Johannes Steininger, BSc.

Data Transfer Methods for Calibration Time Reduction in Sensorymotor Rhythm based Brain-Computer Interfaces

MASTER'S THESIS

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme: Biomedical Engineering

submitted to

Graz University of Technology

Supervisor

Univ.-Prof. Dipl.-Ing. Dr. techn. Gernot R. Müller-Putz

Institute of Neural Engineering

Co-Supervisor
Dipl.-Ing. David Steyrl

Graz, July 2017

AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master‘s thesis.

Date

Signature

Acknowledgements

At this point, I would like to thank all the people who supported me kindly and answered all my questions with great pleasure over the last nine months. I am very thankful for this marvellous company and every single moment.

Foremost, I would like to thank Dipl.-Ing. David Steyrl. Without him, this thesis would not have been possible. He brought up the idea which was later the basis of this thesis and introduced me kindly into this field of research. Whenever I needed help, he had an open ear for my questions and did not hesitate for a second, to provide his assistance.

I also want to thank Dr. Gernot R. Müller-Putz, Head of the Institute of Neural Engineering, for providing the necessary hardware and support, so that I was able to conduct all of my experiments.

With all my love, I want to express my sincere gratitude to my partner Carina, whose comprehension, patience, and love was the support that I needed through all ups and downs of this work.

At last, I want to thank my parents, siblings, friends and all the people who helped me during my final months as a student.

Abstract

Brain-Computer Interfaces (BCIs) extract information from brain signals of users, to control applications or devices. These signals tend to change over time and furthermore depend on the actual condition of the user. Therefore, a calibration phase to adjust the *BCI* system to the current user is unavoidable. This calibration process is often found to be tedious and lengthy by the users. To address this issue, one option is to incorporate data from previous sessions into the calibration process of the *BCI*. However, it is not yet clear, how to include previous data into the process. To gain more insight into this topic, different signal processing methods, along with machine learning and pattern recognition algorithms, were examined in offline simulations. One method is therefore utilized as an advanced bandpass filter and all other methods act as spatial filters and are modifications of the popular Common Spatial Patterns (CSP) algorithm. Among the tested methods, one was selected to be used in an online *BCI* study. In this study, two sessions on two different days were executed. The first session was intended as reference. On the second session, half of the calibration data was recorded and the other half was incorporated from the first session to reduce the calibration time. The performance in terms of the classification accuracy was determined in both sessions. In the offline simulation, the Across Session/Subject Stationarity divCSP (divCSP-AS) was among the best performing methods (82.08% peak-accuracy, 71.34% mean-accuracy) and was therefore selected to be integrated in the online *BCI*. Although the online *BCI* system of the second session was only calibrated in half of the time as in the first session, a peak-accuracy of 82.58% along with a mean-accuracy of 74.55% was achieved in eleven participants. The results suggest, that it is possible to use data from previous sessions to reduce the calibration time of future sessions.

Keywords: Brain-Computer interface (BCI), electroencephalogram (EEG), calibration time reduction, transfer learning, machine learning

Kurzfassung

Gehirn-Computer Schnittstellen (*BCIs*) extrahieren Informationen aus Gehirnsignalen von Nutzern, um Anwendungen oder Geräte zu steuern. Die Signale variieren dabei und sind abhängig von der Zeit und dem aktuellen Zustand des Benutzers. Es ist deshalb notwendig, eine Kalibrierung des *BCIs* durchzuführen. Dies wird dabei oft als mühsam und langwierig beschrieben. Eine Option, dem entgegenzuwirken, ist, bereits aufgenommene Daten von früheren Messungen in den Prozess zu integrieren. Für diesen Zweck wurden Methoden der Signalverarbeitung, sowie des maschinellen Lernens und der Mustererkennung in einer Computer-Simulation untersucht. Eine Methode wird dabei als adaptiver Bandpass eingesetzt. Alle weiteren Methoden sind Modifikationen des bekannten Common Spatial Patterns (CSP) Algorithmus. Eine dieser Methoden wurde anschließend ausgewählt und in einer *BCI* Anwendung integriert. Es wurde eine Studie mit zwei Messungen, an zwei Tagen durchgeführt. Die erste Messung diente dabei als Referenz. Um die Kalibrierungszeit zu verkürzen, wurden in der zweiten Messung nur halb so viele Daten wie in der ersten Messung aufgezeichnet und die zweite Hälfte aus der ersten Messung mitgenommen. Die Funktionsweise des Systems wurde dabei in beiden Messungen anhand der Klassifikationsgenauigkeit ermittelt. Die Ergebnisse der Computer-Simulation zeigten, dass die Across Session/Subject Stationarity divCSP (divCSP-AS) unter den besten Methoden rangiert (82.08% maximal-Genauigkeit, 71.34% mittlere-Genauigkeit) weshalb diese für die *BCI* Studie ausgewählt wurde. Obwohl das System der zweiten Messung in der Hälfte der Zeit der ersten Messung kalibriert wurde, wurden 82.58% maximal-Genauigkeit sowie 74.55% mittlere-Genauigkeit mit elf Probanden erzielt. Die Resultate zeigen, dass es möglich ist, bereits aufgenommene Daten zu verwenden, um die Kalibrierungszeit künftiger Messungen zu verkürzen.

Stichwörter: Gehirn-Computer Schnittstelle (BCI), Elektroenzephalogramm (EEG), Kalibrierungszeit Reduktion, transfer Lernen, maschinelles Lernen

Contents

List of Figures

List of Tables

List of Acronyms

ALS	Amyotrophic Lateral Sclerosis
BCI	Brain-Computer Interface
CSP	Common Spatial Patterns
divCSP	Divergence Common Spatial Patterns
divCSP-AS	Across Session/Subject Stationarity divCSP
divCSP-BS	Between Session Stationarity divCSP
divCSP-MS	Multi Session/Subject Stationarity divCSP
divCSP-WS	Within Session Stationarity divCSP
EEG	Electroencephalography
EEMD	Ensemble Empirical Mode Decomposition
EMD	Empirical Mode Decomposition
EMG	Electromyography
EOG	Electrooculography
EPSP	Excitatory Postsynaptic Potentials
ERD	Event-Related Desynchronization
ERDS	Event-Related (De-)Synchronization
ERP	Event-Related Potential
ERS	Event-Related Synchronization
ETH Zürich	Eidgenössische Technische Hochschule Zürich
FES	Functional Electric Stimulation
fMRI	functional Magnetic Resonance Imaging
IIR	Infinite Impulse Response
IMF	Intrinsic Mode Function
k-fold CV	k-fold Cross-Validation
LOOCV	Leave-One-Out Cross-Validation
MAF	Moving Average Filter

MEMD	Multivariate Empirical Mode Decomposition
NA-MEMD	Noise Assisted - Multivariate Empirical Mode Decomposition
PCA	Principle Component Analysis
SCI	Spinal Cord Injury
sCSP	Shrinkage Common Spatial Patterns
sLDA	Shrinkage Linear Discriminant Analysis
SMR	Sensorymotor Rhythm
SNR	Signal to Noise Ratio
SSA	Stationary Subspace Analysis
ssCSP	Stationary Subspace Common Spatial Patterns
stationaryCSP	Stationary Common Spatial Patterns
SVM	Support Vector Machine
TU Graz	Technische Universität Graz
WGN	White Gaussian Noise
WT	Whitening Transformation
ZCA	Zero-Phase Components Analysis

1 Introduction

Imagine waking up in the morning after a good night sleep, dreaming of playing ball with friends or going for a walk with your beloved and suddenly it comes back to your mind that you are no longer able to do so. An accident, illness or an unexpected event like a stroke and you could be bound to a wheelchair or a bed for the rest of your life.

Paralysed people suffer from a loss of functionality which can vary, depending on the severity of the disorder. In severe cases, the affected people have no, or very low remaining muscle control and are therefore no longer able to move or communicate, but are still totally aware of their surroundings. People who suffer from this so called *Locked-in-Syndrome* [3] are in need of other communication channels independent of the normal output pathways of the brain.

The advance in cognitive neuroscience and the ability to observe the functionality of the brain with different kinds of technology enables us to interface with the human brain directly. In the last decades, researchers have extended these technologies to develop Brain-Computer Interfaces (BCIs). Those are systems for communication and control that monitor the physiological processes in the brain through the use of sensors. By changing these brain signals voluntarily, the user is able to control different devices via a *BCI* [27] [65]. Researchers around the world are currently working to improve the performance of *BCI* applications in terms of convenience, reliability, accuracy and robustness to ease the life of patients in need.

1.1. Background and Related Work

A *BCI* is a communication device that does not use the normal neuromuscular output pathways of the brain, but accepts commands encoded in neurophysiological signals evoked by the user [65]. Figure 1.1 depicts the main components of a *BCI*. The *signal acquisition* block deals with the recording of the brain signals. The *signal processing* block incorporates time-frequency- as well as spatial-filtering methods to increase the performance of the system. The *feature extraction* block transforms the processed brain signals of the user into more distinguishable data values, which carry the important informations. These so called *features* are then used for classification in the *feature translation* block, which also provides the interface to the application. By controlling the application, the user receives feedback which is very important for the user to develop or adapt the skills which are necessary to control the application [65]. All components together form a closed loop system in which the computer parts are calibrated to the user and vice versa, the user adapts also to the system.

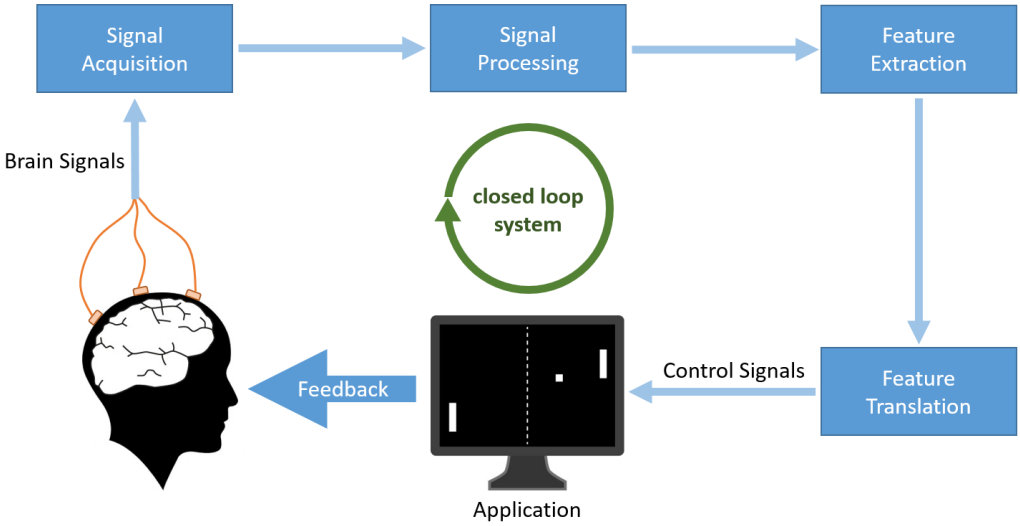


Figure 1.1.: Building blocks of a *BCI* system. All parts together, beginning at the user who intentionally modulates the brain signals up to the application which provides feedback for the user, are forming a closed loop system.

BCI applications include a large variety of technologies to monitor the brain activity like functional Magnetic Resonance Imaging (fMRI) or Electroencephalography (EEG) [65]. The latter was discovered by Dr. Hans Berger in 1924 and has been found to be a result of the superimposed Excitatory Postsynaptic Potentials (EPSP) on apical dendrites of pyramidal cells in the neocortex of the brain. He also found evidence for different rhythms present in the brain [5] and prepared the ground for researchers to investigate the usage of *EEG* in ways to decode the intention of users.

Pfurtscheller et al. [35] [38], Wolpaw et al. [65] as well as Vidal et al. [61] performed wide based research studies to learn more about the changes in the *EEG* caused by the actual execution of specific motoric movements. Later they repeated the experiments by performing just an imagination of these motoric movements. Their work paved the way for the first so called Sensorymotor Rhythm (SMR) based *BCI* [38]. This type of *BCI* utilizes task dependent power changes of different frequency bands in the *EEG* signals, namely Event-Related (De-)Synchronization (ERDS) [36]. A relative power decrease of the signal in comparison to a reference period is therefore associated with a desynchronisation of neural activity of brain areas during e.g. imagination of movements. Desynchronisation is therefore associated with an active state of the brain. Similarly, a power increase is the result of a synchronisation of neural activity and is associated with a resting state of the brain e.g. closing the eyes [36]. Important frequency bands for *SMR* based *BCIs* are the μ (8-13Hz) and the β (13-30Hz) rythms [26].

BCI applications incorporate machine learning approaches to classify the individual thoughts of the user. These machine learning algorithms need to be calibrated for each user due to a high inter-user and also a high inter-session variability of the brain signals [23]. The standard approach [36] of setting up a *BCI* system is to record the *EEG* of the user for different mental tasks. The acquired data is then used to calibrate the system. This calibration process requires time and is therefore annoying and exhausting for the user because it is repetitive and must be redone in every session. Furthermore, it is not possible to use the system for its communication purpose during the calibration process.

1.1.1. Calibration Time Reduction

Due to the issues of large calibration time to set up the *BCI* system, researchers try to reduce the calibration phase to get the technology out of the lab-environment to enable real-life applications. The goal is to shorten the calibration time as much as possible or even create a system without a calibration phase at all. The following points give a short overview about the current approaches which can also be found in [23].

- Regularization
  Regularization is known as a process of utilizing additional information in order to solve mathematical ill-posed problems or to counteract the effect of overfitting [31]. This approach is therefore often used, when only a little amount of calibration data is available. Machine learning algorithms often make use of the so called *covariance matrix* of the data. This matrix needs to be estimated out of the recorded data. If there are too few samples available, the matrix will not represent the underlying data correctly and falsifies the results of the machine learning algorithms. Therefore, regularization of the covariance matrix estimation can be a useful item to overcome this problem. Another approach is to regularize the machine learning procedure itself.

Therefore, the calculation can be guided in a certain direction, which fulfills wanted criteria and makes the results more stable. An overview of current regularization approaches can be found in [25].

- Semi-Supervised Learning
  This approach puts also effort in pushing the training time down by using only a few training trials to learn an initial basic classification model. After a short pre-training phase, the user is able to use the *BCI* system for the desired task. During the application of the *BCI*, the system classifies the mental tasks of the user and collects labeled data on the fly. This data is then used to re-train the system, which theoretically should improve the performance of the system with every update step [11]. The advantage is that the user receives feedback after a short period of time and is able to control an application without the need of recording large amounts of calibration data. However, it is not guaranteed that the classifier improves in every update step which can result in a decrease of performance [11].

- Artificial Data Generation
  Lotte et al. [23] presented a new approach to extend the available calibration data by generating artificial trials with different methods. They segmented the recorded data and recombined the segments to form new trials in the time- and frequency-domain. They also tried to acquire analogies among the trials and used these to create new trials. The results showed an increase in performance, especially when there was very few calibration data available.

- User to User transfer
  Another method to reduce calibration time is to build up a subject independent *BCI* by using the calibration data from other users for the current user. There are different approaches like *Multi-user covariance matrix estimation* or *domain adaptation*. The first approach is another regularization approach, whereas the latter transfers the classifiers or features from other users to the current user. It has been successfully applied to an Event-Related Potential (ERP) based *BCI* [20] and also to some extent to *SMR* based *BCIs* [62]. Gaur et al. [16] introduced a new approach by using a new signal processing method called Empirical Mode Decomposition (EMD) which helped them to utilize the common information of different users to calibrate the system of a new user. However, the *EMD* was only applied to offline data.

1.2. Motivation

One objective of *BCIs* is to enable severely motor impaired persons to control a computer and consequently also Assistive Technologies [27] [49]. The Institute of Neural Engineering of the Technische Universität Graz (TU Graz) is contributing to this research since the beginning and is also eager to move the *BCI* out of the laboratory [28] [56].

In 2014, a new technology assisted championship for persons with severe disabilities like Spinal Cord Injury (SCI) or Amyotrophic Lateral Sclerosis (ALS) was introduced by Dr. Robert Riener and his team of the Eidgenössische Technische Hochschule Zürich (ETH Zürich), namely the *Cybathlon* [46]. The main goal of the competition was to drive the development of Assistive Technologies and to provide a platform for novel technologies which are useful for the daily life. Among the six disciplines (*BCI*-Race, Functional Electric Stimulation (FES)-Bike Race, Leg Prosthetics-Race, Powered Exoskeleton-Race, Powered Wheelchair-Race and Powered Arm Prosthetics-Race) the *BCI*-Race is a discipline in which severely paralyzed pilots control a character in a computer game solely with their thoughts. The character is moving along a race track with a basic velocity. Three different types of action fields are placed on the track. These fields can be triggered by the pilot over the attached *BCI*. When triggered correctly (i.e. when the *BCI* classifies the intended mental task of the pilot right), the character gets a speed boost and jumps further to the next field. Otherwise, the character is slowed down for a certain time as penalty. However, it is also possible to send no commands to the game at all. In that case, the character simply runs with the basic speed along the race track.

To encourage undergraduate/graduate students to learn more about *BCIs*, Prof. Müller-Putz founded the official TU Graz BCI-Racing Team MIRAGE91 (Motor Imagery Racing Graz established 1991) [29]. The team was also meant to compete in the Cybathlon and therefore developed a new screening strategy to find a suitable pilot [57]. The experiences of the team were summerized in [58]. One of this experinces was that developing and controlling a multi-class *BCI* is not an easy task. Especially in terms of collecting calibration data to train the *BCI* system. The calibration itself would therefore take about 40 minutes to collect enough data. Furthermore, controlling a *BCI* can be very exhausting for the user due to its demanding amount of concentration needed for the operation. For this reason, the idea came up to incorporate data from previous sessions to reduce the calibration time of the *BCI* system. However, due to the complexity of the human brain, the signals can significantly change from session to session, which makes it very hard to use the data directly. Therefore, the data was processed by normalizing each *EEG* channel to their resting variance, which was determined on an extra measurement. The normalization coefficients were calculated for each session and were applied to the corresponding data seperately. To reduce the

calibration time by half, the normalized data from the last session was combined with the normalized data of the ongoing session to train the *BCI* system of the current session. This procedure yielded an advantage, because more time of each session could be spent on playing the game for the Cybathlon instead of recording calibration data.

After the Cybathlon, the question came up if there are better methods to decrease the calibration time without any loss of performance. That was the birth of this master thesis. Different methods should be investigated in order to reduce the calibration time by half in comparison to a reference system. As a condition, the performance should not drop significantly due to the reduced amount of calibration data. However, if a significant drop in performance is observed, the average performance of all subjects should be at least above 70% in terms of the classification accuracy. This 70% threshold is the commonly accepted minimum accuracy for useful *BCI* control [22].

Different data transfer methods like the Shrinkage Common Spatial Patterns (sCSP), Divergence Common Spatial Patterns (divCSP) and the Whitening Transformation (WT) as spatial filter methods as well as the *EMD* as time-domain filter method showed promising results in other studies [51] [16] or were thought to be appropriate for the task. To gain insight into the performance of the methods, a simulation should be conducted on an offline data set. The best method among the tested ones should then be incorporated in an online *BCI* system. An online evaluation study should then prove the applicability of the data transfer between two different sessions on two different days. Therefore the first session should consist of a calibration part and a test part were feedback is provided to the participants. This test part is also used to analyse the performance of the participants with a standard approach which is based on the (self-paced) GRAZ-BCI [39]. On the second session only half of the calibration data in comparison to the first session is recorded and the other half is taken from the first session. The system should be trained on the combined dataset and should be tested by providing also feedback to the participants. The performance in terms of the classification accuracy should be obtained for each session.

The hypothesis is defined as follows: A *BCI* system using half of the calibration time in comparison to a reference *BCI* system, yields the same performance as the reference *BCI* system. The reduction of calibration time is therefore performed by incorporating data from previous sessions into the calibration process of the *BCI*.

2 Methods

In this chapter, all data transfer methods (*sCSP*, Across Session/Subject Stationarity divCSP (divCSP-AS), Multi Session/Subject Stationarity divCSP (divCSP-MS), *WT* and Noise Assisted - Multivariate Empirical Mode Decomposition (NA-MEMD)) are described in detail as well as their usage. The software system for the offline evaluation study is explained including the used dataset and the signal processing pipeline with all settings for the different transfer methods. For the online evaluation study, the experimental setup, paradigm, electrode setting and the software system will be described to offer the straight reproduction of all performed experiments.

2.1. Data Transfer Methods

This section is determined for explaining the usage of the different data transfer methods. Therefore, a short historical background is provided for each method to give a better understanding, why the specific method was chosen. Following the historical view, the theoretical background explains the method itself in detail. In addition, also the practical usage of the methods, for data transfer, is discussed.

2.1.1. Shrinkage Common Spatial Patterns - sCSP

Spatial filtering is a very important part of a *BCI*. It is done to act against the highly correlated raw *EEG* data which is an effect of the volume conduction of the scalp [33]. To improve the spatial resolution and the classification accuracy, one option is to apply the Common Spatial Patterns (CSP) method to the band pass filtered data [40] [8]. Therefore, this method maximizes the signal variance for one class and simultaneously minimizes it for the other class. The *sCSP* algorithm is an enhancement of the basic algorithm and will be described in this section.

1. **Historical Background**
   The *CSP* algorithm is a machine learning approach to increase the discriminability of two classes by maximizing the differences in variances of *SMR* patterns of motor imagery tasks [40]. The variance of a bandpass filtered signal is equal to the power of the signal which makes this method effective in discriminating mental tasks that can be characterized by *ERDS* effects [36]. The filtering process corresponds to a transformation of the original data variable $\boldsymbol{x}$ by a matrix $\boldsymbol{w}$ to obtain the filtered data $\boldsymbol{x_{CSP}}$, as can be seen in Equation 2.1.

   $$\boldsymbol{x_{CSP}} = \boldsymbol{w^T x} \tag{2.1}$$

   The filter matrix $\boldsymbol{w}$ can be learned by solving the following optimization term (Rayleigh coefficient)

   $$\boldsymbol{w^*} = \arg\max_{\boldsymbol{w}} \left(\frac{\boldsymbol{w^T \Sigma_1 w}}{\boldsymbol{w^T (\Sigma_1 + \Sigma_2) w}} \right) \tag{2.2}$$

   which can be achieved by solving the generalized eigenvalue problem

   $$\boldsymbol{\Sigma_1 w} = \boldsymbol{\Lambda (\Sigma_1 + \Sigma_2) w} \tag{2.3}$$

   where $\boldsymbol{\Sigma_1}$ and $\boldsymbol{\Sigma_2}$ are the normalized covariance matrices belonging to class 1 and class 2. The obtained spatial filter $\boldsymbol{w}$ correspond to the eigenvectors of the equation with its eigenvalues $\boldsymbol{\Lambda}$. Large values in $\boldsymbol{\Lambda} = diag(\lambda_1, \lambda_2, \ldots, \lambda_d)$ indicate that the corresponding channel or row of the filter matrix $\boldsymbol{w} = [\boldsymbol{w_1}, \boldsymbol{w_2}, \ldots, \boldsymbol{w_d}]$ yields high variance for class 1 and low variance for class 2. By choosing n filters corresponding to extreme eigenvalues (either close to 1 or close to 0) the filtered data

   $$\boldsymbol{x'_{CSP}} = \boldsymbol{w'^T x} \tag{2.4}$$

   will have smaller dimensionality $n < d$ and the two classes will still be maximally separated by their variance. Each entry of $\boldsymbol{w'}$ corresponds therefore to a weighting of the *EEG* channels to discriminate the two classes.

2. **Theoretical Background**
   The empirical covariance matrices for each class need to be estimated out of the training data of the subject. This can cause problems, especially when the data is noisy or the training set is very small [45]. Typically, large eigenvalues of the empirical covariance matrix are estimated too large and small eigenvalues are estimated too small. This error in the estimation decreases the performance of the spatial filter. To address this drawback, regularized versions of the *CSP* have been developed

(see [25] for a review).

Shrinkage is one of these approaches and is well known to overcome this systematic errors. Due to the over- and under-estimation of the eigenvalues, the covariance matrix is regularized towards the identity matrix as can be seen in Equation 2.5.

$$\tilde{\boldsymbol{\Sigma}}(\boldsymbol{\gamma}) = (1-\gamma)\hat{\boldsymbol{\Sigma}} + \gamma\nu\boldsymbol{I} \tag{2.5}$$

The regularization parameter $\gamma \in [0,1]$ and the scalar ν can be calculated analytically which is easy to implement and is very effective compared to other approaches [55].

3. **Application as Transfer Method**
   To use *sCSP* as a transfer method, the data trials of both sessions are concatenated. So the resulting covariance matrix will be a mixture of both covariance matrices containing information of both sessions. It is assumed that the simple combination of the data without further signal processing is already useful for data transfer, because the combination of the data implies already a regularization towards the combined information of both sessions. For the calculation of the *sCSP* filter, a provided Matlab (Mathworks Inc., Natick, USA) implementation from the institute was used.

2.1.2. Divergence Common Spatial Patterns - divCSP

The non-stationary nature of the *EEG* is also a problem in terms of data transfer because the feature distribution changes over time. Many methods arose to handle this problem by filtering out the non-stationary parts or by regularizing towards the stationary parts. When doing so, the feature extraction can be trimmed to be more stable against variations of the *EEG* and can therefore be trained with data from previous sessions. The *divCSP* is a newly developed framework for computing the *CSP* filter and allows also the application of different regularization approaches into the calculation. The following is a description of different methods to counteract the non-stationary nature of *EEG*. The *divCSP*, which includes all listed methods in a single framework, is afterwards explained in detail.

1. **Historical Background**

   Stationary Subspace Analysis - SSA
   Von Bünau et al. [63] decomposed the *EEG* signal into stationary and non-stationary parts by finding a suitable de-mixing matrix such that the stationary parts of the signal are as stationary as possible in means of the first two statistical moments. After rejecting the non-stationary parts, the *CSP* filter were calculated on the obtained stationary signals. They concluded that the performance of a *BCI*

can be increased when a low amount of unstable sources is rejected. However, the amount of stationary sources in the signal is hard to determine and discriminant information can also be discarded in the process [53].

Stationary Common Spatial Patterns - stationaryCSP

The Stationary Common Spatial Patterns (stationaryCSP) is a regularization approach towards stationary parts of the signal [64] [54]. Unlike the *sCSP*, the regularization takes place in the objective function of the *CSP* (Equation 2.2) by extending the denominator with a regularization term $P(\boldsymbol{w}) = (\boldsymbol{w^T K w})$. Formally, the objective function becomes

$$\boldsymbol{w_i^*} = \arg\max_{\boldsymbol{w_i}} \left(\frac{\boldsymbol{w_i^T \Sigma_i w_i}}{\boldsymbol{w_i^T (\Sigma_1 + \Sigma_2) w_i} + \alpha P(\boldsymbol{w_i})} \right) \ , \ i \in \{1, 2\} \tag{2.6}$$

whereby $\alpha \geq 0$ is a user defined regularization parameter. The type of regularization depends on the choice of the matrix $\boldsymbol{K}$. For example: when setting the matrix $\boldsymbol{K} = \boldsymbol{I}$, a simple Tikhonov regularization is obtained (see [25] for a review).

