STUDY OF ADAPTATION METHODS TOWARDS ADVANCED BRAIN-COMPUTER INTERFACES

SIDATH RAVINDRA LIYANAGE

(M.Phil. (Eng.), Peradeniya)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

NUS GRADUATE SCHOOL FOR INTEGRATIVE SCIENCES AND ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2013

Declaration

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any University previously.

Sidath

. . . .

Sidath Ravindra Liyanage

22/01/2013

Acknowledgements

I pay my heart-felt gratitude to my supervisors Prof. Xu Jian-Xin and Prof. Lee Tong Heng who were the twin towers of strength during my time as a graduate student at the National University Singapore. I would like to express my deepest appreciation to Prof. Xu Jian-Xin for his inspiration, excellent guidance, support and encouragements. I am deeply indebted to Prof. Lee Tong Heng for the kind encouragements, timely advise and insightful suggestions without which I might not have met the requirements of my study.

I am also extremely grateful to Dr. Guan Cuntai for letting me work in the Neural Signal Processing laboratory of Institute for Infocomm Research, ASTAR. His erudite knowledge and deep insights in the fields of machine learning and signal processing have been most inspiring and made this research work a rewarding experience. I owe an immense debt of gratitude to him for imparting the curiosity on learning and research in the domain of Brain Computer Interfaces. Also, his rigorous scientific approach, leadership and endless enthusiasm influenced me greatly to achieve the best I could. Without his kind guidance, this thesis and other publications I had during the past four years would have been impossible.

I also would like to thank Prof. Shuzhi Sam Ge for his role as the chair of my Thesis Advisory Committee. A special thanks to Dr. Zhang Haihong and Dr.Kai Keng Ang of Institute for Infocomm Research for guiding me throughout my attachment period at Institute for Infocomm Research. Their day-to-day advices helped me resolve numerous problems that I encountered during my research and specially in preparation of manuscripts.

Thanks also go to NUS Graduate School for Integrative Science and Engineering, for the generous financial support during my pursuit of a PhD.

I am also grateful to all my colleagues and staff at the Control and Simulation Laboratory, National University of Singapore and Brain Computer Interface Laboratory, Institute for Infocomm Research. Their kind assistance and friendship made my life in Singapore a vibrant and memorable one.

Finally, I am deeply indebted to my parents for always being with me in all my academic endeavours. Their selfless contributions, affection and love helped me become everything I am. This thesis, thereupon, is dedicated to them.

Contents

Summary

A Brain-Computer Interface (BCI) is a communication system which enables its users to send commands to a computer using only brain activities. These brain activities are generally measured by ElectroEncephaloGraphy (EEG), and processed by a system using machine learning algorithms to recognize the patterns in the EEG data.

In the first part of the thesis, theoretical foundations of Brain Computer Interfaces are introduced. The specific focus of the study, which is using adaptive machine learning techniques for BCI in order to improve Information Transfer Rates (ITR), is also specified. We attempt to improve the ITR by improving classification accuracies and by increasing the number of different motor imagery tasks classified. Classification in BCI is made more challenging due to the inherent non-stationarity of the EEG data. Therefore, adaptive methods were applied to overcome the problems caused by non-stationarity in EEG.

First, a new multi-class Common Spatial Patterns (CSP) algorithm based on Joint Approximate Diagonalization (JAD) is proposed for feature extraction in multi-class motor motion imagery BCI. The current standard, over-versus-rest (OVR) implementation of simultaneous diagonalization limits the ITR in the multi-class classification setting. The proposed fast Frobenius diagonalization based multi-class CSP is able to jointly diagonalize multiple covariance matrices, thus overcoming the bottleneck created by OVR implementation.

Consequently, a classifier ensemble with a novel adaptive weighting method is proposed to improve the classification accuracies under non-stationary conditions. The proposed classifier ensemble is based on clustering with a novel weighting technique for classifier combination. The optimal classifier combination method used in a stationary setting will not give the best classification results in non-stationary EEG classification. Therefore, clustered training data was used to train classifiers on specific groups of training data. When test data is presented, the similarities to the existing clusters are evaluated to estimate the classification accuracies of the individual classifiers. This estimated classification accuracy measures are used to adaptively weigh the classifier decisions for each test sample.

Error entropy based Kernel adaptation for adaptive classifier training is also proposed. The error entropy criterion accounts for the amount of information in the error distributions. Therefore, the minimization of error entropy considers the error distributions rather than just the error values. The error entropy criterion is used to adapt the width of the Gaussian kernel of the SVM classifier. A subset of data from the subsequent session is used as adaptation data to estimate an error entropy based cost function which is minimized by adapting the kernel width.

Towards the end, adaptation of feature extraction models using feedback training data is proposed, as it is difficult to address the non-stationarity issue only by adapting classifiers. The proposed supervised learning method is able to construct a more appropriate feature space using data from the feedback sessions. The proposed method attempts to account for the underlying complex relationship between feedback signal, target signal and EEG, using a mutual information formulation. The learning objective is formulated as a kernel-based mutual information maximizing estimation with respect to the spatial-spectral filters. A gradient-based optimization algorithm is derived for the learning task.

In conclusion, the future research directions of the proposed methods are unveiled. Possible direct application of the proposed methods to other areas in BCI, such as subject independent EEG classification, and possible extensions to general machine learning applications are outlined.

List of Tables

List of Figures

List of Symbols

Symbol	Meaning or Operation
Adaboost	Adaptive Boosting Algorithm
ALN	Adaptive Logic Network
AWEC	Adaptively Weighted Ensemble Classification
BCI	Brain Computer Interface
BLRNN	Bayesian Logistic Regression Neural Network
BOLD	Blood Oxygenation Level-Dependent
CAR	Common Average Reference
CART	Classification and Regression Tree
CNS	Central Nervous System
CSP	Common Spatial Patterns
DFT	Direct Fourier Transforms
E	Raw EEG data matrix
ECoG	ElectroCorticoGraphy
EEC	Error Entropy Criterion
EEG	ElectroEncephaloGraphy
EP	Evoked Potentials
ERD	Event Related De-synchronisation
ERP	Event Related Potential
ERS	Event Related Synchronisation
FBCSP	filter-bank Common Spatial Patterns
FFDIAG	Fast Frobenius Algorithm for Joint Diagonalization
FFT	Fast Fourier Transform
FIR	Finite Impulse Response filters
FIRNN	Finite Impulse Response Neural Network
fMRI	functional Magnetic Resonance Imaging

