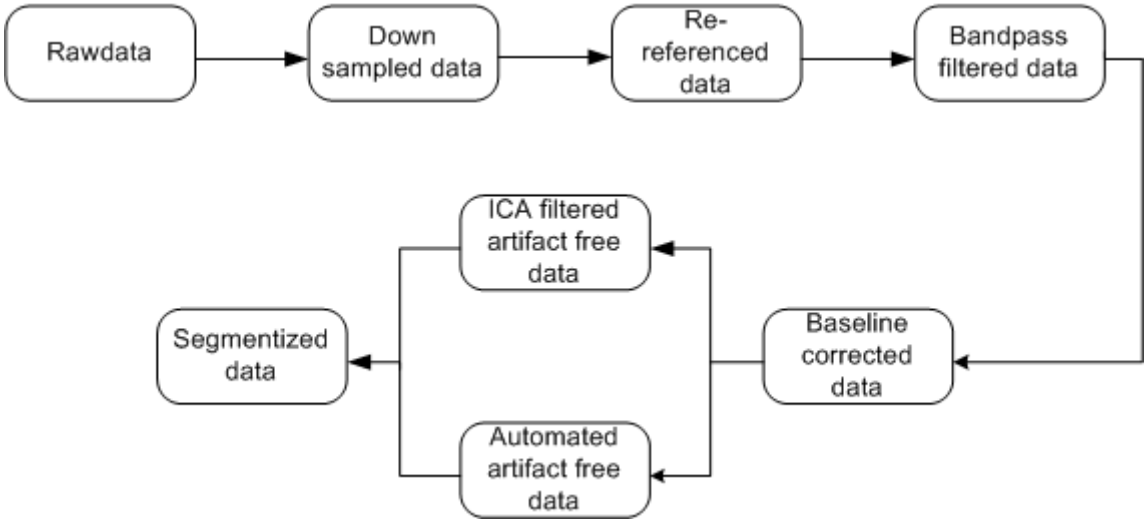


Figure 6.7 Diagram of data flow within the E-Brain pre-processing.

6.6. *Usability*

This section will cover the usability issues and interface design concerning the E-Brain. No user group was available, to interview or inspect. The user is characterized by being a skilled professional concerning both computer interaction and EEG analysis.

In constructing the interface a mix of common sense and basic guidelines have been used in an iterative approach to obtain a usable and intuitive interface.

For the guidelines the "*ten usability heuristics*" by J. Nielsen and the "*Eight Golden rules of interface design*" by B. Schneiderman have been applied in the interface design.
Together these general guidelines amount to:

- Visibility of system status / offer informative feedback.
- Match between system and the real world / design dialogs to yield closure.
- User control and freedom / support internal locus of control.
- Consistency and standards / strive for consistency.
- Error prevention / offer error prevention and simple error handling.
- Recognition rather than recall.
- Flexibility and efficiency of use / enable frequent users to use shortcuts.
- Aesthetic and minimalist design / reduce short-term memory load.
- Help users recognize, diagnose, and recover from errors / permit easy reversal of actions.
- Help and documentation.

The implementation of these guidelines will be covered in section 7.3.13.

6.6.1. Mental model & metaphor

The mental model is the representation of the user concerning the system functions and how it is structured. A mental model is formed by interaction with and explanation of the system. This model is not directly under the designer's control. A conceptual model is used to influence the mental model.
Metaphors make use of existing conceptual models. The metaphor used for the E-Brain is the Windows metaphor. Windows is in itself actually a metaphor, but has become such common knowledge that it can be used as a conceptual model for the interface design.

6.6.2. The interface

The challenge
Since the EEG data is as enormous as are the available options for manipulating this data, it is hard to keep the interface clear and simple. However this goal has always been the starting point; to hide all none-essential options and information and keep the interface intuitive.

The interface is designed to adapt the natural flow of what a user expects: first modality selection, then data, then model specifics.

Like in any normal situation the processes in the E-Brain start in the top left corner and works down to the bottom right. Therefore the information frame is located at the top left, which tells the user exactly what the available options are or what the E-Brain is doing or requests at that moment. The data selection is situated just below and is essentially the same for each model used and always followed by the specific parameters for that model. At the bottom possible result manipulation and the 'exit' can be found.

To keep things clear only one model can be used at a time, this is also in line with the general data flow of the workbench. Any functionality that cannot be used at a certain time is disabled so as not to confuse the user. All the major options are always on top in the drop-down menu like in the Windows metaphor. Furthermore different windows are used to indicate different steps in the analysis process.

Data display
In displaying data a way is searched to be able to compare different tasks easily, the focus is on creating an environment for easy comparison. Therefore spectra are placed vertically with parallel x-axes. And different colours are used to indicate different sessions or tasks. Most of the models offer the possibility to adjust scaling of the plots.

Shortcuts
All the major menu actions can be reached using shortcuts as well as some frequently used actions.

Documentation
Aside from the information display during execution and constant feedback from the program, the E-Brain offers complete documentation concerning the models used and functionality offered. How they function and how they should be applied with the goal to aid the user in operating the E-Brain as good as possible.

Chapter 7
7. E-Brain implementation

This chapter covers the various aspects of the implementation phase of the E-Brain. Aside from the implementation issues, also the setup of the methods, selection of most optimal parameters, screenshots and pseudocode of the various models will be covered here. The goal of this chapter is to provide a detailed view on the E-Brain.

Any third party supplied code or functionality will be mentioned; in all other cases the implementation of the functionality is created for this master thesis except for basic Matlab functionality like for instance matrix manipulation and plot functionality. Testing and evaluation of the E-Brain performance will be covered in the next chapter.

7.1. *Implementation environment*

Matlab from Mathworks (Matlab 7.1 - www.mathworks.com)
Matlab is a high-level language and interactive environment that enables to perform computationally intensive tasks faster than with traditional programming languages. Matlab is the standard in mathematical computing for algorithm development, data visualization, data analysis, and numeric computation; it is stable, very well documented and has a large user community.

The main problem Matlab imposes is that fact that it is based on Java, which is not as fast as for instance a C or C++ based program. Matlab interprets the code on every execution while the latter uses a one time compiled program.

However Matlab has a lot to offer, for instance it is extremely efficient with matrix computation and has an easy to use environment for program and functionality design. One of the main advantages of Matlab is the huge user community which offers support and proven implementations of analysis methods like ICA, CSP and Autoregressive models.

E-Brain
The E-Brain requires only a computer that is running Matlab; no extra toolboxes for this program are required by the workbench. All functionality, aside from the basic functionality offered by Matlab, is offered by the E-Brain.

7.2. *Detailed class diagram*

Figure 7.1 represents the E-Brain in a detailed class diagram, it elaborates on the class diagram from chapter 6, showing the relations between classes and the functionality they represent. The goal is to give an overview on how the E-Brain communicates and what functionality is found where.

Legend for figure 7.1:
The name of the class is shown on top in bold. The next line can contain any variables stored in the class; for simplicity reasons only the very important are shown. The last part of a class represents the main functions which are used within that class. Again for simplicity reasons the parameters that go with the functions are not shown here.

Colours:
- Dark blue represents the main GUI.
- Light blue represents the main functionality GUIs.
- Yellow represents a class offering functionality to any of the main classes.
- Green represents a class which incorporates third party functionality.

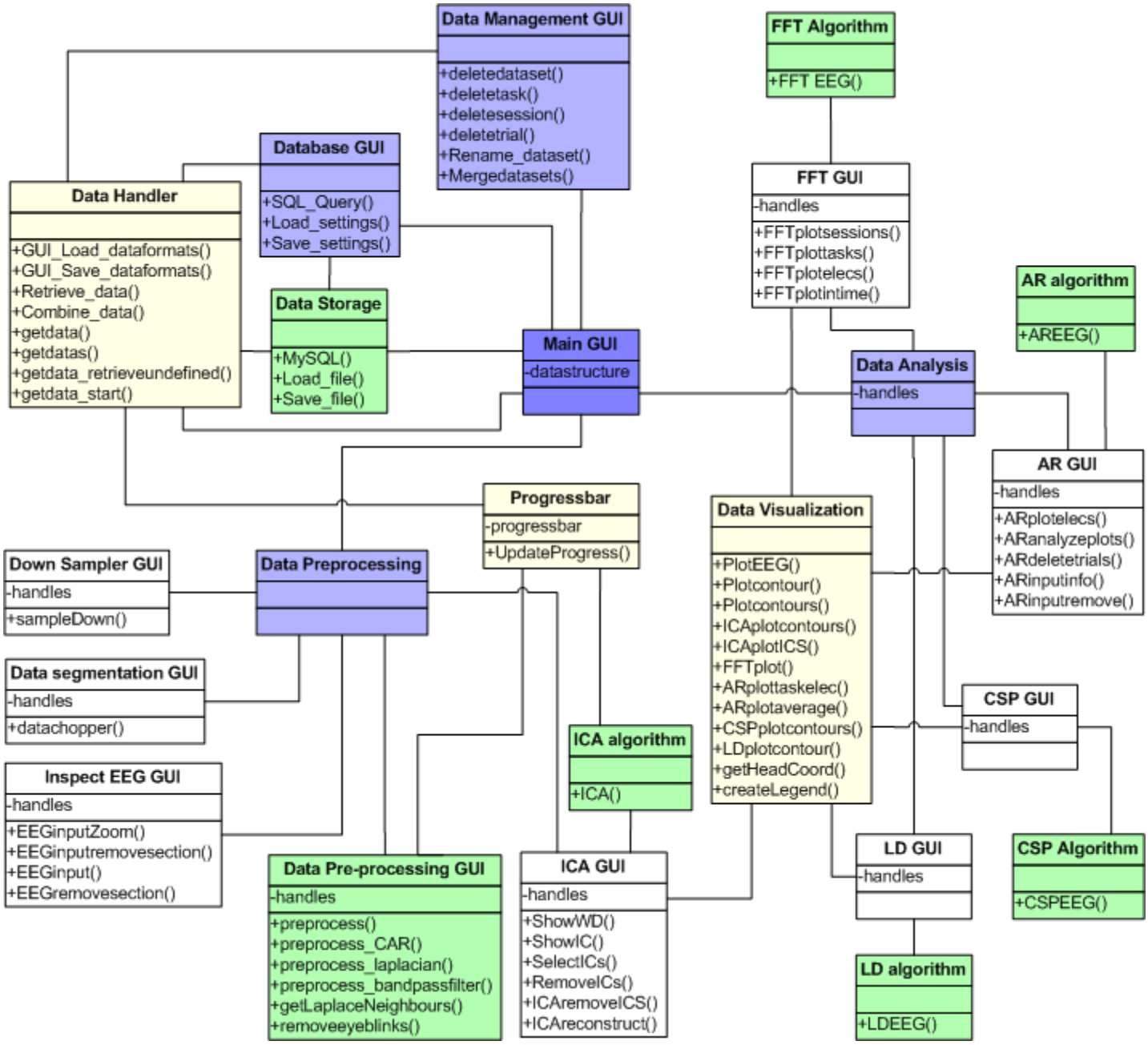


Figure 7.1 Detailed class diagram of the E-Brain.

7.3. *Functionality*

Next the functionality of the E-brain will be covered with explanations of implementation decisions, screenshots and pseudocode. The same order as in chapter 6 is used here.

7.3.1. Datastructure & retrieval

The first option to retrieve data is using a database (see figure 7.2). The database selected for this job is MySQL. Initially the free database Firebird was used for this purpose, however some problems occurred with the communication to this database. Therefore a search for an alternative resulted in the MySQL option, which is free as well as the supporting programs. The connection and queries to the MySQL database are performed using a library provided by MySQL.

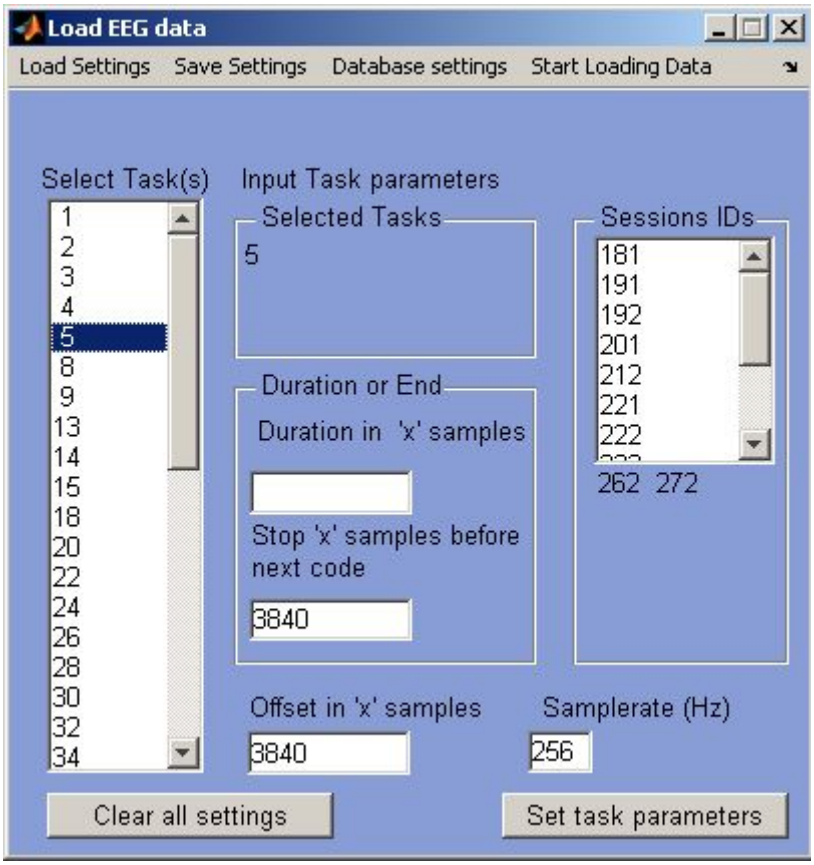


Figure 7.2 Screenshot of the database retrieval component. Left the tasks can be selected and on the right the parameters for each task can be defined. The top-menu offers to select pre-defined- and database settings and start the actual loading process.

The first step is to select the tasks indicated on the left side of figure 7-2; these correspond to the codes inserted during the experiments and are retrieved from the database. The next step is to indicate from where (the offset) until where (duration) the data must be retrieved. This can be done for multiple sessions at once. Or it is also possible to retrieve data from a code (plus offset) until the next code found in the data (minus a certain time period); this option can be used in the case that it is not clear how long the subject performed a given task.

Finally experiment sessions must be selected and possibly the samplerate set.

```
Setup user-defined timepoint query
Retrieve timepoints using database query
for each timepoint:
      Setup data retrieval query based on timepoint and user-defined duration
      Retrieve data using database query
end
Store retrieved data in appropriate format
```

Figure 7-3 Pseudocode of database data retrieval.

Once all parameters are set, they can be saved for easy re-use later on. To communicate correctly with the MySQL database, communication-settings *(host, username, password* and *database*) can be stored in the '*Database settings*'. Finally the loading process can start be clicking '*Start loading data*'.

Aside from database retrieval it is also possible to load an EEG data matrix into the E-Brain. This matrix is a list of electrodes with accompanying samples values. The first row of the matrix can

contain the header names, which will than be used in the program. Alternatively, they can be inserted in the 'Header specification' (see figure 7.4).

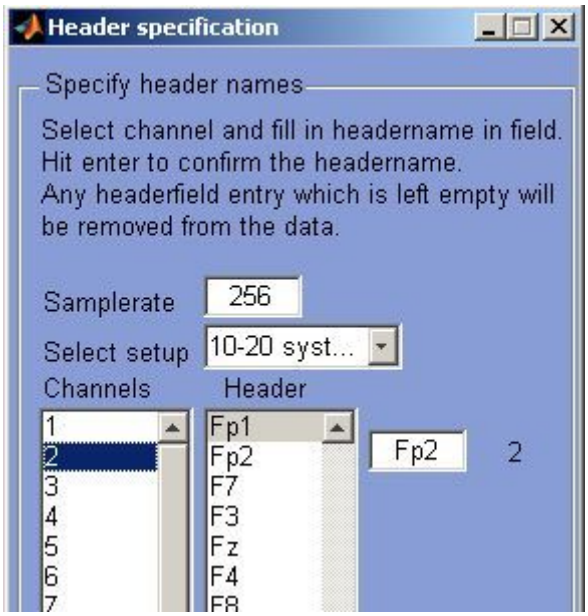


Figure 7.4 Screenshot of the load-matrix component. The header and sample rate must be defined.

Finally previously stored datastructure can off course be loaded back in to the workbench, by simply selecting the file from the hard disk.

The ´struct´ from Matlab is used as a data structure to store the EEG data. The E-Brain will store any additional parameters that are required for correct functioning of the workbench in the main 'struct' of which the datastructures are a part. However if the datastructure is saved to a file, only the actual data is stored. During usage of the workbench all data is stored in the datastructure, it is the user's responsibility to save any data to a file.

7.3.2. Data inspection

In figure 7.5 the data inspection component is shown. In the top left any data can be selected and viewed per trial. The trial will then be shown for all the electrodes on the right of the screen. In principle the entire sample-range is revealed, but the user can set a specific range or manually zoom into a smaller part of the data and navigate using the slider under the graph. The data is automatically scaled for the best fit on the screen, but can be set manually if desired.

Apparent artifacts can be removed manually (*cut)* from the data by selecting a sample range in the plot (see red section in figure 7.5). This action can be undone. Any instructions to the user are shown in the top left frame. All changes to the data can be saved.

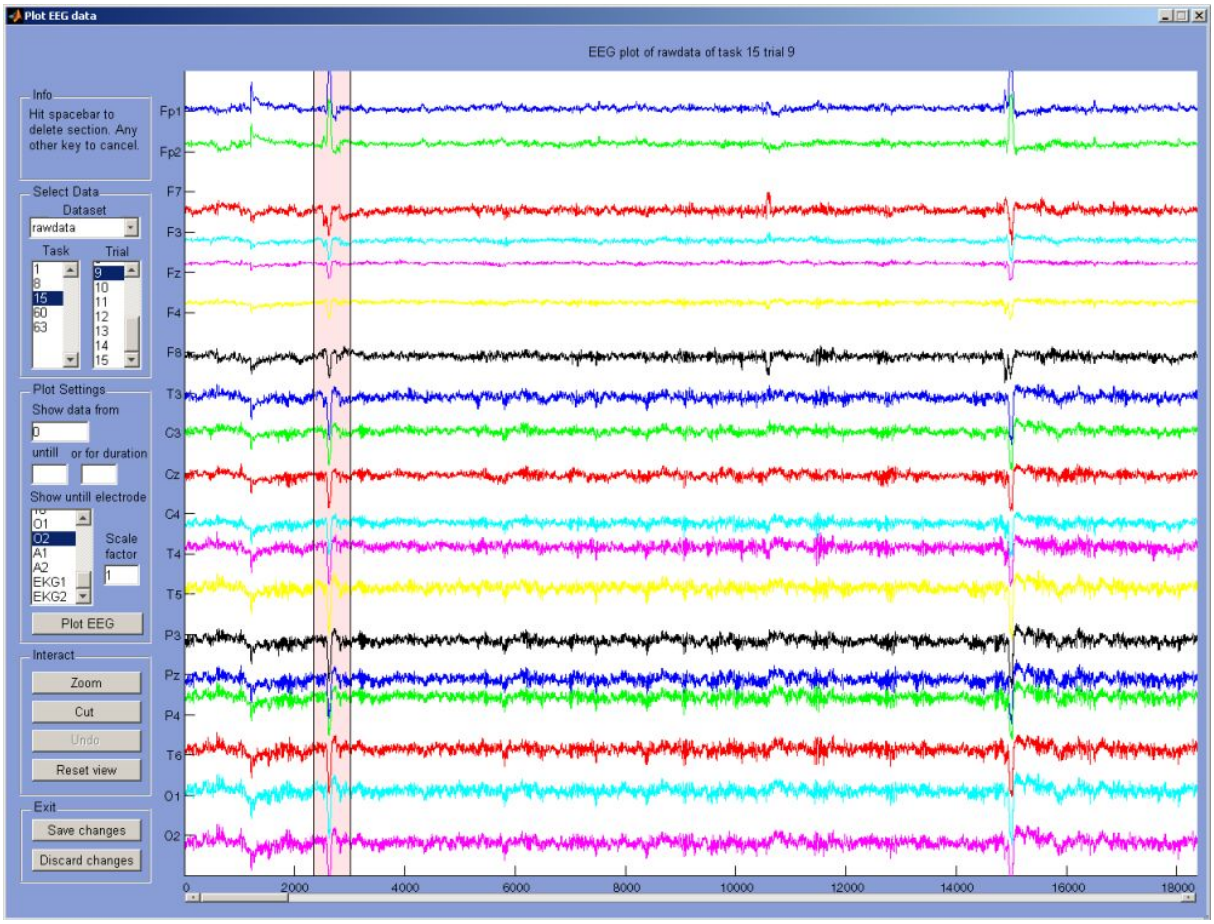


Figure 7.5 Screenshot of the data inspection component. Top left allows for the data selection and information display. The middle left section is for parameter selection. Below that are the data viewing and manipulation. On the right the various electrodes with their samples are shown. The red zone indicates a selected section for deletion.

7.3.3. *Data down sampling*

The data down sampler (see figure 7.6) offers the possibility to lower the samplerate of the data. This is facilitated by stretching a linear space with the size of the new samplerate over the old space.

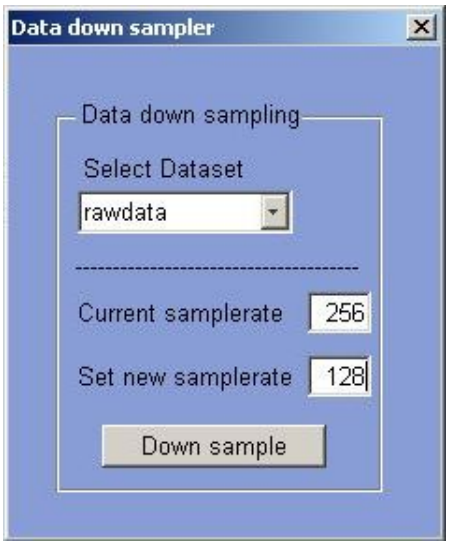


Figure 7.6 Screenshot of data down sampling component. A dataset can

be selected, and the desired samplerate parameter as well.

7.3.4. *Basic pre-processing*

All the basic pre-processing steps can be performed in one sweep (see figure 7.8). Description of the implementation considerations for the basic pre-processing methods follows here:

- Reference filtering, as stated in section 2.6.2.1:
  - Common Average Reference subtracts the average of the complete scalp pattern at sample time *t* from every specific electrode.
  - Small Laplacian Reference subtracts the average of the four or five nearest electrodes at sample time *t* from the specific electrode, using the Euclidian distance (see figure 7.7).
- For Band pass filtering a fourth-order Butterworth digital filter is applied to the data in a specified frequency range using Matlab's *butter* function.
- Baseline correction is performed using Matlab's *detrend* function which removes the best straight line fit from the data.

Automatic removal of eye-blinks (see figure 7.9).
Automatic rejection of eye-artifacts is based on the data recorded in the two frontal electrodes Fp1 and Fp2. The variance of the measured activity is on average very low, however during the event of an eye blink there is a relatively huge peak in the data. The data is searched for those peaks. By comparing the current window's average to the global average it rapidly becomes clear when a peak arises. It is merely a job of fine-tuning the parameters like window size, threshold and rejection size to get an effective rejection for eye blinks.

```
for each electrode
      Calculate Euclidian distance to all other electrodes
      Subtract mean of four nearest neighbours from electrode samples
End
```

Figure 7.7 Pseudocode of small Laplacian re-referencing.

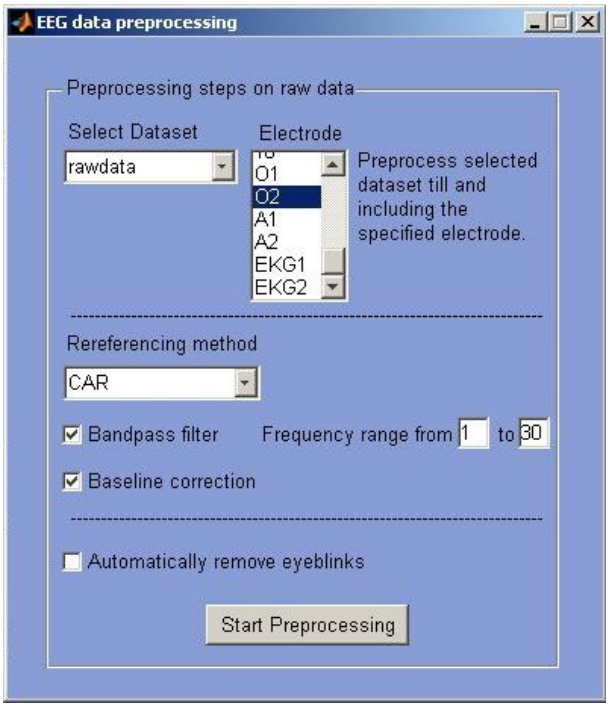


Figure 7.8 Screenshot of the pre-processing component. A dataset can be selected and next the referencing method, bandpass filter and baseline correction. Finally it is possible to allow automatic removal of eyeblinks.

```
while full length of trial
      Calculate mean of current window in frontal electrodes
      if current mean surpasses threshold
            remove window from trial
      end
      update window
end
```

Figure 7.9 Pseudocode of automatic eyeblink removal.

7.3.5. *Advanced pre-processing*

ICA offers the possibility to remove an artifact instead of rejecting the data all together (see figure 7.10). The possible parameter settings are explained in the following sections along with a detailed explanation of the most optimal values for EEG data.

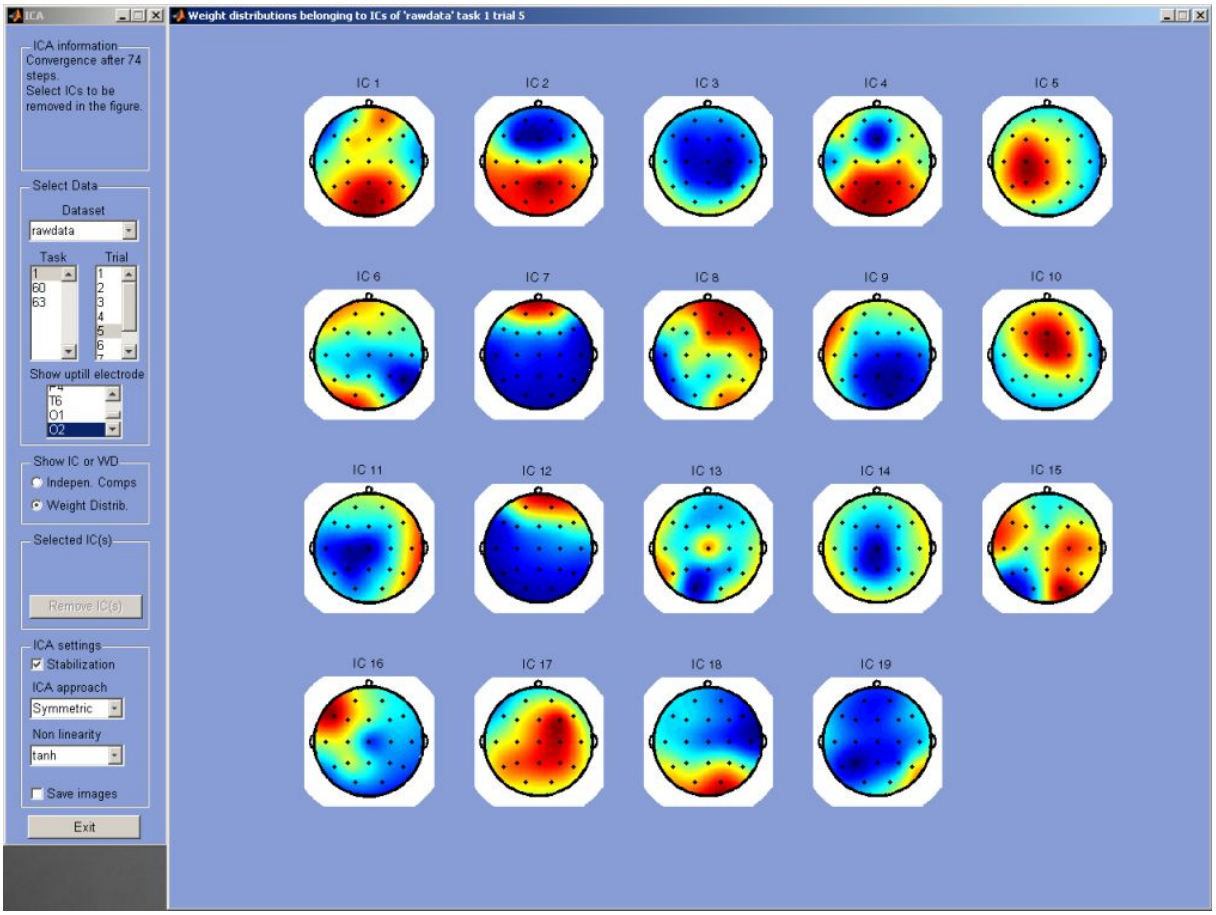


Figure 7.10 Screenshot of the ICA component. Data can be selected in the upper left corner. First the dataset, followed by the task and trial. Down left specific ICA parameters can be set. The right part of the screen shows the intensity distributions of the independent components.