To regularize against non-stationary parts, the signal is divided into N_e epochs, where each epoch can consist out of 1 to n trials from the same class. Non-stationarities are determined by evaluating the differences between the covariance matrices of each epoch $\boldsymbol{\Sigma_i^{(k)}}$ with the global covariance matrix $\boldsymbol{\Sigma_i}$ of class i.

$$\overline{\boldsymbol{\Delta}}_{\boldsymbol{i}} = \frac{1}{N_e} \sum_{k=1}^{N_e} \mathcal{F} \left(\boldsymbol{\Sigma_i^{(k)}} - \boldsymbol{\Sigma_i} \right) \ , \ i \in \{1, 2\} \tag{2.7}$$

$\mathcal{F}$ is an operator to make the resulting matrices positive definite by flipping negative eigenvalues. This ensures that the regularization term is always positive. The final regularization term is determined by setting $\boldsymbol{K} = (\overline{\boldsymbol{\Delta}}_{\boldsymbol{1}} + \overline{\boldsymbol{\Delta}}_{\boldsymbol{2}})$. The objective function for the *stationaryCSP* becomes

$$\boldsymbol{w_i^*} = \arg\max_{\boldsymbol{w_i}} \left(\frac{\boldsymbol{w_i^T \Sigma_i w_i}}{\boldsymbol{w_i^T} \{\boldsymbol{\Sigma_1 + \Sigma_2} + \alpha \left(\overline{\boldsymbol{\Delta}}_{\boldsymbol{1}} + \overline{\boldsymbol{\Delta}}_{\boldsymbol{2}} \right)\} \boldsymbol{w_i}} \right) \ , \ i \in \{1, 2\} \tag{2.8}$$

The regularization term penalizes non-stationary features and α balances discriminativity and stationarity of the features.

The authors demonstrated that the regularization is more robust against outliers and artefacts. In contrast to the Stationary Subspace Analysis (SSA) the method optimizes discriminativity and stationarity in a single objective function. Furthermore, the regularization can be trimmed towards different scales of non-stationarities e.g. by incorporating a larger amount of trials per epoch. In addition, the regularization term could be expanded by including also the epoch

differences of previous sessions or the differences between the current and previous sessions. However, the results depend strongly on the value of the parameter α, which is also difficult to determine. Also the flipping of negative eigenvalues is a direct interference into the calculation and is therefore not optimal [51].

Stationary Subspace Common Spatial Patterns - ssCSP

This method, proposed by Samek et al. in 2013 [52], is an extension of the *stationaryCSP* with the difference that the regularization term against non-stationarities, is learned by incorporating data from other users. The goal is to remove the subspace, which contains the main non-stationarities between the training and testing phase among the subjects. Instead of learning the task-relevant part from other subjects, they assumed that the non-stationary parts between the subjects are related and similar to each other and thus can be transferred.

They achieved this by calculating the eigenvectors $\boldsymbol{v_s^{(1)}} \cdots \boldsymbol{v_s^{(d)}}$ of the difference matrix between training- and testing-phase $\boldsymbol{\Sigma_s^{train}} - \boldsymbol{\Sigma_s^{test}}$ for several subjects s. Subsequently, they are aggregating the largest l eigenvectors, according to the l largest eigenvalues, of every subject s in a matrix $\boldsymbol{\mathcal{P}} = \left[\boldsymbol{v_1^{(1)}} \cdots \boldsymbol{v_1^{(l)}} \cdots \boldsymbol{v_s^{(1)}} \cdots \boldsymbol{v_s^{(l)}}\right]$. This matrix contains now the major changes between the training- and test-phase and can get very large in size. Therefore a dimensionality reduction is performed via the application of Principle Component Analysis (PCA) yielding in a projection matrix $\boldsymbol{\mathcal{P}_v}$. To guide the *CSP* calculation in a direction away from the non-stationary subspace, they calculate the orthogonal complement of $\boldsymbol{\mathcal{P}_v}$. The final regularization term is therefore determined by setting $\boldsymbol{K} = \boldsymbol{\mathcal{P}_v}\boldsymbol{\mathcal{P}_v^T}$ which can be included in the regularized *CSP* calculation (Equation 2.6).

Instead of using the difference between the training- and test-phase, a possible usage for data transfer may be to use differences between two or more sessions of several or the same user. However, the authors used a dataset containing at least five subjects, so the method may fail when only the data of one subject is considered in the calculation.

2. **Theoretical Background**

As described above, the main goal of a *CSP* filter is to maximally separate two classes of data in terms of their variance. An alternative way of achieving a solution for this problem was introduced by Samek et al. in 2014 [51]. They extended their previous methods and also combined them into a single framework in the same step. They showed that the subspace spanned by the *CSP* filter has maximum symmetric Kullback-Leibler divergence [30] between the distributions of both classes. Therefore they transformed the *CSP* calculation into a divergence-maximization problem as

can be seen in Equation 2.9.

$$\boldsymbol{w}^* = \arg\max_{\boldsymbol{w}} \mathcal{D}_{kl}\left(\boldsymbol{w}^T\boldsymbol{\Sigma_1}\boldsymbol{w} \parallel \boldsymbol{w}^T\boldsymbol{\Sigma_2}\boldsymbol{w}\right) \tag{2.9}$$

The Kullback-Leibler divergence $\mathcal{D}_{kl}$ becomes zero when the two distributions are identical and gets larger the more the distributions are different from each other. To solve this maximization problem, a gradient descent algorithm is used. This way of *CSP* calculation was termed *divCSP* by the authors. They showed that the solution of the *divCSP* is equivalent (up to linear transformations within the subspace) to the solution provided by a regular *CSP* algorithm, when maximizing Equation 2.9.

An advantage of this re-formulation is that the framework can easily be unified by extending the objective function (Equation 2.9) with a regularization term as can be seen in Equation 2.10.

$$\begin{aligned} \boldsymbol{w}^* &= \arg\max_{\boldsymbol{w}} \left(\mathcal{L}(\boldsymbol{w})\right) \\ \mathcal{L}(\boldsymbol{w}) &= \underbrace{(1-\lambda)\,\mathcal{D}_{kl}\left(\boldsymbol{w}^T\boldsymbol{\Sigma_1}\boldsymbol{w} \parallel \boldsymbol{w}^T\boldsymbol{\Sigma_2}\boldsymbol{w}\right)}_{CSP-term} - \underbrace{\lambda\Delta}_{Reg.-term} \end{aligned} \tag{2.10}$$

This regularization term can be adjusted in any arbitrary way to tackle different issues like non-stationarity. Note, that the negative sign before the regularization term indicates that it will be trimmed to be as small as possible. Therefore, divergence measures inside the regularization term will tend towards zero, which means that the included distributions will be close to each other.

Different regularization terms have evolved over time.

Within Session Stationarity Divergence CSP - divCSP-WS

To reduce the influence of artifacts or shifts that are present in the training data, the data is divided into epochs, containing 1 to n trials each. The non-stationarity of the epochs is then measured as average divergence between the data distribution of the epochs $\boldsymbol{\Sigma_i^{(k)}}$ and the whole data distribution $\boldsymbol{\Sigma_i}$ for class i.

$$\Delta = \frac{1}{2N_e}\sum_{i=1}^{2}\sum_{k=1}^{N_e}\mathcal{D}_{kl}\left(\boldsymbol{w}^T\boldsymbol{\Sigma_i^{(k)}}\boldsymbol{w} \parallel \boldsymbol{w}^T\boldsymbol{\Sigma_i}\boldsymbol{w}\right) \tag{2.11}$$

Where N_e is the number of epochs. This method is highly related to the *stationaryCSP* with the difference that the eigenvalues are not changed in this process. The flipping of eigenvalues may lead to unwanted side effects and therefore this procedure ensures more robust results [51].

Between Session Stationarity Divergence CSP - divCSP-BS
This regularization reduces the shift of the feature distribution between the training- and testing-phase. The authors utilized data from other subjects s to estimate these changes.

$$\Delta = \frac{1}{2N_s} \sum_{i=1}^{2} \sum_{s=1}^{N_s} \mathcal{D}_{kl} \left(\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{\Sigma}_{\boldsymbol{i,s}}^{\boldsymbol{train}} \boldsymbol{w} \parallel \boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{\Sigma}_{\boldsymbol{i,s}}^{\boldsymbol{test}} \boldsymbol{w} \right) \tag{2.12}$$

Where N_s is the number of other subjects. This method is highly related to the Stationary Subspace Common Spatial Patterns (ssCSP) with the difference that it considers only class unrelated changes, whereas the Between Session Stationarity divCSP (divCSP-BS) evaluates the variations for each class separately.

Across Session/Subject Stationarity Divergence CSP - divCSP-AS
This regularization variant was proposed in order to reduce differences between subjects. The assumption is, that the underlying motor imagery patterns are similar for each subject and hence perform a regularization against differences between the actual subject r and other subjects s.

$$\Delta = \frac{1}{2N_s} \sum_{i=1}^{2} \sum_{s=1}^{N_s} \mathcal{D}_{kl} \left(\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{\Sigma}_{\boldsymbol{i,r}} \boldsymbol{w} \parallel \boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{\Sigma}_{\boldsymbol{i,s}} \boldsymbol{w} \right) \tag{2.13}$$

Where N_s is again the number of other subjects.

Multi Session/Subject Stationarity Divergence CSP - divCSP-MS
This approach uses the regularization term to extend the objective function. Therefore, the regularization term contains the objective functions of other subjects. This means that the current *CSP* filter are calculated for multiple subjects s at once. This can be done by changing the regularization term to the following

$$\Delta = -\frac{1}{N_S} \sum_{s=1}^{N_S} \mathcal{D}_{kl} \left(\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{\Sigma}_{\boldsymbol{1,s}} \boldsymbol{w} \parallel \boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{\Sigma}_{\boldsymbol{2,s}} \boldsymbol{w} \right) \tag{2.14}$$

Note, that the sign of the regularization term is now inverted to maximize the divergence measure for all subjects s included in the calculation.

Beta divergence
The authors introduced also another divergence formulation in contrast to the Kullback-Leibler divergence $\mathcal{D}_{kl}$, namely the *Beta divergence* $\mathcal{D}_{\beta}$. They claim, that the measure is more suitable for *BCI* applications because it is more robust against changes in the data distribution. The scale of the invariance can thereby be changed with an extra parameter $\beta > -0.0115$ (taken from [51]), whereas values between

$-0.0115 < \beta < 0$ penalize single extreme events like artefacts and positive values $\beta > 0$ reduces the effect of more stable changes in the feature distribution. When setting the parameter to $\beta = 0$, *Beta divergence* coincides with the Kullback-Leibler divergence and is therefore a generalization of the Kullback-Leibler divergence [51].

3. **Application as Transfer Method**
As mentioned above, the *divCSP* incorporates all of the stated *CSP* methods, which could be used for performing data transfer between sessions. The *SSA* is a two step approach, namely the removal of non-stationary sources and the subsequent calculation of the *CSP* filter. This removal may be suboptimal as information is rejected in the first step which could be relevant in the second step. In contrast *divCSP* combines these steps by the use of regularization and is therefore a better approach to reduce non-stationarities [51].

Basically, all of the mentioned regularization terms of the *divCSP* can be used to perform a data transfer. However, only the *divCSP-AS* and the *divCSP-MS* were selected for the analysis. The *divCSP-BS* is basically the same method as the *divCSP-AS* with the difference that it considers differences between the testing and training phase of different subjects. In this work, we are only interested in the changes between the sessions. The Within Session Stationarity divCSP (divCSP-WS) was also not selected due to its inter session regularization approach of dividing the data into epochs.

Across Session/Subject Stationarity Divergence CSP - divCSP-AS
Instead of incorporating different subjects, the data from previous sessions is used. Therefore, the data of the same subject from the current run u is compared with previous runs $u-t$, with $t > 0$. The final regularization term can be seen in equation 2.15.

$$\Delta = \frac{1}{2T} \sum_{i=1}^{2} \sum_{t=1}^{T} \mathcal{D}_{kl} \left(\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{\Sigma}_{\boldsymbol{i,u}} \boldsymbol{w} \parallel \boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{\Sigma}_{\boldsymbol{i,u-t}} \boldsymbol{w} \right) \tag{2.15}$$

T is the amount of sessions which should be incorporated for the calculation of the *CSP* filter for the current session. For this study one previous session is utilized in the calculation and therefore $T = 1$.

Multi Session/Subject Stationarity Divergence CSP - divCSP-MS
The regularization term is changed to use data from previous sessions instead of other subjects. The *CSP* filter are calculated by maximizing the distance between the two classes for the current session u and previous sessions $u - t$ simultaneously.

The regularization term becomes

$$\Delta = -\frac{1}{T}\sum_{t=0}^{T} \mathcal{D}_{kl}\left(\boldsymbol{w^T\Sigma_{1,u-t}w} \parallel \boldsymbol{w^T\Sigma_{2,u-t}w}\right) \tag{2.16}$$

As for the other regularization method, one previous session is utilized for the calculation which means that $T = 1$. The regularization term contains now the objective function for both sessions. The regularization parameter is therefore set to $\lambda = 1$ to only consider the regularization term of Equation 2.10.

Because small changes in the feature distribution are assumed between the sessions, the decision fell on using the *Beta divergence* with small positive values to counteract against slow non-stationarities in addition to the regularization.

For the calculation of the *divCSP* filter, the Matlab function **divcsp** (*Wojciech Samek*, 2014), was used.

2.1.3. Whitening Transformation - WT

To enhance the class separability, a whitening step can be applied to the combined data of both classes instead of applying *CSP* filtering. Whitening is also included in the *CSP* calculation and may therefore be feasible for transferring data between sessions.

1. **Theoretical Background**
   Whitening or sphering is a linear transformation which transforms a d-dimensional variable $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ into a new d-dimensional variable

$$\mathbf{z} = (z_1, z_2, \ldots, z_d)^T = \mathbf{W}\mathbf{x} \tag{2.17}$$

with

$$\mathbf{x} = (x_1, x_2, \ldots, x_d)^T$$
$$\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_d)^T$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1^2 & \sigma_1\sigma_2 & \ldots & \sigma_1\sigma_d \\ \sigma_2\sigma_1 & \sigma_2^2 & \ldots & \sigma_2\sigma_d \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_d\sigma_1 & \sigma_d\sigma_2 & \ldots & \sigma_d^2 \end{bmatrix}$$

   where the resulting variable $\mathbf{z}$ has the covariance matrix $\boldsymbol{\Sigma_z} = \boldsymbol{I}$. The square matrix $\mathbf{W}$ is called the whitening matrix. Whitening is a generalization of *standardizing* or *normalization* which is done by

$$\mathbf{z} = \boldsymbol{V}^{-\frac{1}{2}}\boldsymbol{x} \tag{2.18}$$

where $\boldsymbol{V} = diag(\sigma_1^2, \sigma_2^2, \ldots, \sigma_d^2)$ contains the variances $var(x_i) = \sigma_i^2$. This results in $var(z_i) = 1$ but it does not remove the covariances between the dimensions which is the main difference to the *WT*. There are many ways of defining the whitening matrix $\mathbf{W}$ as can be seen in [19]. The most common choices are either the Zero-Phase Components Analysis (ZCA) [4] or the *PCA* method [13]. The decision fell on the *ZCA* approach because the resulting whitened data is as close as possible to the original data in the sense of least squares [4].

2. **Application as Transfer Method**
   To differentiate between the two classes, the classifier needs uncorrelated features. Therefore the whitening matrix is calculated on the imagination periods of all trials for both classes combined. For transfer learning, the trials of both sessions were incorporated in the calculation. The combination of the two classes ensures that the classes are uncorrelated. This is a similar approach as found in the calculation of the *CSP* filter whereas the *CSP* tries to maximize one class and minimizes the other in terms of the variance. The *WT* results in a variance of 1 and uncorrelated channels. Due to this separation, the usage of an additional *CSP* filter is not necessary and was not applied in this study. The whitening matrix is computed with the custom Matlab function **calcWhiteningTrans** written by *Colorado Reed*.

2.1.4. Noise Assisted Multivariate Empirical Mode Decomposition - NA-MEMD

The *EMD* is a signal processing method which decomposes a given time-domain signal into amplitude- and frequency-modulated components, defined as Intrinsic Mode Functions (IMFs). The *EMD* method is therefore adaptive and signal dependent which makes it suitable for the analysis of non-stationary and nonlinear signals like *EEG*.

1. **Historical Background**
   Huang et al. searched for a new way of decomposing a non-stationary and nonlinear signal without the use of the standard approaches of that time. One of this methods was the Fourier Spectral Analysis which has been applied to a broad variety of data since shortly after its introduction. However, the Fourier Spectral Analysis is only valid under specific restrictions like linearity, stationarity and periodicity of the signal. This is often violated in practice and can yield in misleading results, especially with biosignals like *EEG*. Furthermore, the Fourier Spectral Analysis uses a superposition of trigonometric components to decompose the signal. For non-stationary signals, many harmonic components are necessary which can cause energy-spreading throughout the spectrum. The Wavelet Spectral Analysis faces similar problems when analysing nonlinear and non-stationary signals, and is also strongly dependent on the choice of the *mother-wavelet* [18].

Another approach to perform a time-frequency analysis is done by applying the Hilbert Transformation to a real-valued signal $x(t)$ to obtain its complex conjugate $y(t) = \mathcal{H}(x(t))$, where $\mathcal{H}$ represents the Hilbert Transformation. By using this definition, an analytic signal $z(t)$ can be generated like

$$z(t) = x(t) + iy(t) = A(t)e^{i\Theta(t)} \tag{2.19}$$

in which

$$A(t) = \sqrt{x(t)^2 + y(t)^2},\ \Theta(t) = arctan\left(\frac{y(t)}{x(t)}\right) \tag{2.20}$$

With this so called Hilbert Spectral Analysis, a frequency component defined in every time point, namely the *Instanteneous Frequency* ω, can be calculated by using

$$\omega(t) = \frac{d\Theta(t)}{dt} \tag{2.21}$$

However, this representation leads only to valid results, when the signal $x(t)$ is restricted to have zero mean [18]. To apply the Hilbert Spectral Analysis also to functions which do not have a zero mean and furthermore to nonlinear and non-stationary signals, Huang et al. developed a new decomposition of the original signal with a new set of functions. These functions are called *IMFs* and are defined to be locally symmetric with respect to a zero mean level and to have an equal amount of extreme values and zero crossings [18]. With that definition, an *IMF* involves only one mode of oscillation, which can be amplitude- and frequency-modulated and which is not restricted to a narrow band signal. So, an *IMF* can be non-stationary by definition. The nonlinear and non-stationary signal $x(t)$ can now be decomposed into its *IMFs* and the Hilbert Spectral Analysis can be applied to each *IMF* to yield the Instanteneous Frequency for every *IMF* and therefore a time-frequency representation of the signal $x(t)$.

Empirical Mode Decomposition - EMD

To obtain the Hilbert Spectral Analysis, the signal has to be decomposed into one or more *IMFs* first. Huang et al. developed the *EMD* procedure [18] which extracts the *IMFs* out of the signal in a systematic way, designated as *sifting process*.

The main step in the sifting process is the determination of the mean envelope curve of the signal. This is done by identifying all maxima and minima and interpolating them respectively. The created maximum- and minimum-envelope curves are then averaged for each time point to yield the mean envelope curve. An *IMF* is then extracted by subtracting the mean envelope curve from the original signal. This calculation is then repeated as long as the criteria for an *IMF* (resulting function has zero mean and an equal number of extrema and zero crossings) are fulfilled. Figure 2.2 depicts a flow chart of the sifting procedure. Figure 2.1 shows the decom-

position of a non-stationary signal which is composed of three sinusoidal functions with frequencies of $2Hz$, $8Hz$ and $16Hz$. In addition, an exponential function with $\tau = 1$ is added to the signal to add a non-stationary and nonlinear part. The Matlab function **emd**, developed by Rilling et al. [47], was used.

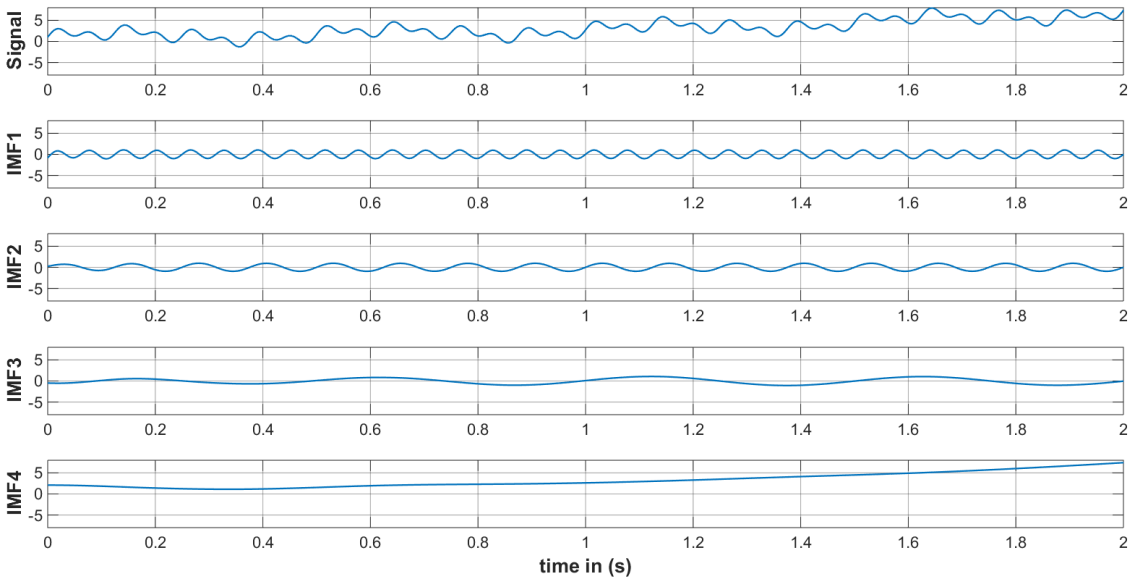


Figure 2.1.: Decomposition of a non-stationary signal using the EMD.

The last *IMF* $c_N(t)$ is often described as the residuum or the trend of the signal $r(t)$. When all *IMFs* are extracted the signal can be reconstructed by using formula 2.22.

$$x(t) = \sum_{n=1}^{N} c_n(t) \tag{2.22}$$

Now the Hilbert Spectral Analysis can be computed on the obtained *IMFs*. The Hilbert Spectral Analysis in combination with the *EMD* is called Hilbert-Huang Transformation [18]. In contrast to the Fourier Spectral Analysis or Wavelet Spectral Analysis, the *EMD* is not based on a theoretical framework. The *EMD* is more like an empirical approach that can be applied to a dataset, rather than a theoretical tool.

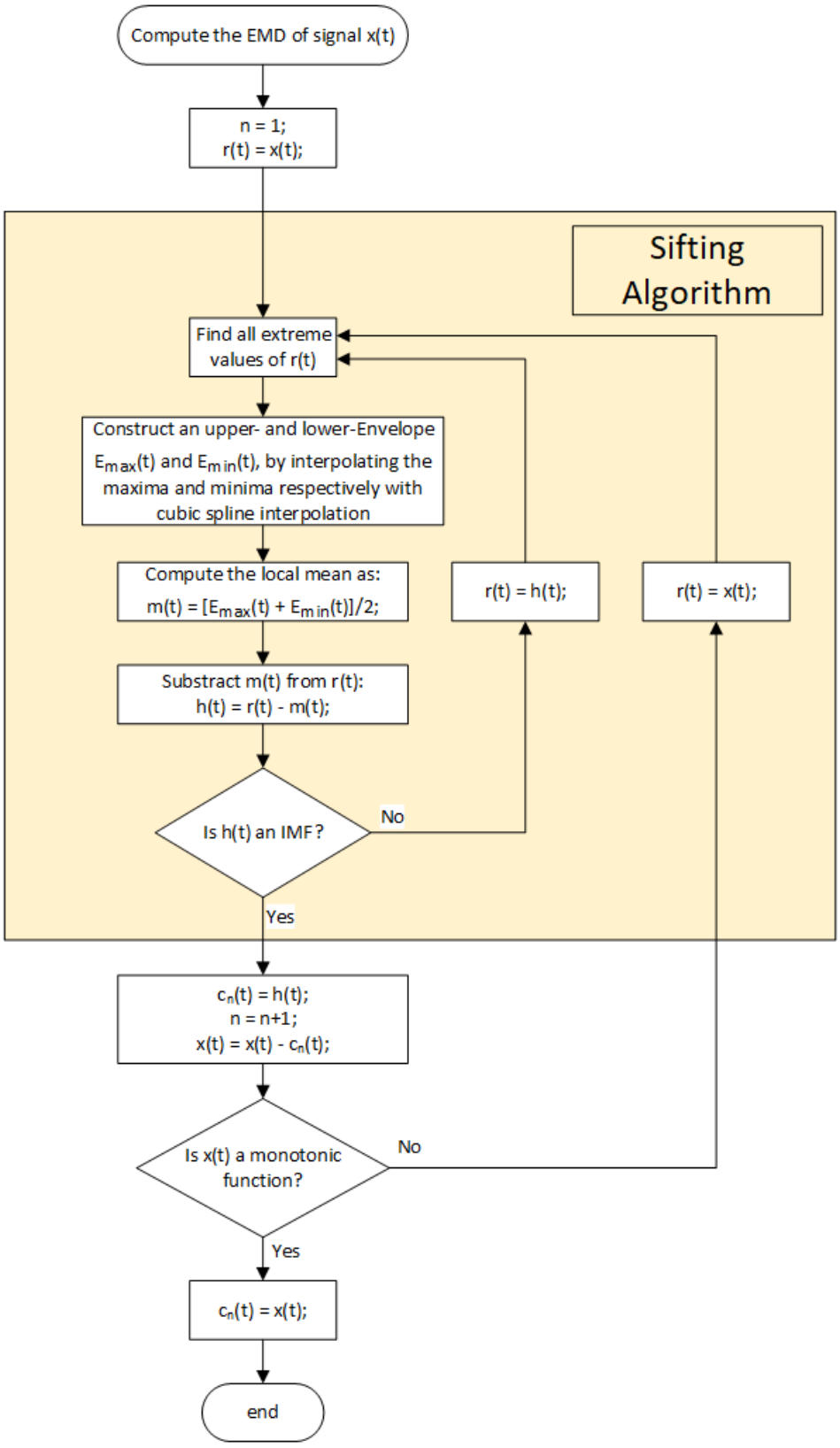


Figure 2.2.: Flowchart of the EMD procedure including the sifting process.

In order to understand how a signal, solely composed out of White Gaussian Noise (WGN), is decomposed into its *IMFs*, Flandrin et al. found quasi dyadic filter bank properties of the *EMD* [12]. So the majority of the frequencies in the range of $\frac{f_s}{2} - \frac{f_s}{4}$ are contained in the first *IMF* and the majority of the frequencies in the range of $\frac{f_s}{4} - \frac{f_s}{8}$ are contained in the second *IMF* and so forth. They concluded that the *EMD* can therefore be seen as an adaptive and signal dependent filter bank.