Symbol	Meaning or Operation
GDNN	Gamma Dynamic Neural Network
H	Entropy
HMM	Hidden Markov Model
I	Identity Matrix
ICA	Independent Component Analysis
IIR	Infinite Impulse Response filters
IP	Information Potential
ITR	Information Transfer Rate
JAD	Joint Approximate Diagonalization
KL	KullbackLeibler divergence
k-NN	k-nearest neighbour
LDA	Linear Discriminant Analysis
LRP	Lateralized-readiness potential
LVQ	Learning Vector Quantization
MAP	Maximum A Posteriori
MCSP	Multiclass Common Spatial Patterns
MDA	Multiple discriminant analysis
MEE	Minimum Error Entropy
MEG	MagnetoEncephaloGraphy
MI	Motor Imagery
MLP	Multi Layer Perceptron
MSE	mean-square-error
NIRS	Near InfraRed Spectroscopy
NN	Neural Network
PAL	Left pre-auricular point
PAR	Right pre-auricular point
PCA	Principal Component Analysis
PeGNC	Probability estimating Guarded Neural Classifier
QDA	Quadratic Discriminant Analysis
RBF	Radial Basis Function
SL	Surface Laplacian

Symbol	Operation Meaning or Operation
SMR	Sensorimotor Rhythms
SSA	Stationary Subspace Analysis
SSEP	Steady State Evoked Potentials
SSVEP	Steady State Visual Evoked Potentials
SVM	Support Vector Machine
TDNN	Time-Delay Neural Network
V	Diagonalization Transformation
ω	Class label
$P(\omega \vert x)$	Conditional Probability of a data x being in class ω
$\mathbb{R}$	set of real numbers
$\subset$	subset of
$\vert \star \vert$	absolute value of a number
$\vert\vert \star \vert\vert_\infty$	Infinite norm of matrix
$\exists$	there exists
$\forall$	for all
$\in$	in the set
$off(\star)$	off-diagonal elements of a matrix

Chapter 1

Introduction

1.1 Brain Computer Interfaces

A Brain Computer Interface (BCI) facilitates online communication between the human brain and peripheral devices. BCI's allow users to by-pass the natural neural pathways to motor neurons and muscles which can be employed to communicate with locked-in patients [1]. Wolpaw [2] has defined a BCI as, a system that measures central nervous system activity and converts it into artificial output that replaces, restores, enhances, supplements, or improves natural central nervous system output and thereby changes the ongoing interactions between the central nervous system and its external or internal environment.

Most BCI's rely on electrical measures of brain activity, and rely on sensors placed over the head to measure this activity. Electroencephalography (EEG) refers to recording electrical activity from the scalp with electrodes. Other types of sensors have also been used for BCI [2]. Magnetoencephalography (MEG) records the magnetic fields associated with brain activity, Functional magnetic resonance imaging (fMRI) measures small changes in the blood oxygenation level-dependent (BOLD) signals associated with cortical activation. Similar to fMRI, near infrared spectroscopy (NIRS) also measures the hemodynamic changes in the brain. NIRS measures the changes in optical properties caused by different oxygen levels of the blood. MEG and fMRI usually come in very large devices and are very expensive. NIRS and fMRI have poor temporal resolution compared to EEG. Therefore, EEG has remained the most popular choice for BCI solutions [2].

EEG equipment is inexpensive, lightweight, and comparatively easy to apply. Temporal resolution, which is the ability to detect changes within a certain time interval, is very good. However, the spatial (topographic) resolution and the frequency range of EEG are limited. EEG signals are also susceptible to artefacts caused by other electrical activities such as eye movements or eye blinks (electrooculographic activity, EOG) and muscles movements (electromyographic activity, EMG). External electromagnetic interferences such as the power line can also contaminate the EEG signals.

It has been found that execution or imagination of limb movements generate changes in rhythmic EEG activity known as sensorimotor rhythms (SMR) [3]. BCI based on SMR extract features and translate the changes in EEG associated with motor imagery tasks and use the resulting output to control BCI applications [4].

There is a rapidly growing interest in modelling and analysis of the brain activities through capturing the salient properties of the brain signals in the machine learning community. BCI techniques are useful in a wide spectrum of brain signal related application areas in bio-medical engineering such as epilepsy detection, sleep monitoring, biofeedback and BCI based rehabilitation. Life-sustaining measures such as artificial respiration and artificial nutrition can considerably prolong the life expectancy of locked-in patients. However, once the motor pathway is lost, any natural ways of communication with the environment is lost. BCI's offer the only channel of communication for such locked-in patients.

A block diagram of an EEG based BCI system with feedback and adaptation is shown in figure (1.1). The acquisition of EEG signals involves an electrode cap and cables that transmit

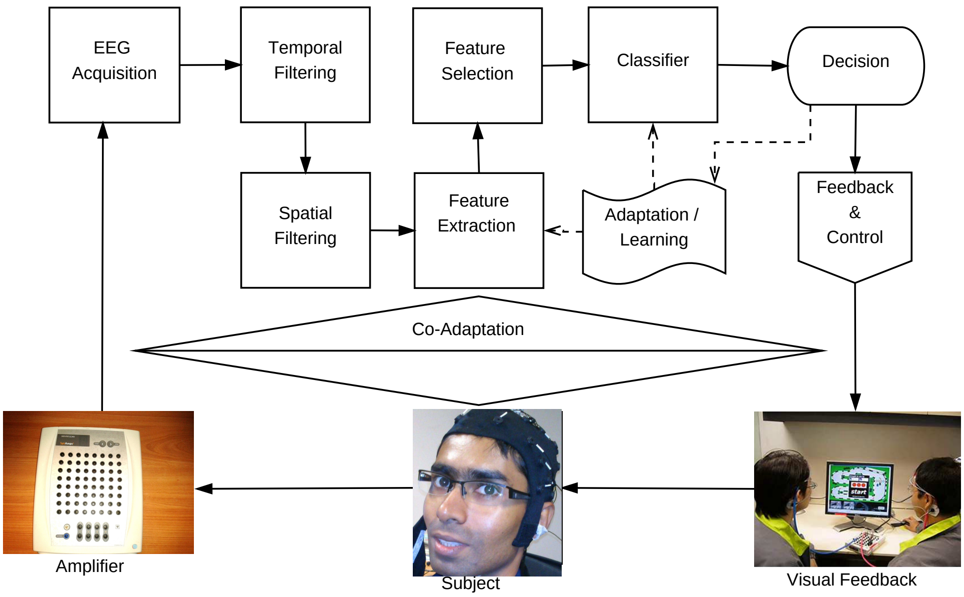


Figure 1.1: A Comprehensive Block Diagram of an EEG based BCI System

Electrode cap measures the electrical changes on the scalp of a user, these signals are converted to digital signals by the amplifier. The acquired EEG signal is pre-processed to filter noise. Feature extraction algorithms and feature selection algorithms are applied to extract and select discriminative features to build a classifier. The classification decision is normally conveyed to the user through a monitor. Adaptation can occur at feature extraction and/or classifier training parts of the system. In systems where the user's brain changes are also considered, co-adaptive learning could take place.