7.3.5.1. *Artifact removal*

The implementation used here is the FastICA implementation, which is a computationally highly efficient estimation of ICA especially written for Matlab provided free of charge by *Helsinki University of technology*. In figure 7.11 a top level representation of the ICA algorithm used in the E-Brain is given. For more information on the algorithm see section 2.6.5.

```
Setup matrix B (random or guessed from  whitening matrix)
for a set number of iterations
      symmetric orthogonalization of matrix B
      define matrix A by product of de-whitening matrix and B
      update B by maximizing nongaussianity based on selected non-linearity
      calculate filter matrix W by product of B' and whitening matrix
end
```

Figure 7.11 Pseudocode of main ICA algorithm (displayed here: the symmetric approach).

To demonstrate the principle of artifact removal by ICA figure 7.12 shows 19 independent components of a 28 second piece of EEG data. Now the IC (number 8) of the third column of the second row seems to represent an eye blinking component.

This evidence is backed up by figure 7.13 which represents the intensity distribution of the independent components of figure 7.12. Red can point to high activity and blue low (or vice versa since magnitude and sign have no meaning in ICA) and the top of the head represents the front. Taking in account both figures it can be concluded that the activity is primarily at the frontal side of the cortex, which indicate ocular activity. Now if this specific component is removed from the array of independent components and the original signal is reconstructed, than the eye blinks are effectively removed and the signal preserved.

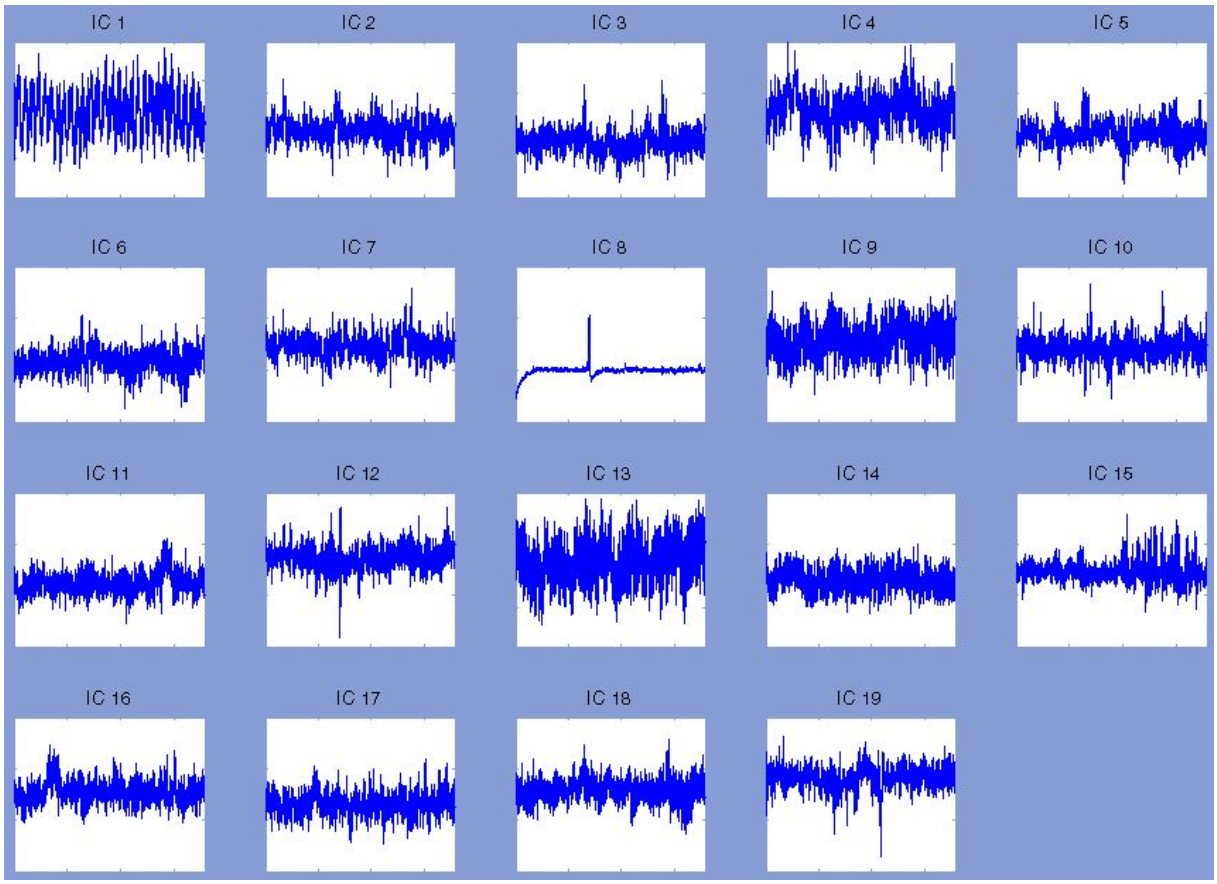


Figure 7.12 Overview of the independent components from the ICA from a single trial. IC 8 represents an eye blink in the data.

The same can be done for the pulse, however because of the large time period used in this example it is not well visible from figure 7.12, is probably the first independent component (first column, first

row). The heartbeat does not have a characteristic distribution like the eye-blink and requires determination via the ICs, not the intensity distributions.
The difficulty in selecting components to be removed is that the user must be absolutely sure that it represents an artifact source. These particular components seem to be clear, but in general it can be a complicated task to define what a component represents. And it goes without saying that removing an independent component which is a relevant source to the EEG activity will render the further analysis next to useless.

One more issue that must be pinpointed here is that the order in which the independent components are returned from the ICA are not fixed, it tells nothing about the importance of an independent component. And will vary between two ICA-runs on exactly the same data. Thus the user always has to manually select the components to be removed.

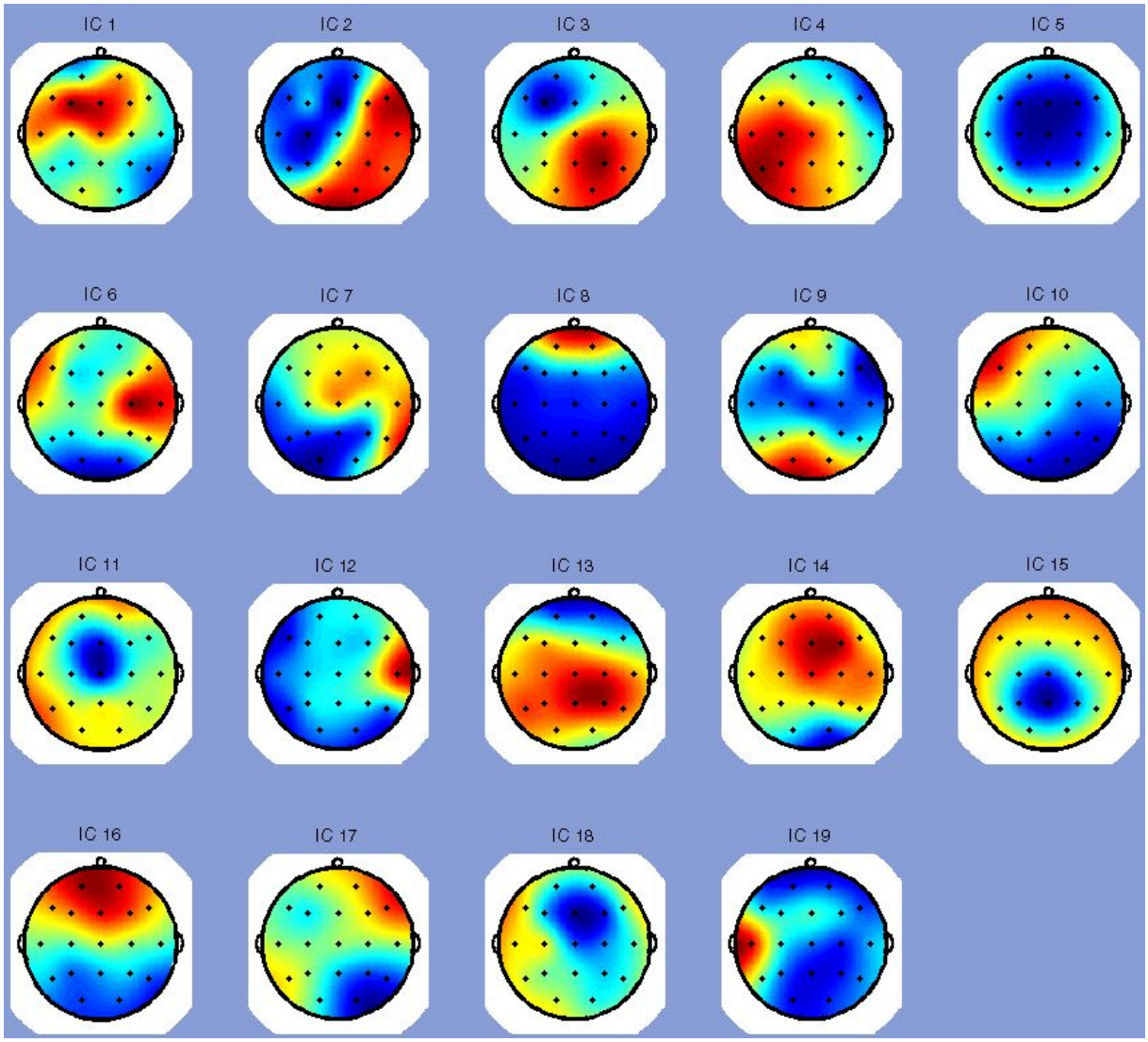


Figure 7.13 Overview of the independent component intensity distribution plots. IC 8 indicates frontal activity.

7.3.5.2. Finding the right ICA

Different parameters have to be set using ICA. To find a satisfying setup for ICA different runs have been made using different parameter settings. The result of ICA can be judged by the quality of the independent components (how much information do they display) and the rate of convergence. The results of these tests are shown in table 7.1 and 7.2.

In table 7.1 the stabilization parameter is on, on average the processes terminated in about 500 steps. After 500 steps the convergence parameter is adjusted by the algorithm and therefore convergence is reached faster. Sometimes convergence is reached before 500 steps.
In table 7.2 no stabilization is used; most of the time this results in no convergence at all. From the tables it can be concluded that either convergence is reached way before the 500 mark or it will not convergence at all on average. Since stabilization is required to have any sensible result at all, it is advised to enable the stabilization parameter.

From the non-linearity functions, both 'tanh' and 'gauss' show good performance. The 'tanh'- function is used to run the ICA algorithm, since this function returns the most distinctive independent components.

Table 7.1 ICA results with stabilization *on*.

Approach is symmetric, stabilization is *on*, number of independent components calculated is 19.				
Nonlinearity	Pow3	Tanh	Gauss	Skew
Average convergence steps (outliers between brackets)	522	531 (89 achieved)	517	518
Description of quality of ICs	Moderate, all very alike. Differences visible	Good, clearly different IC, like blinks, pulse etc	Good, different ICs show the artifacts	Moderate, differences visible, not clear

Table 7.2 ICA results with stabilization *off*.

Approach is symmetric, stabilization is *off*, number of independent components calculated is 19.				
Nonlinearity	Pow3	Tanh	Gauss	Skew
Average convergence steps (outliers between brackets)	No convergence (stop @ 1000 steps)	no convergence (stop @ 1000 steps, 11,76)	no convergence (stop @ 1000 steps, 39)	No convergence (stop @ 1000 steps)
Description of quality of ICs	Moderate: Differences visible. Never convergence reached: uncertainty about correctness	Good: clearly different IC, like blinks, pulse etc	Good: different ICs show the artifacts. But no clear pulse	Moderate. Differences visible. Never convergence reached: uncertainty about correctness

The option 'ICA approach' enables the parallel (*symmetric*) or serial (*deflation*) calculation of the independent components. Although in terms of raw speed no significant differences are perceived, the default is the *deflation* approach as it is supposed to out perform the *symmetric* approach [65].

ICA does not require segmented data, on the contrary, as much data should be available for the algorithm. Since that way the most important independent components can be found for the larger part of the data. Moreover it requires less work to perform the ICA on a single segment than one 10 smaller segments.

7.3.6. Data segmentation

Data segmentation (see figure 7.14) consists of two parts, the segmentation size and the degree of overlap between two segments. The segmentation size is constant; the overlap value will be taken as

an indication. An evaluation will determine which overlap near the desired value allows the segmentation to most optimally fit the total data length.

This is done in order to avoid having a very small (in comparison with the other pieces) last part or having to discard this last piece of data since it's too small.

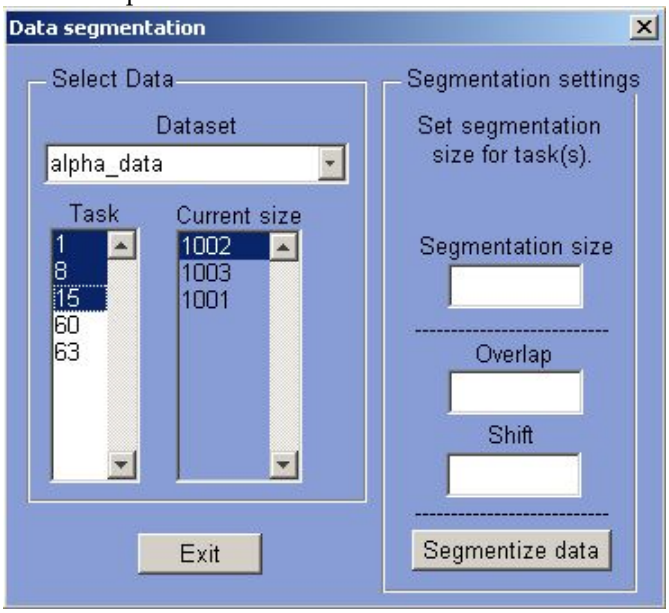


Figure 7.14 Screenshot of the data segmentation component; left concerns the data selection for segmentation. On the right the segmentation size together with the overlap can be inputted.

```
for each trial of the task
      calculate the precise value of the shift based on the trial length
      for length of the trial
            segmentize data and insert into data structure
      end
end
```

Figure 7.15 Pseudocode of data segmentation.

7.3.7. Data analysis

After pre-processing the data is ready for analysis. From this point there is not a single best way to proceed, but the result of the analysis will depend heavily on the user's expertise. The user must decide what to look for and how to look for it and how to interpret the forthcoming results.

All the methods provide the user with the same basic data structure to start from. This structure is a persistent handle in the analysis. However the most important are the tasks and than the trials, therefore the tasks have the priority. The analysis methods provide overviews regarding a task over different sessions and trials and to compare two or more tasks in a single overview.

Another major issue is not only the task but also the location. At the lowest level the data is stored per electrode. To assist the user in searching for valuable characteristics all analysis methods provide the ability to look at single-electrode or multi-electrode overviews to gain better understanding of the task at hand.

7.3.8. FFT

FFT offers the possibility to analysis the frequency spectrum of the EEG data (figure 7.17).

```
Calculate absolute FFT for the data with specific N
```

```
Discard half of the frequencies
Calculate power spectrum density
```

Figure 7.16 Pseudocode of FFT spectrum calculation.

The Fast Fourier Transform algorithm from Matlab (*fft*) is performed on each electrode for the entire data segment made available.

7.3.8.1. Frequency range

Using FFT requires some knowledge of how it is calculated. First of all the highest frequency FFT can reveal from a time series is the Nyquist frequency which is defined as halve the sampling rate. Therefore the sampling rate should be set high enough to provide the desired level of accuracy. What is the desired frequency range? From the literature it shows (see section 2.5.2) that basically all the brain frequencies are in the range from 0 to 40 Hz and moreover the interesting spectrum is from 0 to 25-30Hz, for this range even a sampling rate of 128Hz is still sufficient. The electricity net always adds a sharp frequency peak at about 50Hz. For the analysis here the range from 0 to 25Hz will be sufficient.

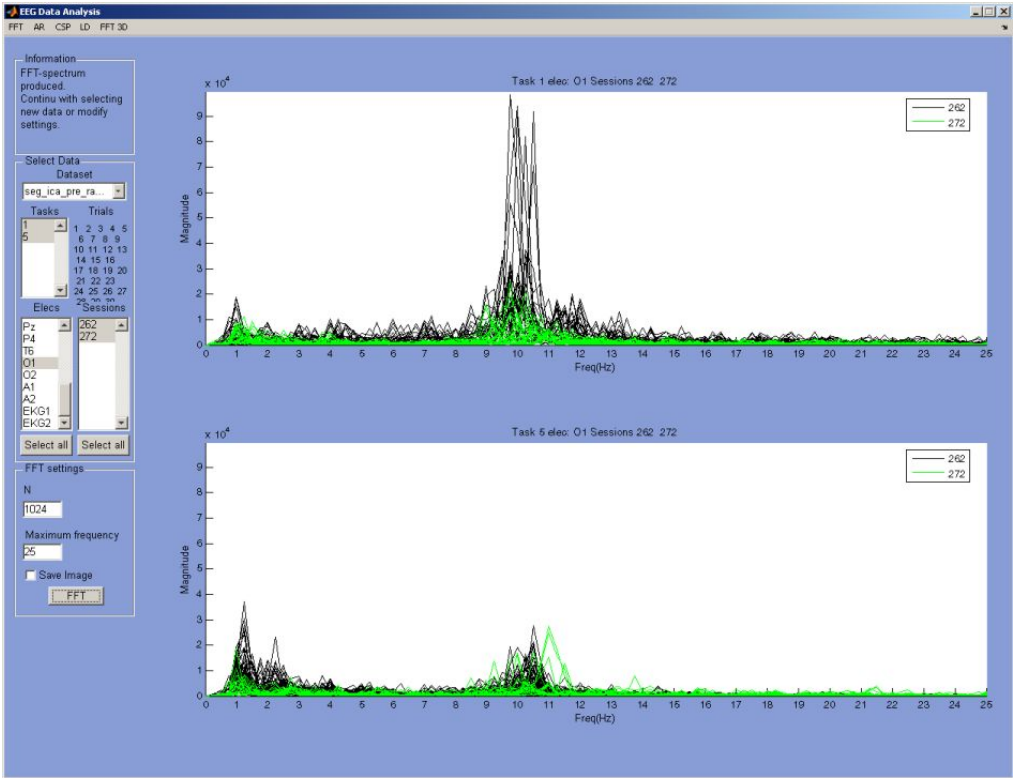


Figure 7.17 Screenshot of the FFT analysis component. Left top: data selection and information display, below parameters can be set concerning the display of the spectrum. To the right the spectra of two different tasks for a single electrode are shown.

The *N* value can be seen as the resolution of the FFT spectrum at the *end-frequency* indicates up to what frequency the spectrum should be shown.

7.3.8.2. Data length

Choosing a specific data length using FFT is not a trivial task. Setting the length to short and the lower frequencies will be represented incorrectly. Settings the length to long and the entire spectrum will flatten and the presented picture will not be very useful either.

Selecting the segmentation size requires the experience and knowledge of the user. Theory and experience show that any segment should at least be about two seconds long (more than 500 samples at 256Hz) and not longer than five seconds (about 1200 samples at 256Hz). In order to obtain a good decomposition of the frequency it is best to try different settings.
Once a data length has been chosen, this length should remain the same for the entire analysis. Or else it will be very hard to compare results.

7.3.8.3. FFT 3D

To achieve a sense of the characteristics of the EEG data, a 3D version of the FFT is also implemented. It allows viewing any data segment FFT spectrum in three dimensions (see figure 7.18). The same principle holds as for the normal FFT. The *window* over which the FFT is calculated as well as the *timejump* or time resolution can be selected. The entire graph can be rotated in any direction afterwards. It reveals the variable nature of EEG.

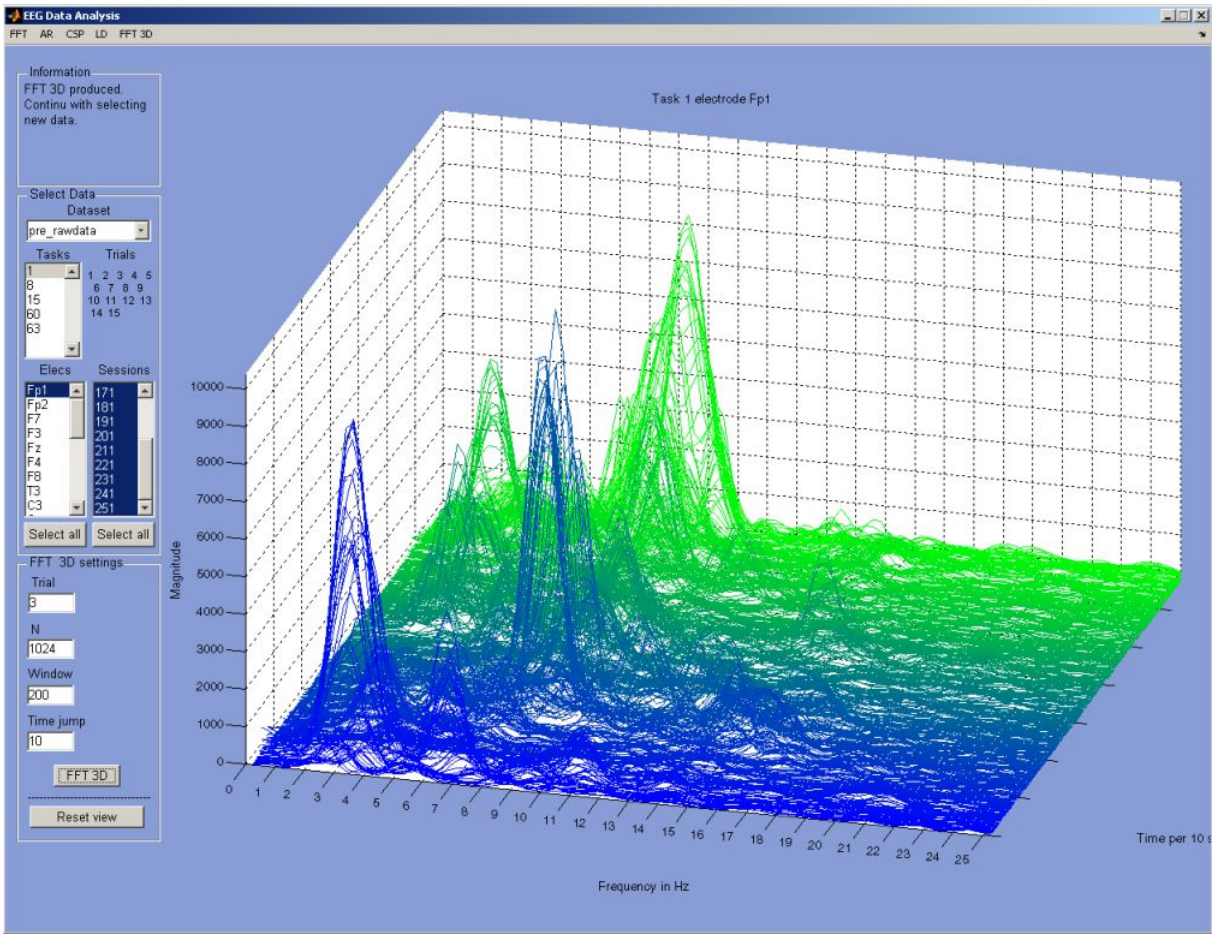


Figure 7.18 3D plot of a FFT-spectrum in time, colours indicate time. On the left side the data can be selected and the parameters inserted.

7.3.9. AR

Autoregressive models are very often used in BCI to extract characteristic features from the EEG data. Background theory on AR models can be found in section 2.10.1.3. The *sig2ar* function from the ARMA-toolbox is used to calculate the autoregressive coefficients. The ARMA-toolbox is provided by *P. M. T. Broersen* from the TU Delft centre for *Systems and Control*.

```
for each data segment (trial-electrode)
      Calculate optimal order based on input parameters for data segment
      Calculate reflections coefficients based on data segment
      Calculate AR model parameters based on reflection coefficient and …
      the optimal order
end
convert AR model parameters to Power Spectral Density spectrum
```

Figure 7.19 Pseudocode of AR spectrum calculation.

7.3.9.1. Model order estimation

As previously mentioned in section 2.10.1.3.3 the AR model order is very important for the performance level of AR. To evaluate the performance of the AR two selection criteria are selected: Schwarz's Bayesian Criterion (SBC) and Akaike's Final Prediction Error (FPE).

The length of the data segments has a significant effect on the error returned for the model order as can be seen in figure 7.20 (segment size is 500 samples) and 7.21 (segment size is 6000 samples). In figure 7.21 the SBC has a minimum at around an order of 30. In figure 7.20 the SBC is almost continuously rising and the FPE returns a minimal error around an order of 40.

In both graphs the SBC is the upper line and the FPE is to lower line.

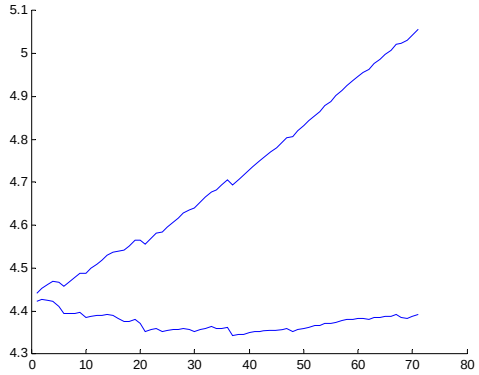


Figure 7.20 Segment size 500 samples, error is vertical and order is horizontal.

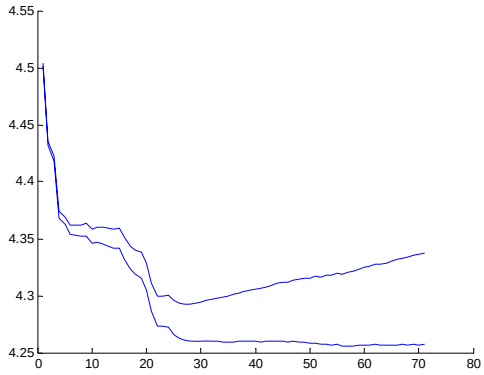


Figure 7.21 Segment size 6000 samples, error is vertical and order is horizontal.

To find the correct order for the AR model, the correct segmentation size must be found. It turns out that the picture shown in figure 7.20 is quite characteristic for the lower segmentation sizes (see figure 7.22). In figure 7.22 the resemblance to figure 7.20 is striking with an ever rising SBC and an almost horizontal FPE. Figure 7.23 returns about the same values for p as figure 7.21, only slightly higher.

In general it can be said that the FPE appears to be more stable in selecting the model order for data with similar properties. Data of segment size 2500 samples (this is about 10 seconds at sample rate of 256Hz) is too large. In the theory however it is stated that these order estimators (SBC and FPE) normally almost never overestimate the order, under estimation is however common if the data-length is not long enough. This is clearly demonstrated by the short explanation above. Theory also states that the performance should be okay (over 80% of correct true order) when the data size is at least 1600 samples.

What conclusions can be drawn? The SBC has some problems reaching a clear minimum even at segment sizes of 2500 samples, while at that size FPE show about the right order every time.

To see anything sensible lets look at a large sample size: 6000 samples. For each criterion (SBC and FPE) we see that it indicates the right order to be around 25-27. This means that the order for much smaller segments of data should be around the same, since the characteristics of the EEG data do not fundamentally change.

In both graphs the SBC is the upper line and the FPE is to lower line.

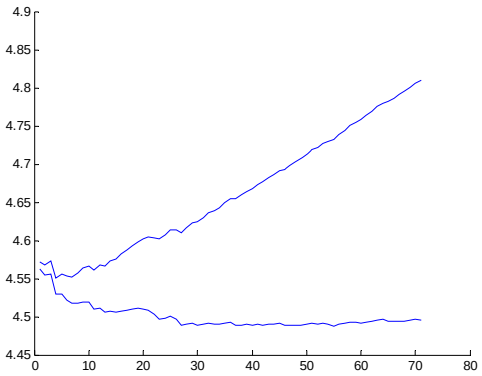


Figure 7.22 Segment size 1200 samples, error is vertical and order is horizontal.