However, there are some drawbacks of the method e.g. finding the extreme values, interpolation errors, stopping criteria and mode-mixing [18]. Mode-mixing is therefore the most severe problem, because it falsifies the results and is comparable to energy leakage in the Fourier Spectral Analysis. Each *IMF* should only contain one mode of oscillation. Mode-mixing is present, when an *IMF* contains more than one mode of oscillation. This effect occurs, when frequency parts are transient and are not present in the whole signal. The *IMFs* contain the frequency parts in a descending order and the sorting between the *IMFs* is disturbed by such transient signals. An example is shown in Figure 2.3, where the original signal consists of four sinusoidal functions with frequencies $2, 8, 16$ and $32Hz$. The $32Hz$ signal is only present between $0.5 - 1.5s$.

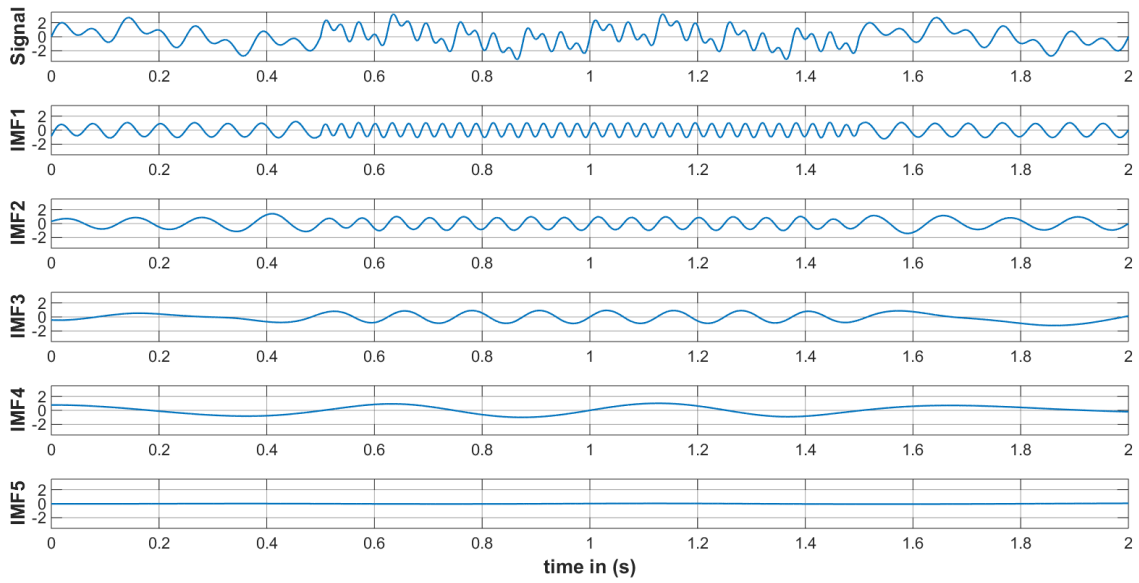


Figure 2.3.: Example of the effect of mode-mixing when decomposing a signal with transient parts. The mode-mixing is clearly visible between 0.5-1.5s.

Ensemble Empirical Mode Decomposition - EEMD

To counteract the problem of mode-mixing, the idea came up, to enforce the dyadic filter bank structure of the *EMD* by adding *WGN* to the signal [12]. By doing so, the original signal parts can be better separated and the mode-mixing problem

will be suppressed. On the contrary, the Signal to Noise Ratio (SNR) will also be reduced due to the added noise. By adding different realizations of *WGN* to the original signal and applying the *EMD* to each of these realizations, the noise will cancel out, when the results are averaged together. This resulting procedure is called Ensemble Empirical Mode Decomposition (EEMD) and was invented by Wu and Huang in 2009 [66]. However, a consequence of adding noise directly to the data is that a trace of residual noise is likely to remain in the result. The power of this residuum depends on the number of realizations or the size of the ensemble. Again the example, with transient functions is used to demonstrate the functionality of the *EEMD* algorithm. The results can be seen in Figure 2.4. The first *IMF* corresponds to the added noise in the signal which is then cancelled out during the process.

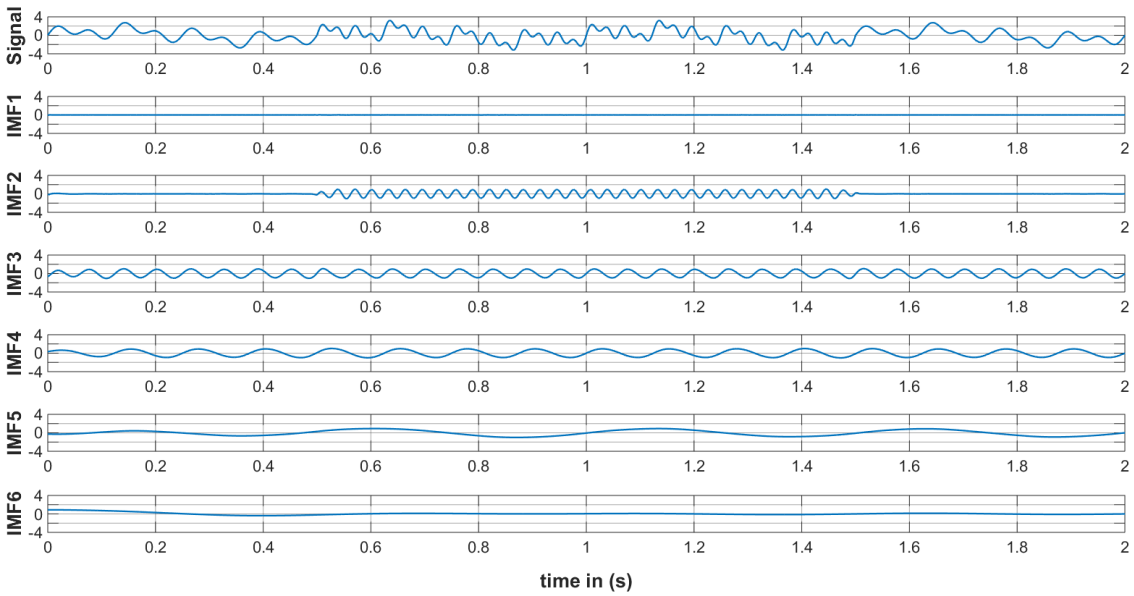


Figure 2.4.: Time signal with transient parts is decomposed with the *EEMD* algorithm. The first IMF contains the residual noise and Mode-mixing is no longer visible.

Multivariate Empirical Mode Decomposition - MEMD

The *EMD* algorithm was originally designed for one dimensional signals. Rilling et al. [48] extended the algorithm for bivariate signals. Therefore, a three dimensional envelope or tube has to be found to extract the two dimensional *IMFs*. For this reason, they projected the two dimensional signal onto the first dimension for different projection directions. The projections are equidistant points along the unit circle. For each direction, they extracted the maxima along the projected signals and calculated an envelope curve by using cubic spline interpolation between those maxima. After obtaining all envelope curves for all directions, the mean envelope is calculated by averaging all envelope curves. All further steps are equal to the one

dimensional case (Figure 2.2).

A further improvement was the development of a trivariate *EMD* algorithm introduced by [41] and then the final extension to multivariate signals [42]. The main problem of this approach was to find the multi dimensional maxima in the signal. Therefore, the method of the bivariate case was generalized for the multivariate case. They showed, that a uniform set of projection directions is not very suitable and introduced a quasi-Monte Carlo sampling of the n-dimensional space. Once, the projections are chosen, the remaining steps are the same as for the bivariate case. Multidimensional *IMFs* behave similar to one dimensional *IMFs* with the difference that the condition for equality of the number of extrema and zero crossings is not imposed, as extrema cannot be properly defined for multivariate signals [42].

As discussed above, mode-mixing is a critical issue of the *EMD*. Mode alignment between different channels is also important when dealing with multivariate data. Especially when data from different observations is analysed, the *IMFs* should be aligned. The Multivariate Empirical Mode Decomposition (MEMD) has this ability as can be seen in Figure 2.5 where a three dimensional synthetic signal was created with frequencies $2, 8, 16$ and $32Hz$. As before, the $32Hz$ part is only partially present in the first channel and is not present in channel three. Additionally, the $2Hz$ part is partially present in channel three and is not present in channel two. Small amounts of mode-mixing are still present, as can be seen in the first *IMF* of the first and third channel.

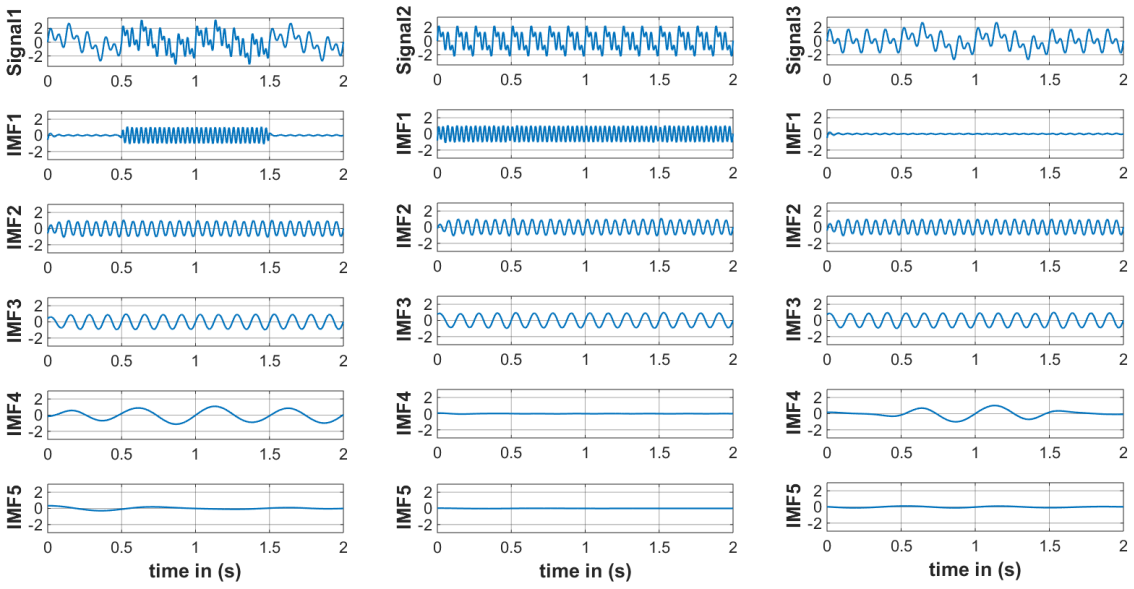


Figure 2.5.: Three dimensional time signal with transient parts is decomposed with the *MEMD* algorithm by using 64 projection directions and a stopping criteria of $[\sigma_1 = 0.05, \sigma_2 = 0.5, \alpha = 0.05]$. The colums correspond to the three channels of the signal and the rows correspond to the extracted *IMFs* for each channel.

2. **Theoretical Background**
   Rehman et al. [43] experimented with the *MEMD* algorithm and *WGN* as input signal and found also a dyadic bandpass filter bank property as with the one dimensional *EMD*. To decrease the amount of mode-mixing, they introduced the *NA-MEMD*. Instead of adding noise directly to the n-channel signal as with the *EEMD*, they added additional l adjacent noise channels with independent realizations of *WGN*. The subsequent application of *MEMD* yields (n + l)-variate *IMFs*, whose integrity is reinforced by the filterbank property of *MEMD* for *WGN*. The *IMFs* of the l-noise channels are then simply discarded, yielding in the n-variate decomposition of the signal. The variance (power) of the noise should be in the range between $2 - 10\%$ of the variance of the original signal (described in [44]). This was realized by calculating the mean variance of all channels, because the channels may have different levels of variance. Figure 2.6 shows the difference of the *MEMD* and the *NA-MEMD* decomposition of channel 3 of the sample signal as used before. The reduction of mode-mixing is now visible in the first *IMF* of channel three.

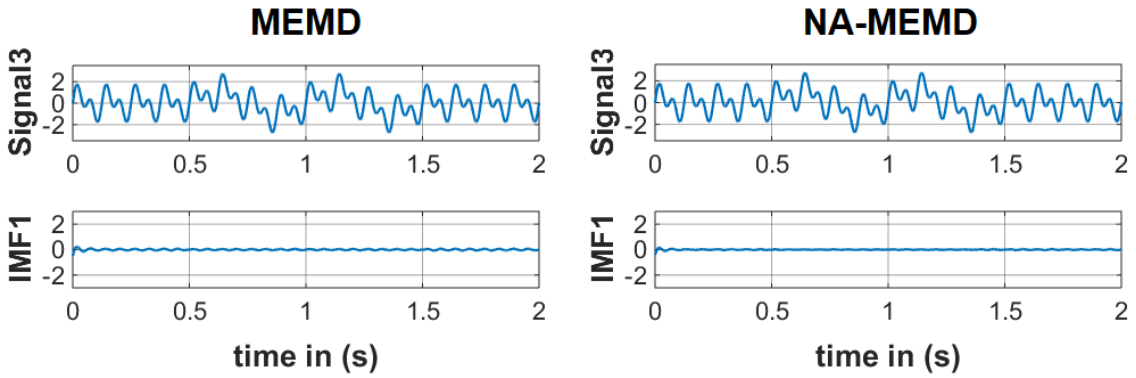


Figure 2.6.: Comparison of the *MEMD* and the *NA-MEMD*. Only *IMF*1 of channel 3 of the three dimensional time signal is shown. The reduction of mode-mixing is clearly visible. The *NA-MEMD* algorithm is applied by using one additional *WGN* channel with a variance of 5% of the input signal, 64 projection directions and a stopping criteria of $[\sigma_1 = 0.05, \sigma_2 = 0.5, \alpha = 0.05]$.

However, this dyadic filter bank property can also lead to mode-mixing, when oscillations are close to each other and are combined in one of these dyadic filter structures [43]. Therefore it is essential to use the *NA-MEMD* with caution to not enhance mode-mixing instead of reducing it.

3. **Application as Transfer Method**
   As stated above, the application of *EMD* on *EEG* signals is a suitable choice to extract the information of interest, namely power changes in the μ and β rhythms. These frequencies can therefore vary in time but are also bound to distinctive frequency intervals [38]. *EEG* is often measured at many scalp positions, which makes

the data also multivariate.

Park et al. [34] showed that the distinction between the different oscillations in *SMR EEG* is possible and that the multivariate algorithms yield a better separation than the one dimensional algorithms. They used the *NA-MEMD* algorithm instead of a bandpass filter for the extraction of *IMFs* and then adding the useful *IMFs*, corresponding to the μ- and β-rhythms, together to yield an enhanced *EEG* signal. Afterwords they applied *CSP* and used a Support Vector Machine (SVM) for classification, resulting in significant better results than with a bandpass filter approach.

Similar results were obtained from Gaur et al. [15], where they classified *SMR* motor imagery based *EEG* data. An extension of this work has been proposed in [16] where they incorporated data from other subjects to train an subject independent *BCI*. This approach should also be possible for incorporating data from other sessions instead of other subjects.

Therefore, the idea came up to combine the *IMFs* of both sessions obtained with the *NA-MEMD*. The obtained *IMFs* were not summed up again to yield an enhanced *EEG* (as in [34] and [15]) but were treated as bandpass filtered signals to mimic a bandpass filter bank approach, similar to [2]. Afterwards, a *sCSP* filter was calculated for every *IMF*, following the calculation of the band power features.

4. **Performance Modification**
   However, in the above mentioned studies, *EMD* was exclusively applied offline, due to a better extraction of *IMFs* (more extreme values) and its time consuming nature. The extraction of *IMFs* is based on finding extreme values in the signal and interpolating them via cubic spline interpolation. In order to gain suitable envelope curves through this interpolation, at least five maxima and five minima should be present in the signal [47]. Rilling et al. introduced an *online* application of the one dimensional *EMD* by applying the method *blockwise* to the signal [47]. Therefore, the same amount of *IMFs* and the same number of sifting operations for each window should be guaranteed. An online implementation of the *EMD* is essential for this work, because there exists the possibility that the *NA-MEMD* is chosen for the online evaluation study.

   Up to now, no algorithm has been developed to calculate an online *MEMD*. For this reason, the Matlab function **memd** (*Naveed ur Rehman* and *Danilo P. Mandic*, October 2009 [42]) was examined in terms of its computational efficiency. Therefore the function was applied to an *EEG* data sample (15 channels, $2s$, $512Hz$ sampling rate) with one additional 5% noise channel. The *MEMD* algorithm with 64 projection directions and a stopping criteria of [0.075,0.75,0.075] required $52s$ for decomposing this signal sample. To determine the primary sources for the time consumption, the

Matlab profiler *Run and Time* was used.

As it turned out, the main effort (nearly 80%) was used for computing the multidimensional spline functions by the Matlab function **spline**. Therefore, the function was modified by sorting out unnecessary code structures without affecting the spline calculation itself. Additionally, a custom *MEMD* function **my_memd** was developed by modifying the function **memd**. This was done by adding an extra parameter for the amount of *IMFs* to be extracted and the amount of sifting iterations per *IMF*.

With these actions, the computational effort was reduced by nearly 50%. The computation of the mentioned signal sample still takes almost $28s$. To apply the method in an online *BCI* application, the *NA-MEMD* must be computed faster than the length of the observed window. Therefore, the performance was evaluated by changing different parameters. The test cases and the results can be seen in Table 2.1. Note, that all calculations were performed on a Sony Vaio Notebook with an Intel i7-2670QM $2.2Ghz$ Dual Core CPU (Intel Inc., Santa Clara, USA), 6GB of RAM and Windows 10 (Microsoft Inc., Redmond, USA) with Matlab 2014b.

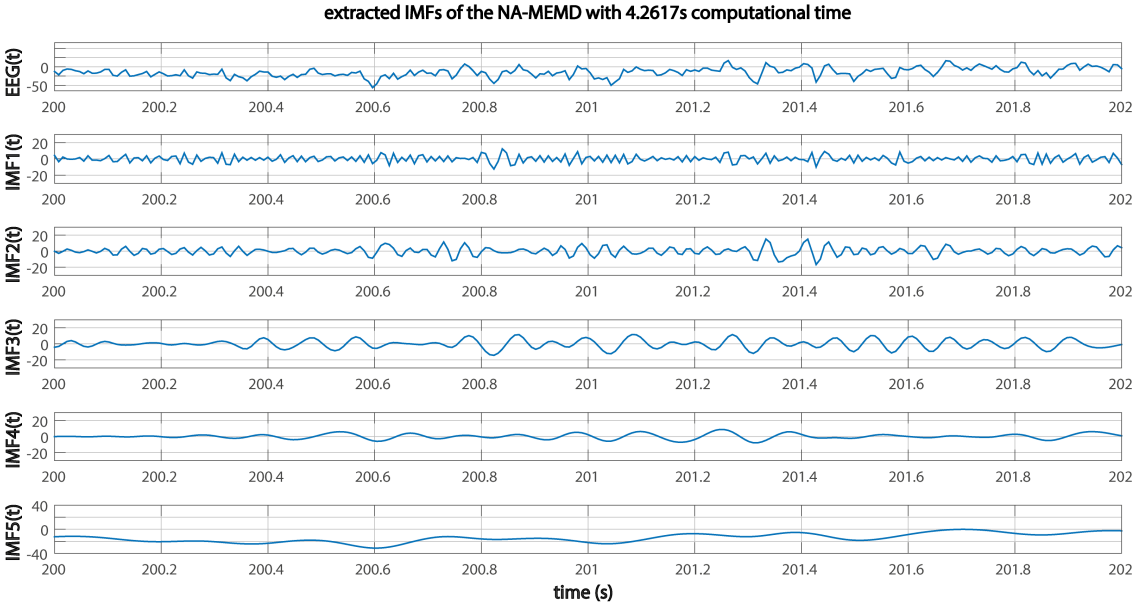
Table 2.1.: Effects of different parameters of the *MEMD* on the computational performance of the method. All tests were performed on the *EEG* data sample (15 channels, $2s$, $512Hz$ sampling rate). *NA-MEMD* algorithm with one additional *WGN* channel with a variance of 5% of the input signal, 64 projection directions and a stopping criteria of $[\sigma_1 = 0.05, \sigma_2 = 0.5, \alpha = 0.05]$.

Parameter	Effects on the computation time
length of the data window	time increases by 2s for each window second
downsampling of the signal	time decreases by a factor of 1.6 for downsampling by 2
amount of input channels	time increases by 0.2s for each channel
amount of Intrinsic Mode Functions	time increases by 1s for each *IMF*
amount of sifting operations per *IMF*	no effect above 100 Iterations
amount of projection directions	time decreases by half with half directions
stoppage criteria	time decreases linearly with higher tolerance values

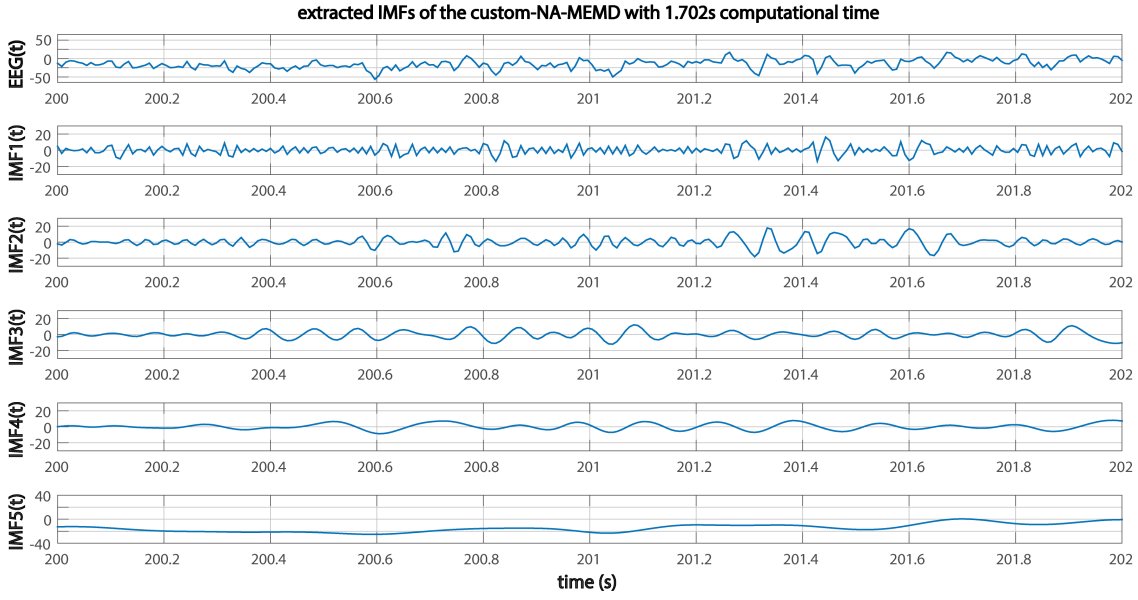
Therefore, the amount of *IMFs* was reduced to a maximum of five. The amount of projection vectors was set to the double of all input channels to a total of 32 projections (minimum, recommended by the authors). This results now in a performance of $8s$ for the proposed data sample. When downsampling the signal by a factor of 4, the algorithm needs $3.4s$ for the decomposition of the $2s$ signal. By setting the stoppage criteria to [0.1,1,0.1], an average computation time of $1.7s$ is reached. When increasing the window size to $6s$, the modified algorithm is now able to compute the signals in $4s$. An online application is now possible with the modified and windowed version of the *NA-MEMD*.

However, a decreased quality of the decomposition is the price paid for this accelerated version of the method. The difference of the original and the custom version

with the enhanced performance is shown in Figures 2.7 and 2.8.

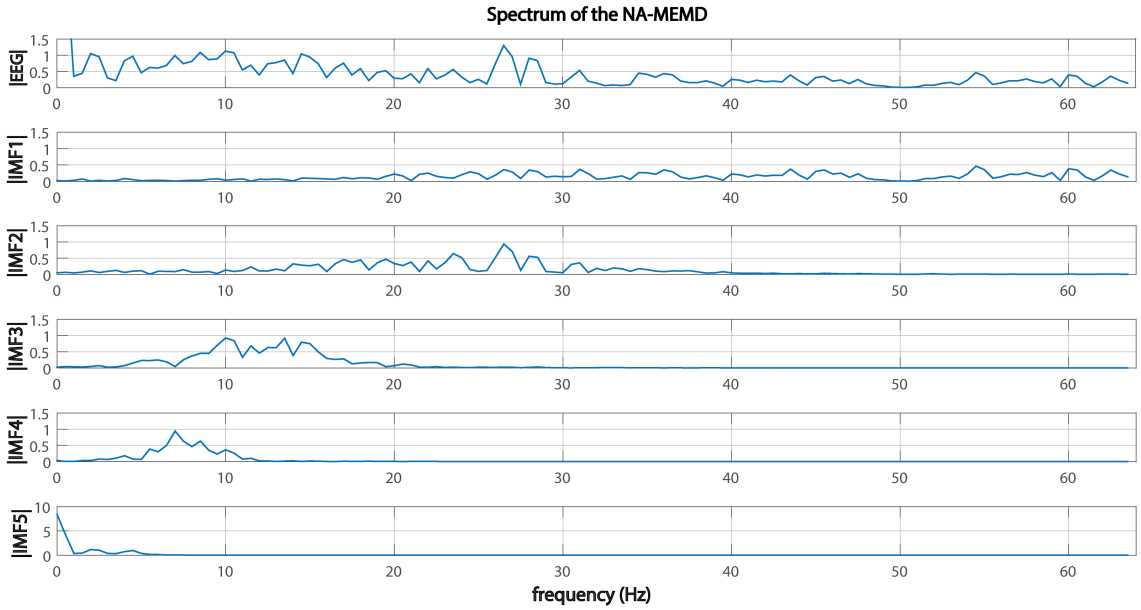


(a) NA-MEMD

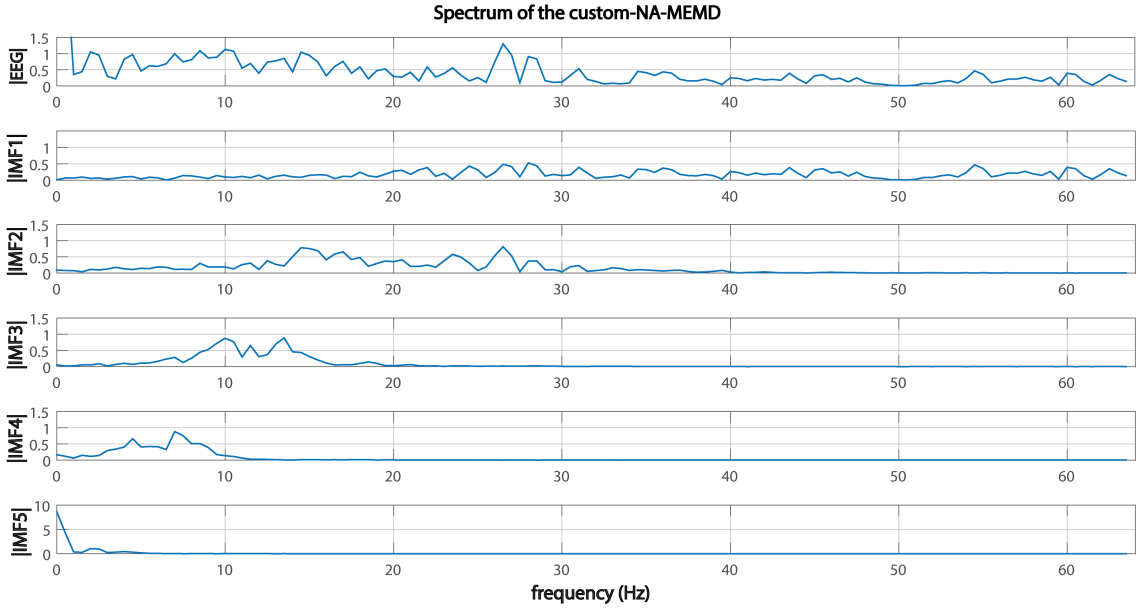


(b) custom-NA-MEMD

Figure 2.7.: Extracted IMFs of the NA-MEMD and the custom-NA-MEMD.