the signals from the electrodes to the bio-signal amplifier. The amplifier converts the EEG signals from analog to digital format.

The acquired EEG signals are pre-processed to filter out the noise and to improve the signal. Temporal and spatial filtering is carried out to enhance the useful components in the signal. Temporal filters such as low-pass or band-pass filters are generally used in order to restrict the analysis to specific frequency bands that are believed to contain the neurophysiological signals. Temporal filters can also remove various undesired effects such as slow variations in the EEG signals and power-line interferences. Spatial filters are also used to isolate the relevant spatial information embedded in the EEG signals and to reduce local background activity.

Feature extraction algorithms and feature selection algorithms are applied to extract and select useful information to build a classifier. There are a number of temporal, frequential and hybrid feature extraction methods used to extract informative features from EEG signals. These are discussed in detail in the next chapter. The goal of classification is to assign a class to the previously extracted features. A wide variety of classification methods are used in BCI's. These will also be considered in detail in the following chapter. The classification decision is usually conveyed to the user via a visual display unit.

In adaptive systems, changes to the feature extraction and classification steps can take place based on the feedback from the system. In systems where the user's brain changes are also accounted for, co-adaptive learning could take place. Such co-adaptive systems need to ensure the stability of the adaptation process by monitoring the changes closely.

1.2 Motivation and Problem Statement

Wolpaw has identified the central task of BCI research as, to determine which brain signals users can best control, to maximize that identified control, and to translate it accurately and reliably into actions that accomplish the users' intentions [6]. BCI operation depends on the interaction of two adaptive controllers: The Central Nervous System (CNS) and the Computer System. The management of this complex interaction between the adaptations of the CNS and the concurrent adaptations of the BCI is among the most difficult problems in BCI [2]. In the ideal case, new users will undergo a one-time calibration procedure and proceed to use the BCI system. The system's performance slowly adapts to the user's brain patterns, reacting only when he or she intends to control it. At each repeated use, the system recalls parameters from previous sessions, so recalibration is rarely, if ever, necessary [7].

Three computational challenges for non-invasive BCI have been identified by Blankertz et al in [7]. Improving information transfer rate (ITR) achievable through Electroencephalography

(EEG), addressing the BCI deficiency problem and integrating an “idle” or “rest” class. The BCI deficiency problem concerns the 20% of population who are not able to generate motor-related mu-rhythm variations capable of driving a BCI system [7]. ITR corresponds to the amount of information reliably received by the system. It is defined as,

$$ITR = \frac{\text{number of decisions}}{\text{duration in minutes}} \cdot \left(p \log_2(p) + (1-p)\log_2\left(\frac{1-p}{N-1}\right) + \log_2(N)\right),$$

where p is the accuracy of a subject in making decisions between N targets.

Other major challenges in BCI have been broadly categorized by Vaadia [8], to be related to theories that explain brain signals and those concerning data acquisition and interpretation. More comprehensive theoretical models of the brain are also needed to explain brain functionality and to decipher the meaning of measured signals. Data acquisition and interpretation methods must also be improved to better listen to the brain. Finding the minimum number of calibration trials needed to achieve moderate performance has also been specified as a secondary challenge in BCI.

Wolpaw has also highlighted that current BCI systems have a relatively low ITR (for most BCI this rate is equal to or lower than 20 bits/min) [2]. This means that with such BCI systems, users need relatively longer time periods in order to send a smaller number of commands. As low ITR is a very important challenge in current BCI systems the focus of this study is to research machine learning techniques to improve ITR. Two aspects can be considered to increase the ITR: increasing the recognition rates and increasing the number of classes used in current SMR based BCI systems.

Increasing the recognition rates

The performances of current systems remain modest, with percentage accuracies of mental states correctly identified rarely reaching 100 %, even for BCI using only two classes (i.e., two kinds of mental states) [6]. A BCI system which makes less mistakes would be more convenient for the user and would provide a higher information transfer rate. Less mistakes from the system would indeed lead to more efficient BCI systems that require less time to correct the mistakes.

The task of increasing ITR rates of current BCI's are impeded by the non-stationarity of the EEG signals. In machine learning, non-stationarity refers to a change in the class definitions over time, which therefore causes a change in the distributions from which the data are drawn [9]. Consider the Bayesian posterior probability of a class ω given instance x belongs, $P(\omega|x) = \frac{P(x|\omega)\cdot P(\omega)}{P(x)}$, non-stationarity is defined as any scenario where the posterior probability changes over time, i.e., $P_{t+1}(\omega|x) \neq P_t(\omega|x)$, where ω is the class to which the data instance x belongs.

The non-stationarity of EEG signals is caused by factors such as, changes in the physical properties of the sensors, variabilities in neurophysiological conditions, psychological parameters, ambient noise, and motion artefacts. Two main factors contributing to non-stationarity as reported in [10,11] are: the differences between the samples extracted from a training session and the samples extracted during an online session, and the changes in the users brain activity during online operation. As a result, the general hypothesis that the signals sampled in the training set follow a similar probability distribution to the signals sampled in the test set from a different session is violated [12]. Therefore, increasing the ITR is a very challenging machine learning problem. Adaptive machine learning techniques provide tools to overcome the issues posed by non-stationarity to improve ITR.