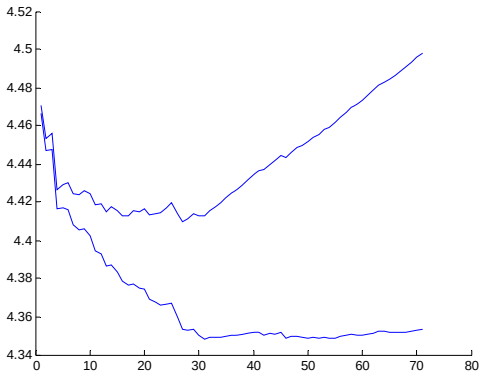


Figure 7.23 Segment size 2500 samples, error is vertical and order is horizontal.

7.3.9.2. Effects on the AR spectrum

Problem with small data segments

Order prediction of small data segments can lead to a very low order (around 10) as the outcome, but as shown this is probably not correct. The problem is that it is very well possible that there are cases in the experiment where the maximal segment is only 5 seconds -> 1280 samples.

The power spectrum for different orders for the same data segment (1280 samples)

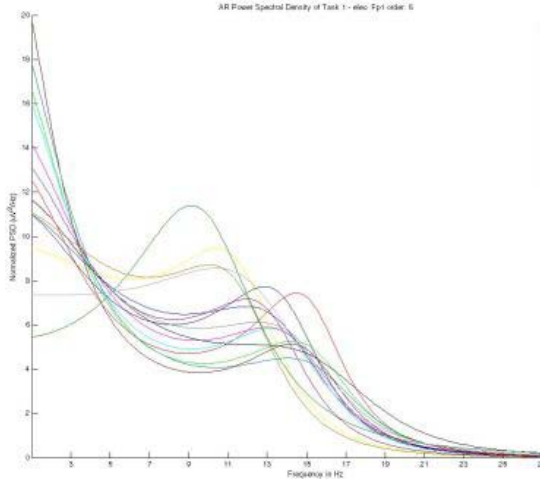


Figure 7.24 AR of order 5.

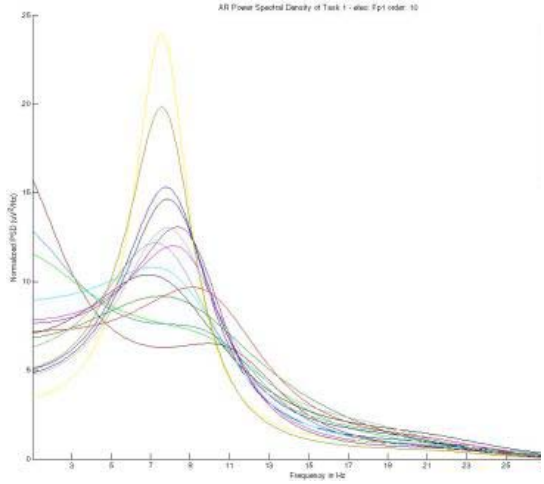


Figure 7.25 AR of order 10.

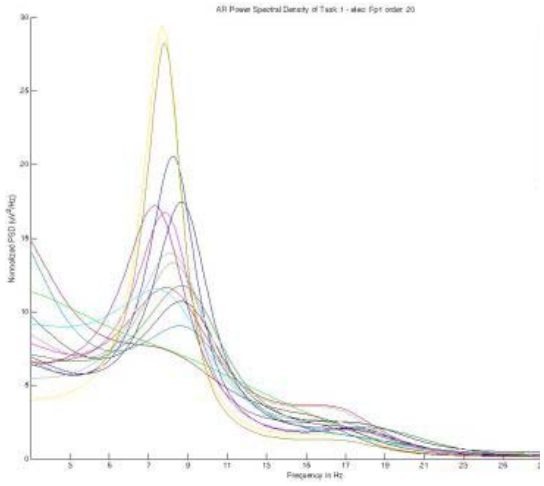


Figure 7.26 AR of order 20.

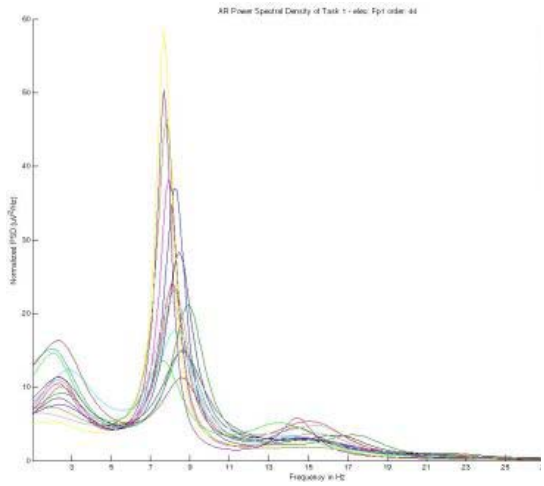


Figure 7.27 AR of order 44.

To counter this phenomena the search for the optimal order should be started at a minimum order. For instance start the search at order 20 and stop at 60, since any higher appears to bring little gain, but requires lots of computational power. This theory is backed up by the following figures 7.24, 7.25, 7.26 and 7.27. The optimal order determined by the algorithm it self is shown in figure 7.27.

Overall it can be said that the assumption of the order being in the range of 20+ will give a satisfying result for EEG data. Increasing it will only require more computational resources and acquire little extra information. Decreasing it will lose valuable information on the signal as is clearly demonstrated in figure 7.24.

7.3.9.3. AR approach

To estimate the AR coefficient several approaches exist, all with their different properties. The performance of these different approaches is very similar, however in certain cases some perform slightly better than others:

- Covariance method (CM); or forward Least Squares approach.
- Modified Covariance method (MCM); or forward-backward least squares approach.
- Burg's method (BM).
- Yule-Walker (YW); or autocorrelation method.

Positive properties of approaches:

- CM: Better resolution for short data segments than Yule-Walker.
- MCM: High resolution for short data segments, no spectral splitting.
- BM: High resolution for short data segments, stable model.
- YW: Stable model.

Negative properties of approaches:

- CM: may produce unstable model.
- MCM: may produce unstable model, peak locations dependent on initial phase.
- BM: peak locations dependent on initial phase, spectral line splitting possible when data is large.
- YW: Performs poorly for shorter data segments.

A stable method means that the poles of the autoregressive transfer function are within the unit circle. What is important for the current cause is that the approach can handle short data segments and is stable. Therefore the Burg's method appears to be a good default candidate.

7.3.9.4. AR analysis

The main aim of data analysis is to find significant differences between tasks and electrode and combinations (see figure 7.28). Therefore it is required to get a sense of when two signals are different, when is the difference of peak levels and average in magnitude at a certain frequency significant enough to state a significant difference?

One issue is that not all trials in the graph contribute to the real characteristics of the data; usually some of the trials simply have minimal activity. On the other hand outliers exist which are significantly higher than all the others. Should these trials be discarded or not? It cannot be said beforehand which is the more interesting data: that which has explicit characteristics, or a complete flat spectrum. Both can be equally valuable. The AR component allows deletion of data lines.

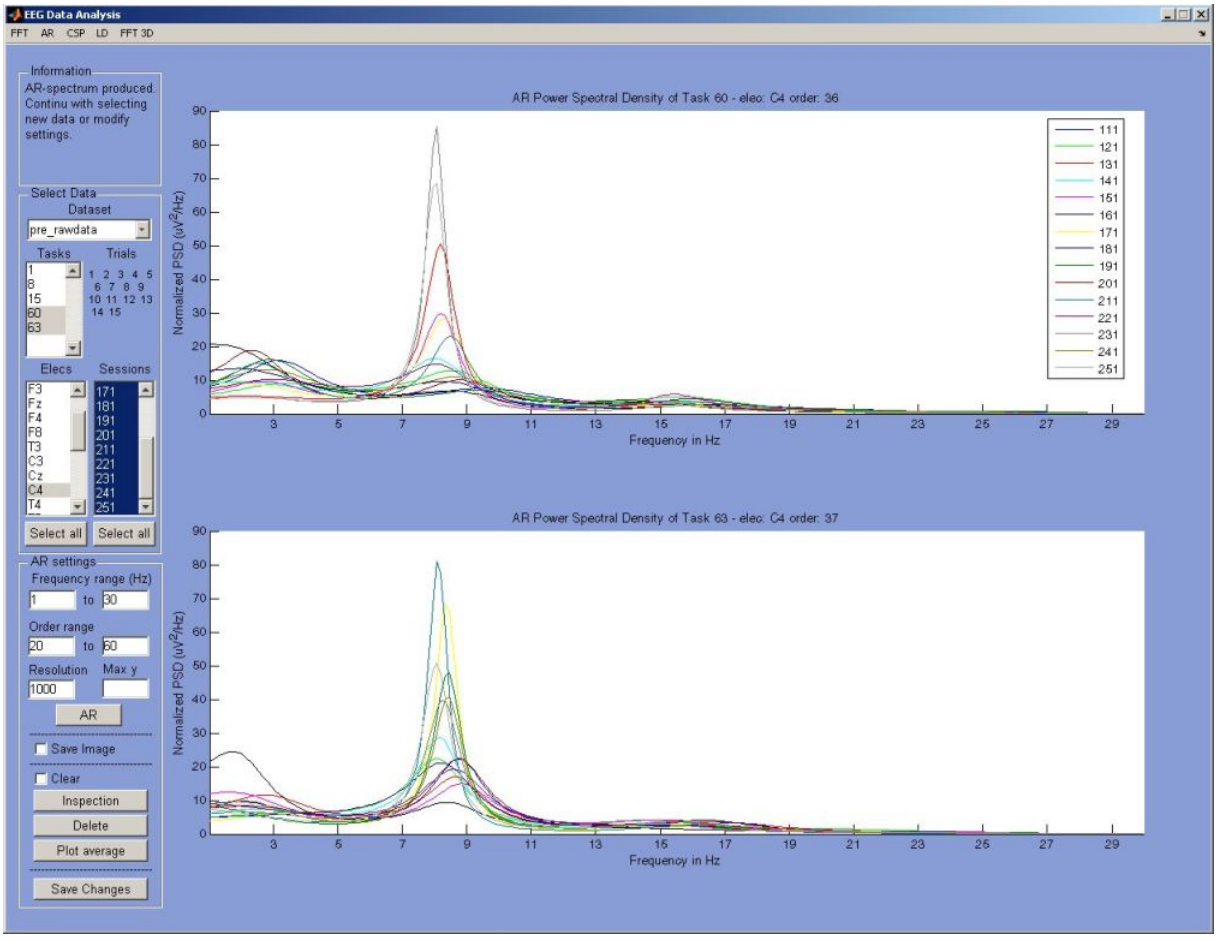


Figure 7.28 Screenshot of the AR component: Left upper side the dataset selection, beneath it the possibility to select AR parameters. Bottom left allows for data inspection. Right side shows the spectra of the AR coefficients.

The directions to search for are magnitude differences and shifts in activity at different frequencies in both tasks and electrodes. The goal is to end up with a picture of an electrode which is the most viable to use in classification.

Inspection

The plotted AR spectra can be inspected by plotting average and maxima into the graphs, simply by selecting the points in the graph.

Deletion
If it is clear that a certain trial is an outlier it can be removed by selecting a range in the plot and indicating whether or not every trial under or above it should be discarded.

7.3.9.5. AR data averaging

Interpreting the results of an AR-average is not as simple as it looks. The graph in figure 7.29 represents the average of the two graphs from figure 7.28. What can be seen is that they characteristics of the two lines are very similar (an obvious problem of averaging), only the peak of task 63 (blue line) is significantly higher at 8Hz.

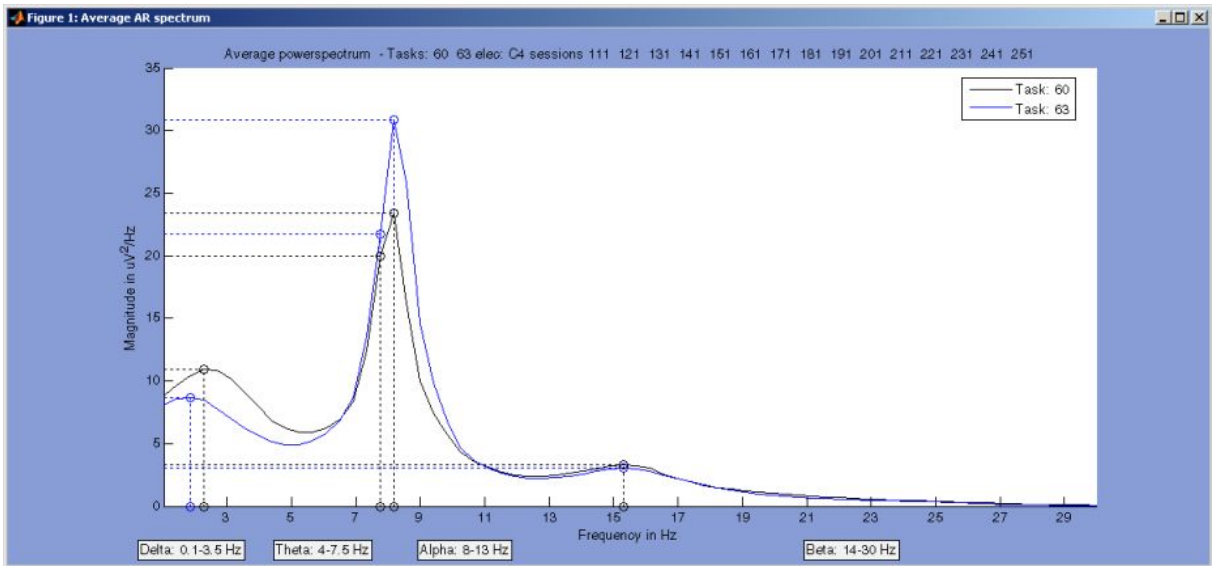


Figure 7.29 The AR average of the different tasks from figure 7.28.

This may seem like a good result, 7 dB is significantly lower. In practice it is not said that every trial (see figure 7.28) will resemble this average. Therefore the sensible use for distinction in practice should be answered.

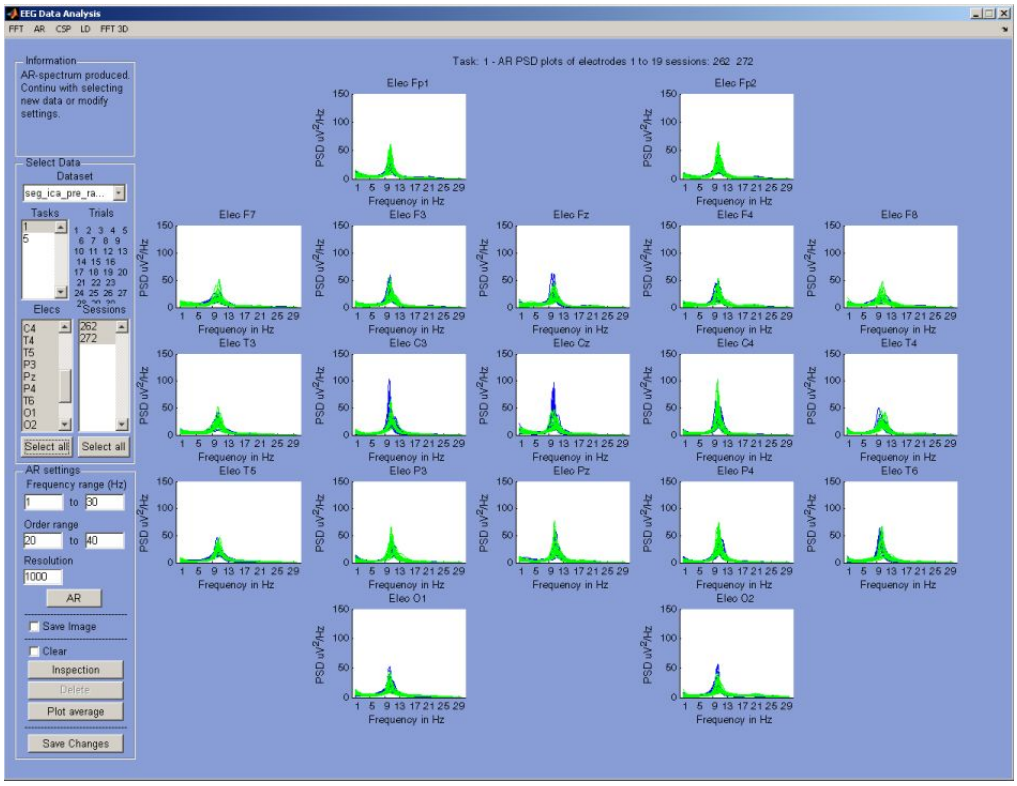


Figure 7.30 Screenshot of the AR component; on the right all the electrodes are drawn for a single task, on the left the parameter specification.

7.3.9.6. Conclusion

Using the AR for data analysis requires even more knowledge from the user concerning what he or she is actually doing than in comparison with FFT. Every step of the way must be justified by sound arguments, since more often than not these very arguments can be used just as easily for the exact opposite case.
The AR has the advantage over FFT that it can be more accurate for very short segments of data (once the order is properly determined) and produce more stable results. However AR is computationally much more demanding in comparison, especially when using higher orders.

7.3.10. CSP

Common spatial patterns are described in section 2.10.1.2. Next the features of the CSP will be discussed.

7.3.10.1. CSP analysis

CSP is used to obtain the most discriminative scalp patterns between two selected tasks. CSP is a stable method; meaning that it will always produce the same scalp patterns for the same data. In practice only about the first and last two patterns produce interesting information (see figure 7.32).

The CSP algorithm is an adapted version of the implementation from H. Ramoser, J. Muller and G. Pfurtscheller (see figure 7.31).

```
for each task
      Remove mean from task
```

```
        Calculate mean of normalized covariance matrix C_task
end
combine mean normalized covariance matrices to matrix C
sort eigenvalues of C in descending order into U
whiten the matrix U into matrix P
transform the C_task using P into S_task
perform simultaneous diagonalization of S into matrix B
calculate the projection matrix W, using B' and P
the common spatial patterns are the inverse of W
```

Figure 7.31 Pseudocode of CSP algorithm.

7.3.10.2. Interpreting CSP results

CSP finds spatial filters that maximizes the variance for one class and minimizes it for the other class [5]. The patterns project the presumed underlying sources to the scalp and are used to verify neurophysiological plausibility.

The goal is to find evidence that the activity is significantly different between two tasks. For instance left and right motor movement could show decreased activity in the right and left hemisphere respectively.

Two basic issues must be kept in mind when evaluating the CSP results. The first is that the patterns returned by the model for a task are always with respect to the other task, meaning that it is not a universal representation of the perceived activity. The second issue is that the absence (blue) of activity in the patterns also tells something about the other task. For instance finding a lack of central activity in the pattern of task *a* versus multiple other tasks, does not automatically mean an increase of activity in the patterns belonging to that task. More on interpreting the CSP results in chapter 8.1.

7.3.10.3. CSP settings

There are not many settings for the CSP algorithm, the most important aspect is the data selection and consequent interpretation of what the results mean. It is possible to set the window length and offset although this is rarely used.

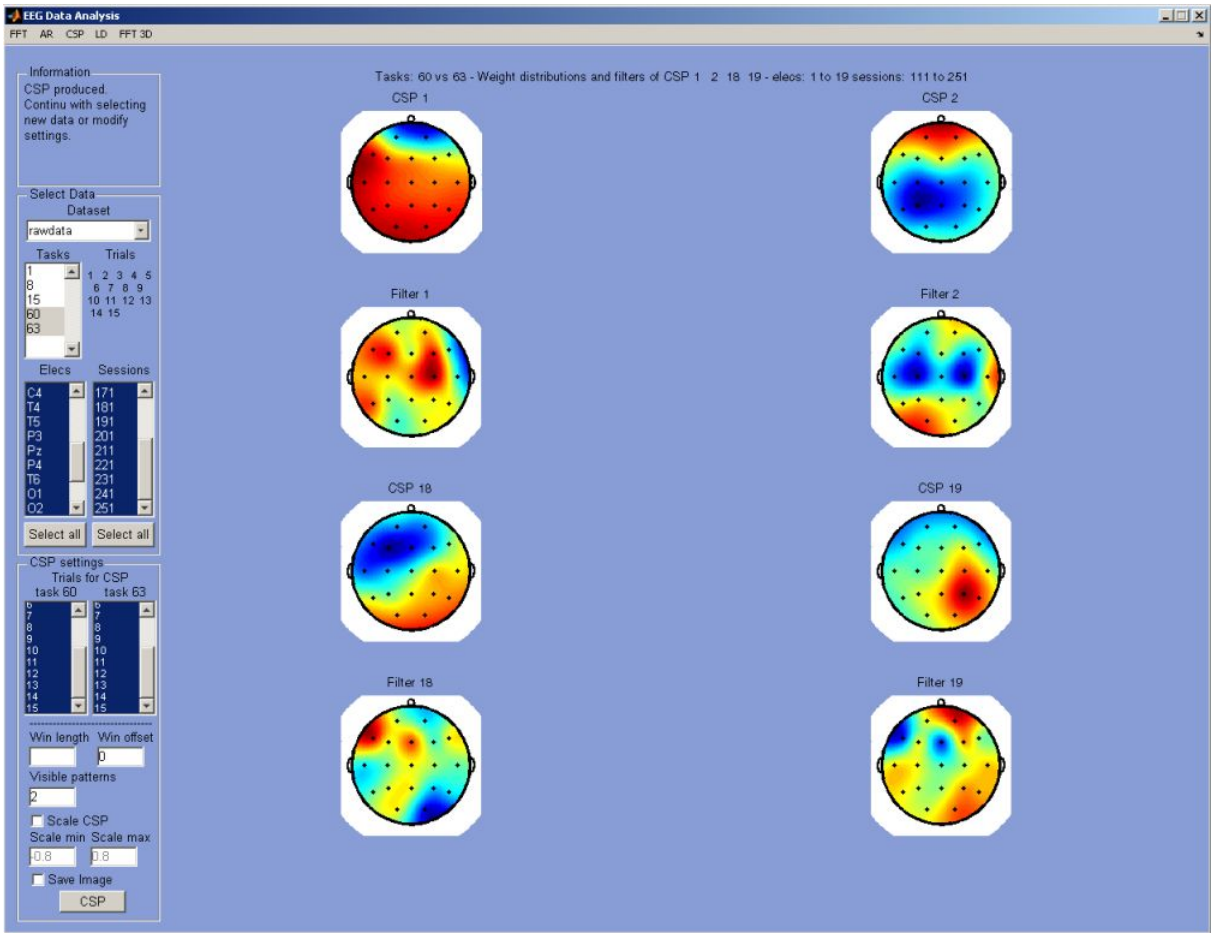


Figure 7.32 Screenshot of the CSP component. On the top left data selection, below that trial selection specific to CSP, continuing downwards with algorithm and display options. On the right the 2 most discriminative CSP and filter are displayed per task.

The visualization options contain of setting the amount of displayed CSP and filters. Theory and practical research point out that a maximum of two CSP for each task will contain the bulk of the information. Scaling can be applied to the plots to allow for better comparison. However it should be pointed out that not to much importance should be appointed to the absolute value of the weights in the CSP and filters.

7.3.11. LD

The basic principle concerning linear discrimination is described in section 2.10.1.4. The LD algorithm is provided by *logist* from A. Gerson and L. Parra as well as the *rocarea* function for calculation of the ROC-curve (see pseudocode 7.33).

The LD performed in the E-Brain uses linear integration as this allows for computation of the spatial distribution [42]. Linear integration is more robust against outliers than Fisher LD [44]. The spatial weighting matrix (returned as the distribution), calculated using logistic regression, represents a hyperplane which maximally separates two classes [34].

7.3.11.1. LD analysis

Linear discrimination produces a single most discriminative pattern between two tasks. LD is stable in that it produces the same results for the same data. LD requires small segments of data, in theory sampling windows as small as 100 samples are used [42], this does mean that it must be certain that

the data contains the specified task. This requirement is somewhat harder to guarantee with 'voluntarily' produced brain activity.

```
Obtain datasets from two tasks and label them
Initialize v to zero
while until a certain convergence level of v or max iterations
      call p the result of a Bernoulli function on the input X and v
      Calculate gradient: X' * (labels – p) – lambda * v
      Calculate increment: X' * p element-wise * (1-p) ...
            element-wise * X plus lambda * I
      Determine new v by v plus the inverse of the increment * the gradient
end
calculate y: X * v
weight distribution A is determined by transpose of the ...
      inverse of y times X
calculate ROC-curve
```

Figure 7.33 Pseudocode of LD algorithm. Lambda is user-defined.

One of the problems with LD is that it tries to find spatial differences, but if presented with a large amount of data, EEG data tends to become very similar on average, resulting in a ROC-surface area between 0.6 and 0.5. The impact of segment size is shown in figures 7.34, 7.35 and 7.36.

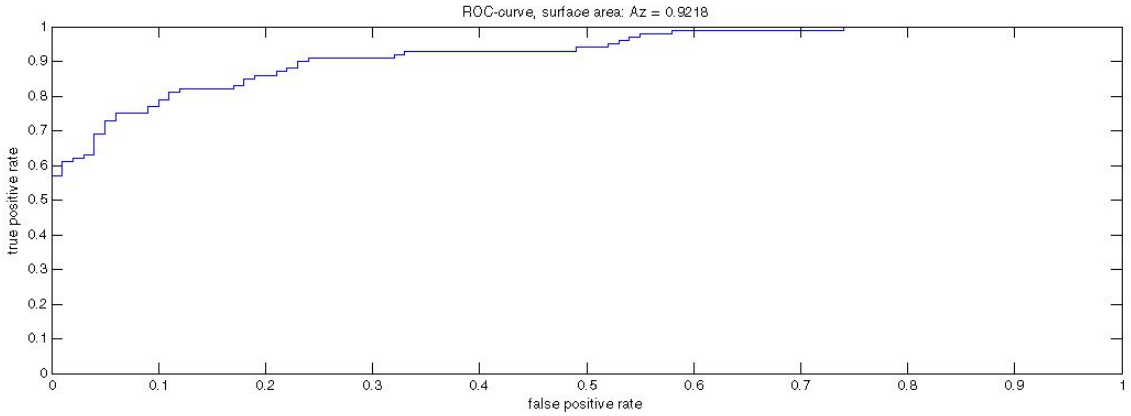


Figure 7.34 ROC-curve belonging to the LD of two 100 sample segments.

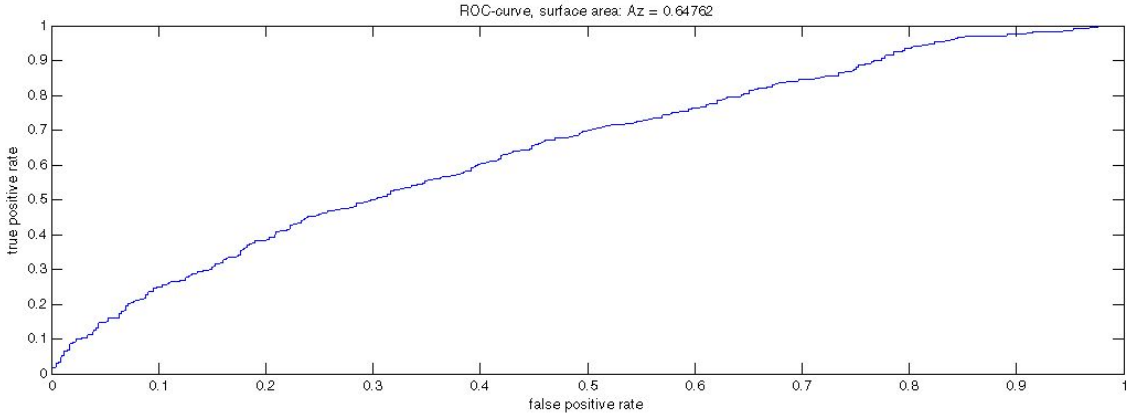


Figure 7.35 ROC-curve belonging to the LD of six 100 sample segments.

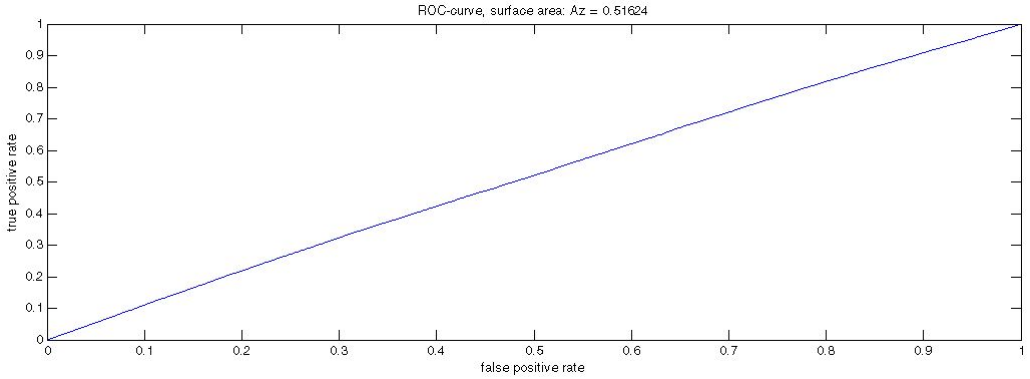


Figure 7.36 ROC-curve belonging to the LD of 400 100 sample segments.