(a) NA-MEMD



(b) custom-NA-MEMD

Figure 2.8.: Spectra of the extracted IMFs of the NA-MEMD and the custom-NA-MEMD.

The differences can be seen best in the spectra of the two methods. With the original *NA-MEMD* method, the frequency bands are better separated as with the custom *NA-MEMD* method. Nevertheless, a peak can be seen in both methods around $8Hz$ in $IMF4$ according to the μ rythm. In $IMF3$, a broader spectrum ranging from $10 - 15Hz$ is visible indicating a mixture of μ and β. But the peaks are similar in both methods. Clearly recognizable is the dyadic bandpass filter bank property of the *NA-MEMD*.

As a final result, the custom Matlab function **slidingMEMD** was developed, which calculates the *NA-MEMD* on subsequent signal windows by using the custom function **my_memd**

2.2. Offline Evaluation

In the previous section, different data transfer methods were introduced and explained. In order to test these methods in terms of their performance, an offline simulation was designed. The evaluation should therefore be comparable to a real, online *BCI* study. The goal is to validate if it is possible to incorporate data from previous sessions into the calibration process of future sessions and to furthermore find a suitable transfer method for an online *BCI*. A Matlab script was written which simulates a *BCI* system. Therefore, the program includes all important parts of a *BCI* like preprocessing, feature extraction and feature translation but not a data acquisition and feedback part, because the data was already recorded in a previous experiment and no user is involved in the simulation. The data transfer is included in the simulation and the performance is evaluated for each transfer method via cross-validation. This section contains information about the used dataset, the signal processing pipeline, the performance evaluation and the software system itself.

2.2.1. Dataset

A dataset, recorded by Friedrich et al. in 2012 [14], was considered for the evaluation. It contains the data of ten subjects, recorded on four independent sessions.

The cue-based paradigm is depicted in Figure 2.9. It defines seven mental tasks namely mental rotation, word association, auditory imagery, mental subtraction, spatial navigation, face imagery and motor imagery of the right hand. The participants were instructed to start the imagery process as soon as they realized the target class, indicated by a visual cue.

For each study-member four sessions were recorded on four different days. Every session consisted of five runs with short breaks in between. One run in particular is comprised out of six trials per class, yielding 30 trials per class per session.

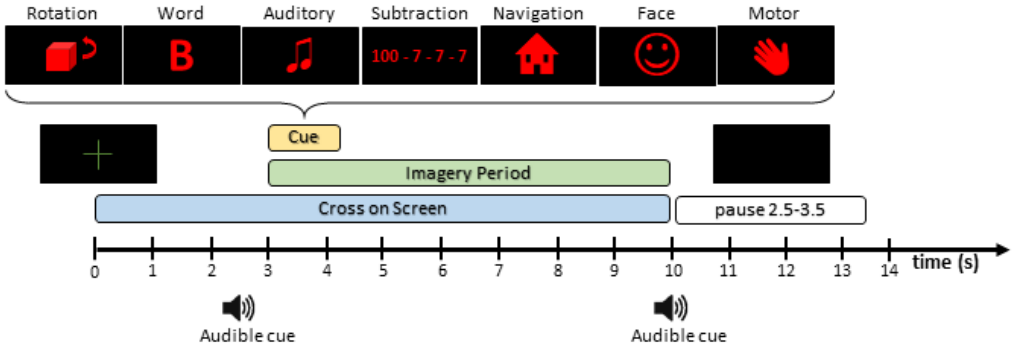


Figure 2.9.: Timing scheme of the cue-based paradigm. Image according to [14].
At $t = 0s$, a fixation cross was presented on the screen (baseline). At $t = 2.5s$, a beep was played. At $t = 3s$, one out of seven symbols was randomly presented for $1.25s$ in the middle of the screen. Users were asked to perform the indicated task for $7s$. At $t = 10s$, a second beep indicated the end of the trial, and the screen remained blank for $2.5 - 3.5s$ before the next trial started.

The *EEG* was recorded from 30 Ag/AgCl electrodes covering the whole scalp according to the international 10-20 system. The signals were sampled at a rate of $256Hz$ and were bandpass filtered between 0.5 and $100Hz$. A $50Hz$ notch filter was employed to reduce the impact of line noise.

The decision was made to choose two classes out of the possible seven. The combination of mental subtraction versus motor imagery of the right hand was selected because this combination showed the best accuracies averaged over all sessions for all subjects.

2.2.2. Signal Processing and Feature Extraction

This section explains the main parts of the signal processing pipeline. The specific settings for all methods will be discussed later in Section 2.2.4.

Electrode Selection

To get as close to the online evaluation setup as possible (see Section 2.3), 17 out of the 30 Ag/AgCl electrodes were selected. These electrodes are: F3, Fz, F4, FC3, FCz, FC4, T3, C3, Cz, C4, T4, CP3, CPz, CP4, P3, Pz, P4 and cover the main motor areas of the brain.

Artefact Handling

The data was inspected visually by an expert and trials containing Electrooculography (EOG) or Electromyography (EMG) were removed [14]. Therefore no extra artefact rejection was implemented.

Signal Processing

The first stage of the signal processing pipeline involves bandpass filtering of the raw *EEG* signals. Depending on the transfer method, one out of three different sets of bandpass filter is applied depending on the spatial filtering method.

- **CSP derived methods**
  For the *CSP* derived methods as *sCSP* or *divCSP*, a filter bank of 15 8^{th}-order Infinite Impulse Response (IIR) Butterworth bandpass filters divides the raw *EEG* data into 15 overlapping sub-bands covering the μ- and β-band, according to [59]. The Butterworth filter-type was chosen because it does not show ripples in the pass- and stop-band. As filter implementation a concatenation of normalized second order structures (SOS) was selected. The functions **fdesign** and **design** of the Matlab DSP-Toolbox were applied to create the filter. The 3dB - frequencies can be seen in Table 2.2.

Table 2.2.: 3dB - frequencies of the 8^{th}-order Butterworth bandpass filter bank for the *CSP* derived methods.

	μ - band							β - band							
filter	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$f_{3db-1ow}$	6	7	8	9	10	11	12	14	17	20	23	26	29	32	35
$f_{3db-high}$	8	9	10	11	12	13	14	19	22	25	28	31	34	37	40

- **Whitening Transformation - WT**
  The *WT* as transfer method is applied instead of the *CSP* filter to make the channels statistically independent from each other. Unlike the *CSP* transformation, no channels can be omitted, so the *WT* yields 17 virtual channels per sub-band. To avoid an *over-fitting* of the Shrinkage Linear Discriminant Analysis (sLDA) classifier, the trial to feature ratio should be close to 1 [7]. For that reason, the raw *EEG* signal is divided into two larger sub-bands covering the μ- and β-band by using two 8^{th}-order *IIR* Butterworth bandpass filter, yielding 34 virtual channels. The 3dB - frequencies can be seen in table 2.3. The whitening matrix is calculated on the combined, concatenated, windowed trials of both classes. Therefore, the signal parts from each trial between $1.5s$ and $3.5s$ after the visual cue were utilized.

Table 2.3.: 3dB - frequencies of the 8^{th}-order Butterworth band pass filter for the *WT*.

	μ - band	β - band
filter	1	2
$f_{3db-1ow}$	8	16
$f_{3db-high}$	16	30

- **Noise Assisted Empirical Mode Decomposition - NA-MEMD**
  In case of the *NA-MEMD* as transfer method, no bandpass filters were applied to the signal. The main reason for that is, that the *EMD* itself acts like an adaptive bandpass filter as explained in Section 2.1.4. Therefore, the extracted *IMFs* are treated as sub-bands and the spatial filtering is directly performed on them.

  The *NA-MEMD* is applied on consecutive windows of the signal with a length of $6s$. For each of the windows, five *IMFs* are extracted. Due to the quasi dyadic filter bank structure of the *NA-MEMD*, the majority of the frequency parts between $\frac{f_s}{2} = 128Hz$ and $\frac{f_s}{4} = 64Hz$ will be present in the first *IMF* [44]. Therefore, the first *IMF* is excluded. Also the *IMFs* relating to the noise channel are omitted. From each channel, four *IMFs* are remaining and are sorted *IMF*-wise to obtain four sub-bands.

After the filter stage, one of the three *CSP* algorithms namely *sCSP*, *divCSP-AS* or *divCSP-MS* is applied to each sub-band respectively. To calculate the filter, the concatenated signal parts of all training trials between $1.5s$ and $3.5s$ after the cue were utilized. The calculated spatial filters according to the two highest and two lowest eigenvalues of each set of *CSP* filters were selected. Hence, one *CSP* calculation per sub-band and four filters per *CSP* resulted in 60 virtual channels for the bandpass filter bank case and 16 virtual channels for the *EMD* case. In case of the *WT*, no *CSP* filter were calculated.

Feature Extraction and Classification

Logarithmic power of the virtual channels was used as features for the *sLDA* classifier. The power or energy of a signal is calculated by using formula 2.23 for each channel respectively.

$$P = \frac{1}{T} \int_{t_0}^{t_0+T} s(t)^2 \, \mathrm{d}t \tag{2.23}$$

The integral is equivalent to the average of the squared signal between t_0 and $t_0 + T$. This calculation is realized by a moving average filter with a length of $1s$. To ensure a gaussian distribution of the features, the base-10 logarithm is applied after averaging. The logarithmic power values $3.5s$ after the cue were then chosen as final features, resulting in a 16-, 34- or 60-dimensional feature vector for each trial, depending on the previous signal processing steps.

The feature translation or classification is carried out by an analytic shrinkage regularized *sLDA* classifier [7] [10], because it is easy to implement and is not as prone to overfitting as more complex classifiers [24]. A Matlab implementation of the institute was used for the calculation of the *sLDA* classifier.

2.2.3. Performance Evaluation

The measure of performance for each transfer method is a critical and non trivial task. A baseline method as reference is therefore needed to compare the results of the transfer methods. In the simulation, the system is trained with calibration data and validated on test data. Idially, the dataset contains a recorded calibration and a recorded test set were the participants received feedback from the trained system. As described in Section 2.2.1, each participant recorded 30 trials per class without any feedback. As a result, the dataset needs to be devided into a calibration and a test part. Often, a k-fold Cross-Validation (k-fold CV) is applied to yield more stable results [21]. Due to the lack of sufficient data trials per session, the Leave-One-Out Cross-Validation (LOOCV) [21] was chosen as validation method. The performances of the reference and all transfer methods are calculated for all subjects.

Reference

To rate the performance of the transfer methods, a reference method had to be found. Therefore, the performance of the standard bandpass filter bank *sCSP - sLDA* system, trained and tested on the calibration data of a single session was selected as reference, because it is widely spread across the community and is seen as a gold standard among the *BCI* community [60] [7].

So the total amount of training data for the reference system is now 30 trials for each class minus the one trial used for testing, yielding 59 trials in total. The accuracy is calculated for the 2^{nd}, 3^{rd} and 4^{th} session without any transfer of data between the sessions. Since the number of available trials for one subject was limited, the one-sided 95% confidence interval for chance level was calculated by using the method discussed in [6] and was determined to be 62.25% for the reference method. The basic principle of the *LOOCV* for the reference method is depicted in Figure 2.10.

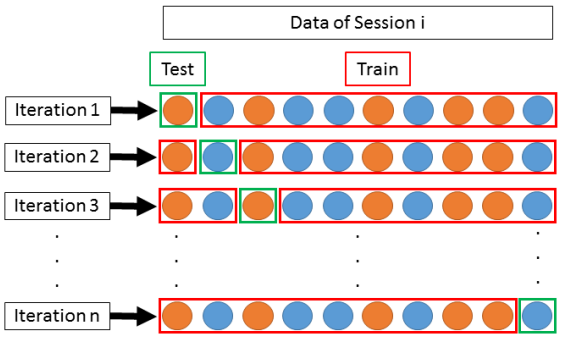


Figure 2.10.: Basic principle of the *LOOCV* for the reference method. Test and training data is taken solely from the current session.

Transfer Methods

In the online evaluation study, trials from the previous session and from the current session are combined to calibrate the system. The performance is then evaluated on test data recorded on the current session. Therefore the offline evaluation should mimic this procedure. The *LOOCV* is again used for the performance evaluation with the difference, that only trials from the current session are used for testing. To not falsify the performance of the transfer methods in comparison with the reference method, 29 trials per class were used for training the system with the main difference, that 15 trials per class were used from the previous session and 14 trials per class were used from the current session. The trials were selected randomly out of the dataset (without the test trial) for each iteration. The accuracy is calculated for the 2^{nd}, 3^{rd} and 4^{th} session, incorporating also training data from one session before the session under test. The one-sided 95% confidence interval for chance level was again calculated by using the method discussed in [6] and was determined to be 62.45% for all transfer methods. The basic principle of the *LOOCV* for the transfer methods can be seen in Figure 2.11.

Figure 2.11.: Basic principle of the *LOOCV* applied in the case of data transfer. Test data is only taken from the current session. Training data is taken from the current and from the previous session to mimic the data transfer.

2.2.4. Software System

As mentioned in Section 2.2, a Matlab script has been developed to mimic a *BCI* in an offline manner. The principle loop of a *BCI* has been described in Section 1.1. The script should therefore load the data from the specific subjects and perform the signal processing, feature extraction and classification as explained in Section 2.2.2.

Structure

As a first step, the parameters for all methods and all subjects are loaded into the system. After that, the order and combination of the bandpass filter-sets and the *CSP* algorithms is defined. The following combinations were considered:

- *EEG* - filter bank (15 filter) - *sCSP* - *sLDA* (as reference, single session)

- *EEG* - filter bank (15 filter) - *sCSP* - *sLDA*
- *EEG* - filter bank (15 filter) - *divCSP-MS* - *sLDA*
- *EEG* - filter bank (15 filter) - *divCSP-AS* - *sLDA*
- *EEG* - filter bank (2 filter) - *WT* - *sLDA*
- *EEG* - *NA-MEMD* (4 *IMFs*) - *sCSP* - *sLDA*

For each subject, the data was loaded into the system and then preprocessed (band pass filtering or *NA-MEMD*) to save computation time. Then the accuracies were determined with the performance measure, explained in Section 2.2.3 for all of the combinations stated above. Along with the accuracies over time, the peak-, median- and the mean-accuracy ($2s$ - $5s$ after the cue) were determined and saved in corresponding mat-files. The structure of the software system for the offline simulation can be seen in Figure 2.12.

Settings
The following lists the different settings and parameters of all transfer methods.

- **Shrinkage Common Spatial Patterns - sCSP**
  In the provided implementation of the institute, the *shrinkage* method has to be chosen for the calculation of the *CSP* filter. Besides that, no more parameters needed to be set.
- **Divergence Common Spatial Patterns - divCSP**
  The initialization of the *divCSP* algorithm was specialized for the individual requirements. The following shows the parameters used for both *divCSP* algorithms.
  - (max_iter = 1000) a maximum of 1000 iterations for the calculation
  - (nreps = 1) calculate *CSP* filters once without repetition
  - (deflation = 0) use subspace algorithm to extract the whole *CSP* subspace at once.
  - (pca = 1) apply PCA in the last step to obtain meaningful filter
  - (csp_init = 1) initialize first repetition with the standard *CSP* solution
  - (sym = 1) apply symmetric divergence for the regularization term
  - (quiet = 1) do not print additional information during the process
  - (beta = 0.01) beta divergence parameter, to reduce the effects of slow, stable changes between the sessions [51].

  For more details relating to the *divCSP* parameter, please see [51] and [50].

- **Across Session/Subject Stationarity Divergence CSP - divCSP-AS**
  - (lambda = 0.3) regularization parameter. Data of the current session, has a weight of 70% in the *CSP* calculation and data of the previous session has a weight of 30%.
  - (mode = 0) change the sign of the regularization term to "$-$"
- **Multi Session/Subject Stationarity Divergence CSP - divCSP-MS**
  - (lambda = 1) use the regularization term only for the calculation
  - (mode = 1) change the sign of the regularization term to "+"
- **Whitening Transformation - WT**
  No parameters needed to be set for the *WT*.
- **Noise Assisted Multi Variate Empirical Mode Decomposition - NA-MEMD**
  For the *NA-MEMD* as transfer method, extra noise channels with white gaussian noise have to be added to the original signal with an energy of 5% of the original signal as explained in [44].
  - (samplefactor = 1) no downsampling of the signal
  - (window = 6) calculate *NA-MEMD* over $6s$ windows
  - (numNoiseChannels = 1) add 1 extra noise channel
  - (numIMFs = 5) decompose signal into 5 *IMFs*
  - (numIterations = 1000) apply a maximum of 1000 sifting operations per *IMF*
  - (numProjections = 36) number of projections for finding the extreme values of the signal. The specified minimum is: 2· (signal channels + noise channel) [44].
  - (stopParams = [0.075,0.75,0.075]) stopping criteria, as used in [44]
  - (usedIMFs = [2,3,4,5]) use all *IMFs* except the first one

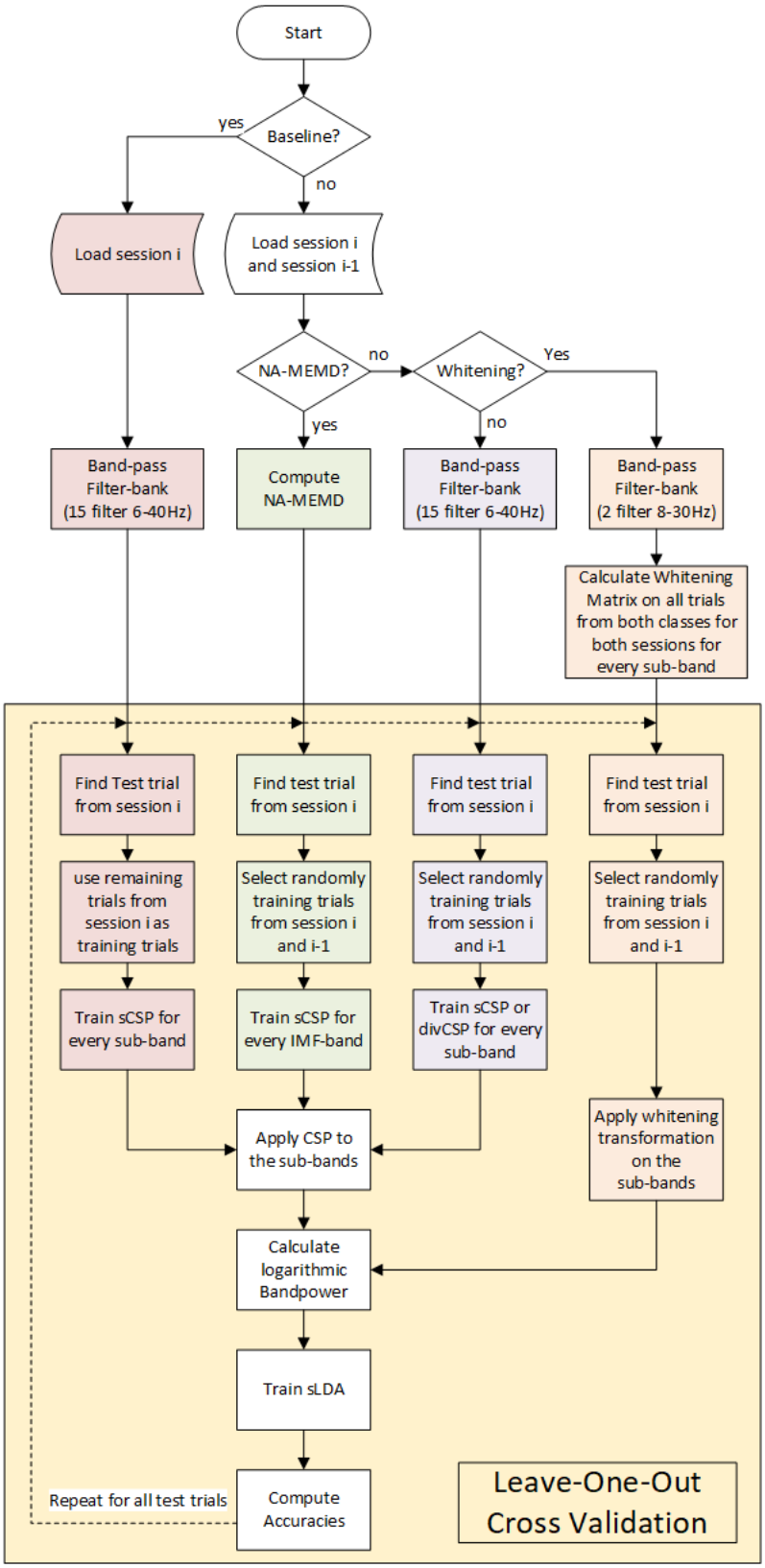


Figure 2.12.: Structure of the software system for the offline evaluation. The flow chart shows all steps for one subject.

2.3. Online Evaluation

In Section 2.2, an offline evaluation study was performed in order to find the best transfer method among the presented ones. The findings indicated that this approach is applicable for *SMR-BCI*. Therefore, the next step was to include these findings in an online *BCI* system. The goal was to asses whether the combination works in an online manner as well as to quantify its performance.

2.3.1. Experimental Setup

A lot of thought was given into the design of the study. To demonstrate the correctness of the hypothesis, at least two sessions on two different days are needed, whereby the *BCI* system of the second session is trained with data from both sessions. The performance has to be evaluated in terms of the classification accuracy on both sessions.

One group of eleven participants (3 female, 8 male), aged between 21 and 28 years, took part in the study. Some of the tested subjects had already experience with *BCIs*, and some had no experience. The measurements took place on two different days between 08:00 in the morning and 18:00 in the evening. The time of the measurement could be chosen from the subjects itself to gain a similar state of well-being on both measurement days. The subjects could also choose the date of the sessions with the restriction that the second measurement takes place at least on the next day and maximally seven days after the first session. The mixture of different measurement times of the day and a different amount of days in between the sessions was preferred to also show the real-world functionality of this approach.

The *divCSP-AS* was chosen as transfer method because of its regularization towards stationary spatial patterns between the sessions for each class individually. The *BCI* system was equal to the simulation system used in the offline evaluation study for the reference and the *divCSP-AS* method (see Section 2.2.3). The online software system is discussed in more detail in Section 2.3.4. To achieve a trial to feature ratio of approximately 1, 60 trials/class are needed to calculate the *CSP* filter and the *sLDA*.

The first session is used as a reference session to gain a basic insight in the performance of the subjects and to later compare the two sessions to observe possible differences. The first session consists of six training runs with ten trials per class, for each run. All subjects were asked after each run, to rate their performance according to concentration, well being, estimated performance and the amount of produced artefacts. The runs, where the subjects were concentrated and produced few artefacts were marked for the second session. The *sCSP-sLDA* system is trained with these six training runs followed by three feedback runs with ten trials per class per run, to evaluate the reference performance.

On the second session, the goal was to reduce the calibration time by half in comparison to the first session. Therefore, three training runs with ten trials per class per run were recorded. The remaining three runs were randomly selected out of the marked runs from the first session. The *divCSP-sLDA* system is then trained with the three runs from the second session and the three selected runs from the first session. After the training, six feedback runs with ten trials per class per run were executed to evaluate the performance of the system with incorporated data from the first session and reduced calibration time. The scheme of the study is depicted in Figure 2.13.

The one-sided 95% confidence interval for chance level was again calculated by using the method discussed in [6] and was determined to be 62.25% for the first session and 58.8% for the second session.

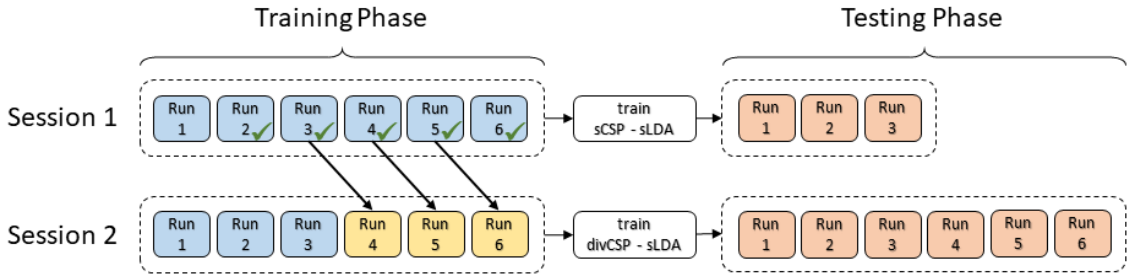


Figure 2.13.: Scheme of the online study. To perform data transfer, three out of all marked runs were chosen from the first session and combined with the three runs from the second session.

2.3.2. Experimental Paradigm

Figure 2.14 shows the sequence of events of a trial, for all training and feedback runs. It was inspired by the cue-guided two class GRAZ-BCI paradigm which is further described here [37]. At $0s$, a green fixation cross appeared on the screen as fixation point for the eyes. The subjects were instructed to focus on the cross with their eyes and to not produce any artefacts until the cross vanishes. Additionally, the subjects should also try to relax in this period. At $2s$, a beep sounded to prepare the subject for the imminent begin of the trial. The cue was displayed from $3s$ to $4.25s$ and was pictured as a red arrow pointing to the right (imagination of right hand movement) or downwards (imagination of feet movement). Participants were instructed to start with motor imagery as soon as they recognized the cue. They should perform sustained kinesthetic motor imagery [32] until the fixation cross vanishes at $8s$. A resting period followed the trial with a random length between $2s$ and $3s$. The subjects were instructed to produce artefacts only in this period. In the training phase, the subjects did not receive any feedback.

Regarding the type of imagination, the subjects were instructed to either think of squeezing an anti-stress ball in a repetitive manner for imagination of right hand movement or they were told to imagine pedal movements with their feet. To help the subjects gain more insight in the training sequence, a short video of the original training paradigm, containing three trials of each class, was shown to them. They were instructed to actually perform the movement during this test run. This allowed the supervisor to observe the correct execution of the motor task. The video was shown a second time with the difference that the subjects had to perform the actual imagination task. This procedure ensured a better preparation of the subjects for the study.

During the feedback runs, feedback was displayed in form of a white colored bar pointing to the right or downwards (Figure 2.15) according to the current classification output. The length of the bar was determined via the distance output of the classifier (measure of certainty). The output of the classifier was presented for positive and negative results in order to create a more real-world environment for the subjects.