Increasing the Number of Classes

The number of classes considered for classification is generally very small for BCI. Most current BCI's are limited to only two class classification. Designing algorithms that can efficiently recognize a larger number of mental states would enable the subjects to use more commands leading to higher information transfer rates [13,14]. However, to significantly increase the information transfer rate, the classification accuracy, (percentage of correctly classified mental states), should also be at a healthy rate while classifying a higher number of classes.

1.3 Objectives and Contributions

This study is focused on developing several machine learning algorithms to improve the information transfer rate. The main contributions lie in the following aspects: joint approximate diagonalization based multi-class common spatial patterns algorithm, a novel adaptive weighting of classifier ensemble in presence of non-stationarity, kernel adaptation by error entropy minimization and adaptive feature selection using feedback training data in self-paced BCI.

Joint approximate diagonalization (JAD) based multiclass common spatial patterns algorithm attempts to overcome the bottleneck created by the one-versus-rest application of two class common spatial patterns algorithm for feature extraction in multiclass class EEG classification. ITR can be increased by increasing the number of effectively classified classes as well as by improving the classification accuracies.

Adaptive BCI mechanisms, where feature selection and classifiers are adapted have been attempted to improve the recognition rates [15]. Adaptive machine learning techniques for BCI are proposed in this study in order to improve classification accuracies and the overall ITR while addressing the non-stationarity problem of the EEG signals. The proposed adaptive weighting of classifier decisions in an ensemble classifier, adaptive training of kernel classifiers and adaptive feature extraction in self-paced BCI all address adaptation at different machine learning tasks associated with the BCI system, with the final objective of increasing the ITR.

The analyses and results presented in this thesis are based on the experiments done on a publicly available dataset and two datasets recorded in the Neural Signal processing laboratory of Institute for Infocomm Research, Agency for Science, Technology and Research, Singapore. All data collections at the Institute for Infocomm Research, Agency for Science, Technology and

Research were carried out in accordance to criteria approved by the Institutional Review Board of the National University of Singapore. The publicly available datasets is BCI Competition IV dataset 2A consisting of right hand, left hand, tongue and foot motor imagery trials.

1.4 Organization of Thesis

(1). In Chapter 2, a review of relevant literature is presented. Explanations of sub-systems of a typical BCI system and state of the art in improving ITR in BCI's are also discussed.

(2). In Chapter 3, joint approximate diagonalization based multi class common spatial patterns algorithms, based on fast Frobenius approximate diagonalization and Jacobi angle methods are presented.

(3). In Chapter 4, a novel adaptively weighted classifier ensemble method for non-stationary BCI is presented.

(4). In Chapter 5, a kernel adaptation approach for adaptive training of SVM classifiers in order to address the non-stationarity in EEG signals is proposed.

(5). A novel supervised learning method that learns from feedback training data for self-paced BCI is presented in Chapter 6.

(6).In conclusion, possible future directions for the applied methods are discussed in Chapter 7.

Chapter 2

Literature Survey

Brain Computer Interfaces measure brain activity, process it, and produce control signals that reflect the users' intent. In this chapter an overview of how brain activity is measured and types of brain signals that are utilized for BCI are discussed. Later in the chapter, current literature on the areas of adaptation and ensemble methods for non-stationary EEG signals are reviewed.

2.1 General Definitions

Several types of different BCI systems can be found in literature. Among these, we will first consider a few contrasting categories. Researchers notably contrast dependent BCI to independent BCI, invasive BCI to non-invasive BCI as well as synchronous BCI to asynchronous (self-paced) BCI. In the following sub-sections, these categories in the general field of BCI are introduced.

2.1.1 Dependent versus independent BCI

One distinction which is generally found in BCI literature concerns dependent BCI versus independent BCI [5]. A dependent BCI is a system which requires a certain level of motor control from the subject whereas an independent BCI does not require any motor control. For instance, some BCI's require the user to control his or her gaze [3]. In order to assist and help severely disabled people who do not have any motor control, a BCI must be independent. However, dependent BCI's are very interesting for healthy persons, in applications such as video games [4]. Furthermore, such dependent BCI's have been found to be more comfortable and easier to use than the independent BCI's [4].

2.1.2 Invasive versus non-invasive BCI

A BCI system can be classified as invasive or non-invasive according to the manner in which the brain activity is measured [1, 16]. If the sensors used for measurement are placed within the brain, i.e., under the skull, the BCI is said to be invasive. On the contrary, if the sensors used for measurement are placed outside the brain, e.g., on the scalp, the BCI is known to be non-invasive.

2.1.3 Synchronous (cue-based) versus Asynchronous (self-paced) BCI

Another distinction that is often found in literature concerns synchronous and asynchronous BCI. It has been recommended to denote asynchronous BCI as "self-paced" BCI in [17, 18]. With a synchronous BCI, the user can interact with the targeted application only during specific time periods, imposed by the system [1, 19, 20]. Hence, the system informs the user about the time periods during which he/she must interact with the application. The user should perform mental tasks during these periods only. If mental tasks are performed outside the specified time periods, the system will not respond.

In a self-paced BCI system, the user can produce a mental task in order to interact with the application at any time [21–24]. The subject can also choose not to interact with the system, by not performing any of the mental states used for control. Self-paced BCI's are the most flexible and comfortable for the user. However, it should be noted that designing a self-paced BCI is much more difficult than designing a synchronous BCI.

Most of the existing BCI systems found in literature are synchronous [1, 25]. Designing an efficient self-paced BCI is presently one of the biggest challenges in BCI and a growing number of groups have started to address this topic [18, 21–23].

2.2 Basic BCI System Framework

The steps involved in classification of EEG data involve a few machine learning techniques. The figure (2.1) shows a block diagram of the basic machine learning tasks in a simple BCI system without any feedback or adaptation.

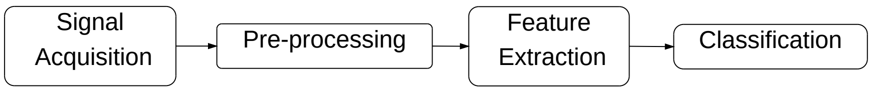


Figure 2.1: Machine Learning Tasks in a Basic BCI System

The first task associated with a BCI system is acquisition of appropriate signals from the brain. After acquiring the signals, the preprocessing step is useful to filter out the noise and improve the signal. The next step of feature extraction is vital for the successful operation of the system as the classifier will be trained on the selected features. Each of these tasks are discussed later in this chapter.