7.3.11.2. LD settings

Several parameters must be set using the linear discrimination (see figure 7.37):

- *Trainingswindow* to specify a smaller section than the entire length.
- *Lambda* (regularize disabled) then this sets the regularization constant for the weight decay.
- *Lambda-search* if the regularize-option is turned on, then a search for the optimal regularization lambda is performed.
- *Eigenvalue-ratio* (minimal eigenvalue / maximum eigenvalue) should be specified if the number of linearly independent rows is smaller than the number of electrodes used. This is in general never the case for EEG data.
- *Regularize* to minimize the effect of outliers on the results.

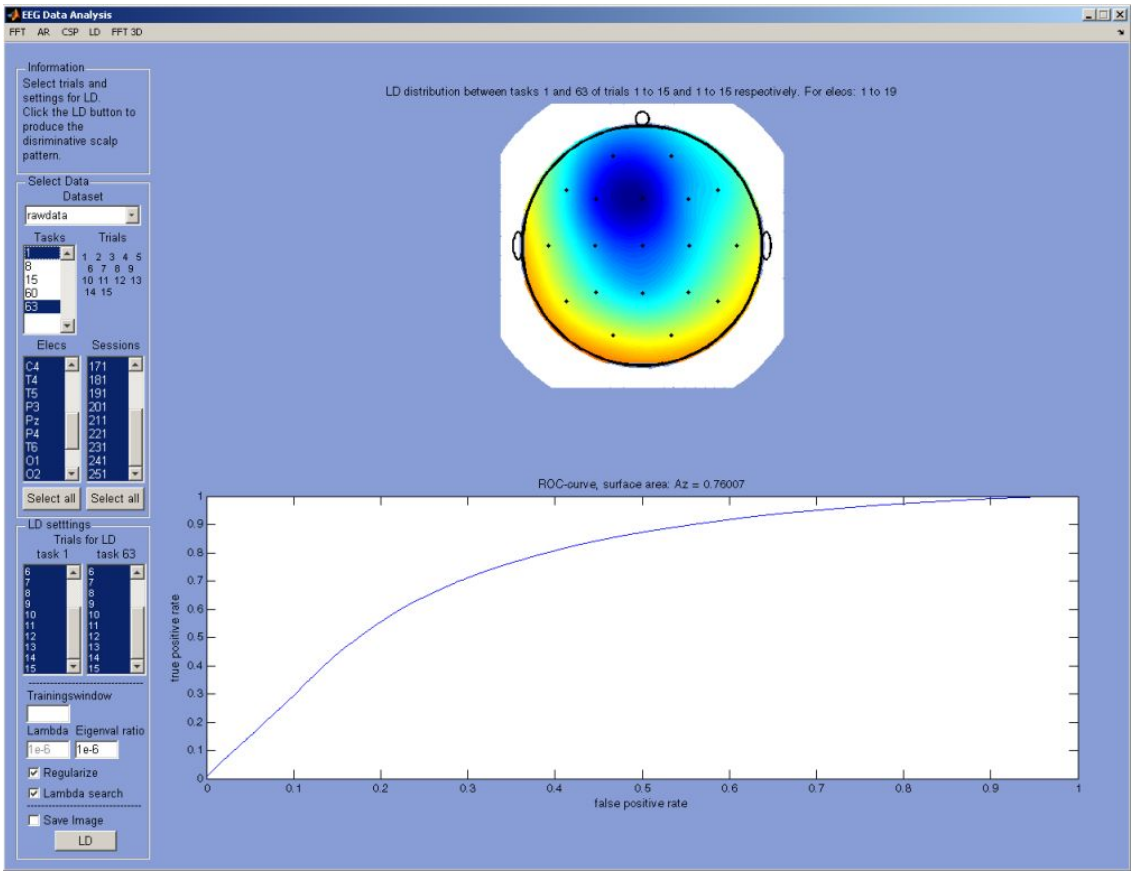


Figure 7.37 Screenshot of the LD component. Left upper side: the data selection, followed by the trial selection. Below this: parameter selection concerning the LD algorithm and visualization. Right the weight distribution and ROC-curve are shown.

It is advised to use the default settings. Whenever EEG data does not correspond to the restrictions put in place by LD, the user will be alerted to change certain parameters.

7.3.12. Data management

During all the pre-processing steps every time a fundamental change is made to the data, a new dataset is created. For instance *rawdata* after pre-processing is called *pre_rawdata* with the original data preserved to be able to easily repeat processing steps and view the results.

To manage the every increasing data pile, the data management component facilitates this process (see figure 7.38).

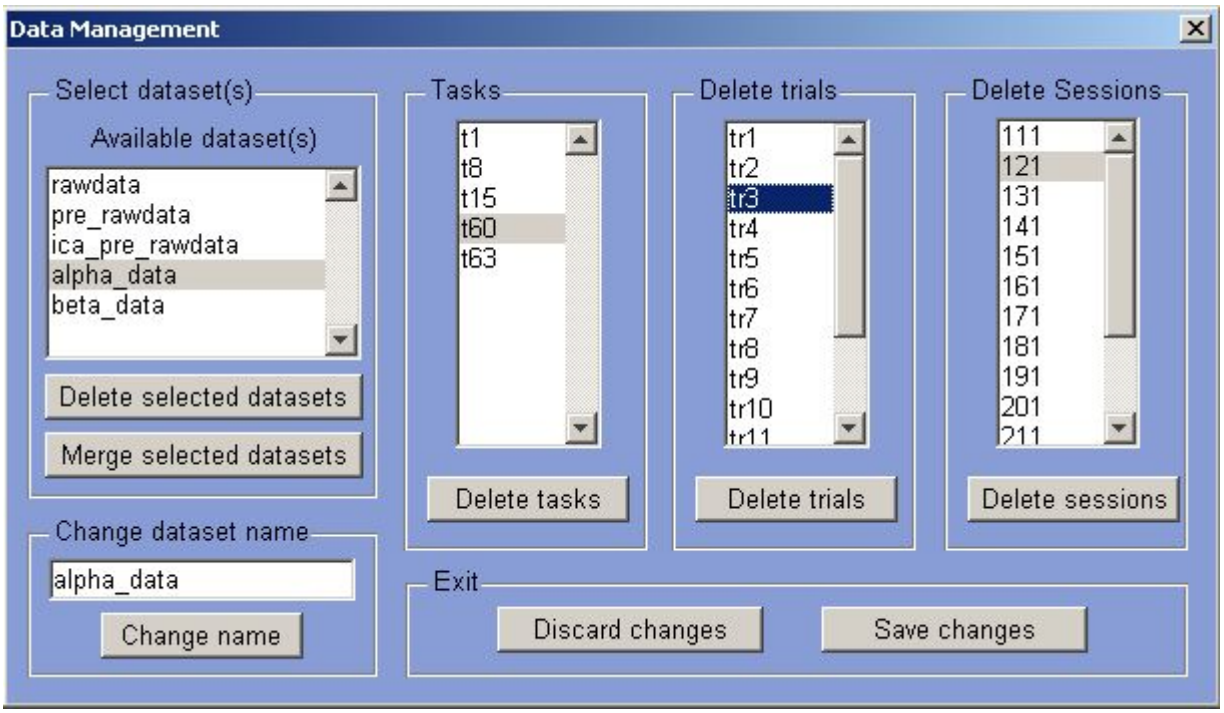


Figure 7.38 Screenshot of the data management component. Top left: selection of dataset and ability to delete or merge datasets. Bottom left name change ability, to the right the ability to delete any part of a selected dataset: tasks, trials or sessions.

The following options are available for data management:

- Deletion of datasets, tasks, trials and sessions.
- Merging of multiple datasets into a single. If two datasets contain the same tasks these are also merged.
- Renaming of datasets.

It is advised for the sake of memory reduction to remove any unnecessary datasets or tasks.

7.3.13. Interface & usability

The final issue covered in this chapter is the interface. The interface is built using Matlab since the program offers good possibilities of constructing a usable GUI facilitated by *'GUIDE' (Graphical User Interface Development Environment)*.

The guidelines stated in section 6.6.2 will be discussed here to indicate how they are implemented. In this explanation figure 7.39 frequently will be mentioned. This figure displays the interaction for the AR model; however it is representative for all the other models.

Visibility of system status / offer informative feedback

At the left top (figure 2.39) the interface starts with the 'information', here the user is continuously informed about the processes and expectations of the E-Brain, it is also used to give instructions to the user concerning both normal operation and error handling. Moreover during calculation the E-Brain shows progress bars and pop-ups to inform the user of the system state.

Match between system and the real world / design dialogs to yield closure

The general data flow in the E-Brain should give the user the feeling that it concerns a real dialog. Therefore first the model and data is selected, followed by model specifics. Finally the results can be viewed and possibly altered. The flow is directed from 'big' decisions to 'small' decisions.

User control and freedom / support internal locus of control

Although the E-Brain tries to guide the user in the flow of the data analysis, the user is actually completely free to go any direction he or she deems necessary. The user is responsible and has the freedom to choose as he or she sees fit.

Consistency and standards / strive for consistency

For each of the models contained within the E-Brain the basic steps are always the same: model selection -> data selection -> parameter selection -> results. This gives the user guidance.

Error prevention / offer error prevention and simple error handling

The E-Brain tries to prevent input errors by offering listboxes of available data selection option (middle of figure 7.39). Where this is not feasible or desirable the user input is checked for correctness. If during the calculations of the models, parameters are not correctly set concerning the data properties, the process is aborted and the user informed on what happened and how to proceed.

Recognition rather than recall

In general the E-Brain recognizes all incorrect input and informs the user about it. In case of incorrect data properties and/or parameters this can only be determined during the actual calculation of the model and the user will be informed accordingly.

Flexibility and efficiency of use / enable frequent users to use shortcuts

For all the major actions within the E-Brain shortcuts are implemented. In the analysis phase a single window contains all the models so that the user does not have to switch continuously and any data and parameters selections and settings are unaltered to quickly jump between models.

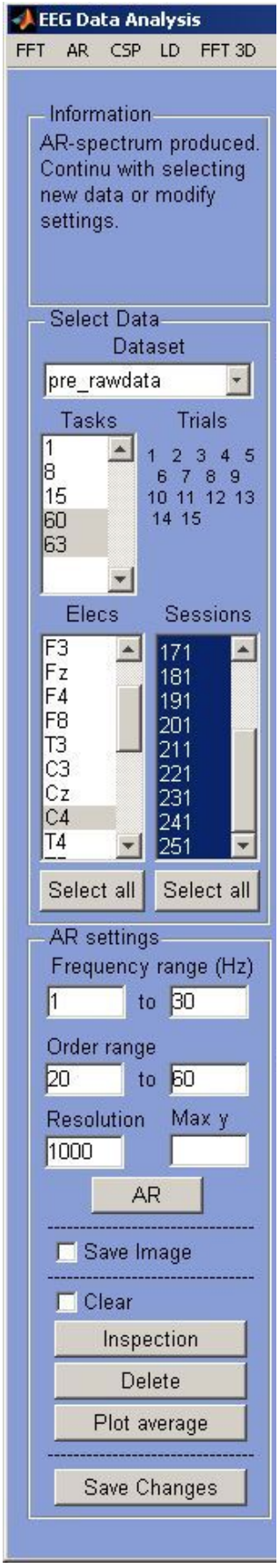


Figure 7.39 The interaction abilities for the AR model.

Aesthetic and minimalist design / reduce short-term memory load

This is one of the fundamental interface design goals set out front, however also the most difficult to implement because of the data characteristics and model properties at hand. Therefore it is chosen to only display that functionality which is necessary at a specific time and reveal more if required. All the models have default values and store the parameters that have been inserted, so that the user does not have to remember them for the entire session.

Help users recognize, diagnose, and recover from errors / permit easy reversal of actions

The entire pre-processing phase is built around the idea of reversal (or repeating) of actions; therefore every step is stored in a new dataset. The original dataset is still available and can be used try different ways of processing. If the data is manipulated the user is always given the ability to either save or discard the changes made or undo the last action. Errors that occur during model calculation will be reported back to the user and the data will be left in the last known good state.

Help and documentation

The user can find documentation on all the functionality found in the E-Brain. Also background theory concerning the models is provided within the workbench (see figure 7.40).

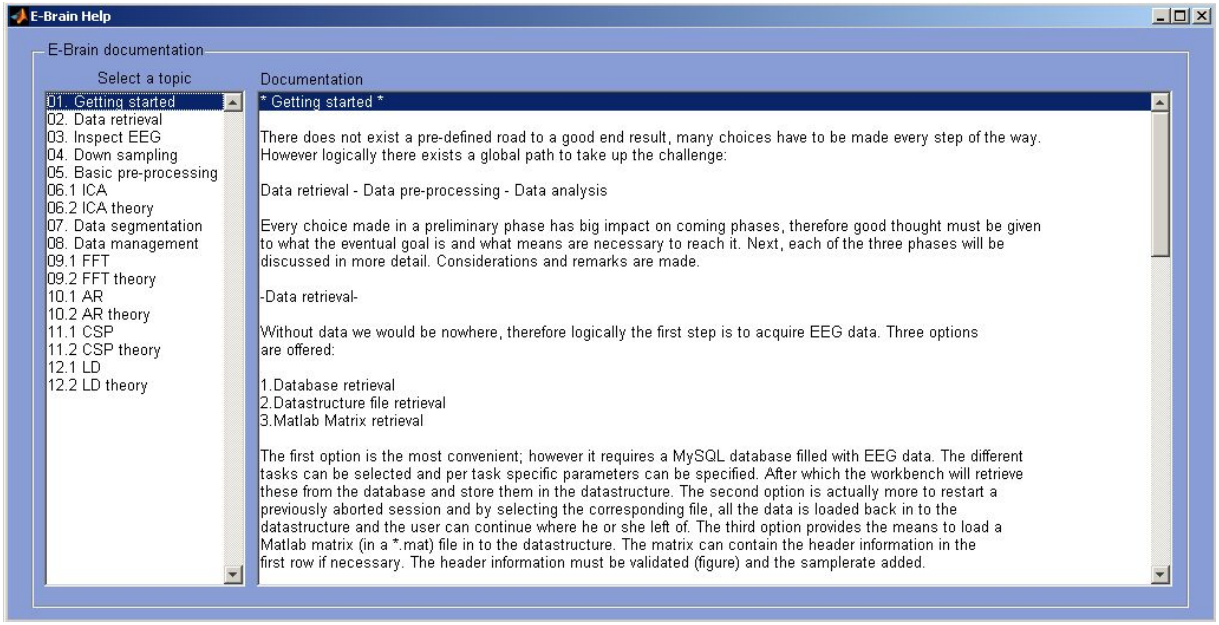


Figure 7.40, Screenshot of the E-Brain help component, on the left a list of topics, on the right the corresponding documentation.

Chapter 8
8. E-Brain evaluation

8.1. *Testing*

Testing is performed to validate the performance of the E-brain (chapter 6 and 7). The testing phase is covered here separately, however this is in fact only in part a separate process, in reality testing was a continuous process during the implementation phase, for some models to a larger extent than for others. During implementation test data is used to check correct functioning of the models and thus answer the fourth research question: *How good is the performance of the data analysis and how useful are the results?*

First the functionality of the models is summarized and after that. The testing sets and testing results are described in the following sections. Different datasets are used to validate the correct functioning of the models in E-Brain. Both artificially created data and third party data is used. First the former will be covered here. This data has unrealistic characteristics; however it contains clear features that can be checked for. The next step is testing the E-Brain with third party data.

To keep the testing coverage within proportion, not all test data used and results generated will be shown. Therefore the examples given here merely give an indication of the testing process.

ICA is not covered here, since the model has been thoroughly tested during the implementation phase (chapter 7) and correct functioning has been validated.

8.1.1. Model functionality overview

In order to interpret the results it must be known how the data is processed and how it is presented to the models. Here the characteristics of the functionality of the models are summarized.

Especially with models using a combination of different tasks as input data, the data selected for analysis should be carefully evaluated. Next on each model a short description of the interpretation of the model resulting output will be given.

FFT
The characteristics of the FFT spectrum reveal both the frequencies of a data segment as well as its magnitude.

AR
The magnitude of the raw EEG data has no reflection on the AR coefficient. AR spectrum only shows the oscillatory properties of the data (see section 2.10.1.3.1).

CSP
CSP shows both the intensity distributions and the filters of two tasks. This information must be evaluated to make decisions regarding the data or in other terms: how to interpret this data. In general it is advised to take only the first two patterns for each task into account, since these two contain the most discriminative information concerning the tasks. After the first two the information-value rapidly decreases.

The distributions (titled CSP) show where one task differs from another, however the second CSP can reveal a completely different picture. This could mean that there is simply another dominant factor (for instance a task which requires more than one active brain area) *or* mean nothing significant at all.

The filter reveals what happens how if the data is filtered: where the data is attenuated and where not. This gives a picture of the resulting important locations of activity.

A simple example to pinpoint an important property of CSP: in the CSP between task *a* & *b* it is revealed where the task *b* differs from *a* in its own CSP. This means that the bulk of activity from *b* with respect to *a* is located at the attenuated spots. And vice versa, at the same time location of absence of activity in task *a* indicate that task *b* probably has increased activity in that area. Especially with CSP it is very important to take into consideration against what data a comparison is made.

LD
LD compares two tasks based on their spatial distributions. Therefore, like CSP, good consideration must be given to what data to compare with the other. However LD produces only a single distribution concerning both tasks. In the result of LD, the colours blue and red are equally important regarding what they tell about the activity localization, the location of the first task is expressed in blue and the second task in red. Therefore the distribution given by LD tells nothing concerning the amplitude of the signals, it merely conveys location information.

8.1.2. Overview test sets

The goal of testing is to validate the functionality described in the previous section. In testing the E-Brain four distinct test sets are used. The first two are artificially created data. The artificial data is created purely for the purpose of testing and does *not* relate directly to common EEG data. It is constructed in such a way that it emphasis the characteristics of the models.

The latter two test sets are third party data sets. The third party data is provided by the '*BCI competition III*'. This is the third version of an international competition between universities and research groups for the best BCI concerning specific problems. The goal of these test sets is to identify the defined tasks within that particular data.

Overview of the test sets used in testing the E-Brain:
- Test set 1: artificial data with high activity in only a part of the available electrodes.
- Test set 2: artificial data with known peak frequencies.
- Test set 3: third party data containing four different tasks.
- Test set 4: third party data containing three different tasks.

8.1.3. Test set 1

The goal of the first test set is two fold; the first is to show the impact of magnitude on the results of the FFT and AR. And the second is to show the discrimination and localization properties of CSP and LD.

Test set 1 contains random noise varying in energy, which is used to validate the correct functioning of the models in the E-Brain. Three different datasets are used (for example of data: figure 8.1 and figure 8.2), which contain high (scaled) energetic data in only a part of the electrodes on the head:
- Upper, frontal area (see figure 8.1).
- Middle, central and temporal area.
- Lower, parietal and occipital area (see figure 8.2).

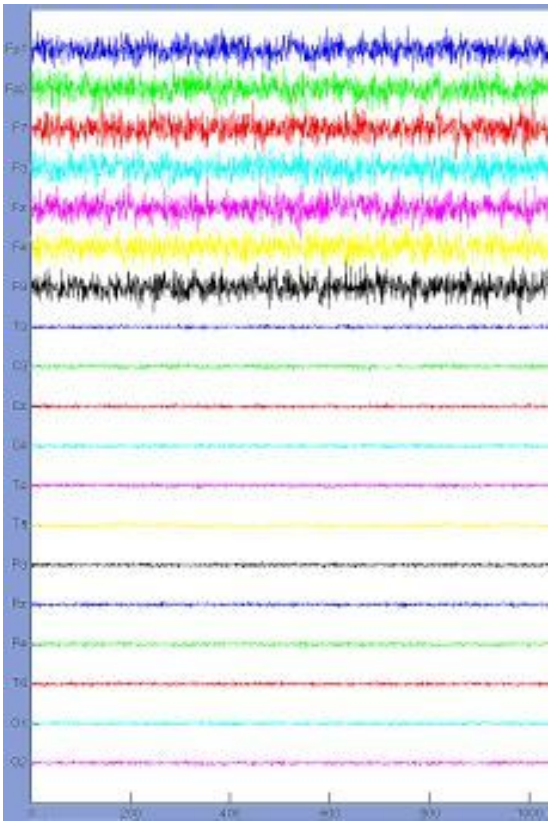


Figure 8.1 Test set 1, plot of raw data: vertically the magnitude and horizontally the time. Highly energetic random noise in *upper* electrodes and low energetic random noise in the rest of the electrodes.

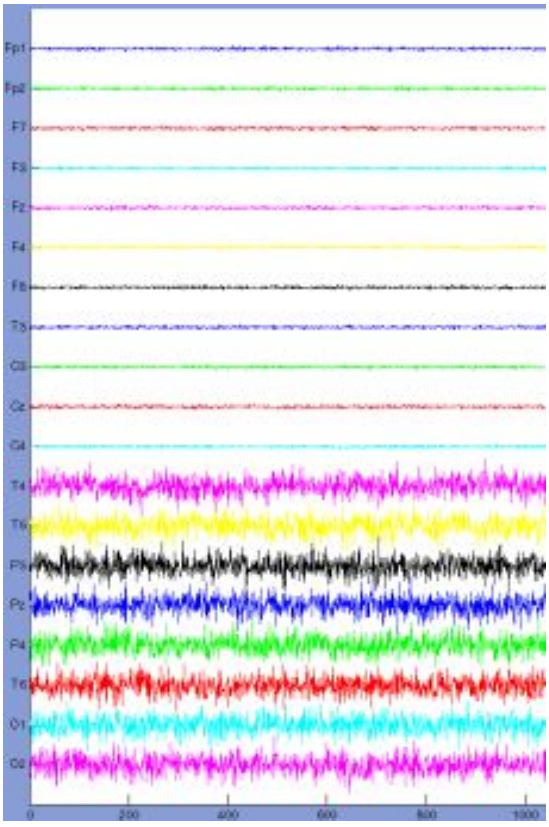


Figure 8.2 Test set 1, plot of raw data: vertically the magnitude and horizontally the time. Highly energetic random noise in *lower* electrodes and low energetic random noise in the rest of the electrodes.

8.1.3.1.Results test set I for FFT & AR

Applying test set I on models FFT and AR, reveals the correct function of their properties. FFT is susceptible for amplitude whereas AR is insensitive to amplitude and only reveals the intrinsic frequencies from the data (figure 8.3 and figure 8.4).

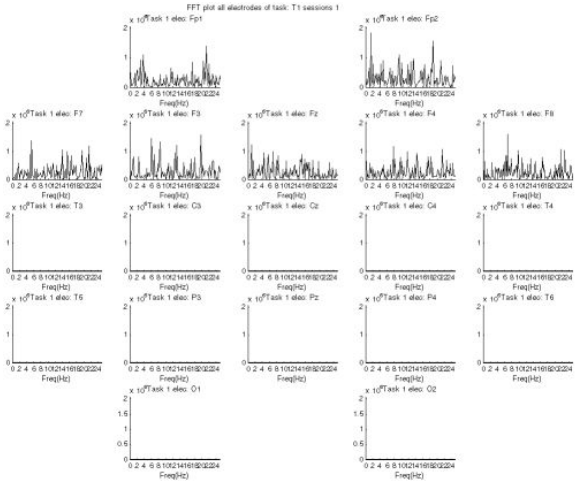


Figure 8.3 Overview of the FFT spectra for 19 electrodes for test set I (upper, figure 8.1). Only (high) activity present in the upper electrodes. Magnitude differences between upper electrodes and the rest. On the vertical axis the magnitude, on the horizontal axis the frequencies.

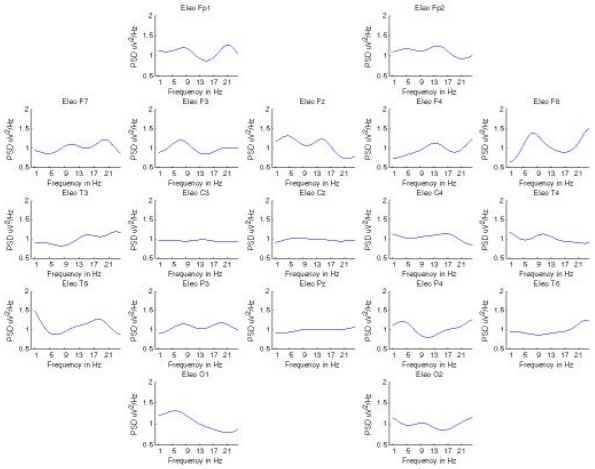


Figure 8.4 Overview of the AR spectra for 19 electrodes for test set I (upper, figure 8.1). No significant magnitude differences between the different electrodes. On the vertical axis the magnitude, on the horizontal axis the frequencies.

8.1.3.2.Results test set I for CSP

Set I is processed using the CSP algorithm to test the discrimination and localization capabilities of the model; a result is shown in figure 8.5.

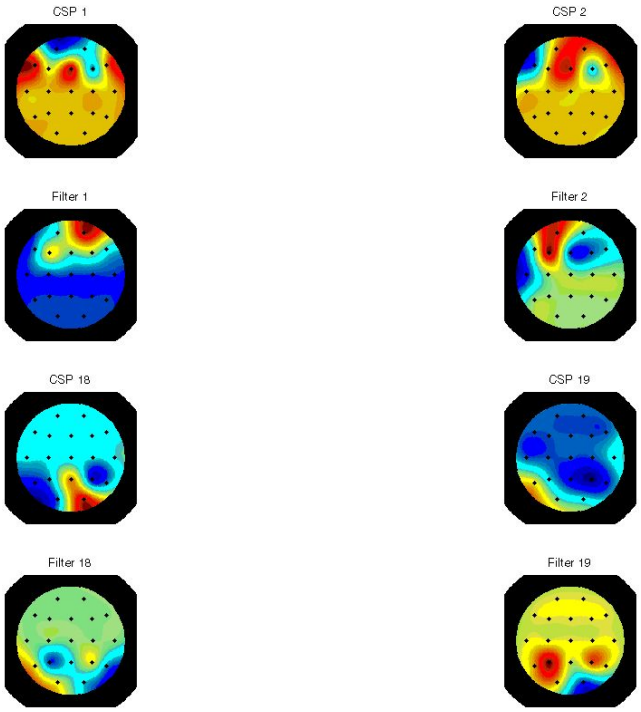


Figure 8.5 Test set 1 in CSP: *upper* test data (figure 8.1) versus *lower* test data (figure 8.2). Both the CSP and the filters shows activity at the correct locations: 1 & 2 (first two rows) for the *upper* test data versus 18 & 19 (last two rows) for the *lower* test data.

It is clear from both the patterns as well as the filters that the intensity is located at the correct locations corresponding to the intensity of set I. The fact that the neither top nor bottom of any of the pictures is completely one color is caused by variations in the randomness of the signal. The example is merely used here to indicate the localization properties of CSP.

8.1.3.3. Results test set I for LD

To test the correct functioning of the LD two slight adjustments are made to test set I. In the first alteration the activity was set high per electrode instead of per group of electrodes (see a result in figure 8.6). In the second (see a result in figure 8.7), test set I has a high but flat spectrum (no random noise).

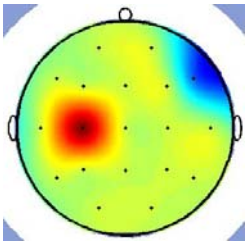


Figure 8.6 Test set 1 (single electrode) in LD: the location of high activity in electrode 7 and 9 is correctly recognized.

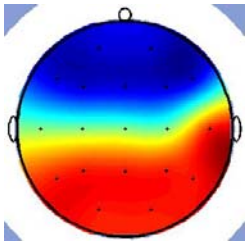


Figure 8.7 Test set 1 (high flat spectrum) in LD: the location of activity of the front versus the back is correctly recognized.

The basic image that is drawn from these tests is that LD finds the most important locations of activity.

What is important to point out is that the first task is represented in blue and the second in red; this has nothing to due with their intrinsic properties and unfortunately is not even linked to the first or second task and is just as easily reversed. Furthermore any location that has nothing to do with activity patterns of either task is shown as yellow-green: meaning no importance (as can be seen in figures 8.6 and 8.7).