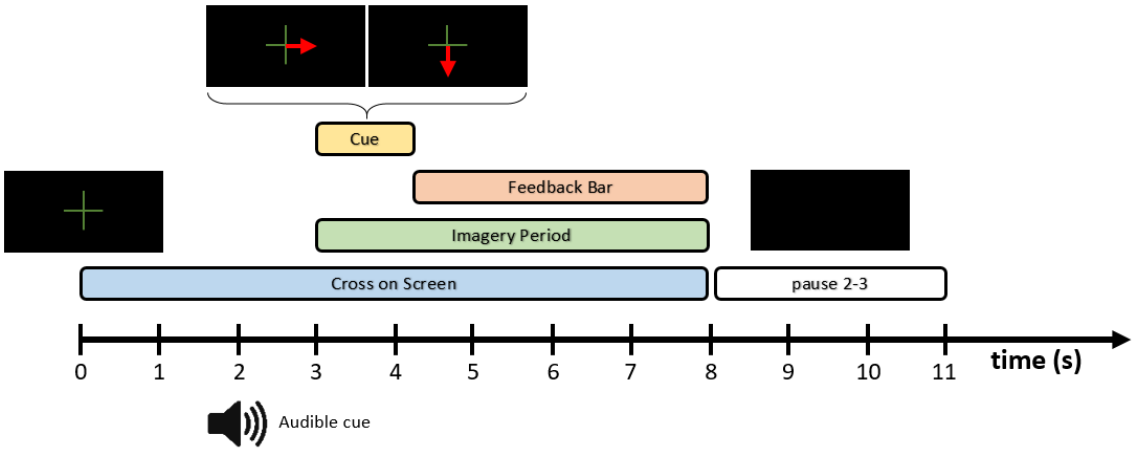


Figure 2.14.: Timing scheme of the cue-guided two class GRAZ-BCI paradigm [37]. At $t = 0s$, a fixation cross was presented on the screen (baseline). At $t = 2s$, a beep was played. At $t = 3s$, a right- or down-arrow was randomly presented for $1.25s$ in the middle of the fixation cross. Users were asked to perform the indicated task for $5s$. At $t = 8s$, the fixation cross disappeared and the screen remained blank for $2 - 3s$ before the next trial started.

2.3.3. Hardware and Data Recording

The online measurements were conducted using a notebook with Windows 7 as operating system, an Intel Core i7 processor at 2.6GHz and 16GB of RAM.

Fifteen active g.tec (g.tec medical engineering GmbH, Schiedlberg, Austria) Lady-bird Ag/AgCl electrodes combined with a g.tec GAMMA box and the g.tec USBamp biosignal

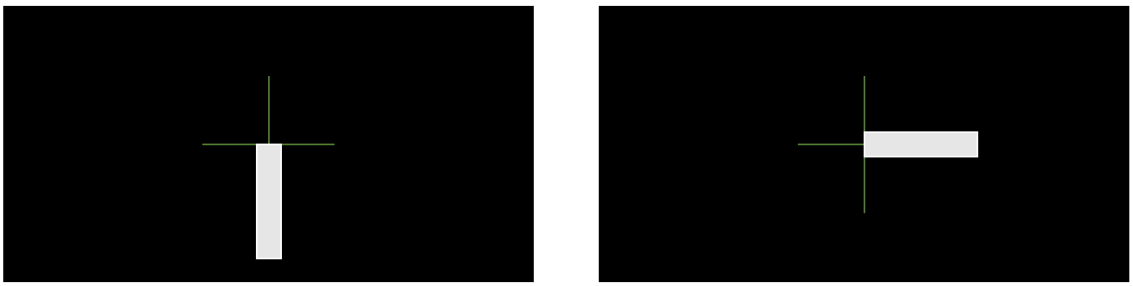


Figure 2.15.: Feedback for good performance of feet (left) and right hand (right) motor imagery.

amplifier were used to measure the *EEG* with a sample rate of $512Hz$. A 8^{th} order Butterworth bandpass filter, to prevent aliasing, was set to $0.1-100Hz$ and an additional notch filter at $50Hz$ was configured to suppress the line noise. The electrodes were mounted in an equidistant manner (2.5cm) at positions FC3a, FCza, FC4a, C5a, C3, C1b, C1a, Cz, C2a, C2b, C4, C6a, CP3a, CPza and CP4a, to cover the main motor areas of the brain (around C3, Cz and C4). The *EEG* signals were recorded monopolar, relative to a common reference electrode which was placed on the left ear lobe. The ground electrode was placed frontal at position AFzb. Figure 2.16 shows the electrode placement on the scalp. Data acquisition from the amplifier is handled by the custom made *TOBI SignalServer Framework* which also provides an interface to Matlab/Simulink where signal processing is performed on the data [28].

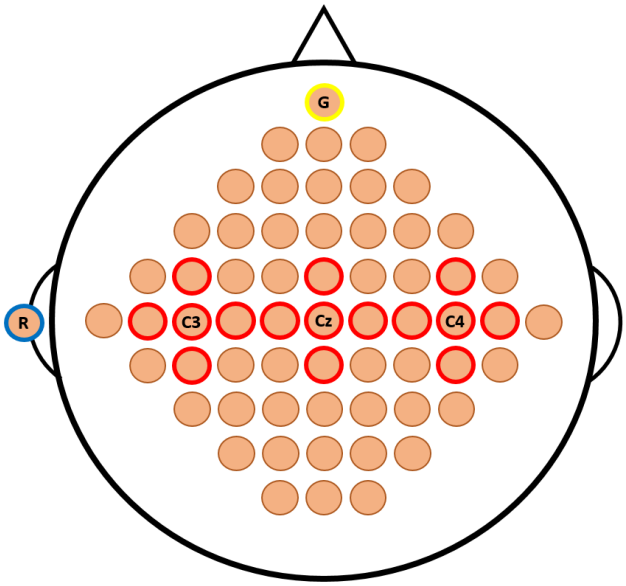


Figure 2.16.: Electrode layout: Only the red encircled electrodes were used in the study. The ground electrode (yellow) is placed frontal at position AFzb and the reference (blue) is clipped to the left ear lobe.

2.3.4. Online System

The system is implemented in Matlab/Simulink 2014b and contains the signal acquisition (TOBI SignalServer, [9]) and the signal processing pipeline as well as the parts responsible for the feedback. The signal processing pipeline is the same in both sessions. Only the calculation of the *CSP* filter changes.

Bandpass Filter Bank
The bandpass filter bank is the same as discussed in Section 2.2.2 (Table 2.2) and consists of 15 8^{th}-order Butterworth filters ranging between $6 - 40Hz$.

Artefact Removal - Outlier Rejection
Before calibrating the system, a statistical outlier rejection (amplitude, threshold, kurtosis, probability) [11] was performed to exclude artefact contaminated trials.

- **Rejection by amplitude**
  Large artefacts as *EOG* can be excluded by thresholding. Therefore, the trial is dismissed, if the amplitude exceeds a certain threshold value ($\pm 100\mu V$).
- **Rejection by variance**
  Trials were rejected if the variance exceeds five times the standard variance, calculated on every channel for all trials.
- **Rejection by probability**
  Trials were rejected according to differences between their probability distribution and the distribution of all trials.
- **Rejection by kurtosis**
  Trials were rejected when their kurtosis exceeds a certain threshold, which was calculated on the kurtosis of all trials.

The interval between $2 - 7s$ of the trials was analysed because it inherits the most important parts for training.

Spatial Filtering
Spatial filtering is performed via a matrix-vector multiplication as shown in Equation 2.1. The 15 *CSP* filter matrices are therefore calculated for each sub-band of the signal, using the obtained training trials. The period between $4.5 - 6.5s$ of the paradigm ($1.5 - 3.5s$ after the visual cue) of all trials, which were not excluded by the outlier rejection, was utilized for the calculation of the covariance matrix. In the first session, the *CSP* filter are calculated by using the *sCSP* method. On the second session, the *CSP* filter are calculated by using the *divCSP-AS* method.

The best two filters corresponding to class 1 and the best two filters corresponding to class 2 were selected as spatial filters yielding in a $15 \cdot 4 = 60$ channel virtual signal.

Feature Extraction
The logarithmic band power features are obtained by squaring the signals and performing an averaging on each of the 60 channels individually. The averaging is performed by using a Moving Average Filter (MAF) over a period of $1s$. After averaging, the base-10 logarithm is applied samplewise.

Classification
Classification was performed with an *sLDA* classifier. The classifier was trained with the logarithmic band power values $6.5s$ after the start of a trial ($3.5s$ after the visual cue). This was done for all training trials, which were not excluded from the outlier rejection.

Feedback
To close the *BCI* loop, feedback is provided from $4.25 - 8s$ after the start of a trial ($1.25 - 5s$ after the visual cue) for each trial in the form of a white arrow pointing in the classified direction (right, down).

Figure 2.17 shows the Matlab/Simulink *BCI* system.

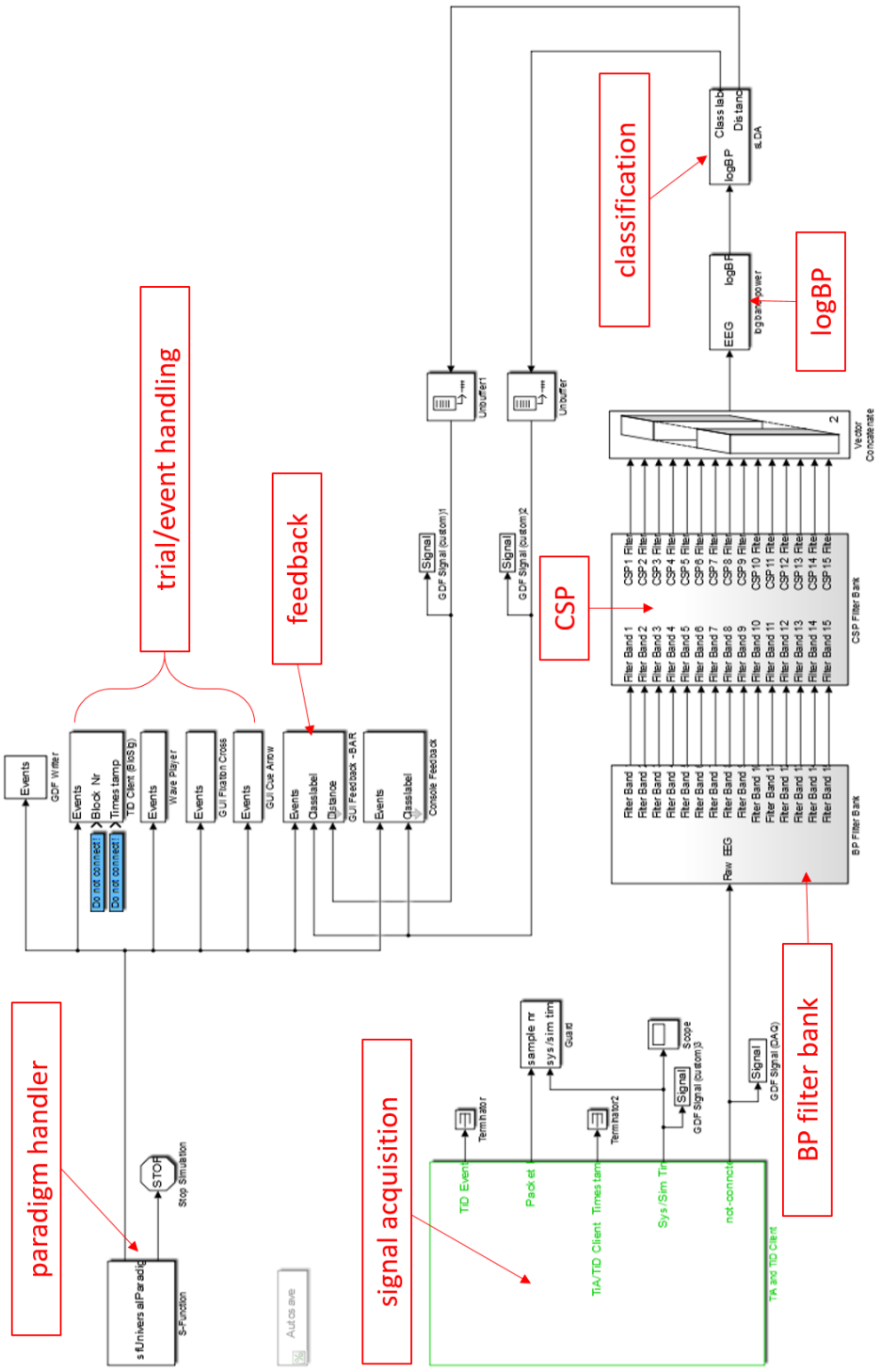


Figure 2.17.: The Matlab/Simulink *BCI* system including the data acquisition, paradigm generation, signal processing and feedback parts.

3 Results

In this chapter, all results of the thesis are stated. It includes the offline evaluation results for all data transfer methods over all sessions and all subjects as well as the online evaluation accuracies of all participants for both sessions.

3.1. Offline Evaluation

3.1.1. Accuracies and Statistical Testing

Tables 3.1 and 3.2 contain the results of the offline evaluation. For each transfer method, the accuracies of each subject were calculated by applying $LOOCV$. The reference system is trained and evaluated solely on the second session and all transfer methods are trained on both sessions and evaluated on the second of the two sessions. The accuracies of the binary classifier were averaged over all subjects for each session and additionally also over all sessions.

Peak refers to the highest achieved accuracy during the feedback period. *Mean* is the averaged accuracy between $2s$ and $5s$ after the cue. *Median* is the median accuracy between $2s$ and $5s$ after the cue.

For an easier comparison of the results, Figures 3.1 and 3.2 contain box-plots, presenting the distribution of the peak- and mean-accuracies of each subject for all session-combinations and each transfer method. A very detailed listing of all accuracies for all subjects over all sessions and each transfer method can be found in Section A.1.

Table 3.1.: Binary validation **peak-accuracies** of the different transfer methods in %. The upper part contains the peak accuracies, averaged over all subjects. The lower part contains the peak-accuracies, averaged over all subjects and sessions.

			Transfer Methods					
Subjects	Sessions	Attribute	Ref.	sCSP	divCSP-MS	divCSP-AS	WT	NA-MEMD
all	1 - 2	Mean	80.89	**84.44**	84.23	83.56	82.12	81.91
		Median	79.49	85.30	**88.72**	85.99	84.29	82.06
		Std	11.65	11.33	10.63	11.99	9.50	10.25
	2 - 3	Mean	**81.39**	79.03	80.57	79.32	78.15	77.13
		Median	84.52	80.22	**84.62**	81.26	79.13	75.85
		Std	12.35	11.63	8.94	9.18	9.00	8.70
	3 - 4	Mean	82.32	**83.54**	80.61	83.37	80.90	78.94
		Median	85.02	82.15	79.12	**86.60**	78.27	76.53
		Std	9.75	10.87	10.05	9.74	7.80	12.64
all	all	Mean	81.53	**82.34**	81.80	82.08	80.39	79.32
		Median	83.01	82.55	84.15	**84.62**	80.56	78.15
		Std	11.25	11.28	9.88	10.30	8.76	10.53

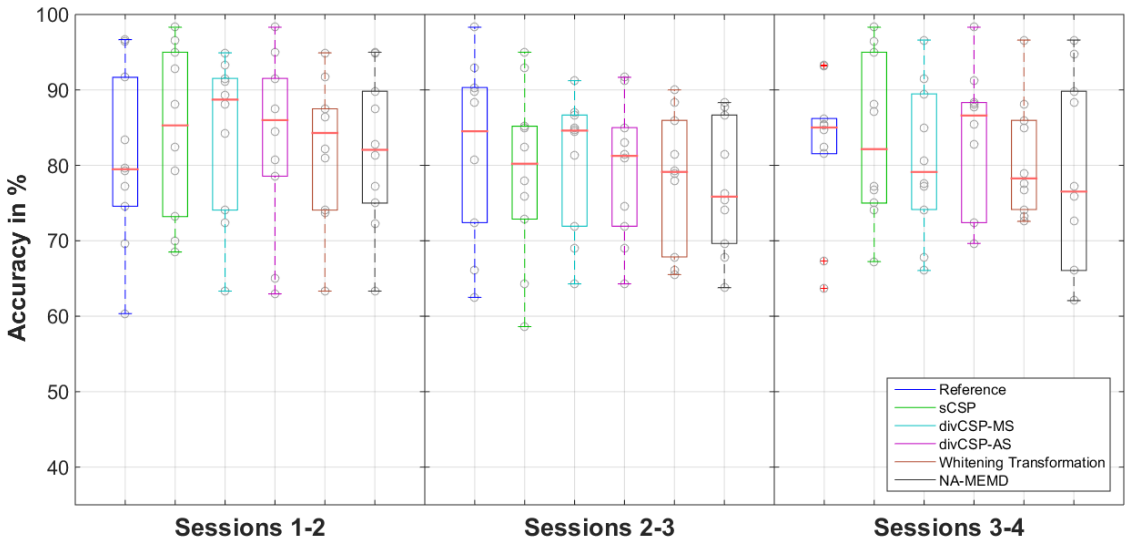


Figure 3.1.: Comparison of the **peak-accuracies** for all transfer methods. The reference system is trained and evaluated solely on the second session and all transfer methods are trained on both sessions and evaluated on the second of the two sessions. Inside the box are 50% of the distribution. The red line inside the box indicates the median of the distribution. The grey circles indicate the accuracies of the subjects. A red cross indicates outlier.

Table 3.2.: Binary validation **mean-accuracies** of the different transfer methods in %. The upper part contains the mean accuracies, averaged over all subjects. The lower part contains the mean-accuracies, averaged over all subjects and sessions. The mean-accuracies are calculated between $2s$ and $5s$ after the cue.

			Transfer Methods					
Subjects	Sessions	Attribute	Ref.	sCSP	divCSP-MS	divCSP-AS	WT	NA-MEMD
all	1 - 2	Mean	70.07	73.11	**73.45**	73.25	70.01	70.54
		Median	67.27	73.82	**73.94**	73.21	72.41	68.91
		Std	13.25	11.48	10.90	12.69	9.64	10.86
	2 - 3	Mean	**71.02**	69.23	69.78	69.41	66.99	66.57
		Median	**73.56**	70.19	70.01	69.91	68.87	65.97
		Std	13.64	12.61	9.67	10.04	10.51	9.25
	3 - 4	Mean	70.26	**72.02**	70.54	71.36	68.21	68.65
		Median	71.67	70.02	69.82	**73.14**	64.85	68.30
		Std	12.92	11.45	11.35	11.83	9.50	13.36
all	all	Mean	70.45	**71.45**	71.26	71.34	68.40	68.59
		Median	70.83	71.34	71.26	**72.08**	68.71	67.73
		Std	13.27	11.85	10.64	11.52	9.88	11.16

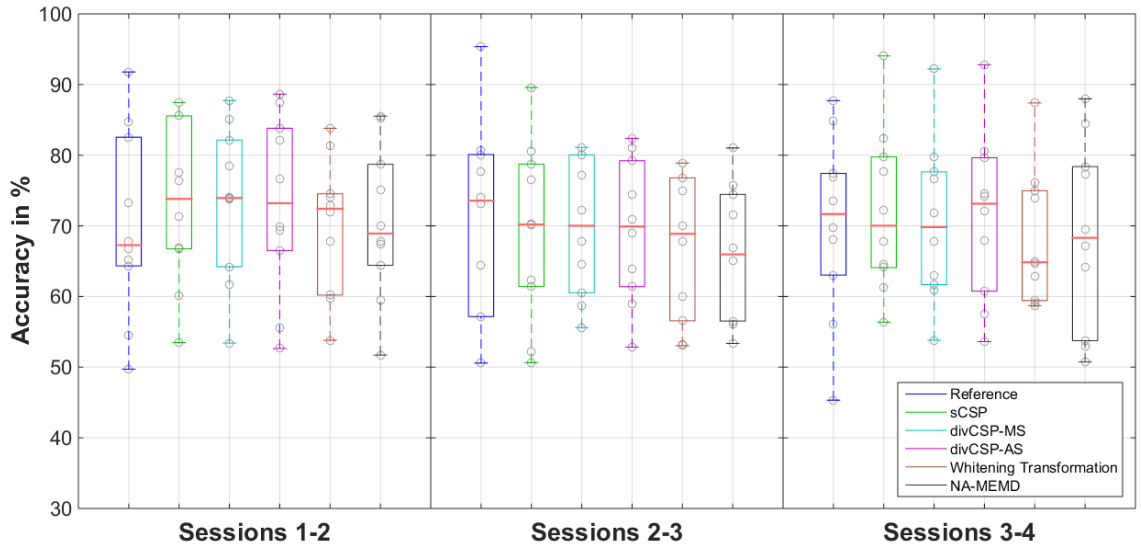


Figure 3.2.: Comparison of the **mean-accuracies** for all transfer methods. The reference system is trained and evaluated solely on the second session and all transfer methods are trained on both sessions and evaluated on the second of the two sessions. Inside the box are 50% of the distribution. The red line inside the box indicates the median of the distribution. The grey circles indicate the accuracies of the subjects. A red cross indicates outlier.

To test the obtained distributions for normality, a *One-sample Kolmogorov-Smirnov test* with a confidence interval of $\alpha = 0.05$ was applied to the data. To indicate differences between the distribution of the reference method and the distribution of a transfer method, a paired-sample t-test was applied with a confidence interval of $\alpha = 0.05$. Table 3.3 shows the related p-values of the statistical tests for each transfer method.

Table 3.3.: Statistical p-values of the one-sample Kolmogorov-Smirnov test for the peak- and mean-accuracy distributions for all methods, to validate their normality (top). Paired t-test to identify differences between each method and the reference (bottom).

		Transfer Methods					
Statistical Test	Attribute	Ref.	sCSP	divCSP-MS	divCSP-AS	WT	NA-MEMD
one-sample KS	Peak Acc.	0.910	0.969	0.729	0.967	0.917	0.926
	Mean Acc.	0.989	0.947	0.711	0.912	0.995	0.716
paired t-test	Peak Acc.	-	0.468	0.792	0.546	0.415	0.137
	Mean Acc.	-	0.386	0.922	0.961	0.132	0.115

3.2. Online Evaluation

3.2.1. Accuracies and Statistical Testing

Table 3.4 contains the results of the online evaluation study of all eleven participants. The participants who had already *BCI* experience were typed *exp* and the participants new to the topic, were typed *nav*. *Peak* refers to the highest achieved accuracy during the feedback period. *Mean* is the averaged accuracy between $2s$ and $5s$ after the cue. *Median* is the median accuracy between $2s$ and $5s$ after the cue. On the first measurement, the *CSP* filter are calculated with the *sCSP* method and on the second measurement, data transfer is achieved with the *divCSP-AS* method. Figure 3.3 shows the accuracies of each subject over time for both sessions. For an easier comparison of the results, Figure 3.4 contains box-plots, presenting the peak- and mean-accuracies of each participant for both sessions.

Table 3.5, shows the amount of days as well as the difference in the day-time between the two measurements for each participant.

To test the obtained distributions for normality, normal probability plots were created. Additionally, a *One-sample Kolmogorov-Smirnov test* with a confidence interval of $\alpha = 0.05$ was applied to the data. To indicate differences between the distributions of both sessions, a paired-sample t-test was applied with a confidence interval of $\alpha = 0.05$.

Table 3.4.: Binary validation accuracies of the different measurements for every participant and every session. The upper part contains information for each participant and the lower part contains the averaged information over all participants.

		Session 1			Session 2		
Participant	type	Peak	Mean	Median	Peak	Mean	Median
P1	nav	98.33	90.60	90.00	99.17	90.14	92.50
P2	nav	65.00	53.78	53.75	61.67	52.66	51.67
P3	exp	81.67	65.96	65.00	85.00	74.89	75.00
P4	nav	83.33	59.96	60.00	83.33	66.29	69.17
P5	exp	76.25	66.88	66.25	85.00	78.09	78.33
P6	exp	100.00	95.85	96.67	97.50	94.82	95.83
P7	exp	80.00	71.33	71.67	77.50	70.52	70.83
P8	nav	80.00	69.77	70.00	81.67	76.83	76.67
P9	exp	90.00	78.14	80.00	80.83	71.70	71.67
P10	exp	96.67	93.06	93.33	92.50	88.01	88.33
P11	nav	62.50	44.60	42.50	64.17	56.09	55.83
	Mean	83.07	71.81	71.74	82.58	74.55	75.08
all	Median	81.67	69.77	70.00	83.33	74.89	75.00
	Std	12.53	16.41	16.98	11.89	13.29	13.81

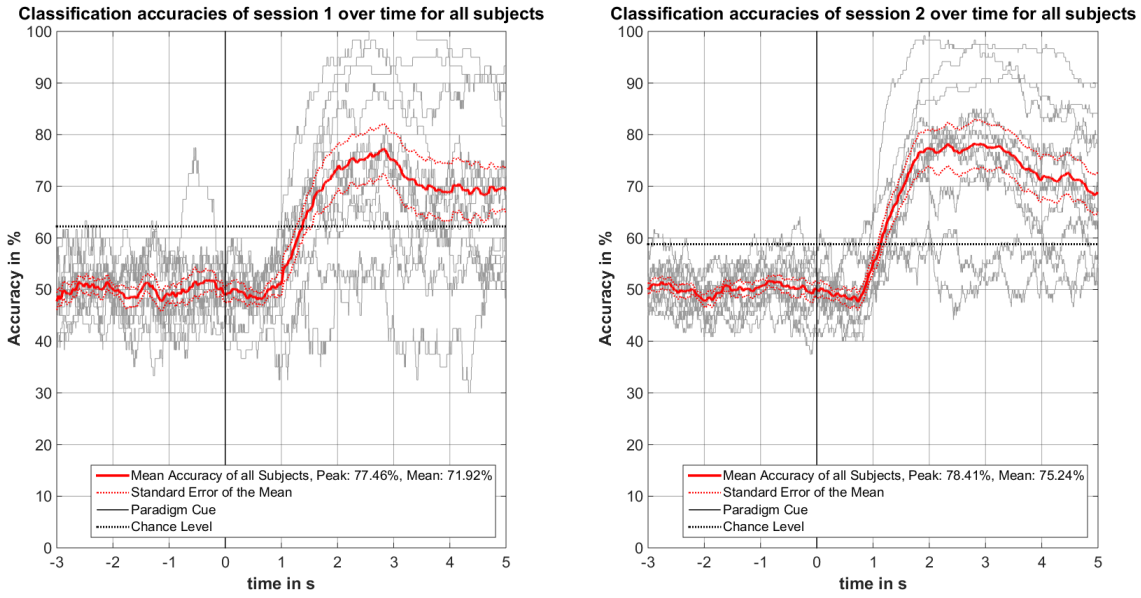


Figure 3.3.: Accuracy curves for all subjects (grey curves) and each of the two sessions. The red curve indicates the mean curve over all subjects along with its standard error of the mean.