One feature of current BCI systems is the use of highly complex feature extraction algorithms compared to the relatively simple (usually linear) classification methods. All forms of available prior knowledge are used to tweak the feature extractors in most practical implementations. Therefore, many different algorithms have been developed for the selection of spatial filters, spectral bands and to extract features.

2.3 Signal Acquisition

The first step required to operate a BCI consists of measuring the subject's brain activity. Up to now, a few different types of brain signals have been identified as suitable to drive a BCI system. These brain signals must be easily observable and controllable in order to drive a BCI effectively [1]. Some of these signals are, MagnetoEncephaloGraphy (MEG) [27,28], functional Magnetic Resonance Imaging (fMRI) [29], Near InfraRed Spectroscopy (NIRS) [30], Electro-CorticoGraphy (ECoG) [31] and implanted electrodes, placed under the skull [16]. However, the most popular brain signal is ElectroEncephaloGraphy (EEG) [25]. As this study considers only the BCI systems driven with EEG signals, the rest of the chapter will focus on steps associated with EEG signal processing.

EEG is relatively cheap, non-invasive, portable and provides good time resolution. Consequently, most current BCI systems use EEG in order to measure brain activities. EEG measures the electrical activity generated by the brain using electrodes placed on the scalp [32]. EEG measures the sum of the post-synaptic potentials generated by thousands of neurons having the same radial orientation with respect to the scalp.

Signals recorded by EEG have weak amplitudes, in the order of microvolts. It is thus necessary to strongly amplify these signals before digitizing and processing them. Typically, EEG signal measurements are performed using a number of electrodes which varies from 1 to about 256, these electrodes being generally attached using an elastic cap. The contact between the electrodes and the skin is generally enhanced by the use of a conductive gel or paste [39]. BCI researchers have recently proposed and validated dry electrodes, which do not require conductive gels [40].

Electrodes are generally placed and named according to a standard model, called the 10-20 international system [33]. This system has been initially designed for 19 electrodes, however,

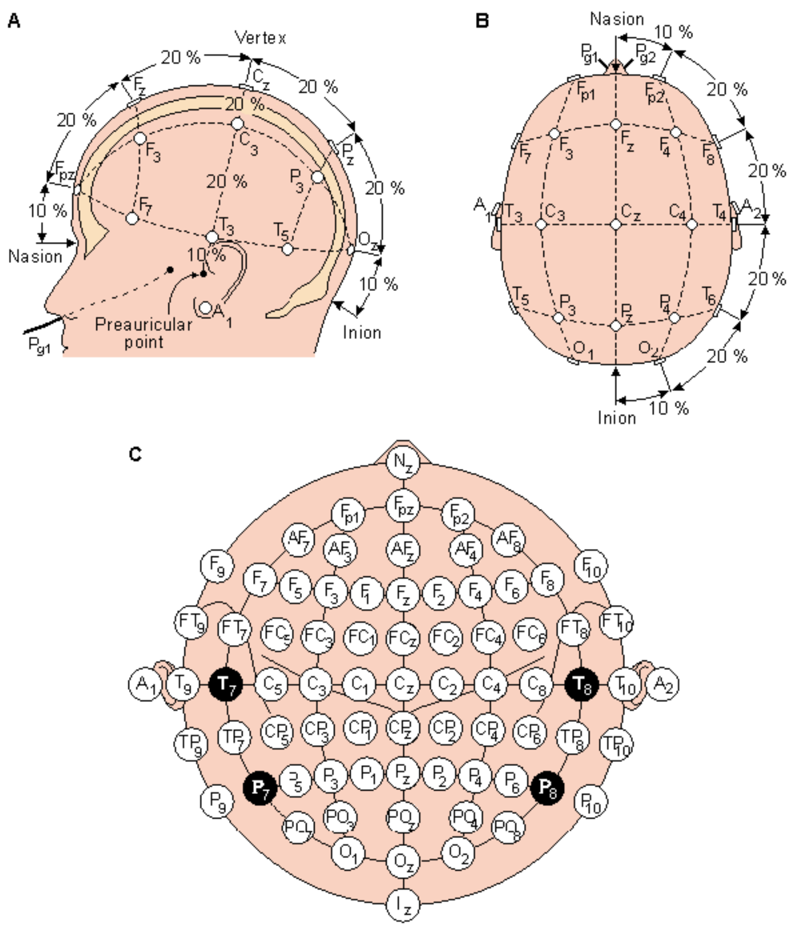


Figure 2.2: The International standard 10:20 montage for electrode placement.

Sub-figure A shows the subdivision of arcs on the scalp starting from craniometric reference points: Nasion (Ns), Inion (In), Left (PAL) and Right (PAR) pre-auricular points. The intersection of the longitudinal (Ns-In) and lateral (PAL-PAR) is named the Vertex. Sub-figure B shows the original 19 electrode positions. Sub-figure C shows the extended version for 70 electrode positions.

extended versions have been proposed to deal with larger number of electrodes [34]. The figure (2.2) shows the positions of electrodes according to the International 10-20 system. It is based on an iterative subdivision of arcs on the scalp starting from craniometric reference points: Nasion (Ns), Inion (In), and Left (PAL) and Right (PAR) pre-auricular points. The intersection of the longitudinal (Ns-In) and lateral (PAL-PAR) is named the Vertex.

The "10" and "20" refer to the fact that the actual distances between adjacent electrodes are either 10% or 20% of the total front-back or right-left distance of the skull as it divides the distance from the nasion and the inion into 10% and 20% segments. The skull perimeters are measured in the transverse and median planes from the nasion and inion points [34]. Each electrode position has a letter to identify the lobe and a number to identify the hemisphere location. The letters F, T, C, P and O stand for frontal, temporal, central, parietal, and occipital lobes, respectively. Note that there exists no central lobe; the "C" letter is only used for identification purposes only. A "z" (zero) refers to an electrode placed on the midline. Even numbers (2,4,6,8) refer to electrode positions on the right hemisphere, whereas odd numbers (1,3,5,7) refer to those on the left hemisphere [32].

2.4 Brain Rhythms

EEG signals are composed of different oscillations named "rhythms" [32]. These rhythms have distinct properties in terms of spatial and spectral localization. There are six classical brain rhythms as shown in figure (2.3) : Alpha, Mu, Delta, Gamma, Beta and Theta with different oscillating frequencies.