8.1.4. Test set 2

The second test set is used to validate the frequency performance of both FFT and AR. The artificial data is randomly normal distributed noise with three peaks at 5, 25 and 76Hz in electrode Fp1. These peaks are shifted circa 1Hz per electrode and the total intensity of the data is increased per electrode. Figures 8.8 (FFT) and 8.9 (AR) show examples of the correct display of the frequency spectrum.

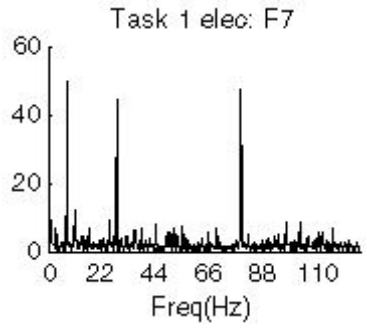


Figure 8.8 Test set 2 in FFT, frequency peaks are correctly displayed. Vertically magnitude, horizontally frequency.

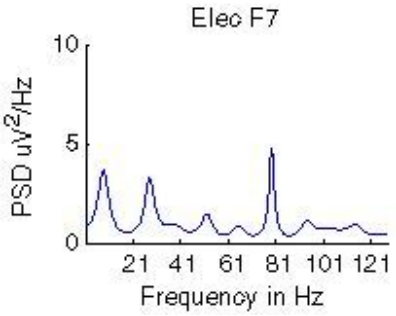


Figure 8.9 Test set 2 in AR, frequency peaks are correctly displayed. Vertically magnitude, horizontally frequency.

The frequencies of both methods are located at exactly the right locations. More over it is evident that the amplitude makes no difference for AR and clearly does for FFT. The slight differences found in the amplitude between different electrodes in figure 8.11 are mainly due to differences in the order selection for each electrode.

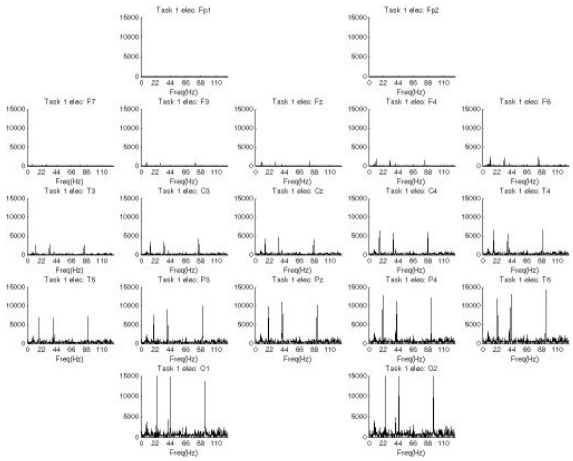


Figure 8.10 Test set 2 processed by FFT for 19 electrodes; it shows the ever increasing amplitudes and the frequencies at the right location.

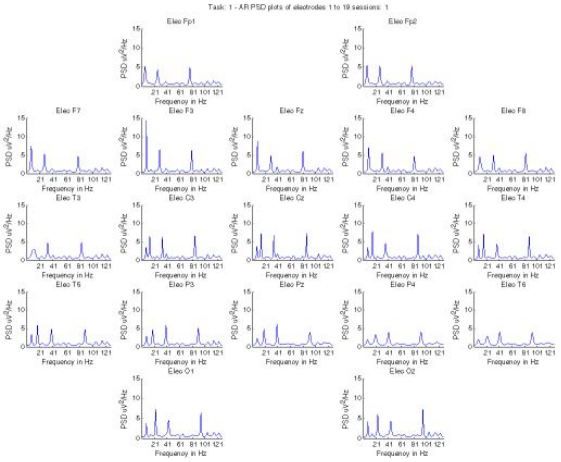


Figure 8.11 Test set 2 processed by AR for 19 electrodes; it shows the correct frequencies at the same amplitude level.

8.1.5. Test set 3

The goal of test set 3 and 4 is to validate the ability of the E-Brain to correctly extract features. From the next two test sets it is known what kind of activity should be found within the data.

The data is explored and searched for these characteristic features. The third party dataset is obtained from the *'BCI competition III'* and concerns dataset IIIa. The data is provided by provided by *'the Laboratory of Brain-Computer Interfaces (BCI-Lab), Graz University of Technology'* and concerns data measured from real human being (non-artificial). This dataset could be classified by the competitors with accuracy between 60% and 80% (the value of classification by chance is zero).

Hypothesis
The experiments concern cued motor imagery (multi-class) with 4 classes (refer to figure 2.4 for localization of body parts over sensorimotor cortex). The expected localization of the tasks:

- Task 1, left hand: localization in the sensorimotor cortex on the right hemisphere.
- Task 2, right hand: localization in the sensorimotor cortex on the left hemisphere.
- Task 3, foot: localization in the sensorimotor cortex in the center.
- Task 4, tongue: localization in the sensorimotor cortex edge of both hemispheres.

Three subjects (ranging from quite good to fair performance) and is measured using an EEG system with 60 channels. The recording was made with a 64-channel EEG amplifier from *Neuroscan*, using the left mastoid for reference and the right mastoid as ground. The EEG was sampled with 250 Hz, it was filtered between 1 and 50Hz with a notch filter.

Modifications
Conversion from the 60-channel data to 19 channels is performed to match the format used in E-Brain (see figure 8.12). Furthermore the dataset is bandpass filter in the alpha (8-13Hz) and beta (14-30Hz) band. The next step is to examine the data in the workbench and have a look at the differences and similarities within the experimental data.

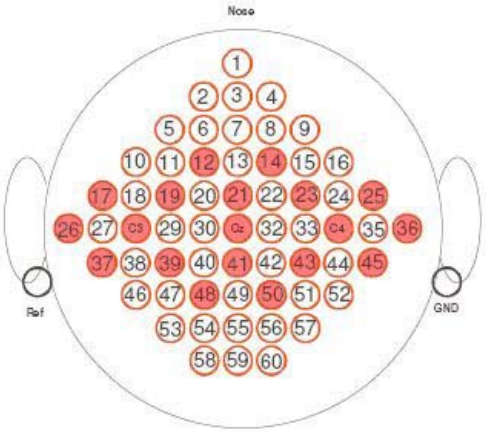


Figure 8.12 Mapping of the 19 electrode 10-20 system to the 60 channel system. The focus is on the sensorimotor cortex: C3, Cz and C4.

8.1.5.1. Results set 3 for CSP

In the analysis of this test set, not only is searched for activity enhancement, but equally so for activity suppression, which is likely for focused motor imagery. Sensor motor activity has two important oscillation-ranges: 8-13Hz (alpha band) and 14Hz and upwards (beta band).

From literature [45, 46] it is shown that alpha activity is suppressed on the contralateral hemisphere. Beta activity can increase on the contralateral hemisphere. Therefore the final results will be increased activity on the ipsilateral side for hand movement.

In table 8.1, the main locations of activity that are found using CSP for a specific task are summarized. The first two of both the CSP and filter are taken into account, where the first has a bigger impact on the final result. In the table *centre* means the actual centre electrodes surrounding

and including *Cz*, *left* and *right* meaning the middle electrodes surrounding *C3* and *C4* respectively, o*utside* meaning the outer rim electrodes 26 and 36. The bias label is just added to indicate the next (much smaller) area of activity (if any).

Table 8.1 Overview of the main activity in the alpha band for each task versus the others (vertically).

Task	**1**	**2**	**3**	**4**
1	x	Right	Centre	Outside
2	Left	X	Centre	Outside; with centre bias
3	Non-centre; left bias	Non-centre; right bias	X	Outside, non-centre
4	Left-Centre	No direction	Centre	X

Looking at the results of table 8.1, it is clear that one has to take into consideration the tasks which are being compared. For instance comparing a task versus the 'centre-activity' task results most of the time in absence of activity for the compared task in the centre. In the next four tables (8.2 to 8.5) each of the tasks is classified according to the witnessed results.

Table 8.2 Task 1 is identified as left hand task: main activity located in the middle of the left hemisphere (electrodes surrounding C3).

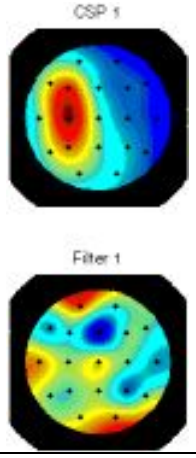
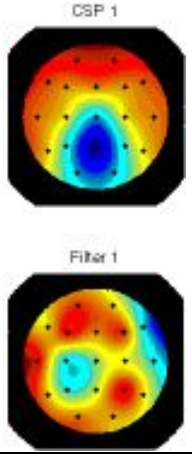
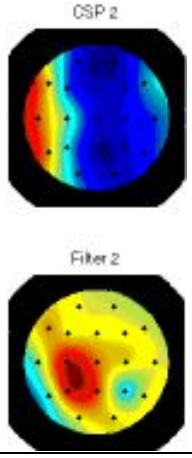
Task 1 In comparison with task 2 the CSP shows left side activity, task 2 is the probable right hand candidate, therefore this info is very important. Versus the third task the CSP is shows clear lack of activity in the centre. Versus task 4 the filter indicates left hemisphere activity. On the whole these images suggest left hemisphere activity.	2 	3 	4 

Table 8.3 Task 2 is identified as right hand task: main activity located in the middle of the right hemisphere (electrodes surrounding C4).

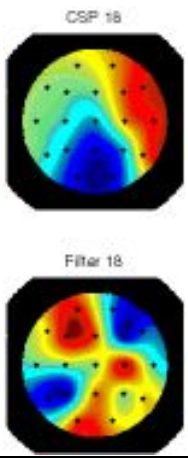
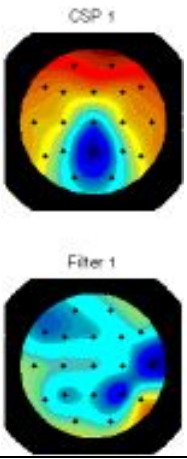
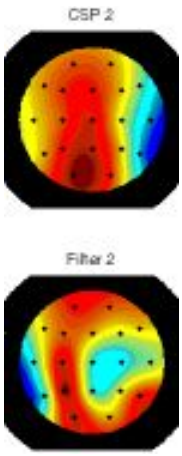
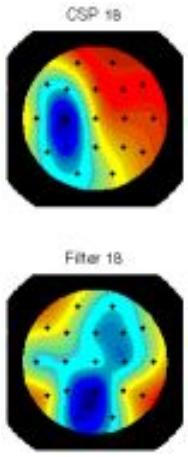
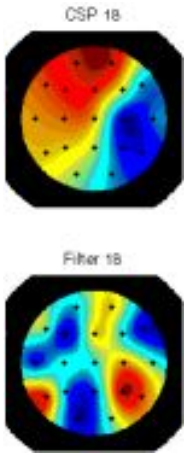
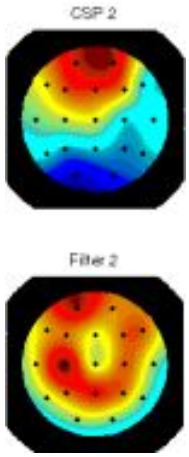
Task 2 In comparison with task 1 (the most probable left hand task) the CSP shows favour for the right side. The CSP reveals absence of activity in the centre versus task 3; however the filter shows mild favour for right activity. Versus task 4 is the focus appears on the left side, but this can be due to high right side activity in task 4. The central activity shows the high 'outside' activity of task 4.	1 	3 	4 

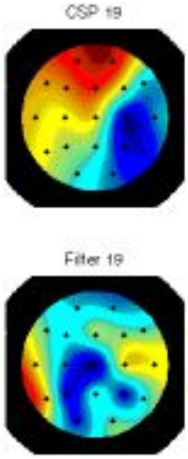
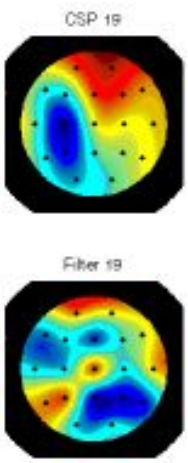
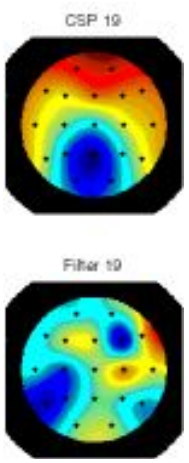
Table 8.4 Task 3 is identified as the foot task: main activity located in the centre of the head (electrodes surrounding Cz).

Task 3 Looking at the tasks shown in the *other* tables it is clear they all present a central gap of activity caused by task 3. Versus task 1 the CSP show an absence of left side activity, which is overly present in task 1. In comparison with task 2 the exact opposite case is presented. And versus task 4 the main activity is in the central region where task 4 has more dominance in the outside regions.	1 	2 	4 

Overall it can be said that it is very important against what kind of task the data is compared. The CSP provides the ability to distinguish between the tasks presented in test set 3.

The evidence found using the CSP model is validated by the FFT and AR in the next section.

Table 8.5 Task 4 is identified as the tongue task: main activity located on the border of both hemispheres (electrodes surrounding 27 and 35).

Task 4 The problem with task 4 is that it is the main activity located at two locations. Often the first CSP would show left activity and the second right, indicating the main activity at both sides This may be not so clear in these pictures. Versus task 1 and 2 the filter shows more activity on the edges. The whole of these pictures seem to re-enforce the tasks with whom they are compared.	1 	2 	3 

8.1.5.2.Results set 3 for AR

Next test set 3 is used with AR to find evidence of the tasks. Next only the average spectra of the AR model, which are produced by E-Brain, are shown concerning electrode activity (figure 8.13 to 8.15). Because the tasks concern the sensorimotor cortex, only three central electrodes are presented here as an example, since they provide the best indication on the tasks.

The spectra in figure 8.13 and 8.14 clearly show the suppression of the activity in the alpha band for tasks 1 and 2. And for each task the activity is even more suppressed at the contralateral hemisphere, completely complying with the hypothesis. In figure 8.15 the spectra of the central electrode *Cz* is given, it is very clear that the activity from task 3 is suppressed in comparison with the other tasks.

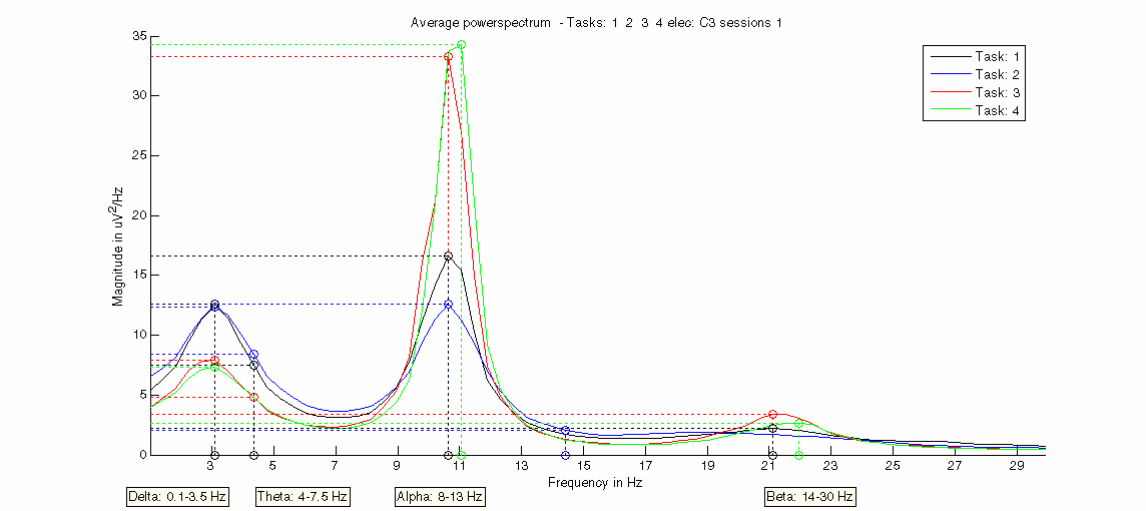


Figure 8.13 Test set 3 in AR: Average AR spectrum at electrode **C3** of the four tasks. It clearly shows the suppression of activity from tasks 1 and 2, with still higher activity of task 1. This complies with the hypothesis.

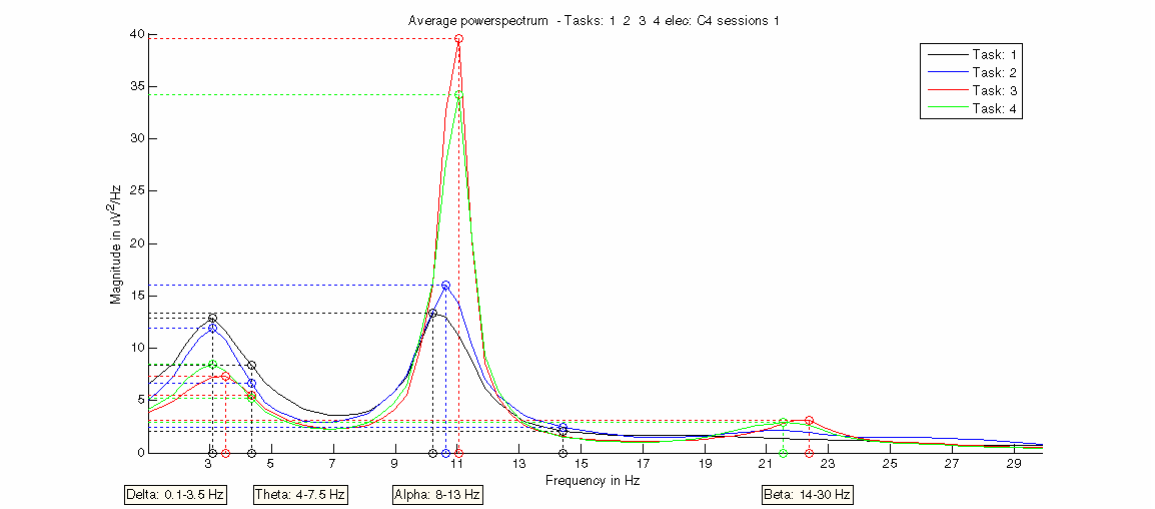


Figure 8.14 Test set 3 in AR: Average AR spectrum at electrode **C4** of the four tasks. Clearly shows the suppression of activity from tasks 1 and 2, with still higher activity of task 2. This complies with the hypothesis.

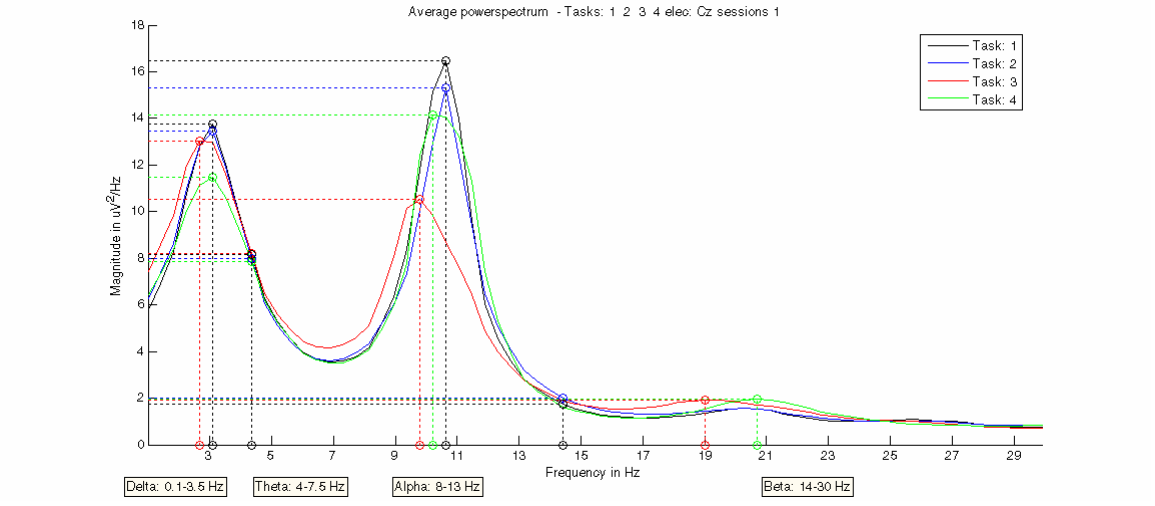


Figure 8.15 Test set 3 in AR: Average AR spectrum at electrode **Cz** of the four tasks. Clearly shows the suppression of activity for task 3. This complies with the hypothesis.

These spectra validate the results found using CSP. The four classes which are supposed to be there according to the data description are actually found using the E-brain.
LD produces similar results, because the colours are not defining the tasks clearly, there exists some ambiguity, since there is only a single LD pattern for both tasks.

8.1.6. Test set 4

The fourth test set is another dataset from the '*BCI competition III*' and concerns dataset V, it is provided by the *IDIAP Research Institute*. This dataset could be classified by competitors with accuracy up to circa 70% (33.3% if by chance). Test set 4 is measured using different equipment, subjects and experiments from test set 3.

Hypothesis
This test set contains data from 3 normal subjects during 4 non-feedback sessions. There are 3 tasks within this test set, but only the two imagery movement tasks are used in this particular test:

- Task 1, imagination of repetitive self-paced left hand movements, (*left*, class 2): localization in the sensorimotor cortex on the right hemisphere.
- Task 2, imagination of repetitive self-paced right hand movements, (*right*, class 3): localization in the sensorimotor cortex on the left hemisphere.
- Task 3, generation of words beginning with the same random letter, (*word*, class 7).

All 4 sessions of a given subject were acquired on the same day, each lasting 4 minutes with 5-10 minutes breaks in between them. The subject performed a given task for about 15 seconds and then switched randomly to another task at the operator's request. Samplerate is 512 Hz.

Modifications
Similar to test set 3, conversion from the 32-channel data to 19 channels by selection is performed to match the format used in E-Brain again focusing on the sensorimotor cortex. After removing artifacts using ICA, the dataset is bandpass filtered in the range from 8-13 (alpha) and 14-30Hz (beta). The next step is to examine data in the workbench and have a look at the differences.

8.1.6.1. Results set 4 for AR & FFT

For task 1 and 2 (left and right hand respectively) alpha should be suppressed at the contralateral side and this will result in a more dominant ipsilateral side (similar to the previous test set).

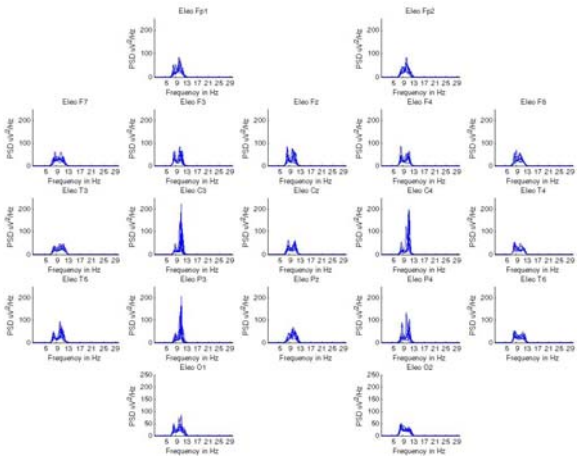


Figure 8.16 Test set 4 in AR alpha range of task 1 for all electrodes: clear activity in central electrodes: C3, C4 and P3.

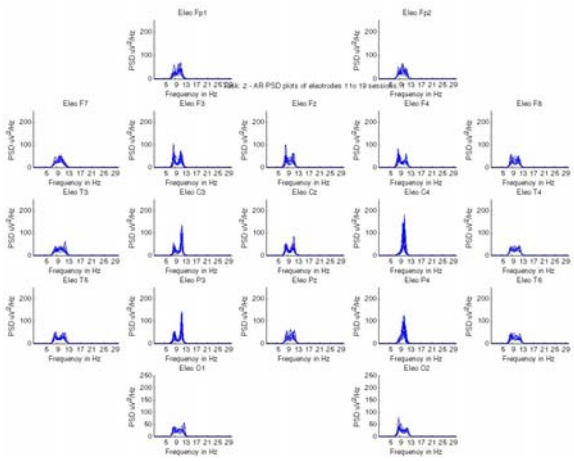


Figure 8.17 Test set 4 in AR alpha range of task 2 for all electrodes:, clear activity on right side: C4.

This is demonstrated by figure 8.16 and 8.17 for AR. And even in figure 8.18 and 8.19 for FFT.

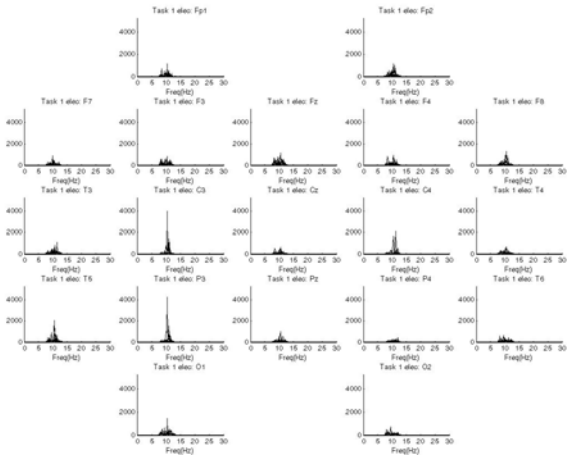


Figure 8.18 Test set 4 in FFT alpha range of task 1 for all electrodes: clear activity in left side: C3 and P3.

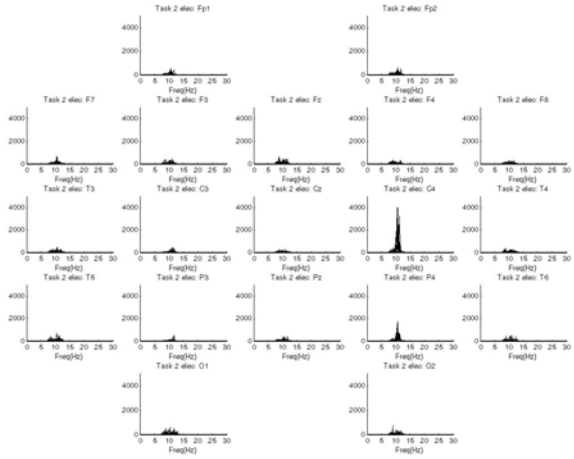


Figure 8.19 Test set 4 in FFT alpha range of task 2 for all electrodes: clear activity in right side: C4.

8.1.6.2.Results set 4 for CSP

The picture drawn by AR and FFT of the two tasks is supported by the results of the CSP, figure 8.20 and 8.21. Clearly the left hand task (CSP 1 and 2) indicates left activity (right hemisphere de-synchronization). And CSP 19 favours the right side. According to the hypothesis there could be an increase in the beta-band for both tasks, however no significant differences where found between the two. The results of the analysis of test set 4 comply with the hypothesis concerning these tasks.

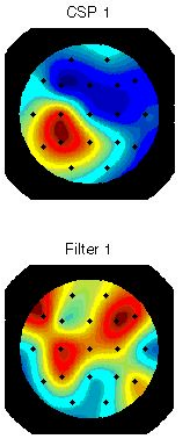


Figure 8.20 Test set 4 in CSP of left versus right hand task. CSP 1 shows the left hemisphere activity.

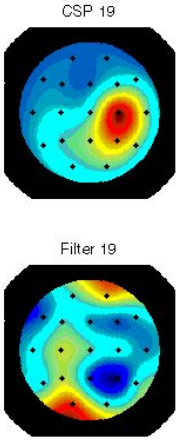


Figure 8.21 Test set 4 in CSP of right versus left hand task. CSP 19 shows the right hemisphere activity.

8.2. *Workbench review*

Only the four analysis models are mentioned in the testing phase. All the methods used in the pre-processing phase have been validated during the implementation. ICA (as can be seen in chapter 7) performs exactly as it should and enables the possibility to remove eye-movement related artifacts as well as pulse artifacts and reconstruct the data accordingly.

FFT and AR return correct results for data with known frequencies, using a bandpass filter (known Matlab performance) the results comply with the theoretical characteristics of the models (see section 8.1.4). The same holds for the sensitivity to magnitude: FFT represents these properties in its spectra, AR does not (section 8.1.3.1). FFT and AR also perform as expected for the third party data.

The insensitivity to magnitude makes AR more stable to interpret, however it is less robust against the variable EEG signals, on the other hand finding absolute values of activity with FFT can also be valuable.

CSP is very sensitive to artifacts and any data fed to the model should be filtered very carefully. Using CSP in analysis requires good understanding of what is compared to what, since this has a big impact on the witnessed results. Therefore it is advised to draw conclusions on multiple tasks to filter out any odd outlier effects. The discrimination ability of CSP is validated in section 8.1.3.2, 8.1.5.1 and 8.1.6.2.

The ROC-curve of the LD will in general be between 0.5 and 0.6. The main cause for this low value is that the EEG data seen over 19 channels for an extended period of time is very similar. Therefore a high difference in a single channel compared to the other task will not result in too much difference over 19 electrodes. A solution to this problem would be to use very small data segments. This however requires it to be quite certain that the task is really performed during this short time segment.