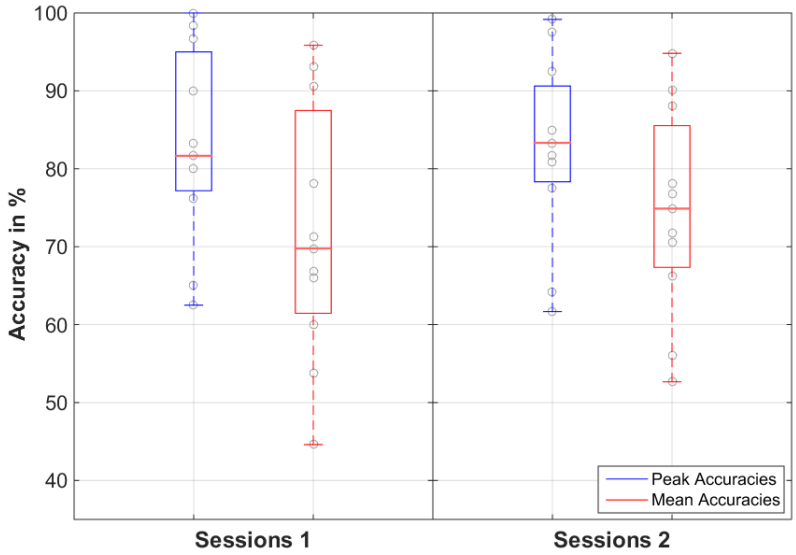


Figure 3.4.: Comparison of the peak- and mean-accuracies for both sessions and all participants. Inside the box are 50% of the distribution. The red line inside the box indicates the median of the distribution. The grey circles indicate the accuracies of the participants. A red cross indicates outlier.

Table 3.5.: Listing of the study details for each participant.

Participant	days between the measurements	day-time difference
P1	1	no
P2	1	no
P3	1	pm to am
P4	1	no
P5	5	no
P6	1	pm to am
P7	1	no
P8	1	no
P9	1	no
P10	2	no
P11	1	no

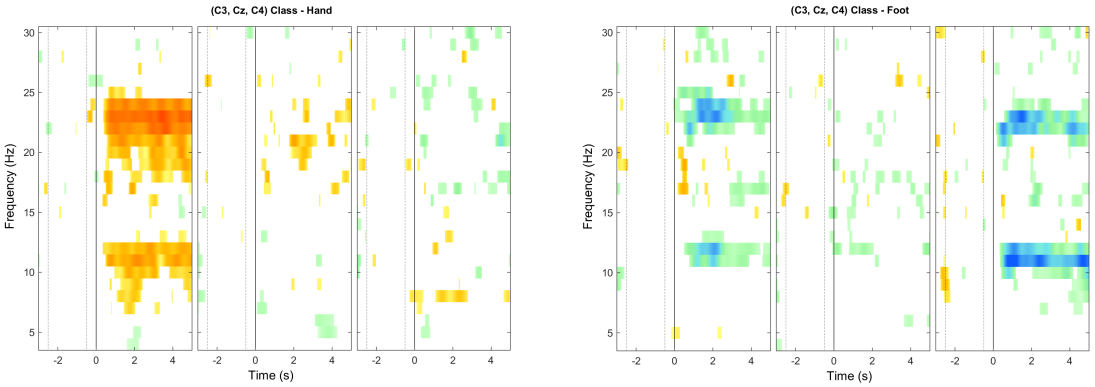
Table 3.6.: Statistical p-values of the one-sample Kolmogorov-Smirnov test for the peak- and mean-accuracy distributions for both sessions, to validate their normality (left). Paired t-test to identify equality between the sessions (right).

One-sample Kolmogorov-Smirnov test				paired t-test	
Session 1		Session 2			
Peak Acc.	Mean Acc.	Peak Acc.	Mean Acc.	Peak Acc.	Mean Acc.
0.974	0.941	0.862	0.990	0.733	0.189

3.2.2. ERDS Maps

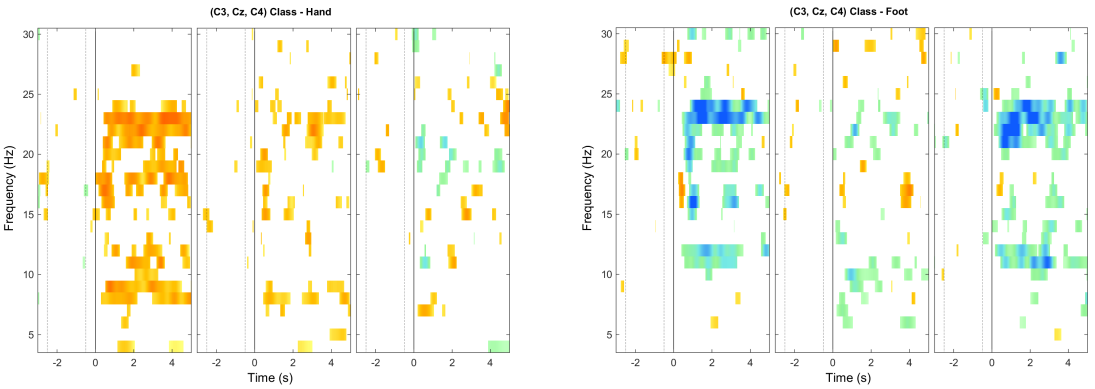
Time-frequency maps were calculated for all subjects from the laplace filtered channels C3, Cz and C4. Data was bandpass filtered between 4 and 30 Hz and the parts between -3 seconds to 5 seconds after the visual cue were considered in the *ERDS* analysis [17] with respect to a reference interval of -2.5 to -0.5 seconds before the visual cue. Statistical significance of the *ERDS* data was ensured by applying a t-percentile bootstrap algorithm with a significance level of alpha = 0.05. Yellow to red colours indicate weak to strong Event-Related Desynchronization (ERD), green to blue colours mark weak to strong Event-Related Synchronization (ERS).

Figure 3.5 and 3.6 shows an exemplary result of subject P10 from session 1 and session 2 for both classes respectively. Figure 3.7 and 3.8 shows an exemplary result of subject P11 from session 1 and session 2 for both classes respectively. The remaining *ERDS* maps of all subjects can be seen in Section A.2



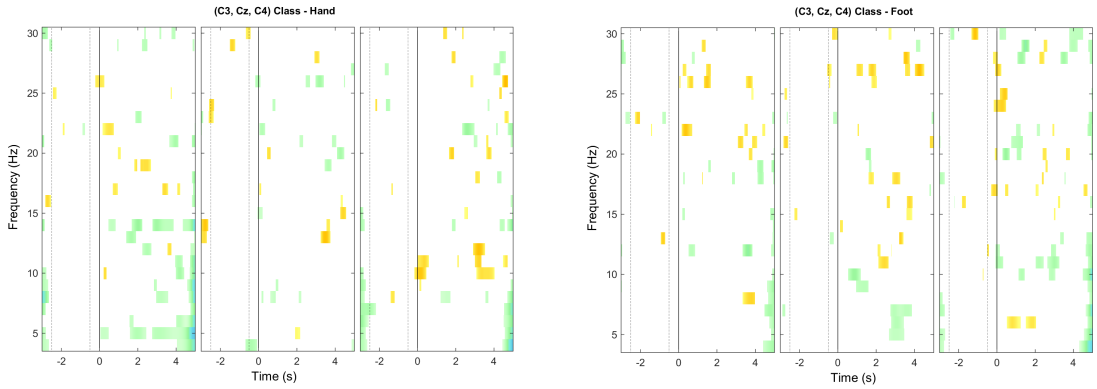
(a) class right hand **(b)** class both feet

Figure 3.5.: ERDS Maps of subject P10 for the training data of session 1 for both classes and electrodes C3, Cz and C4



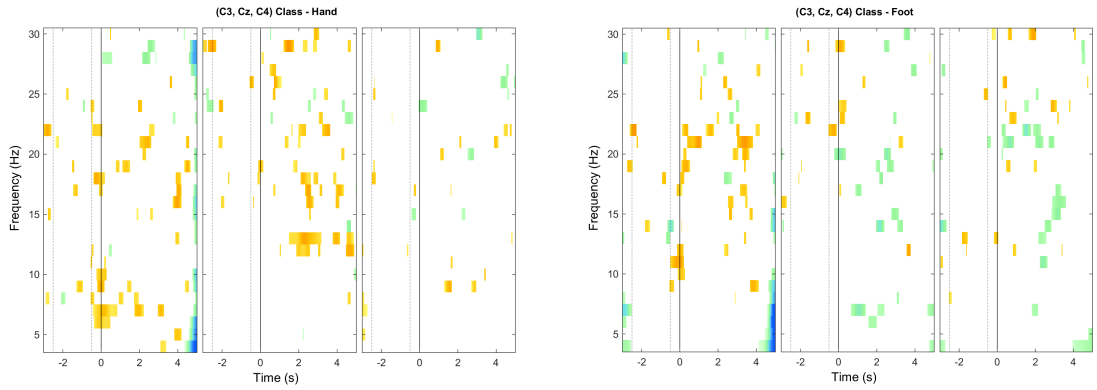
(a) class right hand **(b)** class both feet

Figure 3.6.: ERDS Maps of subject P10 for the training data of session 2 for both classes and electrodes C3, Cz and C4



(a) class right hand **(b)** class both feet

Figure 3.7.: ERDS Maps of subject P11 for the training data of session 1 for both classes and electrodes C3, Cz and C4



(a) class right hand **(b)** class both feet

Figure 3.8.: ERDS Maps of subject P11 for the training data of session 2 for both classes and electrodes C3, Cz and C4

4 Discussion

In this chapter, the obtained results of the offline evaulation as well as of the online evaluation study are discussed in detail and are questioned towards their validity.

4.1. Offline Evaluation

The system of the offline evaluation was developed to be as close to the online system as possible. The performance evaluation is not the same due to the lack of feedback runs and available training data (30 trials per class per session). Therefore, *LOOCV* was implemented to evaluate the performance of the data transfer methods. The training data for the system was therefore created out of two subsequent sessions, whereby the performance was solely evaluated on the latter session. The recorded electrodes were selected to cover the scalp areas over motor regions of the brain and that they are placed on similar positions as in the online electrode setup. In addition, the signal processing pipeline and the parameters used for every transfer method were chosen accordingly to their usage in the latter online study.

All methods were not significantly different in comparison to the reference method, which was solely trained and tested on the posterior of the used sessions. Some methods as the *sCSP* or *divCSP* performed numerically better and other methods as the *WT* or *NA-MEMD* performed numerically worse than the reference method. Due to that findings, one can conclude, that data transfer is possible with all methods and without a significant loss of performance.

Shrinkage Common Spatial Patterns - sCSP
The *sCSP* method is easy to implement due to the available software and is more and more used on our institute. The combination of data from different sessions is already a form of regularization because the covariance matrix is calculated on the concatenated trials of both sessions and will therefore be a mixture of the covariance matrices of the single sessions. The *sCSP* does not consider any regularization against non-stationarities and should therefore work well under conditions where the patterns of the user hardly change between the sessions. The *sCSP* had the highest peak- and mean-accuracies among all methods in terms of the averaged mean performance over all subjects and sessions. The method is not significantly better or worse than the reference method or any of the other methods. The averaged peak- and mean-performance was numerically better and the median performance was numerically lower than the *divCSP* methods. Using this method alone without any special regularization approaches is very suitable for performing data transfer between sessions, due to its simple application.

Across Session/Subject Stationarity Divergence CSP - divCSP-AS
This method performs a regularization against non-stationarities between the signal distributions of the two sessions. This regularization is influencing the *CSP* calculation with a weight of 30%. This weight was selected as a tradeoff between the regularization and the calculation of the filter itself. Therefore, regularization should not influence the calculation too much. The performance was numerically better as with the *divCSP-MS*. When averaging the performance over all sessions and subjects, the *sCSP* had a higher mean value but a lower median and a higher standard deviation which indicates that the method had a higher variability in the performance and was not as stable as the *divCSP-AS*. The methods were not significantly different from the reference method and hence also not between each other. The method may had a better influence on poor performing subjects due to its lower weighting of data from the previous session and because of the regularization against non-stationarities.

Multi Session/Subject Stationarity Divergence CSP - divCSP-MS
The *divCSP-MS* method provided also high numerical accuracies, compared to the reference. In comparison with the *divCSP-AS*, the performance was numerically lower. The method calculates the *CSP* filter by maximizing the *CSP* equation for both sessions simultaneously. Therefore each session had the same weight in the calculation. A suggestion to yield a better performance may be to change the regularization term. It should therefore contain the *CSP* equation of the first session instead of both and the *CSP* term should consist of the *CSP* equation of the current session. To weight the actual session more than the last session, the regularization parameter λ can be used. This method is considered equal to the *divCSP-AS* and the *sCSP* method because all methods are not significantly different than the reference method and hence also not between each other.

Whitening Transformation - WT
The *WT* performed slightly worse than the reference method but can be considered equal to the reference. The *WT* was calculated on the same signal windows as the *CSP* filter. It was therefore no surprise that the *WT* performed not as good as the *CSP* methods, because the *CSP* calculation implies a whitening step and is therefore an enhancement of the *WT*. Also the fact that the *WT* is only applied on two bandpass filtered sub-bands instead of 15 reduces the separability of the frequency based features. The reduction of sub-bands was necessary due to the otherwise badly placed trial to feature ratio. Nevertheless, the whitening step alone yields also feasible performances.

Noise Assisted Multivariate Empirical Mode Decomposition - NA-MEMD
The *NA-MEMD* performed numerically worst of all methods in terms of the overall peak- and mean-accuracies. The method is usually applied to offline data and was therefore transformed into a windowed version which can also be applied to online signals. This process included a number of limitations due to the immense computational effort of the method. The windowed approach with window durations of 6s was therefore necessary to finish the *NA-MEMD* calculation faster than the duration of the actual window to ensure functionality in an online *BCI*. As a result of the longer window durations, the different *IMFs* should be easier to extract due to the higher amount of extreme values contained in the window. The quality of the extracted *IMFs* was reduced in consequence of the configured limitations of the method, which were:

- reduced number of extracted *IMFs*
- lower amount of projection directions to extract extreme values
- lower amount of sifting iterations per *IMF*
- higher tolerance in the stoppage criteria for an *IMF*

These limitations may had an influence on the performance of the method but further studies especially designed on the *EMD* and its usage in online *BCIs* are needed to verify these points. The *NA-MEMD* was the best choice among the *EMD* methods due to its multidimensional nature and because of the suppression of mode-mixing, which is a critical point in this windowed approach. The *NA-MEMD* has also a dyadic filter bank property which can lead to mode-mixing if the oscillation frequencies are close to each other. This may also lead to a decreased performance. The biggest drawback in comparison to the bandpass filter bank *CSP* is the limited amount of sub-bands. The bandpass filter bank devides the signal into 15 sub-bands whereas the *NA-MEMD* separates the signal into only four *IMFs*. The results are still not significantly different than the results of the reference method. Furthermore, Gaur et al. [15] obtained accuracies about 80-82% (unknown if peak- or mean-accuracy) by combining the *IMFs* corresponding to motor rhythms to

create an enhanced *EEG*. In a follow up study [16], they yielded accuracies about 70-73% (unknown if peak- or mean-accuracy) with the same signal processing methods but additionally they incorporated training data from different users. The results from these studies are similar to those obtained in this evaluation.

4.2. Online Evaluation

Eleven participants took part in the online evaluation study. They were asked to control an online *BCI* by the use of motor imagery tasks on two different days.

4.2.1. Experimental Setup

The first session was meant as a reference to gain insight into the performance level of the participants. Six training runs to collect 60 trials per class and three feedback runs to determine the performance of the participants were conducted. During the second session, three training runs from the first session were selected according to the subjective rating of the participants (concentration, well being, estimated performance, amount of produced artefacts). Furthermore, the first run of naïve subjects was not used because they often needed more time to adapt to the paradigm. The last run was also often not incorporated, due to fatigue of the subjects.

Some participants mentioned difficulties with the feedback because the feedback bar could also alternate between the classes. Over time, the participants got used to the feedback and were more encouraged to concentrate on the task.

81% of the second measurements took place on the next day and the same time of the day. The participants were asked whether they were in the same condition as on the first session to avoid a bias towards better or worse results.

4.2.2. Performance

The 95% confidence interval for chance level was determined to be 62,25% in the first and 58,8% in the second session. A subject was considered to perform better than random, when its mean accuracy was greater than chance level. Participants P2 and P11 were therefore not able to control the *BCI* system because their mean accuracy was lower then the chance level. The *ERDS* maps of P11 and P2 are also showing no strong, distinguishable patterns. A possible reason is that the named participants are among the 30% of people, who are not able to control a *BCI* because their *ERDS* patterns cannot be detected in their *EEG* [1].

The overall peak performance of the second session was lower than in the first session. On the contrary, the mean and median performances increased and the

standard deviation decreased. The increase in the mean accuracy can be seen in Figure 3.3 with the prolonged plateau phase of the feedback period in the second session.

Among the naïve participants, P1, P4 and P8 increased their performance in the second session whereas P1 had already one of the best performances from the beginning. P1 and P4 had more pronounced *ERDS* patterns in the second session than in the first session which could explain the boost in performance. The patterns of P8 were not changing which indicate that the boost could be due to a learning step of the participant.

Among the expert participants, only P3 and P5 increased their performance. Participant P3 had the second session on the following day in the morning and reported that the concentration was better in the morning, which could be the explanation of the performance boost. P5 had its second session five days after the first session but on the same time of day. P5 reported that the concentration was also better on the second session which may explain the boost. Also the *ERDS* pattern of P3 and P5 were more pronounced in the second session than in the first session.

Participants P6, P7 and P10 kept their level of performance. They reported the same state of well being and concentration for both sessions. P6 had less *ERS* over C3 and C4 and more *ERD* over Cz for feet imagination in the second session. The patterns for imagination of the right hand stayed the same. P7 had a *ERD* change over C4 for both classes between the sessions. The patterns of P10 stayed the same over the sessions. The small deviations in the *ERDS* maps can be explained with the decreased training set of 30 trials per class for the second session. Nevertheless, participants P6, P7 and P10 had a small decrease of performance, which can be explained by a normal variation of the results between different days or with an effect of the *divCSP-AS*. The *divCSP-AS* regularizes against the differences between sessions for each class. When the differences are not pronounced, the regularization term will vanish and the *CSP* calculation will transform into a normal, unregularized version. This may be a limitation of this method.

All other experts dropped in performance. P9 dropped 9% in peak and 6.5% in mean but reported that the concentration was worse on the second session which is also visible in the *ERDS* maps, where the patterns changed considerably.

The one-sample Kolmogorov-Smirnov test yielded in p-values above 0.05 for the accuracy distributions. That means, that the null-hypothesis can not be rejected and that it is very likely that all data samples follow a normal distribution. Therefore a paired t-test can be applied to the paired data values. The p-values are above 0.05 which means that the null-hypothesis can not be rejected. It is therefore very unlikely, that the accuracy distributions of the two sessions do not originate from the same distribution.

This indicates, that no loss of performance has been obtained, by incorporating data from the previous session and thus reducing the calibration time.

4.2.3. Limitations

A critical point is to compare the performance of the second session with the performance of the first session without neglecting the improvement and learning of the subjects between the sessions. The best solution would be the comparison of two groups, where the system of the first group is trained solely on training data from the current session and the system of the second group is trained on a combination of the data from both sessions. To gain a statistical certainty about the results, at least 10 subjects should be in a group which would lead to a total of 20 subjects and 40 *BCI* measurements in total. This would go beyond the scope of this thesis, due to the main focus lied on the functionality of the *BCI* system, with a shortened calibration time. To overcome this limitation, only expert users could be measured due to their stabilized *ERDS* patterns. Expert users are hard to find and even then it is not guaranteed that they can perform equally good in both sessions due to the complexity of the mental condition.

5 Conclusions and Future Work

5.1. Conclusions

The goal of this thesis was to find a suitable method to achieve data transfer from one session to another in order to reduce the calibration time of the *BCI* system. The hypothesis to be tested was that such a system using only half of the data, obtained in the calibration phase, works as good as a standard *BCI* system (averaged performance $> 70\%$). Several different data transfer methods, which yielded promising results were therefore tested against a reference method in an offline evaluation study to gain insight into the performances of the methods and to select a suitable candidate among the tested for an online evaluation study. The *divCSP-AS* method achieved promising results and was therefore implemented in an online *BCI* system. To evaluate online performance, a group of eleven subjects was tested on two different days in an online study: the first measurement day was to assess the performance of the subjects by using a standard *BCI* system while on the second measurement, only half of the calibration data was recorded and the other half was incorporated from the first measurement by using the data transfer method.

Results of the two measurements showed that the starting hypothesis can be accepted: The averaged performance of the second session was above 70% (Peak Accuracy: 82.58% , Mean Accuracy: 74.55%) and there were no significant differences in mean or peak accuracies between the two measurements with a significance level of $\alpha = 0.05$. Therefore the results suggest, that it is possible to use data of previous sessions to reduce the calibration time in future sessions.

5.2. Future Work

The results of this work suggest that a calibration time reduction by incorporating calibration data from previous sessions is useful for *SMR* based *BCIs*. An obvious next step would be to test other methods like the *sCSP* or *divCSP-MS* and test them against the results obtained in this study.

As explained above, the *divCSP-AS* may perform worse if the *ERDS* patterns are very similar between the sessions. Therefore this could be tested by conducting a similar study with only expert *BCI* users, with the *divCSP-AS* and the *sCSP* as transfer methods.

This procedure to reduce calibration time can be used in every *BCI* application. However the combination with adaptive *BCI* systems, as semi-supervised systems, seems appropriate to cancel the initial calibration phase by using the data from previous sessions. This could lead to a *BCI* system with no need of a calibration phase and could make a difference in making *BCIs* more practicable and bringing the systems out of the lab towards real-world applications.

Bibliography

[1] Allison, B. Z. and Neuper, C. (2010). Could anyone use a BCI? In *Brain-Computer Interfaces: Applying our Minds to Human-Computer Interaction*, pages 35–54. Springer. (page 56)

[2] Ang, K. K., Chin, Z. Y., Wang, C., Guan, C., and Zhang, H. (2012). Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b. *Frontiers in neuroscience*, 6:39. (page 24)

[3] Bauer, G., Gerstenbrand, F., and Rumpl, E. (1979). Varieties of the locked-in syndrome. *Journal of neurology*, 221(2):77–91. (page 1)

[4] Bell, A. J. and Sejnowski, T. J. (1997). The independent components of natural scenes are edge filters. *Vision research*, 37(23):3327–3338. (page 16)

[5] Berger, H. (1929). Über das Elektroenkephalogramm des Menschen. *European Archives of Psychiatry and Clinical Neuroscience*, 87(1):527–570. (page 2)

[6] Billinger, M., Daly, I., Kaiser, V., Jin, J., Allison, B. Z., Müller-Putz, G. R., and Brunner, C. (2013). Is it significant? Guidelines for reporting BCI performance. In *Towards Practical Brain-Computer Interfaces*, pages 333–354. Springer. (page 32, 33, 38)

[7] Blankertz, B., Lemm, S., Treder, M., Haufe, S., and Müller, K.-R. (2011). Single-trial analysis and classification of ERP components-a tutorial. *NeuroImage*, 56(2):814–825. (page 30, 31, 32)

[8] Blankertz, B., Tomioka, R., Lemm, S., Kawanabe, M., and Muller, K.-R. (2008). Optimizing spatial filters for robust EEG single-trial analysis. *IEEE Signal processing magazine*, 25(1):41–56. (page 7)

[9] Breitwieser, C., Neuper, C., and Müller-Putz, G. R. (2011). A concept to standardize raw biosignal transmission for brain-computer interfaces. In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*, pages 6377–6380. IEEE. (page 41)

[10] Dwinnell, W. and Sevis, D. (2010). LDA: Linear Discriminant Analysis. *Matlab Central File Exchange*, 29673. (page 31)

[11] Faller, J., Vidaurre, C., Solis-Escalante, T., Neuper, C., and Scherer, R. (2012). Autocalibration and recurrent adaptation: Towards a plug and play online ERD-BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 20(3):313–319. (page 4, 41)

[12] Flandrin, P., Gonçalves, P., and Rilling, G. (2005). EMD equivalent filter banks, from interpretation to applications. *Hilbert-Huang transform and its applications*, pages 57–74. (page 20)

[13] Friedman, J. H. (1987). Exploratory projection pursuit. *Journal of the American statistical association*, 82(397):249–266. (page 16)

[14] Friedrich, E., Scherer, R., and Neuper, C. (2012). The effect of distinct mental strategies on classification performance for Brain-Computer interfaces. *International Journal of Psychophysiology*. (page 28, 29)

[15] Gaur, P., Pachori, R. B., Wang, H., and Prasad, G. (2015). An empirical mode decomposition based filtering method for classification of motor-imagery EEG signals for enhancing brain-computer interface. In *Neural Networks (IJCNN), 2015 International Joint Conference on*, pages 1–7. IEEE. (page 24, 55)

[16] Gaur, P., Pachori, R. B., Wang, H., and Prasad, G. (2016). A multivariate empirical mode decomposition based filtering for subject independent BCI. In *Signals and Systems Conference (ISSC), 2016 27th Irish*, pages 1–7. IEEE. (page 4, 6, 24, 56)

[17] Graimann, B., Huggins, J., Levine, S., and Pfurtscheller, G. (2002). Visualization of significant ERD/ERS patterns in multichannel EEG and ECoG data. *Clinical Neurophysiology*, 113(1):43–47. (page 50, 73)

[18] Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., Yen, N.-C., Tung, C. C., and Liu, H. H. (1998). The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. In *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 454, pages 903–995. The Royal Society. (page 16, 17, 18, 20)

[19] Kessy, A., Lewin, A., and Strimmer, K. (2015). Optimal whitening and decorrelation. *arXiv preprint arXiv:1512.00809*. (page 16)

[20] Kindermans, P.-J., Tangermann, M., Müller, K.-R., and Schrauwen, B. (2014). Integrating dynamic stopping, transfer learning and language models in an adaptive zero-training ERP speller. *Journal of neural engineering*, 11(3):035005. (page 4)

[21] Kohavi, R. et al. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Stanford, CA. (page 32)

[22] Kübler, A., Kotchoubey, B., Kaiser, J., Wolpaw, J. R., and Birbaumer, N. (2001). Brain-computer communication: unlocking the locked in. *Psychological bulletin*, 127:358–375. (page 6)

[23] Lotte, F. (2015). Signal processing approaches to minimize or suppress calibration time in oscillatory activity-based brain–computer interfaces. *Proceedings of the IEEE*, 103(6):871–890. (page 3, 4)

[24] Lotte, F., Congedo, M., Lécuyer, A., Lamarche, F., and Arnaldi, B. (2007). A review of classification algorithms for EEG-based brain-computer interfaces. *Journal of neural engineering*, 4:R1–R13. (page 31)

[25] Lotte, F. and Guan, C. (2011). Regularizing common spatial patterns to improve BCI designs: unified theory and new algorithms. *IEEE Transactions on biomedical Engineering*, 58(2):355–362. (page 4, 9, 10)

[26] McFarland, D. J., Miner, L. A., Vaughan, T. M., and Wolpaw, J. R. (2000). Mu and beta rhythm topographies during motor imagery and actual movements. *Brain topography*, 12(3):177–186. (page 3)

[27] Millán, J. d. R., Rupp, R., Müller-Putz, G., Murray-Smith, R., Giugliemma, C., Tangermann, M., Vidaurre, C., Cincotti, F., Kubler, A., Leeb, R., et al. (2010). Combining brain–computer interfaces and assistive technologies: state-of-the-art and challenges. *Frontiers in neuroscience*, 4:161. (page 1, 5)