- Alpha rhythm: These are oscillations, located in the 8-12 Hz frequency band, which appear mainly in the posterior regions of the head (occipital lobe) when the subject has closed eyes or is in a relaxation state.
- Beta rhythm: This is a relatively fast rhythm, belonging approximately to the 13-30 Hz frequency band. It is a rhythm which is observed in awake and conscious persons. This rhythm is also affected by the performance of movements, in the motor areas [35].
- Delta rhythm: This is a slow rhythm (1-4 Hz), with a relatively large amplitude, which is mainly found in adults during deep sleep.

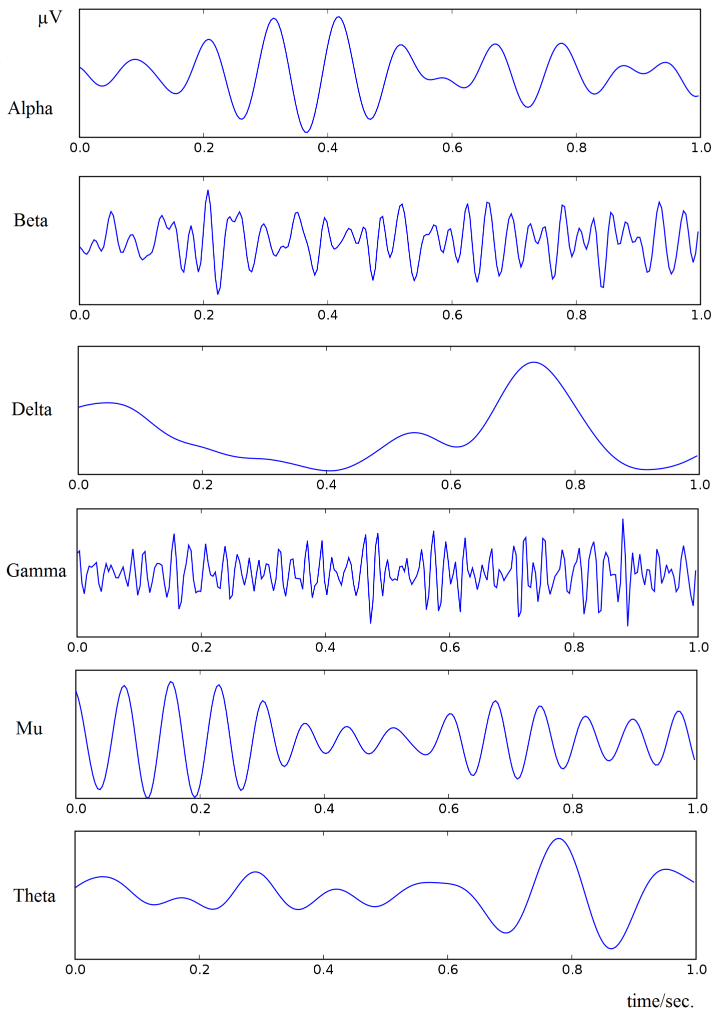


Figure 2.3: Brain Rhythms

- Gamma rhythm: This rhythm mainly concerns frequencies above 30 Hz. This rhythm is sometimes defined as having a maximal frequency around 80 Hz or 100 Hz. It is associated with various cognitive and motor functions.

- Mu rhythm: These are oscillations in the 8-13 Hz frequency band, located in the motor and sensorimotor cortex. The amplitude of this rhythm varies when the subject performs movements. Consequently, this rhythm is also known as the "sensorimotor rhythm".

- Theta rhythm: This a slightly faster rhythm (4-7 Hz), observed mainly during drowsiness and in young children.

2.5 Neurophysiological Signals in EEG for BCI

Various signals in EEG have been studied and some of them have been identified as relatively easy to be controlled by the user. These signals have been divided into two main categories as evoked signals and spontaneous signals [1, 36].

- Evoked signals are generated unconsciously by the subject when he/she perceives a specific external stimulus. These signals are also known as Evoked Potentials (EP).
- Spontaneous signals are voluntarily generated by the user after an internal cognitive process without any external stimuli.

2.5.1 Evoked potentials

The main advantage of evoked potentials is that, contrary to spontaneous signals, evoked potentials do not require a specific training for the user, as they are automatically generated by the brain in response to a stimulus. As such, they can be used efficiently to drive a BCI since the first use [1, 36]. Nevertheless, as these signals are evoked, they require using external stimulations, which can be uncomfortable, cumbersome or tiring for the user.

In the category of evoked potentials, the main signals that are used in BCI are the Steady State Evoked Potentials (SSEP) and Event Related Potentials (ERP) [1, 36].

Steady State Evoked Potentials

Steady State Evoked Potentials (SSEP) are brain potentials that appear when the subject perceives a periodic stimulus such as a flickering picture or a sound modulated in amplitude. SSEP are defined by an increase of the EEG signal power in the frequencies being equal to the stimulation frequency or being equal to its harmonics and/or sub-harmonics [3, 37, 38]. Various kinds of SSEP are used for BCI, such as Steady State Visual Evoked Potentials (SSVEP) [3, 39–41], which are by far the most used, somatosensory SSEP [38] and auditory SSEP [37]. SSEP appear in the brain areas corresponding to the sense which is being stimulated, such as the visual areas when a SSVEP is used. Not requiring training and ability to have large number of commands make it an attractive research area in BCI [42–47].

Event Related Potentials

An event related potential (ERP) is a measured response that is directly the result of a sensory, motor, or cognitive event. Figure (2.4) shows several ERP components associated with visual stimuli. P1 and N1 components are generated when information flows along the visual system and visual analysis. Attention to peripheral targets in the visual field evokes N2 components. N2 and P300 (P3) components are associated with categorization of the visual stimulus, indexing and maintaining working memory encoding.

Other than these ERP's, elicited during the selection and preparation of the motor response the process continues even after the motor response. Components such as error-related negativity could be triggered if the subject realizes that an error has occurred during the trial and lateralized-readiness potential(LRP) components which are associated with preparation for motor movement.

ERPs are calculated by averaging the EEG signals over multiple trials. The minimum number of trials needed to average out the noise is different for each component. Generally, to get a good measure of P1 and N1 ERP's 300-1000 trials per condition are required. However, P300 (P3) requires only around 30 trials per condition; therefore it is a very useful type of ERP component.