Interpreting LD is quite a hassle in its own right, since it is unknown which colour depicts which task. Therefore it is advised not to base any conclusions solely on the LD results. It can be used to verify results found by the other modalities. Or obtain search directions concerning the data.

Practical recommendations

A computer with a large amount of memory would improve the E-Brain's speed, since it is working with relatively large blocks of memory. The E-Brain does offer the possibility to remove irrelevant data. Loading all available data into the E-Brain and database is not advisable: since it is useless, from a practical point of view, to compare all tasks and subjects at the same time. A small selection should be made concerning the tasks to be inspected.

Especially calculating the AR spectra for a large number of trials and electrodes can take a while, defining an optimal order can significantly increase the performance of AR, especially if it is not set too high. Another measure to counter calculation time can be to down sample the data.

If in the future the E-Brain is extended to an online BCI system, then a state-of-the-art computer will be a necessity for the intensive real-time requirements requested by such a system.

8.3. *Conclusion*

All models give to expected results for the performed tests according to the theory. Only the results for the third party test sets of LD are somewhat ambiguous and hard to interpret 'stand-alone'. So accept for the usefulness of LD, no fundamental problems are found during the testing phase. The problem that arose with LD is the ambiguity of the interpretation of the weight distribution of two tasks. Therefore it is recommended to be careful drawing any conclusion based on the LD results.

The E-Brain is capable of extracting characteristic features from the data in test sets 3 and 4, which comply with the stated hypothesises in sections 8.1.5 and 8.1.6.

Overall the E-Brain meets all the objectives stated at the beginning of chapter 6. The workbench is reliable both in terms of stability and re-producible results, whilst offering a solid structured analysis approach for the user. The only objective which is hard to quantify here is the effectiveness, since this would at least require a comparison with similar tools like E-Brain and moreover is not wholly dependent on the workbench but also on the qualities and experience of the user. At least for the third party datasets tested, the correct features were found during testing.

The next step is classification of the results obtained using the E-Brain. E-Brain also meets these non-functional requirements, in that it should be relatively easy to extent and modify the workbench for this purpose, since it offers a well documented approach up until the feature classification phase. And the data from the feature extraction phase is readily available for the next step.

Chapter 9
9. Results of experiments

This chapter covers the analysis of the results from the BCI experiments (chapter 4) using the E-Brain (chapter 6, 7 & 8). In part this answers the fourth research question: *How good is the performance of the data analysis and how useful are the results?* And helps to answer the final research question: *How to proceed in future experiments and BCI research?*

The goal of the analysis is to globally survey the quality of the data and search for distinguishable differences within the data and to be able to give an indication on the quality of the data. Therefore not all results are shown in this chapter, but merely a selection which should make the process of analysis clear and re-enforce the conclusions.

First a description of the experiment results is given, after which the general analysis approach is explained together with the results from the analysis. The chapter is concluded with discussion, conclusion and recommendation concerning both this chapter and the fourth chapter.

9.1. *Experiment result description*

The gathered data is summarized in this section. Table 9.1 presents an overview of a single session of the experiment. It covers task-IDs, task descriptions, duration of tasks and number of repetitions corresponding to the tasks mentioned in chapter 4.

Table 9.1 Single experiment/session overview

Task-ID	Task description	Time (sec)	Trials
T1	Baseline task - eyes open	30	1
T2	Baseline task - eyes closed	30	1
T3, T13	Mathematical Task: summation - (eyes closed, eyes open)	n/a	1
T4, T14	Mathematical Task: subtraction - (eyes closed, eyes open)	n/a	1
T5, T15	Mathematical Task: multiplification - (eyes closed, eyes open)	n/a	1
T8, T18	Geometrical figure rotation - (eyes open, eyes closed)	30	1
T9	Hyperventilation	60	1
T20	Visual self selection – Yellow Triangle (eyes open)	5	+/- 5
T22	Visual self selection – Green Dot (eyes open)	5	+/- 5
T24	Visual self selection – Red Cross (eyes open)	5	+/- 5
T26	Visual self selection – Blue Lines (eyes open)	5	+/- 5
T30	Presentation – Yellow Triangle (eyes open)	5	5
T32	Presentation - Green Dot (eyes open)	5	5
T34	Presentation – Red Cross (eyes open)	5	5
T36	Presentation – Blue Lines (eyes open)	5	5
T40, T41	Auditive evoked Yellow Triangle – (eyes open, eyes closed)	8	4
T42, T43	Auditive evoked Green Dot – (eyes open, eyes closed)	8	4
T44, T45	Auditive evoked Red Cross – (eyes open, eyes closed)	8	4
T46, T47	Auditive evoked Blue Lines– (eyes open, eyes closed)	8	4
T50	Visual and auditive evoked – Yellow Triangle (eyes open)	5	5
T52	Visual and auditive evoked – Green Dot (eyes open)	5	5
T54	Visual and auditive evoked – Red Cross (eyes open)	5	5
T56	Visual and auditive evoked – Blue Lines (eyes open)	5	5
T60, T61	Imagery movement : Left hand circle - (eyes open, eyes closed)	20	1
T63, T62	Imagery movement : Right hand circle - (eyes open, eyes closed)	20	1
T64, T65	Imagery movement : Left hand grab - (eyes open, eyes closed)	20	1
T67, T66	Imagery movement : Right hand grab - (eyes open, eyes closed)	20	1

This amounts to a total of 39 tasks per session. However certain tasks are performed multiple times (trials) during a session.

The experiments are performed on two test persons:

- Subject one: Mark Wessel, 24 year old male.
- Subject two: Pavel Hrubeš, 30 year old male.

Both have undergone 20 one hour experiments with over 1 hour of preparation with a maximum of 2 experiments per day. Any more will simply lead to useless results because of concentration problems and fatigue. Each single experiment is labelled using consecutive numbers for each following experiment at the end the subject number is appended. For example the 14th experiment of subject 2 is labelled *142.*

9.2. *Experiment analysis*

The following tasks are chosen to be analyzed:

- Baseline task (**T1**), this task is examined globally to understand its characteristic features. Since this is the task against which all others must be compared in case of an asynchronous BCI. Therefore it is insightful to know where the other tasks differ from the baseline task.
- Imagery Movement (**T60 & T63**), both right and left movement and it is opted to examine the 'circle'-movement task.
- Mathematical equation (**T15**), from the three possible tasks, it is opted to analyze the multiplification. From this task it is known that the user was continuously calculating and moreover is not disturbed in anyway during the performance of the task. This is not the case for the other two tasks: summation and subtraction.
- Mental rotation (**T8**).

For all these tasks, the version with the eyes open is used for analysis, since this is the most functional and logical version. The reason for choosing these tasks is that they (theoretically) differ the most amongst each other of all the tasks performed during the experiment (see section 4.7). The other tasks require either muscle movement or have great similarity to the baseline task/mental rotation.

The data from the experiment is analyzed using the E-Brian (see chapter 6 to 8). For subject one the sessions 111 to 251 (total of 15) and for subject two the sessions 232 to 272 (total of 5) have been selected.

9.2.1. Pre-processing & analysis approach

After the data has been downloaded from the database, it is pre-processed using a CAR filter, a 1-30Hz bandpass filter and finally baseline correction. Then next step is to remove obvious artifacts at the start of each trial (like for instance drift and eye blinks) using the manual EEG inspection component. Especially task 1 has some serious drift in the first part of the trial for subject one.

After this step ICA is used to further remove artifacts from each trial. Aside from the obvious eyeblink artifacts, also very a-symmetric ICs are removed. For instance a completely average IC with at the end a large spike: although visible neither in the rawdata nor clear evidence in weight distributions this probably indicates an artifact. Pulse is harder to identify for subject one, especially for data with over 4000 samples, because the IC will be too compressed to be able to see the oscillating heartbeat. The heartbeat does not have a distinctive pattern in the weight distribution like eye-artifacts with subject one. When applying ICA during pre-processing, the pulse artifact from subject two does not only have a characteristic IC, but also (and unlike subject one) has a characteristic pulse weight distribution. It concerns a diagonal pattern from the top left to down right. This could be due to a greater impact of the heartbeat of subject two on its physiology.

Although it is tempting to discard odd-looking IC, care must be taken with removal, since discarding a 'none-artifact IC' could be worse than keeping an (small) artifact IC.

After ICA the data is divided into an alpha (7-13Hz) band and beta (14-30Hz) band. The final step of the pre-processing is the segmentation into 50% overlap on 256*4 sample segments in all tasks.

General analysis notes

Using FFT the tasks are searched for outlying sessions with respect to the other sessions, these are sessions with huge peaks that cannot possibly be original or appropriate signals. The sessions that were out of line for subject 1 are: 241 and 121 and none for subject 2.

Next both FFT and AR is used to obtain an overview of each of the task by plotting the complete data range for all electrodes. Using AR, the bands (alpha and beta) can show different behaviour in comparison with the total 1-30 band, this is due to the way AR calculates the coefficients and the optimal order used. It is not possible to say that either one is flawed; they are simply different views on the data. Using these modalities together with the theory (table 4.2) an overview is generated concerning which tasks differ in which locations from each other that are of importance.

Finally using AR, CSP and LD different tasks are compared at these specific locations. Using the AR average quickly conclusion can be drawn about the proposed sites of activity. CSP is used to validate any patterns found using FFT and AR.

Hypothesis for T1

The baseline task (T1) is thought to be the same all the time (not meaning the same activity in all electrodes) and can thus be generated at will. Therefore no specification of the activity regions is given.

9.2.2. Analysis subject one

Hypothesis for T1 vs. T8 vs. T15

The main activity for task 8 (mental rotation) should be found in occipital lobe. For task 15 (math task) it should be in the parietal lobe (see table 4.2).

Locations of interest for T1 vs. T8

AR shows overall the same image for *T1* as *T8*, except for the electrodes Fz, F4, T6, O1 and O2 where *T8* and in Cz, T4, P3, Pz and P4 where *T1* has increased alpha activity.

In combination with table 4-2, which reveals important locations of activity in the parietal and occipital lobe and part of the temporal lobe, closer inspection into electrodes: Cz, T4, P3, Pz, P4, T6, O1, and O2 will be performed.

Locations of interest for T1 vs. T15

The first thing that is noticeable is the overall increase for *T15* concerning the alpha activity, most in the frontal region F3, Fz and F4 and at the back: O1 and O2. In combination with table 4-2, which reveals important locations of activity in the parietal lobe, closer inspection into electrodes: P3, Pz, P4, O1 and O2 will be performed.

Locations of interest for T8 vs. T15

Overall picture of more activity in *T15* in frontal (F) and central region (T & P), however slightly more activity for *T8* for O1 and T6. In combination with table 4.2 closer inspection into electrodes: T3, T4, T5, P3, Pz, P4, T6, O1 and O2.

Analysis of locations of interest T1 vs. T8 vs. T15

Table 9.2 shows the results of the comparison of the electrodes selected in the first step. The table is based on the average AR spectra for instance shown in figure 9.1 to 9.4. If within 1 magnitude, two tasks are assumed equal.

Table 9.2 Ranked overview of inspected electrode between task T1, T8 and T15 concerning the average AR spectra. Between brackets: estimation of the magnitude in squared microvolt per Hertz.

Electrode	Rank 1	Rank 2	Rank 3
T3	T15 = T8 (16)		T1 (13)
C3	T8 (27)		T1=T15 (22)
Cz	T15 (30)	T1 (25)	T8 (15)
C4	T15 (25)	T8 (22)	T1 (16)
T4	T15 (27)		T1=T8 (16)
T5	T15=T8 (15)		T1 (12)
P3	T1=T15 (24)		T8 (19)
Pz, see figure 9.1	T15 (28)	T1 (19)	T8 (16)
P4, see figure 9.2	T15 (27)		T1=T8 (20)
T6	T8 (25)	T15 (21)	T1 (15)
O1, see figure 9.3	T8 (19)		T1= T15 (15)
O2, see figure 9.4	T8 (29)	T1 (18)	T15 (15)

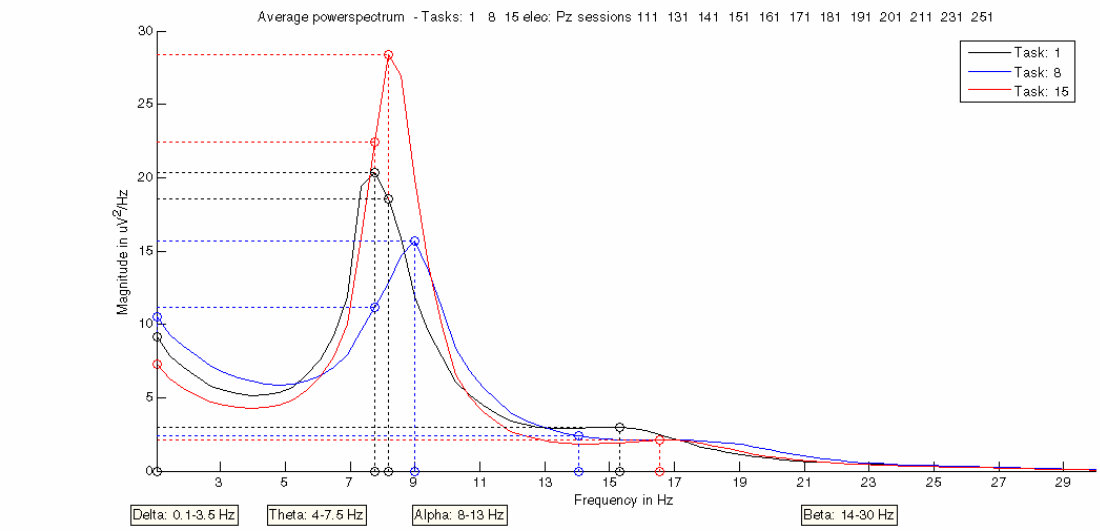


Figure 9.1 Subject one: average AR spectrum of tasks T1 (black), T8 (blue) and T15 (red) for electrode **Pz**.

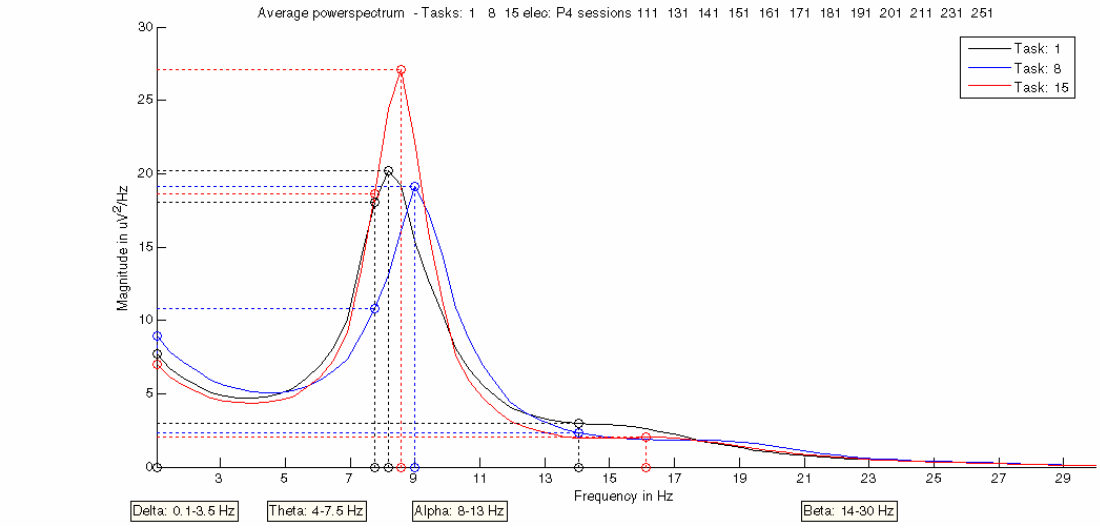


Figure 9.2 Subject one: average AR spectrum of tasks T1 (black), T8 (blue) and T15 (red) for electrode **P4**.

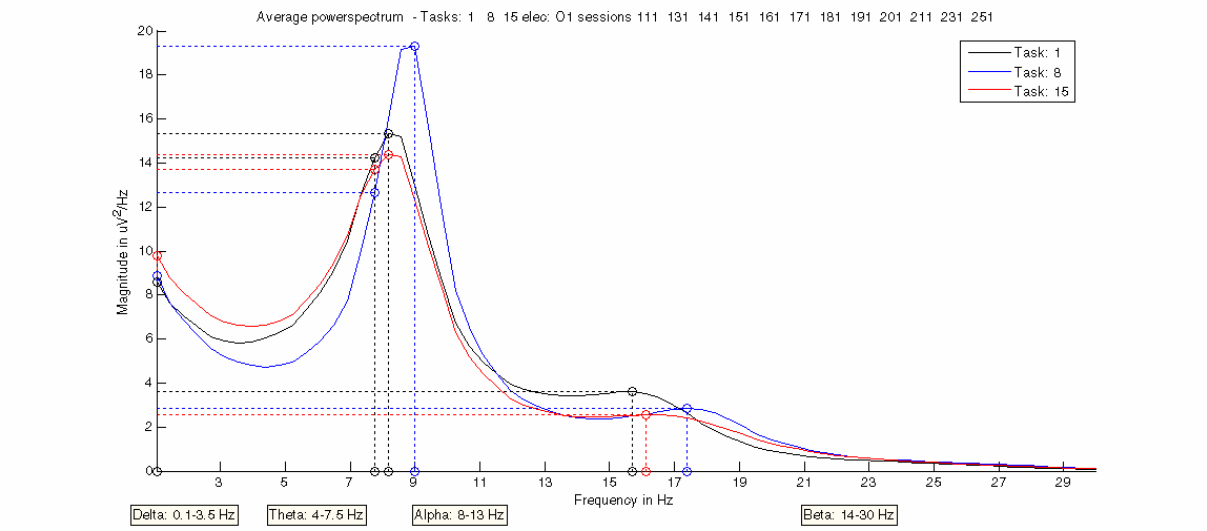


Figure 9.3 Subject one: average AR spectrum of tasks T1 (black), T8 (blue) and T15 (red) for electrode **O1**.

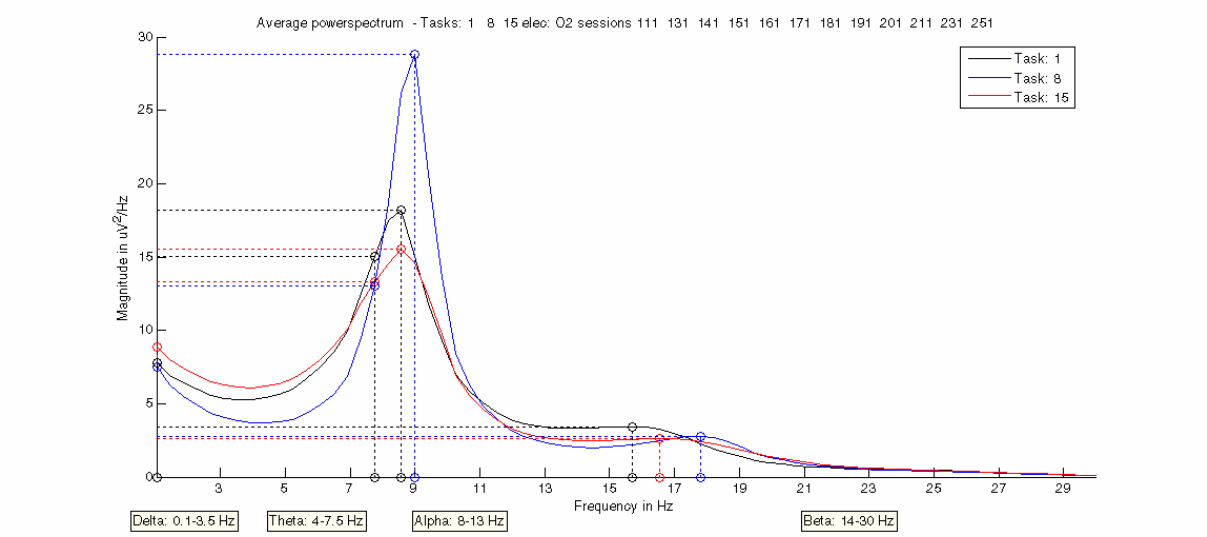


Figure 9.4 Subject one: average AR spectrum of tasks T1 (black), T8 (blue) and T15 (red) for electrode **O2**.

The analysis shows dominance of *T8* in occipital C3, T6, O1 and O2 and parietal dominance of *T15* in Cz, C4, T4, Pz, P4, see figure 9-5. This supports the hypothesis.

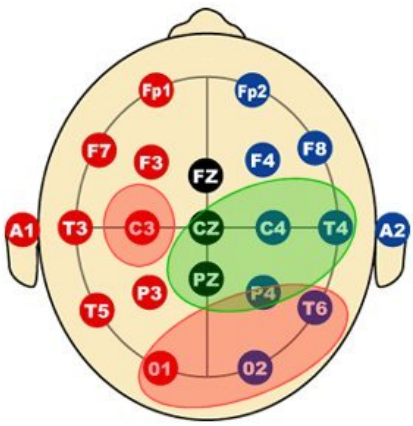


Figure 9.5 Overview of activity dominance areas of T8 (red) and T15 (green).

CSP shows very different results for different trials. No clear picture is drawn from this model. On top of this CSP appears to recognize eye-artifacts from ICA that perhaps is applied to loose on the data. LD also finds some evidence for the pattern found by AR, but this is very dependent on the trial selection.

Synopsis T1 vs. T8 vs. T15
Clear differences in the alpha activity distribution can be found over the electrodes *on average*. Still large and contradictory results can be found for trials and even sessions. Using these global results it should be possible to distinguish a task, but only on average and not acting as a reliable online system. Except for electrode C3, the results of the analysis appear to comply with the hypothesis.

Hypothesis T1 vs. T60 & T63
The main activity for task 60 (left) should be found in C3 and for task 63 (right hand) it should be C4 (see table 4.2).

Locations of interest T1 vs. T60 vs. T63
Analysis focus for task *T60* and *T63* will be on C3 and C4 and to lesser extent also on P3 and P4 which showed different activity versus *T1* in the AR and FFT.

Analysis of locations of interest of T1 vs. T60 & T63
Table 9.3 shows the results of the comparison of the electrodes selected in the previous step. The table is based on the average AR spectra shown in figure 9.6 to 9.9. If within 1 magnitude, two tasks are assumed equal.

Table 9.3 Ranked overview of inspected electrode between task T1, T60 and T63 concerning the average AR spectra. Between brackets: estimation of the magnitude level in squared microvolt per Hertz.

Electrode	Rank 1	Rank 2	Rank 3
C3, see figure 9.6	T1 (20)	T63 (16)	T60 (14)
C4, see figure 9.7	T63 (19)	T1 (15)	T60 (12)
P3, see figure 9.8	T1 (22)		T60=T63 (20)
P4, see figure 9.9	T63 (25)	T1 (20)	T60 (17)

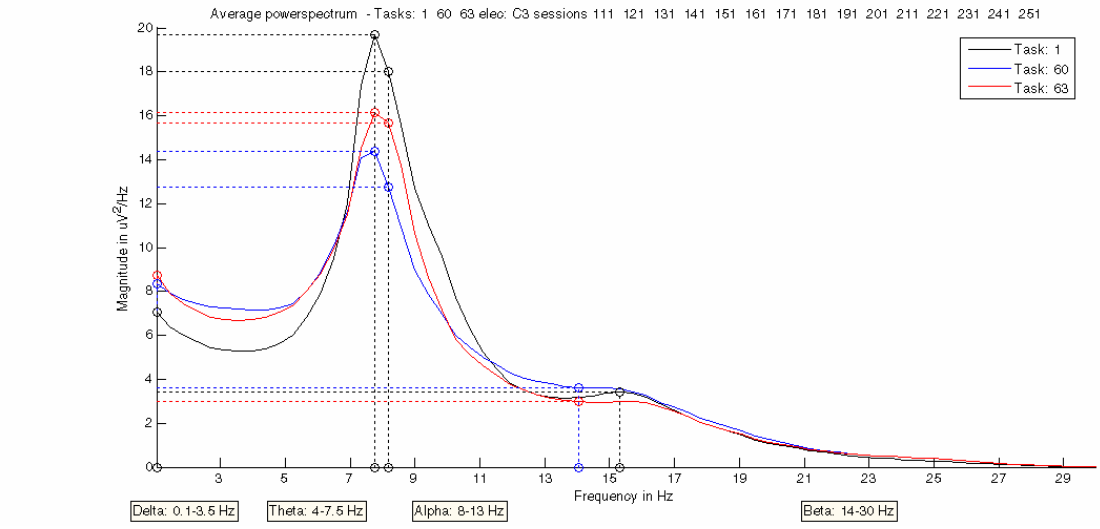


Figure 9.6 Subject one: average AR spectrum of tasks T1 (black), T60 (blue) and T63 (red) for electrode **C3**.

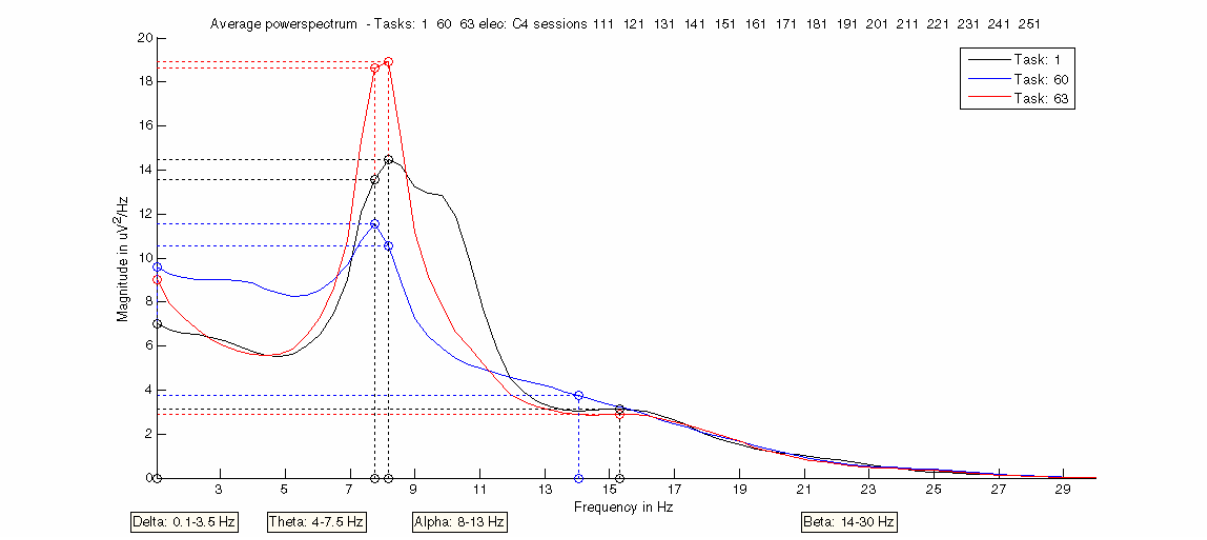


Figure 9.7 Subject one: average AR spectrum of tasks T1 (black), T60 (blue) and T63 (red) for electrode **C4**.

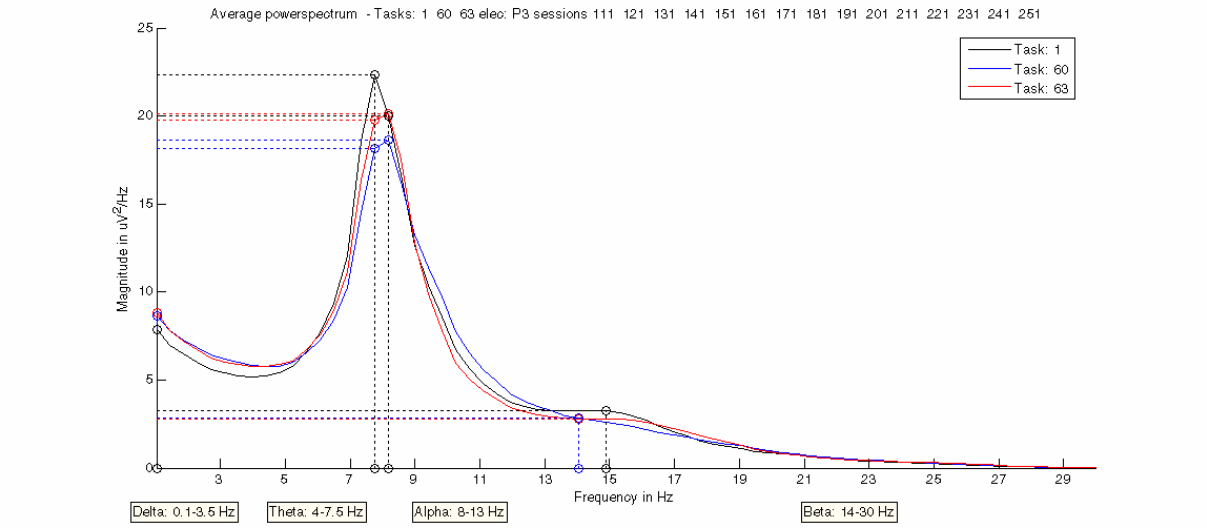


Figure 9.8 Subject one: average AR spectrum of tasks T1 (black), T60 (blue) and T63 (red) for electrode **P3**.