[28] Müller-Putz, G., Leeb, R., Tangermann, M., Höhne, J., Kübler, A., Cincotti, F., Mattia, D., Rupp, R., Müller, K.-R., and Millán, J. d. R. (2015). Towards noninvasive hybrid brain–computer interfaces: framework, practice, clinical application, and beyond. *Proceedings of the IEEE*, 103(6):926–943. (page 5, 40)

[29] Müller-Putz, G., Schwarz, A., and Steyrl, D. (2016). Mirage91: The Graz BCI-Racing Team-making students familiar with BCI research. *Proceedings of the 6th International Brain-Computer Interface Meeting, organized by the BCI Society.* (page 5)

[30] Murata, N., Takenouchi, T., Kanamori, T., and Eguchi, S. (2004). Information geometry of u-boost and bregman divergence. *Neural Computation*, 16(7):1437–1481. (page 11)

[31] Neumaier, A. (1998). Solving ill-conditioned and singular linear systems: A tutorial on regularization. *SIAM review*, 40(3):636–666. (page 3)

[32] Neuper, C., Scherer, R., Reiner, M., and Pfurtscheller, G. (2005). Imagery of motor actions: Differential effects of kinesthetic and visual–motor mode of imagery in single-trial eeg. *Cognitive brain research*, 25(3):668–677. (page 38)

[33] Nunez, P. L., Srinivasan, R., Westdorp, A. F., Wijesinghe, R. S., Tucker, D. M., Silberstein, R. B., and Cadusch, P. J. (1997). EEG coherency: I: statistics, reference electrode, volume conduction, Laplacians, cortical imaging, and interpretation at multiple scales. *Electroencephalography and clinical neurophysiology*, 103(5):499–515. (page 7)

[34] Park, C., Looney, D., ur Rehman, N., Ahrabian, A., and Mandic, D. P. (2013). Classification of motor imagery BCI using multivariate empirical mode decomposition. *IEEE Transactions on neural systems and rehabilitation engineering*, 21(1):10–22. (page 24)

[35] Pfurtscheller, G. and Aranibar, A. (1977). Event-related cortical desynchronization detected by power measurements of scalp EEG. *Electroencephalography and clinical neurophysiology*, 42(6):817–826. (page 3)

[36] Pfurtscheller, G. and Lopes Da Silva, F. H. (1999). Event-related EEG/MEG synchronization and desynchronization: Basic principles. (page 3, 8)

[37] Pfurtscheller, G. and Neuper, C. (2001). Motor imagery and direct brain- computer communication. *Proceedings of the IEEE*, 89(7):1123–1134. (page 38, 39)

[38] Pfurtscheller, G., Neuper, C., Flotzinger, D., and Pregenzer, M. (1997). EEG-based discrimination between imagination of right and left hand movement. *Electroencephalography and clinical Neurophysiology*, 103(6):642–651. (page 3, 23)

[39] Pfurtscheller, G., Neuper, C., Muller, G., Obermaier, B., Krausz, G., Schlogl, A., Scherer, R., Graimann, B., Keinrath, C., Skliris, D., et al. (2003). Graz-BCI: state of the art and clinical applications. *IEEE Transactions on neural systems and rehabilitation engineering*, 11(2):1–4. (page 6)

[40] Ramoser, H., Muller-Gerking, J., and Pfurtscheller, G. (2000). Optimal spatial filtering of single trial eeg during imagined hand movement. *IEEE transactions on rehabilitation engineering*, 8(4):441–446. (page 7, 8)

[41] Rehman, N. and Mandic, D. P. (2010a). Empirical mode decomposition for trivariate signals. *IEEE Transactions on signal processing*, 58(3):1059–1068. (page 22)

[42] Rehman, N. and Mandic, D. P. (2010b). Multivariate empirical mode decomposition. In *Proceedings of The Royal Society of London A: Mathematical, Physical and Engineering Sciences*, volume 466, pages 1291–1302. The Royal Society. (page 22, 24)

[43] Rehman, N. and Mandic, D. P. (2011). Filter bank property of multivariate empirical mode decomposition. *IEEE Transactions on Signal Processing*, 59(5):2421–2426. (page 23)

[44] Rehman, N., Park, C., Huang, N. E., and Mandic, D. P. (2013). EMD via MEMD: multivariate noise-aided computation of standard EMD. *Advances in Adaptive Data Analysis*, 5(02):1350007. (page 23, 31, 35)

[45] Reuderink, B. and Poel, M. (2008). Robustness of the common spatial patterns algorithm in the BCI-pipeline. (page 8)

[46] Riener, R. and Seward, L. J. (2014). Cybathlon 2016. In *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*, pages 2792–2794. IEEE. (page 5)

[47] Rilling, G., Flandrin, P., Goncalves, P., et al. (2003). On empirical mode decomposition and its algorithms. In *IEEE-EURASIP workshop on nonlinear signal and image processing*, volume 3, pages 8–11. IEEER. (page 18, 24)

[48] Rilling, G., Flandrin, P., Gonçalves, P., and Lilly, J. M. (2007). Bivariate empirical mode decomposition. *IEEE signal processing letters*, 14(12):936–939. (page 21)

[49] Rohm, M., Schneiders, M., Müller, C., Kreilinger, A., Kaiser, V., Müller-Putz, G. R., and Rupp, R. (2013). Hybrid brain–computer interfaces and hybrid neuroprostheses for restoration of upper limb functions in individuals with high-level spinal cord injury. *Artificial intelligence in medicine*, 59(2):133–142. (page 5)

[50] Samek, W., Blythe, D., Müller, K.-R., and Kawanabe, M. (2013a). Robust spatial filtering with beta divergence. In *Advances in Neural Information Processing Systems*, pages 1007–1015. (page 34)

[51] Samek, W., Kawanabe, M., and Muller, K.-R. (2014). Divergence-based framework for common spatial patterns algorithms. *IEEE Reviews in Biomedical Engineering*, 7:50–72. (page 6, 11, 12, 13, 14, 34)

[52] Samek, W., Meinecke, F. C., and Müller, K.-R. (2013b). Transferring subspaces between subjects in brain–computer interfacing. *IEEE Transactions on Biomedical Engineering*, 60(8):2289–2298. (page 11)

[53] Samek, W., Müller, K.-R., Kawanabe, M., and Vidaurre, C. (2012a). Brain-computer interfacing in discriminative and stationary subspaces. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 2873–2876. IEEE. (page 10)

[54] Samek, W., Vidaurre, C., Müller, K.-R., and Kawanabe, M. (2012b). Stationary common spatial patterns for brain–computer interfacing. *Journal of neural engineering*, 9(2):026013. (page 10)

[55] Schäfer, J. and Strimmer, K. (2005). A Shrinkage Approach to Large-Scale Covariance Matrix Estimation and Implications for Functional Genomics. *Statistical Applications in Genetics and Molecular Biology*, 4(1):32. (page 9)

[56] Scherer, R., Schloegl, A., Lee, F., Bischof, H., Janša, J., and Pfurtscheller, G. (2007). The self-paced Graz brain-computer interface: methods and applications. *Computational intelligence and neuroscience*, 2007. (page 5)

[57] Schwarz, A., Steyrl, D., and Müller-Putz, G. (2016). Brain-computer interface adaptation for an end user to compete in the cybathlon. *Proceedings of the 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2016)*. (page 5)

[58] Statthaller, K., Schwarz, A., Steyrl, D., Kobler, R., Höller, M. K., Brandstetter, J., Hehenberger, L., Bigga, M., and Müller-Putz, G. (2017). Cybathlon experiences of the Graz BCI Racing Team MIRAGE91. *Journal of Neuro Engineering and Rehabilitation*. (page 5)

[59] Steyrl, D., Scherer, R., Faller, J., and Müller-Putz, G. R. (2016). Random forests in non-invasive sensorimotor rhythm brain-computer interfaces: a practical and convenient non-linear classifier. *Biomedical Engineering/Biomedizinische Technik*, 61(1):77–86. (page 30)

[60] Steyrl, D., Scherer, R., Förstner, O., and Müller-Putz, G. R. (2014). Motor imagery brain-computer interfaces: random forests vs regularized lda–non-linear beats linear. In *6th International Brain-Computer Interface Conference*, pages 1–4. (page 32)

[61] Vidal, J. J. (1973). Toward direct brain-computer communication. *Annual review of Biophysics and Bioengineering*, 2(1):157–180. (page 3)

[62] Vidaurre, C., Sannelli, C., Müller, K.-R., and Blankertz, B. (2011). Machine-Learning-Based Coadaptive Calibration for Brain-Computer Interfaces. (page 4)

[63] Von Bünau, P., Meinecke, F. C., Király, F. C., and Müller, K.-R. (2009). Finding stationary subspaces in multivariate time series. *Physical review letters*, 103(21):214101. (page 9)

[64] Wojcikiewicz, W., Vidaurre, C., and Kawanabe, M. (2011). Stationary common spatial patterns: towards robust classification of non-stationary eeg signals. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 577–580. IEEE. (page 10)

[65] Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., and Vaughan, T. M. (2002). Brain-computer interfaces for communication and control. *Clinical neurophysiology : official journal of the International Federation of Clinical Neurophysiology*, 113:767–791. (page 1, 2, 3)

[66] Wu, Z. and Huang, N. E. (2009). Ensemble empirical mode decomposition: a noise-assisted data analysis method. *Advances in adaptive data analysis*, 1(01):1–41. (page 21)

A

Supplementary Material

A.1. Detailed Measurement Results of the Offline Evaluation

Table A.1.: Peak accuracies of every subject for every transfer method and every session.

		Transfer Methods					
Subject	Sessions	Ref.	sCSP	divCSP-MS	divCSP-AS	WT	NA-MEMD
S1	1 - 2	96.67	**98.31**	91.53	91.53	86.44	89.83
	2 - 3	**98.31**	95.00	86.67	85.00	90.00	86.67
	3 - 4	93.22	**98.31**	96.61	**98.31**	96.61	96.61
S2	1 - 2	**96.49**	95	93.33	95	91.67	95
	2 - 3	88.33	**92.98**	91.23	91.23	85.96	87.72
	3 - 4	93.33	**95.00**	85.00	88.33	85.00	88.33
S3	1 - 2	74.58	88.14	88.14	**91.53**	86.44	81.36
	2 - 3	**90.32**	77.97	84.75	83.05	77.97	76.27
	3 - 4	86.21	**87.10**	80.65	85.48	72.58	72.58
S4	1 - 2	79.66	82.46	**84.21**	80.70	73.68	77.19
	2 - 3	72.41	72.88	**81.36**	74.58	66.10	67.80
	3 - 4	**81.54**	67.24	74.14	72.41	74.14	62.07
S5	1 - 2	69.64	68.52	**74.07**	62.96	**74.07**	72.22
	2 - 3	62.50	64.29	64.29	64.29	67.86	**69.64**
	3 - 4	67.27	**76.79**	66.07	69.64	73.21	66.07
S6	1 - 2	91.67	96.61	94.92	**98.31**	94.92	94.92
	2 - 3	**92.98**	85.00	85.00	91.67	88.33	88.33
	3 - 4	85.29	**96.49**	89.47	91.23	85.96	94.74
S7	1 - 2	60.34	**70.00**	63.33	65.00	63.33	63.33
	2 - 3	66.07	58.62	**68.97**	**68.97**	65.52	63.79
	3 - 4	63.64	75.00	67.86	69.64	**76.79**	66.07
S8	1 - 2	83.33	**92.86**	89.29	87.50	82.14	87.50
	2 - 3	**89.83**	85.19	87.04	81.48	81.48	81.48
	3 - 4	82.46	88.14	**91.53**	88.14	88.14	89.83
S9	1 - 2	79.31	73.21	**91.07**	78.57	87.50	75.00
	2 - 3	80.70	75.86	**84.48**	81.03	79.31	74.14
	3 - 4	84.75	77.19	77.19	**87.72**	78.95	77.19
S10	1 - 2	77.19	79.31	72.41	**84.48**	81.03	82.76
	2 - 3	72.41	**82.46**	71.93	71.93	78.95	75.44
	3 - 4	85.45	74.14	77.59	**82.76**	77.59	75.86

Table A.2.: Averaged peak accuracies of every subject for every transfer method.

			Transfer Methods					
Subjects	Sessions	Attribute	Ref.	sCSP	divCSP-MS	divCSP-AS	WT	NA-MEMD
S1	all	Mean	96.07	**97.20**	91.60	91.61	91.01	91.04
		Median	96.67	**98.31**	91.53	91.53	90.00	89.83
		Std	2.59	1.91	4.97	6.65	5.16	5.07
S2	all	Mean	92.72	**94.33**	89.85	91.52	87.54	90.35
		Median	93.33	**95.00**	91.23	91.23	85.96	88.33
		Std	4.11	1.17	4.33	3.34	3.61	4.04
S3	all	Mean	83.70	84.40	84.51	**86.68**	78.99	76.73
		Median	86.21	**87.10**	84.75	85.48	77.97	76.27
		Std	8.16	5.60	3.75	4.37	6.99	4.41
S4	all	Mean	77.87	74.19	**79.90**	75.89	71.30	69.02
		Median	79.66	72.88	**81.36**	74.58	73.68	67.80
		Std	4.82	7.69	5.19	4.30	4.51	7.63
S5	all	Mean	66.47	69.86	68.14	65.63	**71.71**	69.31
		Median	67.27	68.52	66.07	64.29	**73.21**	69.64
		Std	3.64	6.36	5.21	3.54	3.36	3.09
S6	all	Mean	89.98	92.70	89.79	**93.73**	89.73	92.66
		Median	91.67	**96.49**	89.47	91.67	88.33	94.74
		Std	4.11	6.67	4.97	3.97	4.64	3.75
S7	all	Mean	63.35	67.87	66.72	67.87	**68.54**	64.39
		Median	63.64	**70.00**	67.86	68.97	65.52	63.79
		Std	2.88	8.39	2.99	2.51	7.22	1.47
S8	all	Mean	85.20	88.73	**89.28**	85.70	83.92	86.27
		Median	83.33	88.14	**89.29**	87.50	82.14	87.50
		Std	4.03	3.87	2.25	3.67	3.67	4.31
S9	all	Mean	81.58	75.42	**84.24**	82.44	81.92	75.44
		Median	80.70	75.86	**84.48**	81.03	79.31	75.00
		Std	2.83	2.03	6.94	4.74	4.84	1.57
S10	all	Mean	78.35	78.63	73.97	**79.72**	79.19	78.02
		Median	77.19	79.31	72.41	**82.76**	78.95	75.86
		Std	6.60	4.20	3.14	6.80	1.73	4.11
all	1 - 2	Mean	80.89	**84.44**	84.23	83.56	82.12	81.91
		Median	79.49	85.30	**88.72**	85.99	84.29	82.06
		Std	11.65	11.33	10.63	11.99	9.50	10.25
	2 - 3	Mean	**81.39**	79.03	80.57	79.32	78.15	77.13
		Median	84.52	80.22	**84.62**	81.26	79.13	75.85
		Std	12.35	11.63	8.94	9.18	9.00	8.70
	3 - 4	Mean	82.32	**83.54**	80.61	83.37	80.90	78.94
		Median	85.02	82.15	79.12	**86.60**	78.27	76.53
		Std	9.75	10.87	10.05	9.74	7.80	12.64
all	all	Mean	81.53	**82.34**	81.80	82.08	80.39	79.32
		Median	83.01	82.55	84.15	**84.62**	80.56	78.15
		Std	11.25	11.28	9.88	10.30	8.76	10.53

Table A.3.: Mean accuracies of every subject for every transfer method and every session.

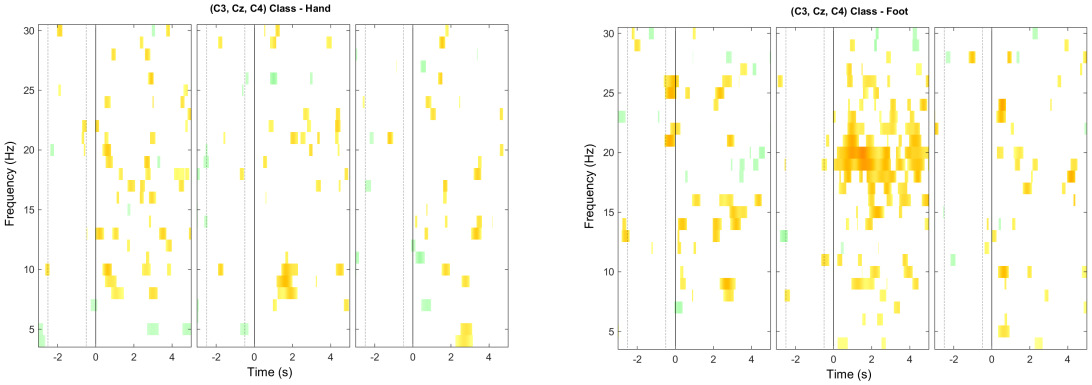
		Transfer Methods					
Subject	Sessions	Ref.	sCSP	divCSP-MS	divCSP-AS	WT	NA-MEMD
S1	1 - 2	**91.78**	85.68	82.14	83.81	74.53	78.71
	2 - 3	**95.36**	89.58	81.10	79.21	78.88	75.70
	3 - 4	87.72	**94.05**	92.20	92.80	87.39	87.97
S2	1 - 2	84.76	85.56	85.15	**87.41**	81.32	85.29
	2 - 3	80.09	80.55	80.09	**82.39**	76.79	74.45
	3 - 4	**84.78**	82.38	76.63	80.50	76.12	77.33
S3	1 - 2	65.26	77.55	78.48	**82.16**	72.02	69.97
	2 - 3	**77.70**	70.18	72.24	74.48	59.94	66.87
	3 - 4	76.86	**77.70**	71.85	72.09	58.70	64.20
S4	1 - 2	64.33	71.30	**73.91**	69.33	59.75	67.39
	2 - 3	57.10	61.44	**67.77**	61.39	53.18	56.10
	3 - 4	**69.76**	56.36	61.68	60.76	59.43	53.00
S5	1 - 2	54.58	60.07	**61.64**	55.55	60.20	59.44
	2 - 3	50.58	52.19	55.59	52.83	**56.56**	56.53
	3 - 4	56.05	**61.30**	53.81	57.57	59.02	53.74
S6	1 - 2	82.54	87.46	87.73	**88.64**	83.81	85.52
	2 - 3	80.66	78.74	80.04	**81.07**	78.84	81.03
	3 - 4	77.42	79.75	79.82	79.64	73.88	**84.49**
S7	1 - 2	49.71	53.49	53.40	52.64	**53.80**	51.71
	2 - 3	57.16	50.63	58.68	**58.95**	53.04	53.35
	3 - 4	45.33	**64.58**	60.93	53.61	62.87	50.74
S8	1 - 2	73.24	76.34	73.98	**76.59**	72.81	75.06
	2 - 3	74.05	76.48	**77.22**	70.87	74.96	71.60
	3 - 4	63.03	72.19	77.64	74.53	74.97	**78.39**
S9	1 - 2	66.70	66.94	73.84	69.82	**74.06**	64.41
	2 - 3	**73.07**	62.32	64.58	68.95	67.78	65.07
	3 - 4	73.58	67.84	67.79	**74.18**	64.97	69.51
S10	1 - 2	67.84	66.74	64.21	66.51	67.79	**67.85**
	2 - 3	64.39	**70.21**	60.53	63.91	69.96	65.03
	3 - 4	**68.02**	64.09	63.02	67.92	64.72	67.09

Table A.4.: Averaged mean accuracies of every subject for every transfer method.

			Transfer Methods					
Subjects	Sessions	Attribute	Ref.	sCSP	divCSP-MS	divCSP-AS	WT	NA-MEMD
S1	all	Mean	**91.62**	89.77	85.15	85.27	80.27	80.79
		Median	**91.78**	89.58	82.14	83.81	78.88	78.71
		Std	3.82	4.19	6.13	6.91	6.54	6.39
S2	all	Mean	83.21	82.83	80.62	**83.43**	78.08	79.02
		Median	**84.76**	82.38	80.09	82.39	76.79	77.33
		Std	2.70	2.54	4.28	3.57	2.83	5.61
S3	all	Mean	73.27	75.14	74.19	**76.24**	63.55	67.01
		Median	76.86	**77.55**	72.24	74.48	59.94	66.87
		Std	6.95	4.30	3.72	5.26	7.36	2.89
S4	all	Mean	63.73	63.03	**67.79**	63.83	57.45	58.83
		Median	64.33	61.44	**67.77**	61.39	59.43	56.10
		Std	6.35	7.60	6.12	4.78	3.70	7.57
S5	all	Mean	53.74	57.85	57.01	55.32	**58.59**	56.57
		Median	54.58	**60.07**	55.59	55.55	59.02	56.53
		Std	2.83	4.94	4.10	2.38	1.86	2.85
S6	all	Mean	80.21	81.98	82.53	83.12	78.84	**83.68**
		Median	80.66	79.75	80.04	81.07	78.84	**84.49**
		Std	2.59	4.77	4.50	4.84	4.97	2.35
S7	all	Mean	50.73	56.23	**57.67**	55.07	56.57	51.93
		Median	49.71	53.49	**58.68**	53.61	53.80	51.71
		Std	5.98	7.37	3.87	3.40	5.47	1.32
S8	all	Mean	70.11	75.00	**76.28**	74.00	74.25	75.02
		Median	73.24	76.34	**77.22**	74.53	74.96	75.06
		Std	6.14	2.44	2.00	2.90	1.24	3.40
S9	all	Mean	**71.12**	65.70	68.74	70.98	68.94	66.33
		Median	**73.07**	66.94	67.79	69.82	67.78	65.07
		Std	3.83	2.96	4.70	2.80	4.65	2.77
S10	all	Mean	66.75	67.01	62.59	66.11	**67.49**	66.66
		Median	**67.84**	66.74	63.02	66.51	67.79	67.09
		Std	2.05	3.07	1.88	2.03	2.63	1.46
all	1 - 2	Mean	70.07	73.11	**73.45**	73.25	70.01	70.54
		Median	67.27	73.82	**73.94**	73.21	72.41	68.91
		Std	13.25	11.48	10.90	12.69	9.64	10.86
	2 - 3	Mean	**71.02**	69.23	69.78	69.41	66.99	66.57
		Median	**73.56**	70.19	70.01	69.91	68.87	65.97
		Std	13.64	12.61	9.67	10.04	10.51	9.25
	3 - 4	Mean	70.26	**72.02**	70.54	71.36	68.21	68.65
		Median	71.67	70.02	69.82	**73.14**	64.85	68.30
		Std	12.92	11.45	11.35	11.83	9.50	13.36
all	all	Mean	70.45	**71.45**	71.26	71.34	68.40	68.59
		Median	70.83	71.34	71.26	**72.08**	68.71	67.73
		Std	13.27	11.85	10.64	11.52	9.88	11.16

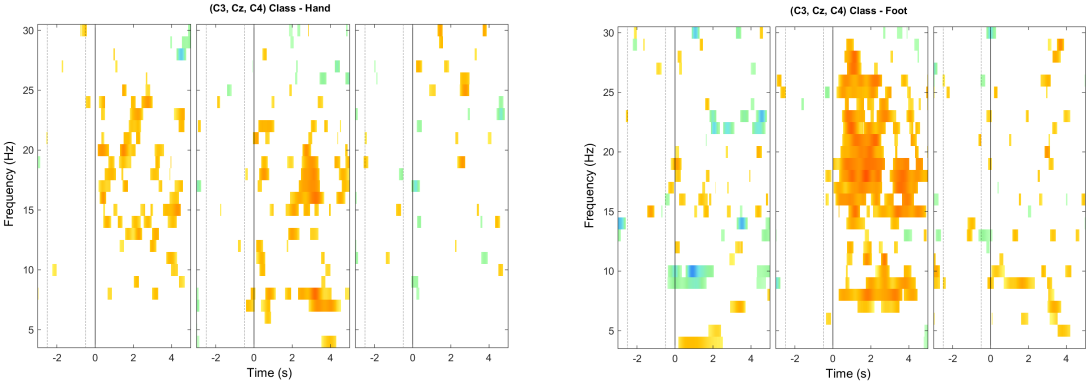
A.2. ERDS Maps of the Online Study

Time-frequency maps were calculated for all subjects from the laplace filtered channels C3, Cz and C4. Data was bandpass filtered between 4 and 30 Hz and the parts between -3 seconds to 5 seconds after the visual cue were considered in the *ERDS* analysis [17] with respect to a reference interval of -2.5 to -0.5 seconds before the visual cue. Statistical significance of the *ERDS* data was ensured by applying a t-percentile bootstrap algorithm with a significance level of alpha = 0.05. Yellow to red colours indicate weak to strong glserd, green to blue colours mark weak to strong *ERS*.