The P300 (P3) consists of a positive waveform appearing approximately 300 ms after a rare and relevant stimulus (see Figure (2.4)) [48]. It is typically generated through the "odd-ball" paradigm, in which the user is requested to attend to a random sequence composed of two kinds of stimuli with one of these stimuli being less frequent than the other. If the rare stimulus is relevant to the user, its actual appearance triggers a P300 observable in the user's EEG. This potential is mainly located in the parietal areas. P300 is quite attractive as it is consistently detectable, is elicited by precise stimuli and is evoked in nearly all subjects. Due to these reasons P300 has become a very popular ERP signal to drive Brain Computer Interfaces. The P300 is mostly used in speller applications [48–52].

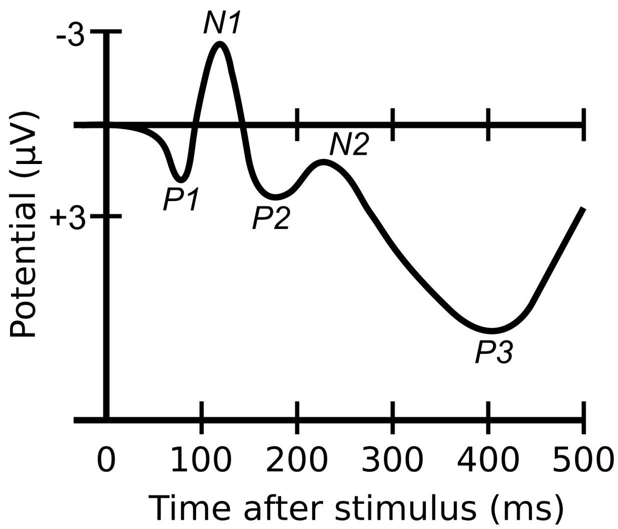


Figure 2.4: ERP generated for a visual stimuli

2.5.2 Spontaneous signals

Under the category of spontaneous signals, which are voluntarily generated by the user without any external stimuli, the most used signals are the sensorimotor rhythms (SMR).

Motor and sensorimotor rhythms

Sensorimotor rhythms are brain rhythms related to motor actions, such as arm movements. These rhythms, which are mainly located in the $\mu\,(\approx 8-13Hz)$ and $\beta\,(\approx 13-30Hz)$ frequency bands, over the motor cortex, can be voluntarily controlled by a user. The role of feedback is essential in operant conditioning type of learning, as it enables the user to understand how he/she should modify his/her brain activity in order to control the system. Generally, in BCI based on operant conditioning, the power of the μ and β rhythms in different electrode locations are linearly combined in order to build a control signal which will be used to perform 1D, 2D or 3D cursor control [53, 54].

Motor imagery

A user performing motor imagery involves imagining movements of his/her own limbs or muscles (hands, feet or tongue for instance) [17, 20, 53]. The resultant signals generated by performing or imagining a limb movement have very specific temporal, frequential and spatial features, which makes them relatively easy to recognize automatically [17, 56, 57]. For instance, imagining a left hand movement is known to trigger a decrease of power, known as, Event Related Desynchronisation (ERD) in the μ and β rhythms, over the right motor cortex [58].

In motor imagery based BCI, the motor imagery task is associated with a specific command such as controlling a cursor etc. [20,59,60]. Using a motor imagery-based BCI generally requires a few runs of training before being efficient enough for test classification [16]. However, using advanced signal processing and machine learning algorithms enables the use of such BCI with almost no training [61, 62, 105].

2.5.3 Pre-processing

Most BCI systems use simple spatial or temporal filters as pre-processing steps in order to increase the signal-to-noise ratio of the EEG signals. Temporal filters such as low-pass or band-pass filters are generally used in order to restrict the analysis to specific frequency bands that are believed to contain the neurophysiological signals. Temporal filters can also remove various undesired effects such as slow variations in the EEG signals and power-line interferences. Temporal filters that are used in general include, Direct Fourier Transforms (DFT), Finite Impulse Response filters (FIR) and Infinite Impulse Response filters (IIR).

In DFT, the signal is first converted into the frequency domain. All coefficients $S(f)$ that do not correspond to target frequencies are set to zero. Then the signal is represented as a sum of oscillations at different frequencies f. The signal is then transformed back to time domain by inverse DFT. DFT is also known as Fast Fourier Transform (FFT) due to its fast execution speed [64].

Finite Impulse Response (FIR) filters use a few last samples of a raw signal in order to determine the filtered signal [65]. On the other hand, Infinite Impulse Response filters (IIR) are linear, recursive filters. In addition to a last few samples as used in FIR, the IIR make use of the outputs of a few last filters also. IIR filters can perform filtering with a much smaller number of coefficients than FIR filters.

Spatial filters are also important pre-processing tools in processing EEG signals. Various spatial filters are used to isolate the relevant spatial information embedded in the EEG signals. This is achieved by selecting or by weighting the contributions from the different electrodes [65]. Popular spatial filters include Common Average Reference (CAR) and Surface Laplacian (SL) filters [65]. These spatial filters can also reduce local background activity.

Common Spatial Patterns

A very popular spatial filtering method in BCI is Common Spatial Patterns (BCI). The Common Spatial Patterns (CSP) algorithm was first presented by Koles [66] as a method to extract the abnormal components from EEG, using a set of patterns that are common to both the normal and the abnormal recordings and have a maximally different proportion of the combined variances. Later CSP was used to create features for classification in EEG caused by imagined movements. The first and last few CSP components (the spatial filters that maximize the difference in variance) are selected as features to classify the trials. CSP is currently considered as the gold standard for ERD based BCI [7]. It has been extended to multi-class problems in [211], and further extensions and robustifications using simultaneous optimization of spatial and frequency filters have been proposed in [123, 124, 138].

The CSP algorithm computes the transformation matrix W to yield features whose variances are optimal for discriminating 2 classes of EEG measurements by solving the eigen value decomposition problem

$$\Sigma_1 W = (\Sigma_1 + \Sigma_2)\, W\Delta, \tag{2.1}$$

where Σ_1 and Σ_2 are estimates of the covariance matrices of band-pass filtered EEG measurements of the respective motor imagery actions, and Δ is the diagonal matrix that contains the eigen values of Σ_1. Spatial filtering is performed by linearly transforming the EEG measurements using

$$Z_i = W^T E_i, \tag{2.2}$$

where $E_i \in \mathbb{R}^{ch\times t}$ denotes the single-trial EEG measurement of the ith trial, $Z_i \in \mathbb{R}^{ch\times t}$ denotes E_i after spatial filtering, $W \in \mathbb{R}^{ch\times ch}$ denotes the CSP projection matrix, ch is the number of channels, t is the number of EEG samples per channel, and T denotes transpose operator.