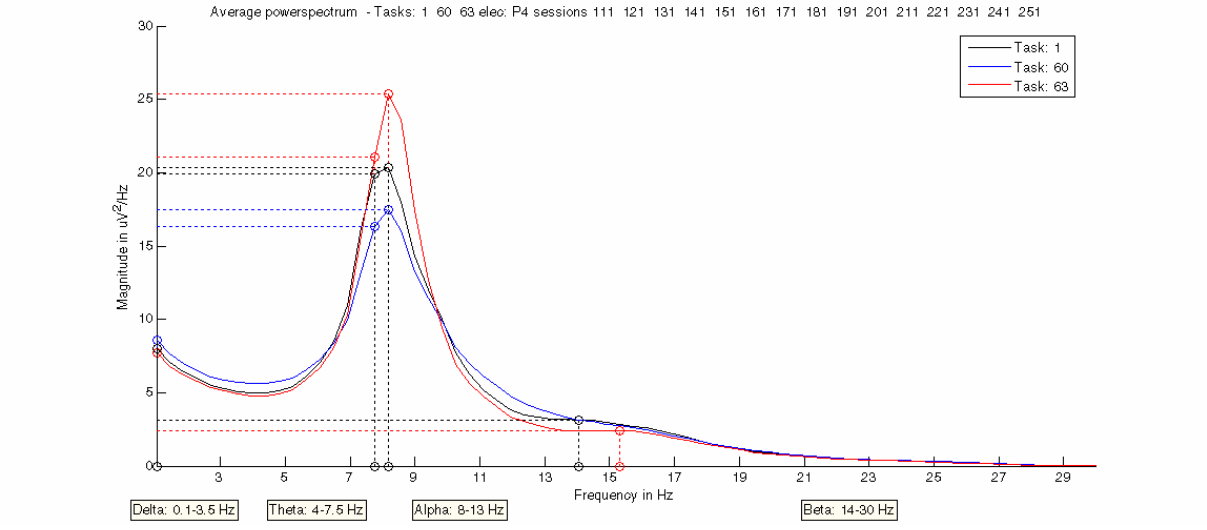


Figure 9.9 Subject one: average AR spectrum of tasks T1 (black), T60 (blue) and T63 (red) for electrode **P4**.

The overall picture is high T1 alpha activity everywhere, but in T63 the right side does not change and the left side activity is suppressed, re-enforcing the hypothesis. Overall T63 is higher on the right side versus its own activity and versus T1. T60 is higher on the left side (C3, P3) and lower on the right (C4, P4), but only versus itself and not versus T1 or T63. T1 appears to exhibit exactly the same characteristics as T60 for C3-C4 and P3-P4.

Analysis T60 vs. T63

Although when comparing T60 with T63, T63 is always higher in activity than T60, the tasks are characterized by the fact that they both have higher activity on the ipsilateral versus the contralateral hemisphere. This is schematically shown in figure 9.10.

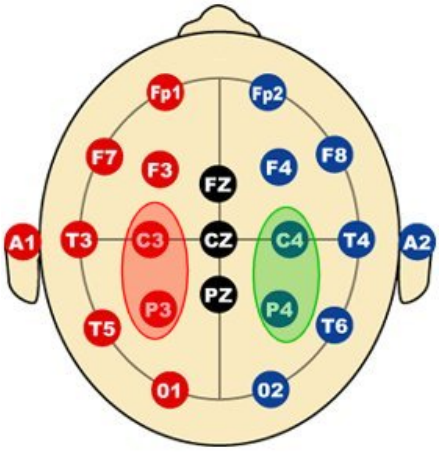


Figure 9.10 Overview of activity areas of T60 (red) and T63 (green).

CSP

The two most discriminating patterns were used during analysis. Looking at the CSP of T60 versus T63 (for all 15 sessions), it appears that T60 is biased to the left and T63 biased to the right.

The two tasks (T60 & T63) have also been compared per session (CSP is very sensitive to changes in electrode position, within one session they are assumed not have moved). Not all the tasks show the same picture, however on the whole the same pattern seems to emerge. The signal itself is highly variable and comparing *two* tasks contributes even more to this effect. This does support the trend found using AR. Large differences exist however between single sessions.

LD

The ROC provided with LD generally shows an Az-value just over 0.5 up to 0.6. Comparing T60 & T63 this is not a real surprise, since globally these tasks do not differ much at all, and should theoretically only differ in a small area (the sensorimotor cortex).

Synopsis T1 vs. T60 vs. T63

Evidence exists that the right hand task (T63) is actually visible in the data, showing suppressed activity in C3 (left hemisphere) and unchanged activity in the right hemisphere (C4). The left hand task (T60) shows similar (opposite) characteristics although not as intensively as T63 and moreover versus the baseline task (T1) no significant differences are found.

This leads to the conclusion that T60 versus T63 are distinguishable amongst each other. But adding the baseline task adds ambiguity to the process, one way to circumvent this problem could be to establish T1 as the benchmark, since it has the highest activity in all electrodes and simply registering a downfall in activity. After that point the C3 versus the C4 can be checked on activity amongst the left and right hand task (using for instance the CSP). Electrode P3 and P4 display the same characteristics as C3 and C4 only slightly less emphasised.

9.2.3. *Analysis subject two*

For subject two exactly the same tasks are analyzed using the same hypothesis as stated in the previous section. Contrary to the analysis of subject one, the data of subject two revealed very little differences between the tasks. The only major difference is that T1 has abundance in activity over all electrodes in comparison with the other tasks. Not even between task T60 and T63 is there any significant difference. The cause for this is unknown.

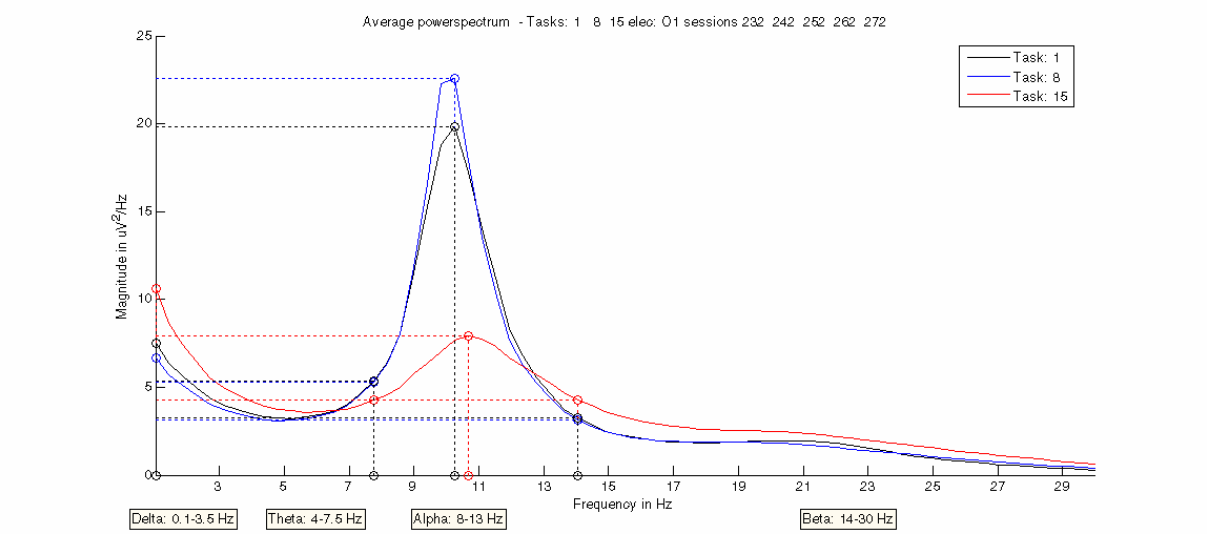


Figure 9.11 Subject two: average AR spectrum of tasks T1 (black), T8 (blue) and T15 (red) for electrode **O1**.

Analysis T1 vs. T8 vs. T15

Except for the occipital region (see figure 9.11) T1 exhibits more alpha activity than T8. T15 appears to have suppressed activity throughout all electrodes in subject two. It is not clear what caused this; it can point to a highly focussed mind which suppresses all activity.

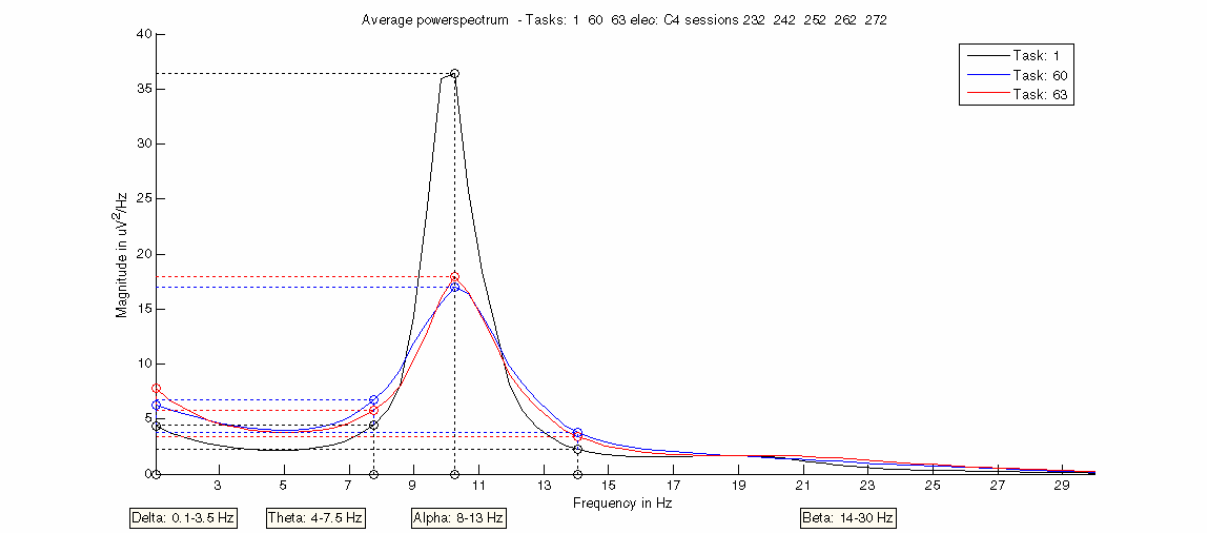


Figure 9.12 Subject two: average AR spectrum of tasks T1 (black), T60 (blue) and T63 (red) for electrode **C4**.

Analysis T1 vs. T60 vs. T63

In the AR comparison there appears to be next to no difference between task T60 and T63 (see figure 9.12), they are both suppressed relative to T1. No real suppression differences found between

hemispheres (C3, P3 versus C4, P4). Both T60 and T63 show more characteristics of the opposite meaning T60 looks like task T63 and vice versa.

For none of the tasks the CSP returned consistent images of the distributions. When applying only sessions 272 to the CSP reasonable results, complying with the hypothesis, are returned concerning task T60 and T63 (also versus T1).

9.3. *Experiment evaluation*

9.3.1. Discussion

The analysis

The overall picture for subject one reveals that *on average* the two tasks (T60 and T63) can be distinguished on the basis of picture drawn of the AR spectrum in electrodes C3, P3 versus C4, P4. It must be stressed however that there exist huge inter-trial differences and that these differences mentioned here concern the average of 15 sessions for subject one. The same goes for T8 and T15, both appear to comply with the hypothesis regarding their localization and it should be possible to classify them correctly. Task 8 also shows increased activity in electrode C3, although this does not comply with the hypothesis, however it cannot be stated that the measurement is therefore incorrect. The brain is one gigantic hive of linked activity and performing a task in one part of the brain certainly has effects on other parts.

One of the first things that stands out after analyzing subject one *and* two is the differences between two subjects, who have performed exactly the same tasks. The data of subject one does reveal usable differences between the performed tasks, whereas these are not found in the data of subject two.

A major uncertainty is the fact that it is not known whether or not every trial actually represents the intended task. The subject could have done something different or perhaps not be concentrated on the task for the complete trial period. It would be useful if the information-value (in the sense of the sessions actually containing the specified activity) could be determined, so that none-informative sessions can be discarded.

It could well be possible that subject two was not performing the tasks as he should have done and hence render all the data useless.

LD is hardly mentioned in the analysis, since it is very hard to give funded interpretation to the results produced (see chapter 8).

Motor imagery seems to be the most applied task in BCI. There exist valid reasons for this, since they are among the most easy to generate voluntary *and* their location is very precisely defined. Therefore it is easier to recognize these brain patterns.

The experiment

The analysis of the experiments does result in some differences between tasks, their significance and applicability is however questionable. The global picture obtained from this work is the poor data quality (for subject two even more than for subject one), caused by the high resemblance of the different tasks to each other.

One of the main causes of the poor quality of the data is the incorrect placement of the ground and reference electrode. In the initial experiment setup the ground and reference were attached to the mastoid bone (bone located behind the ear), since these are relatively stable points on the head. However due to heavy pulse artifacts in the data with subject two, it was deemed necessary to use the ground and reference electrode located on EEG cap, next to the Pz and Fz respectively. This resulted in the disappearance of the heavy artifact. However this caused a major instability in the measurement and thus heavily influencing the quality and interpretability of the results, because the ground and

reference are instable if positioned at those locations. For sake of comparison the same locations were chosen for subject one.
Another contributor to the decrease of the data quality is the fact that the spatial resolution is very poor. The use of just 19 electrodes is simply not enough: for instance in the current setup there is only a single electrode on any of the hemispheres of the sensorimotor-cortex. Even a slight error will render a trial useless, since there is no evidence in the other electrodes concerning that area of the brain. In addition increased spatial resolution allows for capturing different forms of motor activity, which is impossible with the current resolution.

A third factor which has impact on the data quality lies within the experimental design. A major problem for any BCI experiment is to know for sure that a particular piece of data contains the activity which it should (the subject was actually concentrated performing the task). This chance should be maximized using relatively short time frames for the activity to take place in.

The fourth impact factor is caused by the fact that for measuring the voltages. The electrodes have to be filled with a conducting gel. Aside from being an arduous task, this caused the resistance within the electrodes never to be at the same level between different electrodes and certainly not between different sessions and thus inflicting a hard to quantify error to the measurements.

A final factor which may have influence on the results is the fact that the electrodes are not on exactly the same location between different sessions, especially for a method like CSP this can have a big impact on the results.

Most of the factors mentioned above can be countered or minimized. For the first factor, incorrect placement of ground and reference electrode, the ground and reference electrode should be placed on the mastoid bone behind the ear.

The effect of the second, fourth and fifth factor: poor spatial resolution, gel filled electrodes and location shift over sessions respectively can be reduced by using a different system of measuring brain activity with a higher spatial resolution, a substitute for gel filling and a better system to copy previous electrode settings. The third factor can be minimized by designing an experiment that validates subject activity and uses relatively short time frames.

It is impossible to be completely objective when interpreting the experiment results. The researcher always has ideas concerning the outcome of the experiments and will probably use these prejudices to some extent when looking at the data and searching for relations.

Practical notes
These experiments have been conducted with the help and experience of Pavel Hrubeš. He also had ideas concerning the *design* of the experiment which differed from my own. That's why some of the experiment tasks may seem of little out of place in light of the experimental design approach stated in section 4.3.1.

Initially the idea was to create a communication system from the combination of BCI with the eye-tracker. However soon it turned out that superior systems based solely on eye-tracking for control are available and therefore the focus was shifted to more fundamental BCI research. For more on this topic see section 5.2.2.

9.3.2. Recommendations

The recommendations made here are both aimed at the practical performance of EEG experiments as well as the design of future experiments and data analysis.

EEG experiment performance recommendations:

- Increase the spatial resolution by using an EEG-cap containing more electrodes.
- Minimize artifacts by muscle movement during experiment.
- Use EMG to record any limb muscle movement and ECG to measure eye-movement.
- Use non head mounted ground and reference point, preferable mastoid bone or earlobe.

- Use absolutely quiet experiment environment, ridden of any distractions.
- Use different system than gel for voltage conductance.
- Try to place the cap in exactly the same position over sessions.
- Try to create similar environment for every session: same posture subject, same room temperature etc.

Experimental design recommendations:

- Construct experiment to obtain lots of short time frames with activity rather than a few very long ones, since this will increase the quality of the data; the amount of trials with correct activity will be higher. In practice it is hard to fully concentrate on a task for an extended period of time.
- Use methods that allow validation of the subject's performance of the experiment, in that the subject is actually performing the task at hand. For instance in the mental rotation task, ask the subject to rotate a complex object and afterwards question the subject to validate his or her rotation.
- Focus on tasks that have a theoretically fixed location, like for instance the motor imagery.

EEG data analysis recommendations:

- Avoid strong noise and outliers in the data
- Know in advance what kind of activity is searched for in the brain and also where.

9.3.3. Conclusion

Differences are found, however it cannot be said for certain how significant these differences are. It is nevertheless very encouraging to find such results despite the problems encountered in the voluntary tasks. In the evoked tasks no difference where found at all, this is probably due to the fact that the presented objects do not create distinctive brain states.

Review of the goals stated at the beginning of this chapter.
The first goal was to obtain knowledge and experience about EEG measurement. Errors have been made, but this is simply inherent to research.

The second goal was to generate data for the analysis and construction of the EEG workbench. More than enough data has been produced for this purpose and this made the construction of the workbench possible.

The third goal was to analysis the experiments and by that verifying whether or not the next BCI step is possible and moreover to make and acquire invaluable recommendations for future experiments, both practical and in design. Part of the last goal was to continue in creating a working BCI, this indeed is to far fetched to complete in light of this master thesis and the work that has been done. This will be an issue of future research.

It is always remarkable how much time and effort one must put into the start of a new subject matter. What at a later stage appears to be rather simple can be quite a hassle to solve at first. Some of the mistakes that were made which could perhaps have been avoided, however these are off course inherent to new research.

The EEG equipment offers a spatial resolution which is too low for use of localization of activity; it can be used to identify larger scale phenomena in the brain, like for instance microsleeps, fatigue and arousal. This is where the equipment is used for in research at the technical university of Prague.

Chapter 10
10. Conclusion & recommendations

This final chapter will answer the research questions and cover the objectives as stated in chapter 1 and discuss the various problems and issues encountered during the process, as well as make recommendation concerning future research.

10.1. *Conclusion & discussion*

The overarching goal of this thesis is to make start in the field of BCI and setup of suitable environment for further research. Is this goal accomplished?

Research question 1 *What is the current state of BCI?*

First considerable amounts of knowledge have been gained concerning the extensive field of BCI. This knowledge is concentrated in chapter 2. The models used in the workbench are further elaborated in the following chapters 6 and 7. This was a necessity to continue with the experiments and construction of the workbench.

This satisfies *Objective 1:* Obtaining necessary and sufficient theoretical knowledge concerning Brain Computer Interfaces.

Research question 2 *How to construct and perform an EEG-based BCI experiment?*

The EEG equipment was installed, set up and tested. EEG-based BCI experiments have been designed and conducted (see chapter 4). Programs to support and control the experiments have been constructed. Problems were encountered during and after the experiments, some were solved on the spot, others were only identified in a later stage, this is inherent to research.

This satisfies *Objective 2:* Initial measurements and setup of EEG and consequently designing and performing EEG-based BCI experiments.

Research question 3 *How to process and analyze the data?*

An analysis workbench has been designed, implemented for the purpose of structured analysis and support programs have been constructed and implemented as well; this is covered in a part of chapter 5 and chapter 6 and 7.

This satisfies *Objective 3:* Design and implement an EEG data analysis tool for a structured analysis approach of EEG-based BCI data.

Research question 4 *How good is the performance of the data analysis and how useful are the results?*

The performance of the E-Brain has been established and validated by testing covered in chapter 8. These tests show that the E-Brain is capable of finding similar patterns in the *third party* data as other research groups and stated by the hypothesis.

Due to the mediocre quality of the EEG data from the conducted experiments, the results from the analysis are less convincing than those of chapter 8. Despite the shortcomings of the data quality, differences were found complying with the hypothesis concerning the performed task of subject 1. This is covered in chapter 9.

This satisfies *Objective 4:* Evaluation of BCI experiments and EEG workbench.

Research question 5 *How to proceed in future experiments and BCI research?*

The experience obtained from this master thesis and the completion of *objective 4*, enables to give indications on further research. In the recommendation-sections 9.3.2 and 10.2 indications are given on what pitfalls to avoid and in which direction to continue with the current research.

It can be stated that all the requirements for this thesis have been met. However there remain some issues to be discussed.

The workbench
The E-Brain meets its requirements; this is covered in section 8.3. The only issue encountered is the usefulness of the LD algorithm in the analysis phase of the workbench. Since it produces ambiguous results it is recommended not to use this method to reach conclusions.

It should be relatively easy to extend the E-Brain with a classifier, since every aspect required up to this point is available to create an offline BCI.

The experiment
The data quality of the performed EEG experiments appears to be mediocre at best, caused by several reasons. The two most important being of practical nature: the incorrect placement of ground and reference electrode and the poor quality offered by the EEG equipment used. This results in a poor spatial resolution, location shifts between sessions and different resistance levels in the electrodes caused by gel-filling, see for extended description: *Experiment evaluation* (section 9.3).

By using an EEG measurement system more focussed on BCI research many of the causes for the poor data quality can be countered. Further more good thought must be given to the experimental design. The idea of the performed experiments was to create a voluntary asynchronous BCI, however not time-locking the events increases the complexity of the system to an higher level which does not contribute that much to the research at hand. For BCI research performed in light of this master thesis however this does not matter. It will become important in constructing a working online BCI system in the future.

Unfortunately there was no time to construct a working BCI, this was not a objective of the thesis in the first place, however it would have been extremely insightful to have some form of a online BCI to learn even more on BCI systems.

An encouraging fact is that even though many problems were encountered during the construction of this thesis, already significant differences were found between tasks of subject one (section 9.3.3).

All the objectives that have been stated in the first chapter concerning this thesis have been achieved. By completing these objectives all the research questions have been answered. And thereby the start of a BCI research line at the TU Delft and a suitable environment for further research is achieved.

10.2. *Recommendations & future work*

In order to learn from this thesis and not fall into the same pitfalls, this section covers recommendations for future research. Most of the practical recommendations concerning experiment execution are covered in the experiment discussion (section 9.3.1 and 9.3.2).
The BCI experiment must be designed in such a way that it maximizes the chance that the subject is actually performing the correct task. This can be done by:

- Returning some kind of feedback to the user concerning his or her actions. This does require an online BCI system to be available.

- Using very short time periods for the intended activity; it is easier for the subject to stay concentrate and moreover more trials can be performed during a single session. The time periods should maximally be stretched up to 3 or 4 seconds.
- Using some sort of verification; the subject has to answer some sort of question regarding the performed task to verify that he or she actually performed the task.

All these recommendations will contribute to gaining a higher degree of knowledge concerning what is concealed within the measured data.

The direction of BCI in which to proceed is to:

- Impose the main load of the learning task on the machine. Improving the learning capabilities of the machine should have higher importance over improving those of the user [39].
- Work with event related potentials to get a more active (responsive) system by time-locking the system. However evoked potentials are generally less desired in the face of self control. It is advised to start with synchronous BCI. And when enough knowledge and experience is present continue with asynchronous BCI.

The E-Brain should be extended to any other modalities and into the classification phase. Recommendations concerning this next step:

- It is very important to avoid over fitting of highly dimensional data when using classifiers [39].
- Using classification to create a measure for the quality of the data.
- Not only the feature selection, but also the pre-processing step has to be taken into account within the cross-validation of the classification [39].

The next step after this thesis must be to implement a classifier in the E-brain, to construct an offline BCI and eventually using this information to create an online (real-time) BCI.

Most of these recommendations are specifically aimed at EEG-based BCI and try to convey a direction in which to proceed in the future. It should be kept in mind that research is to a certain degree also trial-and-error, meaning that errors are made and these opportunities should be used to review the current situation and search for new direction to proceed with this information. So even though this thesis provides a way to continue in all the aspects, other ways to proceed could be just as promising.

The future

Despite the recommendations mentioned above it would be nice to gaze into the future and draw a picture of the ideal (within reason and practicality) BCI. This BCI would react on the user's will (asynchronously) and would require only very short commands (time frames of brain activity), it would be universally used after a minimal training period and should be as easily applied as putting on a pair of glasses.

Science fiction?

The ultimate and futuristic (and unrealistic) BCI would on top of this not only have two-way communication; it would also be able to extract actual intentions and thoughts. And allow control just as easily and unconsciously as offered by muscle movement, fitting seamlessly in to the human body.

Bibliography

[1] H. Berger, "Uber das Electrenkephalogramm des Menchen." in *Arch f.Psychiat Nervenkr*, vol. 87, pp. 527–570, 1929.

[2] G. Birch, S. Mason, J. Borisoff, "Current Trends in Brain-Computer Interface Research at the Neil Squire Foundation." in *IEEE transactions on neural systems and rehabilitation engineering,* vol. 11, no. 2, June 2003.

[3] G. Blanchard, B. Blankertz, "BCI Competition 2003: Data Set IIa: Spatial Patterns of Self-Controlled Brain Rhythm Modulations." in *IEEE transactions on biomedical engineering*, 100-104, 2003.

[4] B. Blankertz, K. Müller, G. Curio, et al, "The BCI Competition 2003: Progress and Perspectives in Detection and Discrimination of EEG Single Trials." in *IEEE transactions on biomedical engineering,* vol. 03, no. 4, 2004.

[5] B. Blankertz, G. Dornhege, M. Krauledat, K. Müller, G. Curio, "The Berlin Brain-Computer Interface: Report from the Feedback Sessions" in *FIRST report*, January 2005.

[6] B. Bruegge, A. Dutoit, "Object-Oriented Software Engineering", 2002.

[7] M. Cheng, W. Jia, X. Gao, S. Gao, F. Yang, "Mu rhythm-based cursor control: an offline analysis." in *Clinical Neurophysiology,* 115, 745–751, 2004.

[8] F. Cincotti, A. Scipione, et al, "Comparison of Different Feature Classifiers for Brain Computer Interfaces." in *Proceedings of the 1st International IEEE EMBS Conference on Neural Engineering,*Capri Island, Italy, 20-22 March 2003.

[9] F. Cincotti, D. Mattia et al, "The Use of EEG Modifications Due to Motor Imagery for Brain–Computer Interfaces." in *IEEE transactions on neural systems and rehabilitation engineering,* vol. 11, no. 2, June 2003.

[10] L. Citi, R. Poli, C. Cinel, F. Sepulveda, "Feature Selection and Classification in Brain Computer Interfaces by a Genetic Algorithm." in *Proceedings of the Genetic and Evolutionary Computation Conference,* Seatle, 2004.

[11] M. Cohen, S. Kosslyn, H. Breiter, G. DiGirolamo, W. Thompson et al, "Changes in Cortical Activity During Mental Rotation: *A mapping study using functional magnetic resonance imaging"* in *Brain* 119:89-100, 1996.

[12] S. Coyle, T. Ward, C. Markham, G. McDarby, On the suitability of near-infrared (NIR) systems for next-generation brain–computer interfaces." in *Physiological measurement,* 25, 815–822, 2004.

[13] G. Dornhege, B. Blankertz, G. Curio, "Speeding up classification of multi-channel Brain-Computer Interfaces: Common spatial patterns for slow cortical potentials." in *Neural Information Processing Systems Foundation*, 2003.

[14] G. Dornhege, B. Blankertz, G. Curio, K. Müller, "Increase information transfer rates in BCI by CSP extension to multi-class." in *Neural Information Processing Systems Foundation,* 2003.

[15] G. Dornhege, B. Blankertz, G. Curio, K. Müller, "Boosting bit rates in non-invasive EEG single-trial classifications by feature combination and multi-class paradigms." in *IEEE transactions on biomedical engineering,* vol. 03, no. 6, 2004.

[16] G. Fabiani, D. McFarland, J. Wolpaw, G. Pfurtscheller, "Conversion of EEG Activity into Cursor Movement by a Brain–Computer Interface (BCI)." in *IEEE transactions on neural systems and rehabilitation engineering,* vol. 12, no. 3, September 2004.