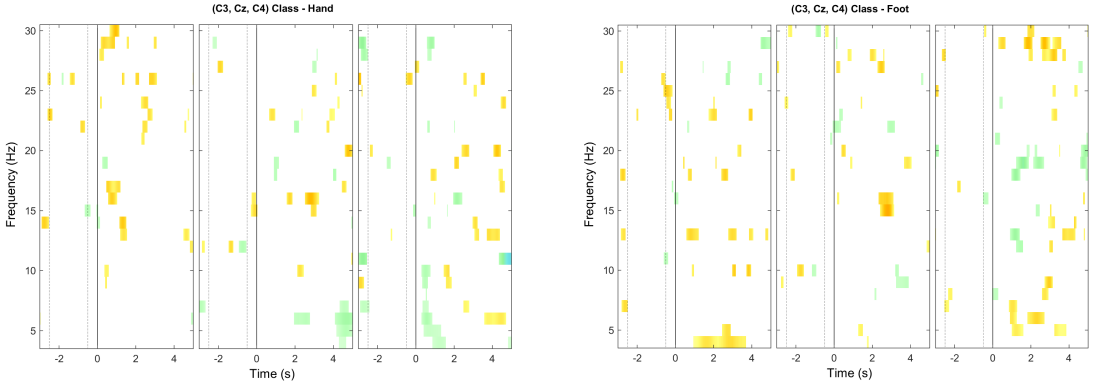
(a) class right hand **(b)** class both feet

ERDS Maps of subject P1 for the training data of session 1 for both classes and electrodes C3, Cz and C4



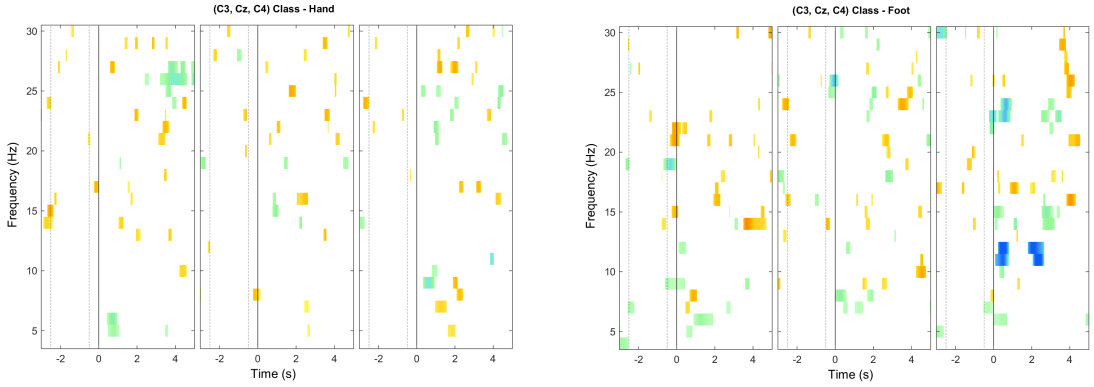
(a) class right hand **(b)** class both feet

ERDS Maps of subject P1 for the training data of session 2 for both classes and electrodes C3, Cz and C4



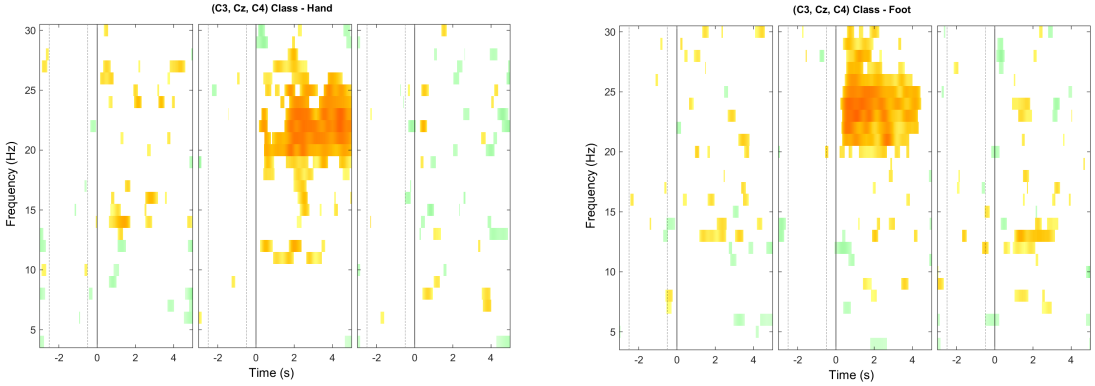
(a) class right hand **(b)** class both feet

ERDS Maps of subject P2 for the training data of session 1 for both classes and electrodes C3, Cz and C4



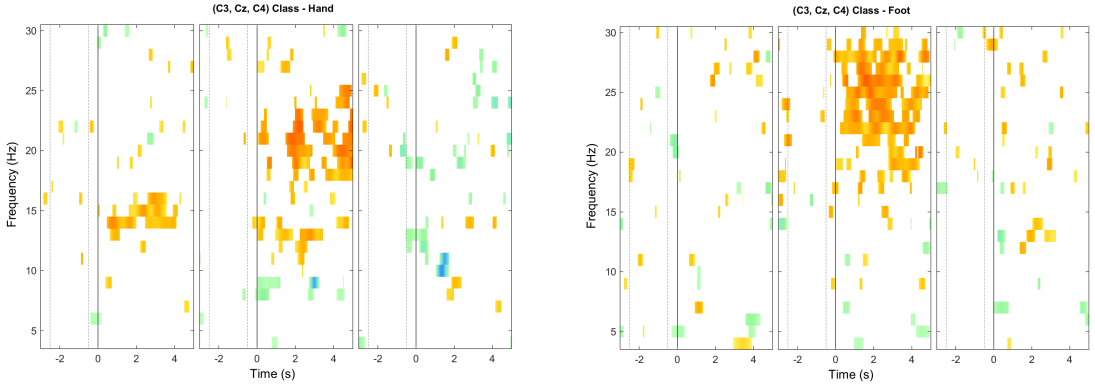
(a) class right hand **(b)** class both feet

ERDS Maps of subject P2 for the training data of session 2 for both classes and electrodes C3, Cz and C4



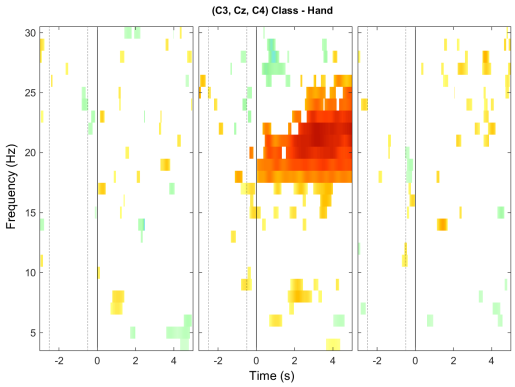
(a) class right hand **(b)** class both feet

ERDS Maps of subject P3 for the training data of session 1 for both classes and electrodes C3, Cz and C4

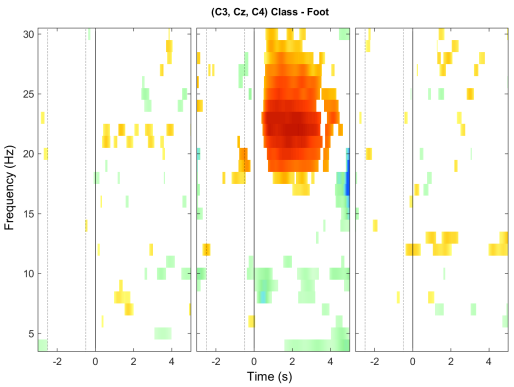


(a) class right hand **(b)** class both feet

ERDS Maps of subject P3 for the training data of session 2 for both classes and electrodes C3, Cz and C4

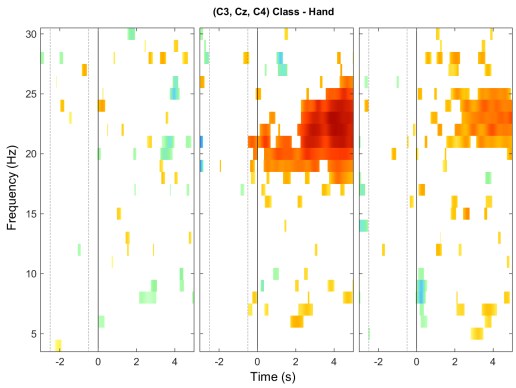


(a) class right hand

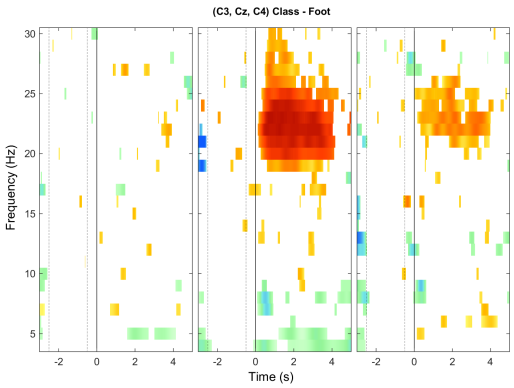


(b) class both feet

ERDS Maps of subject P4 for the training data of session 1 for both classes and electrodes C3, Cz and C4

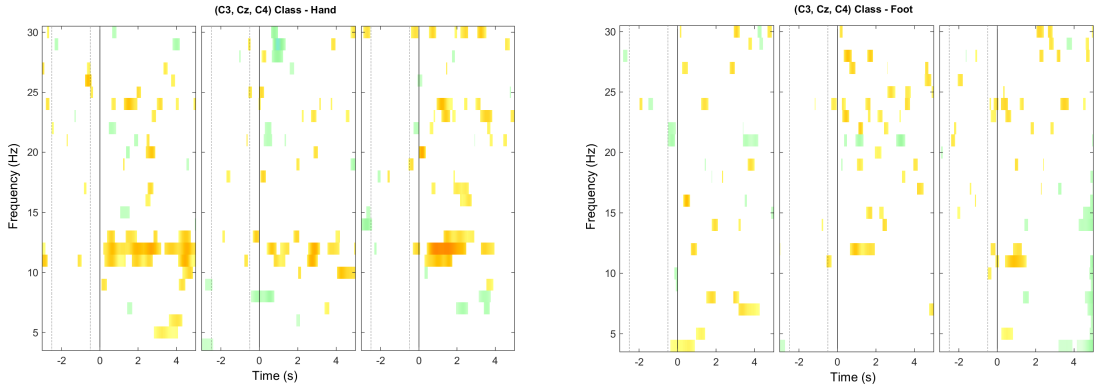


(a) class right hand



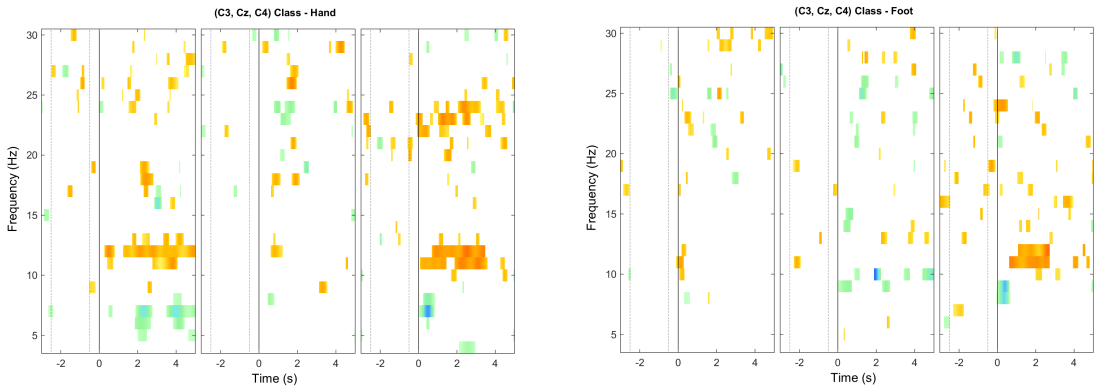
(b) class both feet

ERDS Maps of subject P4 for the training data of session 2 for both classes and electrodes C3, Cz and C4



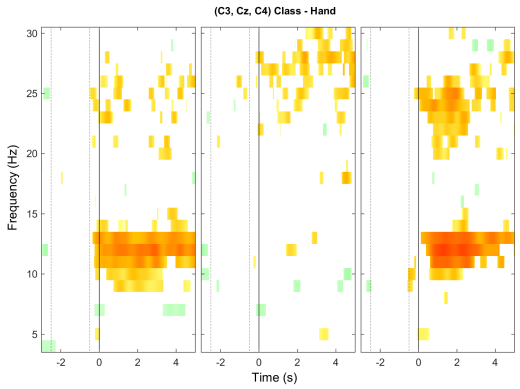
(a) class right hand **(b)** class both feet

ERDS Maps of subject P5 for the training data of session 1 for both classes and electrodes C3, Cz and C4

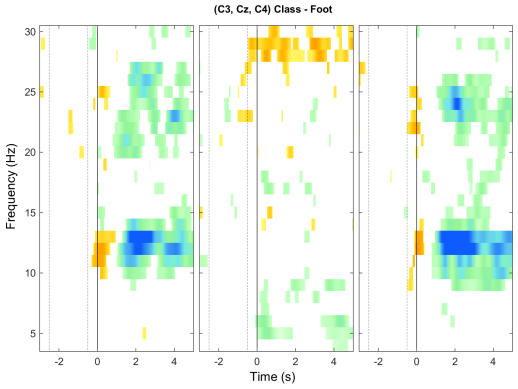


(a) class right hand **(b)** class both feet

ERDS Maps of subject P5 for the training data of session 2 for both classes and electrodes C3, Cz and C4

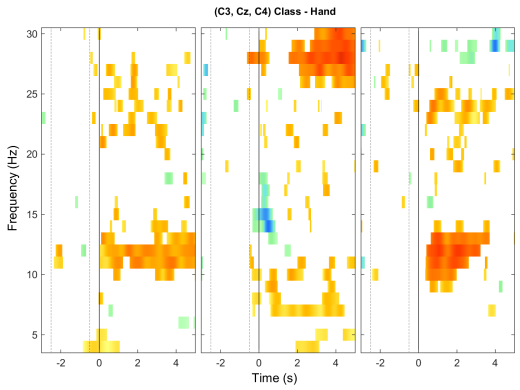


(a) class right hand

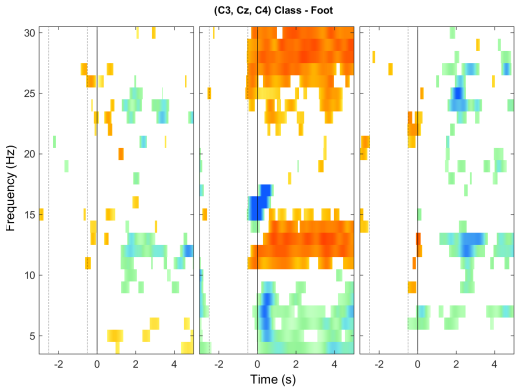


(b) class both feet

ERDS Maps of subject P6 for the training data of session 1 for both classes and electrodes C3, Cz and C4

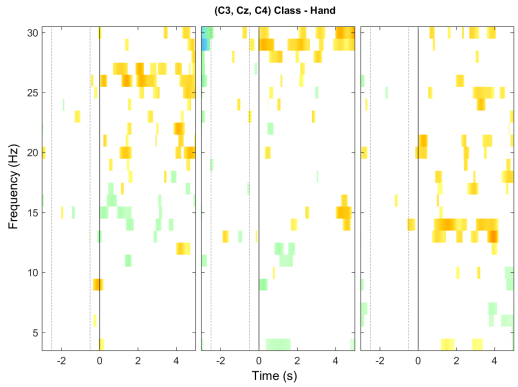


(a) class right hand

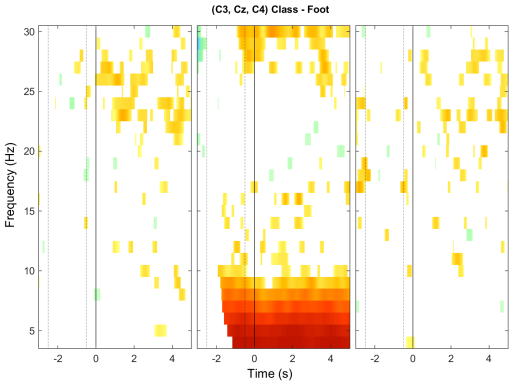


(b) class both feet

ERDS Maps of subject P6 for the training data of session 2 for both classes and electrodes C3, Cz and C4

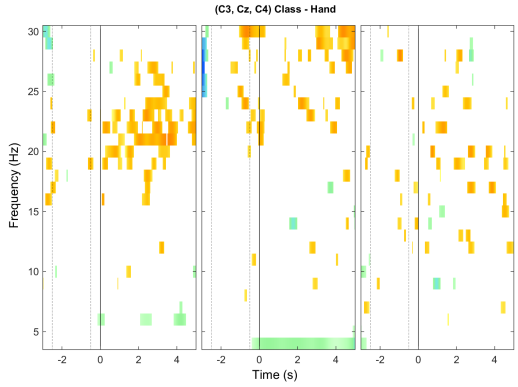


(a) class right hand

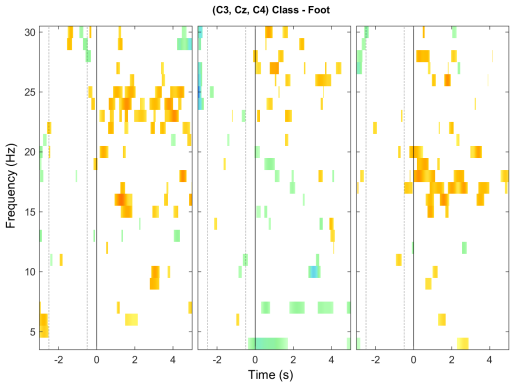


(b) class both feet

ERDS Maps of subject P7 for the training data of session 1 for both classes and electrodes C3, Cz and C4

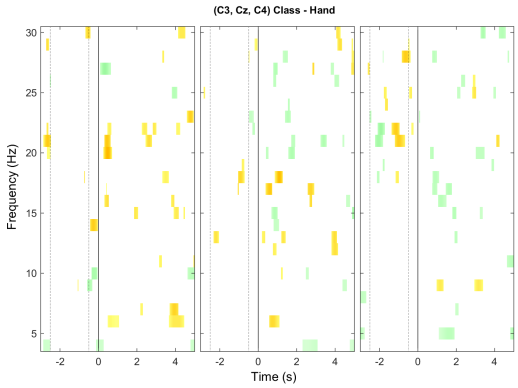


(a) class right hand

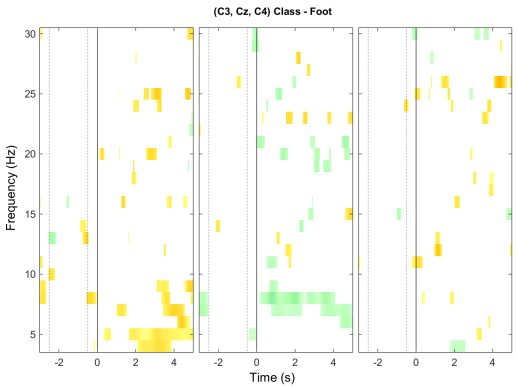


(b) class both feet

ERDS Maps of subject P7 for the training data of session 2 for both classes and electrodes C3, Cz and C4

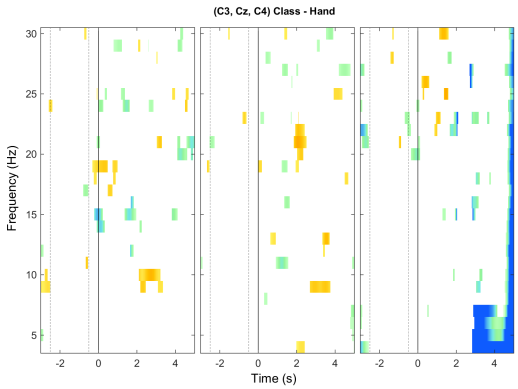


(a) class right hand

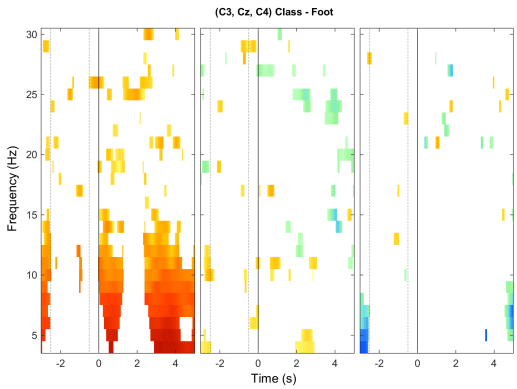


(b) class both feet

ERDS Maps of subject P8 for the training data of session 1 for both classes and electrodes C3, Cz and C4

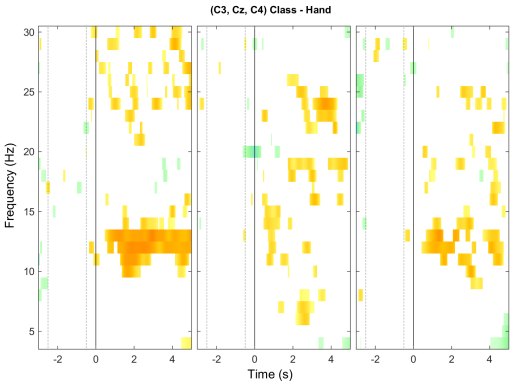


(a) class right hand

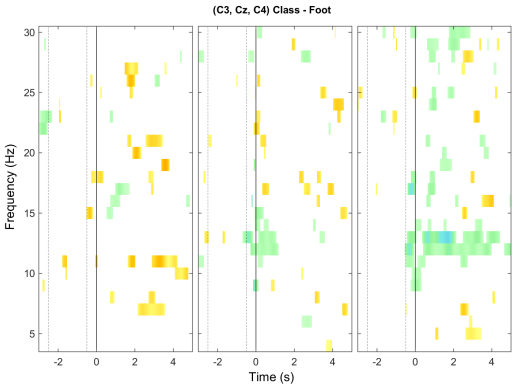


(b) class both feet

ERDS Maps of subject P8 for the training data of session 2 for both classes and electrodes C3, Cz and C4

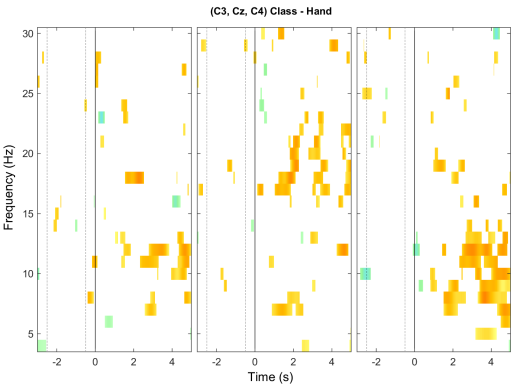


(a) class right hand

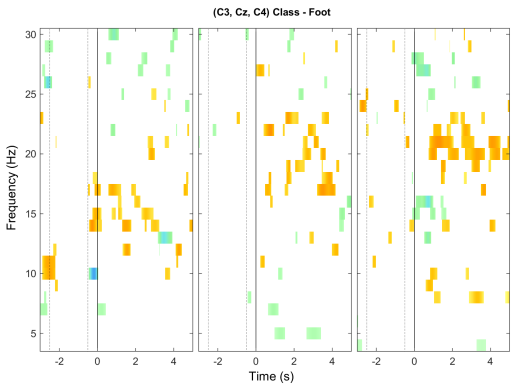


(b) class both feet

ERDS Maps of subject P9 for the training data of session 1 for both classes and electrodes C3, Cz and C4

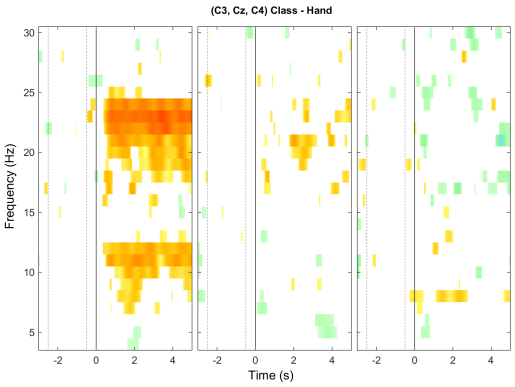


(a) class right hand

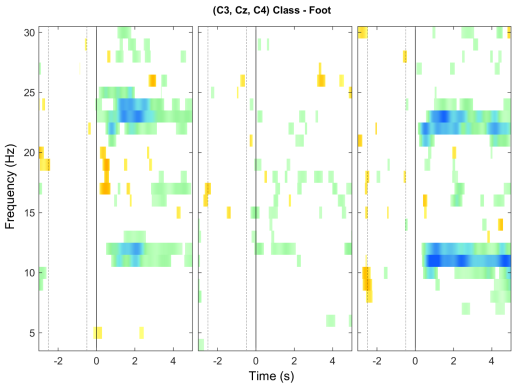


(b) class both feet

ERDS Maps of subject P9 for the training data of session 2 for both classes and electrodes C3, Cz and C4

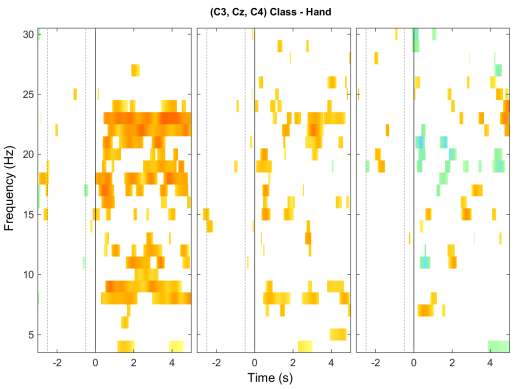


(a) class right hand

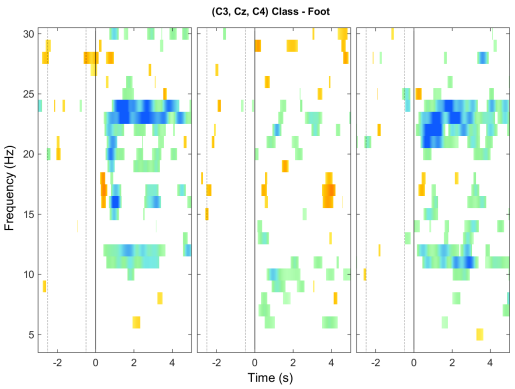


(b) class both feet

ERDS Maps of subject P10 for the training data of session 1 for both classes and electrodes C3, Cz and C4

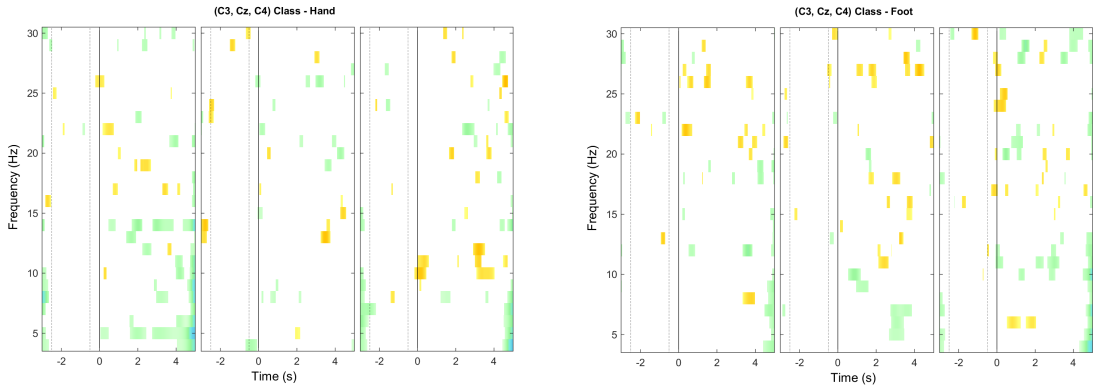


(a) class right hand



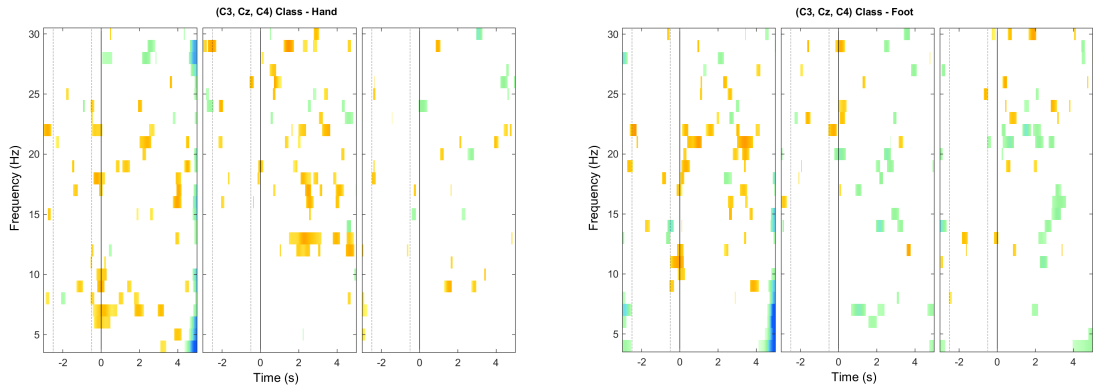
(b) class both feet

ERDS Maps of subject P10 for the training data of session 2 for both classes and electrodes C3, Cz and C4



(a) class right hand **(b)** class both feet

ERDS Maps of subject P11 for the training data of session 1 for both classes and electrodes C3, Cz and C4



(a) class right hand **(b)** class both feet

ERDS Maps of subject P11 for the training data of session 2 for both classes and electrodes C3, Cz and C4