[17] R. Fulbright, D. Molfese, A. Stevens et al, "Cerebral Activation during Multiplication: A Functional MR Imaging Study of Number Processing" in *AJNR Am J Neuroradiology* 21:1048–1054, June/July 2000.

[18] M. Greiner, D. Pfeiffer, R. Smith, "Principles and practical application of the receiver-operating characteristic analysis for diagnostic tests" in *Preventive Veterinary Medicine,* 45, 23-41, 2000.

[19] C. Guger, H. Ramoser, G. Pfurtscheller, "Real-time EEG Analysis with Subject-Specific Spatial Patterns for a Brain-Computer Interface (BCI)." in *IEEE transactions on rehabilitation engineering,* vol. 8, no.4, December 2000.

[20] C. Guger, G. Edlinger, W. Harkam, G. Pfurtscheller, "How Many People are Able to Operate an EEG-Based Brain-Computer Interface (BCI)?" in *IEEE transactions on neural systems and rehabilitation engineering,* vol. 11, no. 2, June 2003.

[21] T. Hanakawa, I. Immisch, K. Toma, M. Dimyan, P. van Gelderen, M. Halettie, "Functional Properties of Brain Areas Associated with Motor Execution and Imagery" in *J Neurophysiology* 89: 989–1002, 2003.

[22] J. Hanley, B. McNeil, "The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve" in *Radiology* 143: 29-36, April 1982.

[23] I. Harris, G. Egan, C. Sonkkila, H. Tochon et al, "Selective right parietal lobe activation during mental rotation" in *Brain* 123:65-73, 2000.

[24] T. Hinterberger, A. Kubler, J. Kaiser, N. Birbaumer, "A brain–computer interface (BCI) for the locked-in: comparison of different EEG classifications for the thought translation device." in *Clinical Neurophysiology,* 114, 416–425, 2003.

[25] T. Hinterberger, N. Weiskopf, N. Birbaumer, "An EEG-Driven Brain-Computer Interface Combined With Functional Magnetic Resonance Imaging." in *IEEE transactions on neural systems and rehabilitation engineering,* vol. 51, no. 6, June 2004.

[26] M. Hoeve, B. van der Zwaag, M. van Burik, K. Slump, R. Jones, "Detecting Epileptic Seizure Activity in the EEG by Independent Component Analysis" from *ProRISC (Program for Research on Integrated Systems and Circuits)*, 373-377, 2004.

[27] N. Huan, R. Palaniappan, "Neural network classification of autoregressive features from electroencephalogram signals for brain–computer interface design." in *Journal Neural Engineering*, 1, 142–150, 2004.

[28] A. Hyvärinen, "Survey on Independent Component Analysis" in *Neural Computing Surveys,* Vol. 2, pp. 94-128, 1999.

[29] A. Hyvärinen, E. Oja, "Independent Component Analysis: Algorithms and Applications" in *Neural Networks,* 13 (4-5):411-430, 2000.

[30] T. Jung, C. Humphries, T. Lee, S. Makeig, M. McKeown, V. Iragui, T. Sejnowski, "Extended ICA removes artifacts from electroencephalographic recordings" in *Advances in Neural Information Processing systems,* vol. 10, 894-900, 1998.

[31] T. Jung, S. Makeig, M. Mckeown, A. Bell, T. Lee, T. Sejnowski, "Imaging Brain Dynamics Using Independent Component Analysis" in *Proceedings of the IEEE,* vol. 89, no. 7, July 2001.

[32] S. Kelly, D. Burke, P. de Chazal, R. Reilly, "Parametric Models and Spectral analysis for Classification in Brain-Computer Interfaces." in *Proceedings of 14th International Conference on Digital Signal Processing*, Greece, July 2002.

[33] T. Lal, M. Schröder, T. Hinterberger, J. Weston, M. Bogdan, N. Birbaumer, B. Schölkopf, "Support Vector Channel Selection in BCI" in *IEEE transactions on biomedical engineering,* vol. 51, no.6, June 2004.

[34] A. Luo, A. Gerson, P. Sajda, "Comparison of supervised and unsupervised linear methods for recovering task-relevant activity in EEG." in *IEEE*, 1012-1015, 2004.

[35] S. Mason, G. Birch, "A General Framework for Brain–Computer Interface Design." in *IEEE transactions on neural systems and rehabilitation engineering,* vol. 11, no.1, March 2003.

[36] D. McFarland, L. McCane, S. David, J. Wolpaw, "Spatial filter selection for EEG-based communication" in Electroencephalography and clinical Neurophysiology, 103, 386-394, 1997.

[37] D. McFarland, W. Sarnacki, J. Wolpaw, "Brain-computer interface (BCI) operation: optimizing information transfer rates" in *Biological Psychology* 63, 237-251, 2003.

[38] K-R. Müller, S. Mika, G. Ratsch, K. Tsuda, B. Scholkopf, "Introduction to Kernel-based learning algorithms" in *IEEE transaction on neural networks*, vol. 12, no. 2, March 2001.

[39] K.-R. Müller, M. Krauledat, G. Dornhege, G. Curio, B. Blankertz, "Machine learning techniques for Brain-Computer Interfaces." in *Biomedical. Technology,* 49(1):11-22, 2004.

[40] M. Novak, J. Faber, Z. Votruba, "*Problems Reliability in Interactions between Human Subjects and Artificial Systems"*, 2004.

[41] M. Novak, "*Neurodynamic and neuroinfoamatics studies"*, 2005.

[42] L. Parra, C. Alvino, A. Tang et al, "*Linear Spatial Integration for Single-Trial Detection in Encephalography*" in *NeuroImage* 17**,** 223–230, 2002.

[43] L. Parra, C. Alvino, A. Tang, B. Pearlmutter et al, "Single-trial detection in EEG and MEG: Keeping it linear" in *Neurocomputing,* 52–54, 177-183, 2003.

[44] L. Parra, C. Spence, A. Gerson, P. Sajda, "Recipes for the linear analysis of EEG" in *NeuroImage*, 28 , 326-341, 2005.

[45] G. Pfurtscheller, C Neuper, "*Motor imagery activates primary sensorimotor area in humans"* in *Neuroscience letters* 239, 65-68, 1997.

[46] G. Pfurtscheller, C Neuper, "*Motor Imagery and Direct Brain–Computer Communication"* in *proceedings of the IEEE*, vol 89, no. 7, July 2001.

[47] G. Pfurtscheller, C. Neuper, G. R. Müller, B. Obermaier et al, "*Graz-BCI: state of the art and clinical applications*" in *IEEE transactions on neural systems and rehabilitation engineering*, vol. 11, no. 2, 2003.

[48] J. Pineda, D. Silverman, A. Vankov, J. Hestenes, "Learning to Control Brain Rhythms: Making a Brain-Computer Interface Possible. " in *IEEE transactions on neural systems and rehabilitation engineering,* vol. 11, no. 2, June 2003.

[49] J.C. Principe, N.R. Euliano and W.C. Lefebvre, "Neural and adaptive system, fundamentals through simulations", 1999.

[50] L. Qin, L. Ding, B. He, "Motor imagery classification by means of source analysis for brain–computer interface applications." in *Journal Neural Engineering*, 1, 135–141, 2004.

[51] H. Ramoser, J. Müller-Gerking, G. Pfurtscheller, "Optimal Spatial Filtering of Single Trial EEG During Imagined Hand Movement." in *IEEE transactions on rehabilitation engineering*, vol. 8, no. 4, December 2000.

[52] G. Schalk, D. McFarland, T. Hinterberger, N. Birbaumer, J.Wolpaw, "BCI2000: A General-Purpose Brain-Computer Interface (BCI) System" in *IEEE transactions on biomedical engineering,* vol. 51, no. 6, June 2004.

[53] A. Schoegl, K. Lugger, G. Pfurtscheller, *"*Using autoregressive parameters for brain-computer interface experiment.*"* in *Proceedings - 19th International Conference - IEEE/EMBS* Chicago USA, October 30 - November 2 1997.

[54] B. Storey, "Computing Fourier Series and Power Spectrum with MATLAB" in *Olin college of engineering,* fall 2002.

[55] P. Sykacek, S. Roberts, M. Stoke, "Adaptive BCI Based on Variational Bayesian Kalman Filtering: An Empirical Evaluation." in *IEEE transactions on neural systems and rehabilitation engineering,* vol. 51, no. 5, May 2004.

[56] K. Tavakolian, A. Nasrabadi, S. Rezaei, "Selecting better EEG channels for classification of mental tasks." in *IEEE*, 537-540, ISCAS, 2004.

[57] M. Teplan, "Fundamentals of EEG measurement", in *Measurement Science* Review, Vol. 2, Sec. 2, 2002.

[58] C. Thomaz, D. Gillies, R. Feitosa, "A New Quadratic Classifier applied to Biometric Recognition" in *Proceedings of the International ECCV 2002 Workshop Copenhagen on Biometric Authentication,* 2002.

[59] O. Tomori, M. Moore, "The Neurally Controllable Internet Browser (BrainBrowser)" in *CHI* 2003, April 5-10 2003.

[60] T. Vaughan, W. Heetderks et al, "Brain–Computer Interface Technology: A Review of the Second International Meeting" in *IEEE transactions on neural systems and rehabilitation engineering,* vol. 11, no. 2, June 2003.

[61] R. Vigário, J. Särelä, V. Jousmäki, M. Hämäläinen, E. Oja, "Independent Component Approach to the Analysis of EEG and MEG Recordings" in *IEEE transactions on biomedical* engineering, vol. 57, no 5, May 2000.

[62] M. Vinther, "Independent Component Analysis of Evoked Potentials in EEG", *DTU,* 8 December 2002.

[63] T. Wang, B. He, "An efficient rhythmic component expression and weighting synthesis strategy for classifying motor imagery EEG in a brain–computer interface." in *Journal Neural Engineering*, 1, 1–7, 2004.

[64] J. Wolpaw, N. Birbaumer et al, "Brain–Computer Interface Technology: A Review of the First International Meeting" *in IEEE transactions on rehabilitation engineering*, vol. 8, no. 2, June 2000.

[65] V. Zarzoso, P. Comon, "*Block deflation ICA algorithms"* in *IEEE International conference on Acoustics, Speech and signal Processing (ICAP)*, May 14-19 2006.

[66] L. Zhukov, D. Weinstein, C. Johnson, "Independent Component Analysis for EEG Source Localization" in *IEEE engineering in medicine and biology*, May/June 2000.

Websites

[67] "*www.alsa.org*", the site of the American ALS association.

[68] "*www.strokecenter.org*", the site of the Internet stroke centre, a non-profit educational service.

[69] "www.ucp.org", the site of the United Cerebral Palsy association.

Appendix A - Use cases

This appendix covers the use cases not displayed in chapter 6.

E-Brain use case diagram
E-Brain (figure A.1) represents the user interaction on the highest level with E-Brain and has the purpose to show the main options for the user. Each of the use cases in *E-Brain* will be explained in a detail use case diagram.

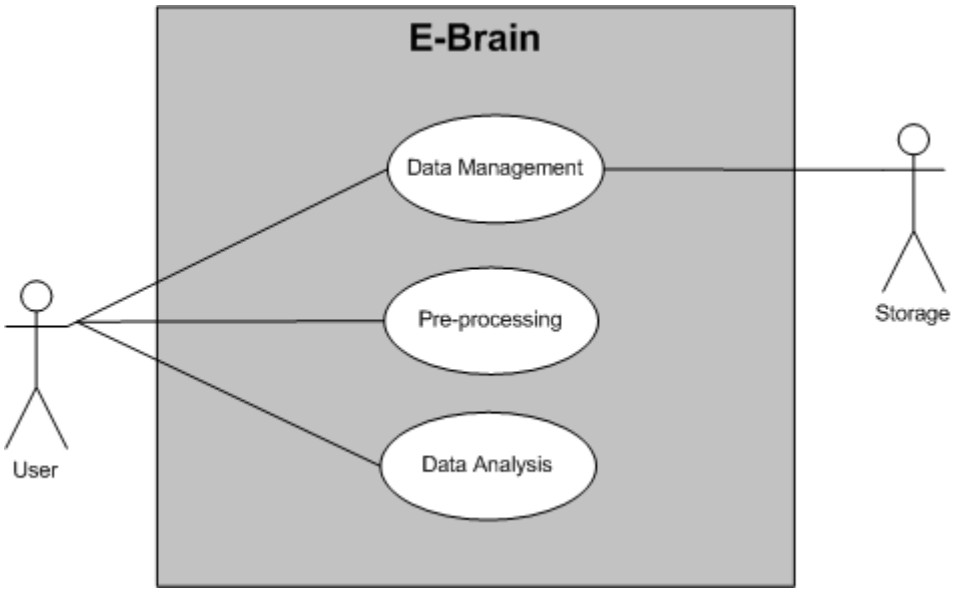


Figure A.1 Use case diagram E-Brain.

Use cases:

- *Data Management*
  - Description: this case covers every aspect which has to do with data handling.
  - Actors: *User* and *Storage.*
- *Pre-processing*
  - Description: this case covers every aspect which has to do with pre-processing the data.
  - Actors: *User.*
- *Data Analysis*
  - Description: this case covers every aspect which has to do with the data analysis.
  - Actors: *User.*

Data Management use case diagram
This case (see figure A.2) identifies all the interaction available for the actors concerning retrieving, managing and saving the data used in E-Brain.

Use cases:

- *Retrieve Data*
  - Description: The *User* can select data from either the database or a file which will be loaded into E-Brain.
  - Extensions:
    - *Database*, the *User* retrieves data from the database.
    - *File,* the *User* retrieves data from a selected file.
  - Actors: *User* and *Storage.*

- *Manage Data*
  - Description: The *User* has the abilities to delete any part of the datastructure or rename datasets.
  - Uses:
    - *Delete Data*, the *User* has the option to delete datasets, tasks, trials and sessions.
    - *Rename Data*, the *User* has the ability to rename any dataset.
  - Actors: *User.*
- *Save Data*
  - Description: The *User* has the ability to save the current datasets.
  - Actors: *User* and *Storage.*

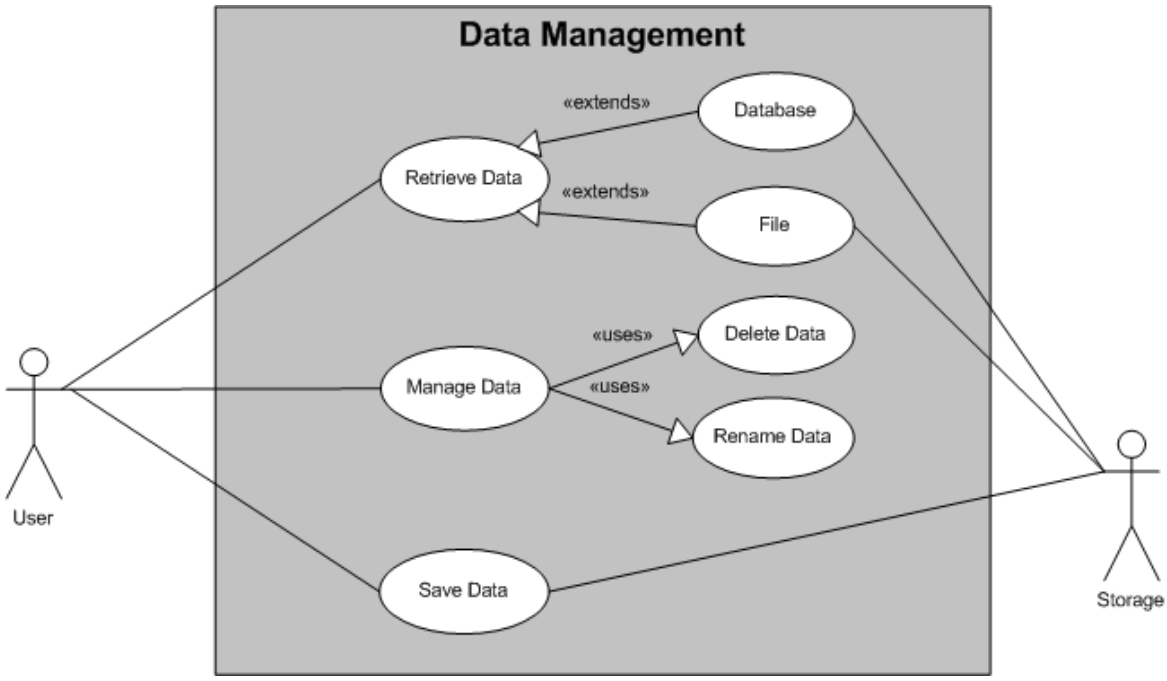


Figure A.2 Use case diagram Data Management.

Pre-processing use case diagram
This case (see figure A.3) identifies all the interaction available for the actors concerning the pre-processing of the data before the actual analysis.

Use cases:

- *Basic Pre-processing*
  - Description: offers three basic pre-processing methods to the *User* to apply to the raw data.
  - Actors: *User.*
  - Uses:
    - Re-reference: the *User* performs a re-referencing filter on the data, either CAR or small Laplacian.
    - Bandpass Filter: the *User* performs a bandpass filter on the data in a specified frequency range.
    - Baseline Correction: the *User* performs a baseline correction on the data.
    - Data Selection.
    - Save Data.
- *Inspect EEG*

- Description: the *User* can view any part of the data by specifying parameters and zooming using mouse input. Furthermore the *User* can delete any part of the data and save changes to the data.
- Actors: *User.*
- Uses: Data Selection & Save Data.

- *Perform ICA*
  - Description: the *User* can remove artifacts from the data using the ICA algorithm. After selecting desired data the Independent Components are generated and the undesired components can be selected for removal. The *User* can reconstruct the signal without these components.
  - Actors: *User.*
  - Uses: Data Selection & Save Data.
- *Data segmentation*
  - Description: the *User* can segmentize the selected data into specified pieces with specified overlap, after which the data is saved into the datastructure.
  - Actors: *User.*
  - Uses: Data Selection & Save Data.
- *Data Selection*
  - Description: Select specific data from *Storage* for pre-processing.
  - Actors: *Storage.*
- *Save Data*
  - Description: Save the data to the datastructure in *Storage*.
  - Actors: *Storage.*

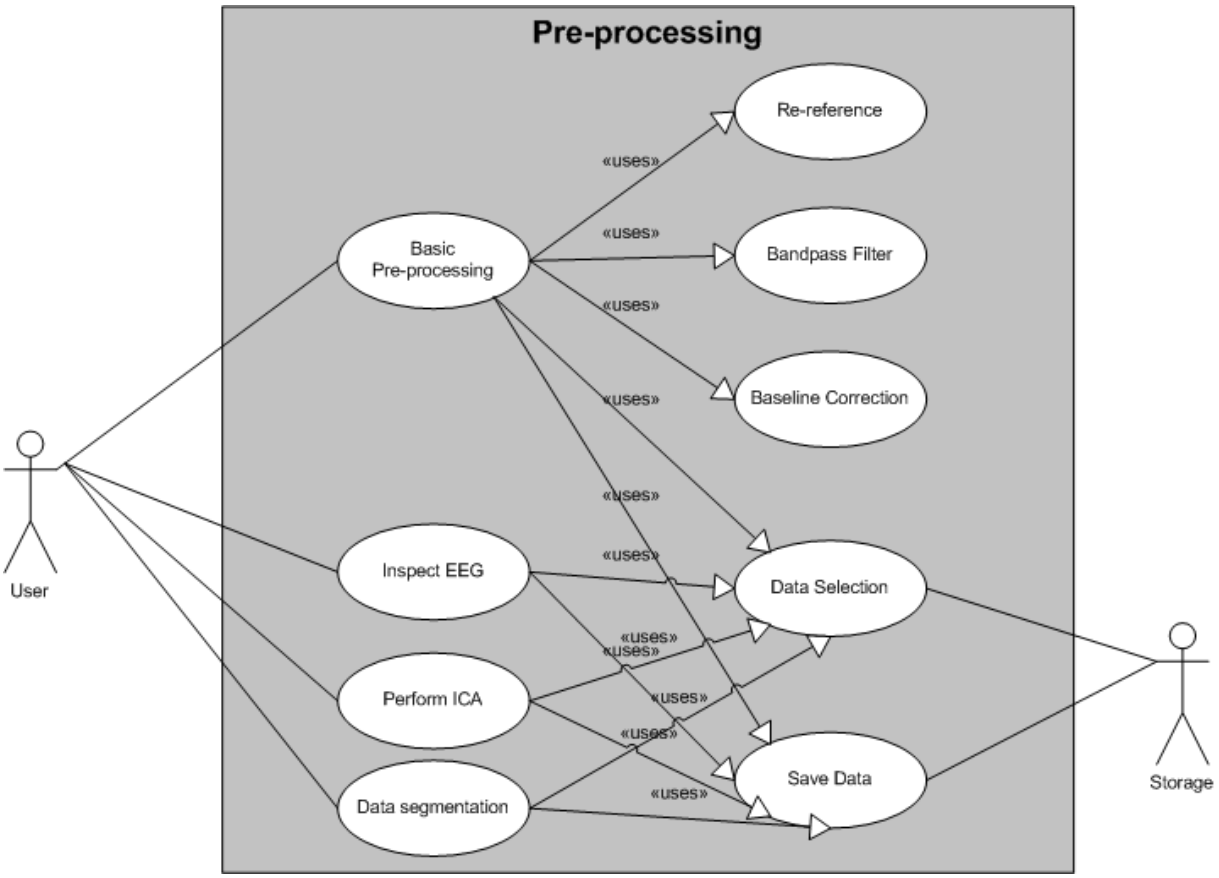
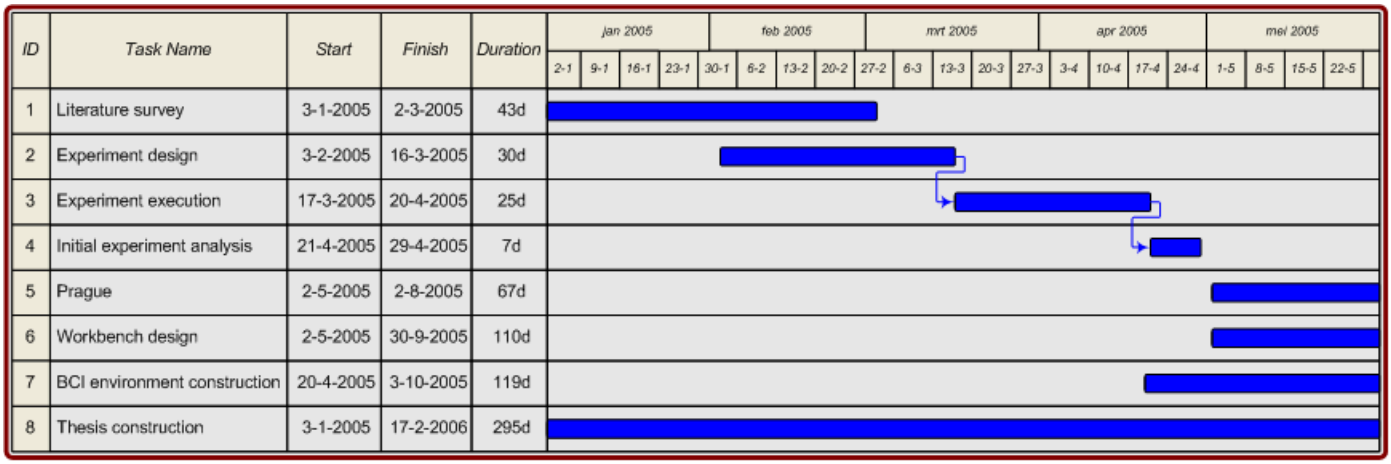


Figure A.3 Use case diagram Pre-processing.

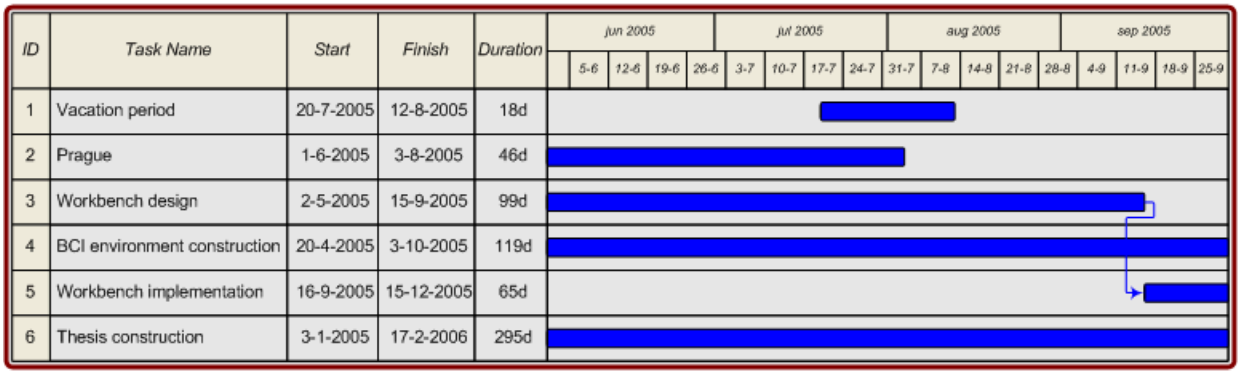
Appendix B – Planning

The Gant-chart in figure B.1 shows the time course of the various activities that have taken place during this master thesis. It must be noted that during the entire period approximately 1 to 1.5 days per week was spent for a side job. This amounts to a total of about three months. This is not displayed in the figure.

ID	Task Name	Start	Finish	Duration
1	Literature survey	3-1-2005	2-3-2005	43d
2	Experiment design	3-2-2005	16-3-2005	30d
3	Experiment execution	17-3-2005	20-4-2005	25d
4	Initial experiment analysis	21-4-2005	29-4-2005	7d
5	Prague	2-5-2005	2-8-2005	67d
6	Workbench design	2-5-2005	30-9-2005	110d
7	BCI environment construction	20-4-2005	3-10-2005	119d
8	Thesis construction	3-1-2005	17-2-2006	295d



ID	Task Name	Start	Finish	Duration
1	Vacation period	20-7-2005	12-8-2005	18d
2	Prague	1-6-2005	3-8-2005	46d
3	Workbench design	2-5-2005	15-9-2005	99d
4	BCI environment construction	20-4-2005	3-10-2005	119d
5	Workbench implementation	16-9-2005	15-12-2005	65d
6	Thesis construction	3-1-2005	17-2-2006	295d



ID	Task Name	Start	Finish	Duration
1	BCI environment construction	8-8-2005	31-10-2005	61d
2	Workbench implementation	3-10-2005	11-1-2006	73d
3	Workbench testing	12-1-2006	26-1-2006	11d
4	Experiment analysis	27-1-2006	10-2-2006	11d
5	Thesis construction	8-8-2005	17-2-2006	140d
6	Finish & Presentation	20-2-2006	15-3-2006	18d

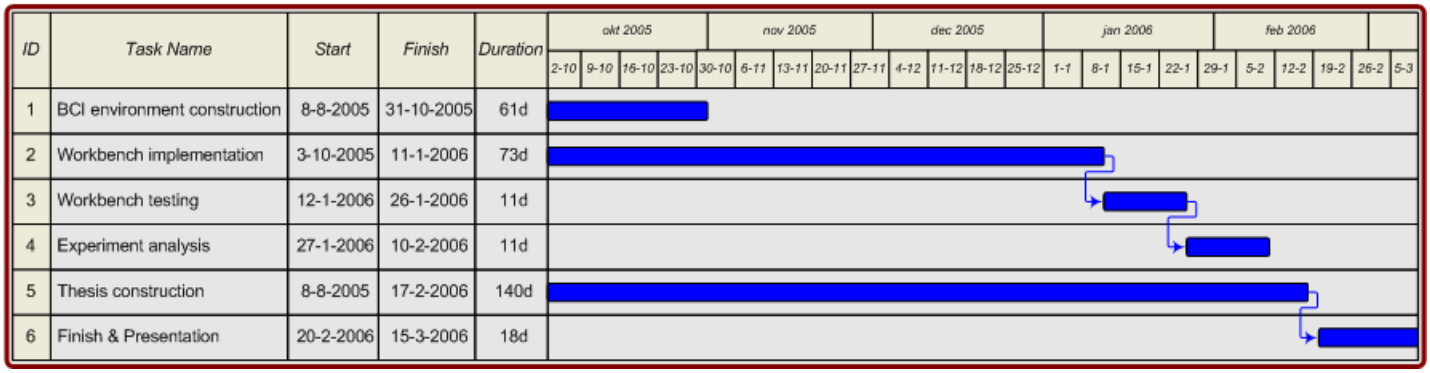


Figure B.1 Time line of master thesis.

Furthermore the task 'BCI environment construction' is displayed continuously over several months, however work has not been conducted continuously over that period. This is merely a simplification of the figure. The same goes for the vacation period, which is actually split up in several parts.