The CSP features of the ith trial are then given by

$$x_i = \log \frac{\mathrm{diag}\left(\bar{W}^T E_i E_i^T \bar{W}\right)}{\mathrm{tr}\left[\bar{W}^T E_i E_i^T \bar{W}\right]}, \tag{2.3}$$

where $x_i \in \mathbb{R}^{2m}$ are CSP features, $\bar{W}$ represents the first m and the last m columns of W, $diag(\cdot)$ returns the diagonal elements of the square matrix, and $tr[\cdot]$ returns the sum of the diagonal elements in the square matrix.

2.5.4 Feature Extraction

Measuring brain activity through EEG leads to the acquisition of a large amount of data. EEG signals are generally recorded with a large number of electrodes varying from 8 to 256. Sampling frequencies ranging from $100Hz$ to $1000Hz$ are normally used in collecting data. In order to ensure satisfactory performances under these conditions it is necessary to work with a smaller number of values that include the most informative parts of the signals. These values are known as “features”. Such features can be, for instance, the power of the EEG signals in different frequency bands. Features are generally aggregated into a vector known as “feature vector”. Thus, feature extraction can be defined as an operation which transforms one or several signals into a feature vector.

Identifying and extracting good features from signals is a crucial step in the design of a reliable BCI system. If the features extracted from the EEG are not relevant and do not describe the corresponding neurophysiological signals adequately, the classification algorithm which depends on such features will have trouble predicting the correct class of these features, i.e., the mental state of the user. As a result, the recognition rates of mental states will be low, leading to an inconvenient BCI system or even a system failure. Numerous feature extraction techniques have been studied and proposed for BCI [68, 69, 72].

These feature extraction techniques can be divided to three main groups. Firstly, there are methods that exploit the temporal information embedded in the signals [70, 71, 75]. The Second type of methods is based on frequential information [35, 76, 77]. Finally there are hybrid methods that are based on time-frequency representations. These hybrid methods exploit both the temporal and frequential information [78, 79].

Temporal Feature Extraction Methods

Temporal methods for feature extraction use variations of the signal time series. These methods are particularly useful to identify specific neurophysiological signal components with precise time signatures such as the P300 or ERD [70,75]. Amplitude of raw EEG signals, auto-regressive parameters and Hjorth parameters [80] can be identified under temporal methods for feature extraction.

Frequential Feature Extraction Methods

Frequential methods used for feature extraction make use of the specific oscillations in the EEG known as rhythms. Performing a given mental task (such as motor imagery or another cognitive task) makes the amplitude of these different rhythms vary. Moreover, signals such as steady state evoked potentials are defined by oscillations with frequencies synchronized with the stimulus frequency. Band power features and power spectral density features are used to extract features under this category.

Hybrid Feature Extraction Methods

Other than the above two major categories of feature extraction methods, hybrid methods combining both time and frequency domains are available. Time-frequency representations are able to can catch relatively sudden temporal variations of the signals, while still keeping frequential information. These methods include short-time Fourier transform and wavelets [81, 82].

2.5.5 Classification

The third key step in processing neurophysiological signals is translating the features into commands [69, 73]. The goal of classification is to assign a class to the previously extracted feature vectors. This end can be achieved using a few different techniques. A wide variety of classification methods are used in BCI's. Prevailingly, Linear classifiers, Bayesian classifiers, neural networks, nearest neighbour classifiers and combined classifiers are the main groups of classifiers currently used in BCI research [226]. In addition to these classifiers, in this study we considered the k-nearest neighbour classifier and the Classification and Regression Tree (CART) classifier.

Linear Classifiers

Linear classifiers are discriminant algorithms that use linear functions to distinguish classes. They are probably the most popular algorithms for BCI applications. Two main kinds of linear classifiers have been used for BCI design, namely, Linear Discriminant Analysis (LDA) and Support Vector Machines (SVM).

LDA Classifier

The aim of LDA is to use hyperplanes to separate the data representing the different classes [81]. The separating hyperplane is obtained by seeking the projection that maximizes the distance between the means of the two classes and minimizes the interclass variance [81]. This can be achieved by maximizing the ratio of between-class scatter to within class scatter given by Eq. (2.4),

$$J(w) = \frac{w^T S_B w}{w^T S_W w}, \tag{2.4}$$

$$S_B = (m_1 - m_2)(m_1 - m_2)^T\,, \tag{2.5}$$

$$S_W = S_1 + S_2, \tag{2.6}$$

where S_B is the between class scatter matrix for two classes as shown in Eq. (2.5), S_W is the within class scatter matrix for two classes given in Eq. (2.6), w is an adjustable weight vector or projection vector.

The low computational cost of this method makes it suitable for online BCI systems. LDA has been used in a number of BCI systems such as motor imagery based BCI, P300 speller, multi-class and asynchronous BCI [59, 78]. The main drawback of LDA is its linearity which could sometimes give rise to poor results when handling complex non-linear data.

SVM Classifier

SVM also uses a discriminant hyperplane to separate the classes [83]. In SVM, the selected hyperplane is the one that maximizes the margins, i.e., the distance from the nearest training points. For a linear SVM, the large margin (i.e. the optimal hyperplane w) is realized by minimizing the cost function on the training data

$$J(w, \xi) = \frac{1}{2}||w||^2 + C\sum_{i=1}^{n} \xi_i, \tag{2.7}$$

under the constraints,

$$y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad \forall i = 1, \cdots, n, \tag{2.8}$$

where $x_1, x_2, \cdots, x_n$ are the training data, $y_1, y_2, \cdots, y_n \in [1, +1]$ are the training labels, $\xi_1, \xi_2, \cdots, \xi_n$ are the slack variables, C is a regularization parameter that controls the trade-off between the complexity and the number of non-separable points, and b is a bias. The slack variables measure the deviation of data points from the ideal condition of pattern separability. The parameter C can be user-specified or determined via cross-validation.

Maximizing the margins is known to increase the generalization capabilities [83, 86]. SVM classifier has been successfully applied to a relatively large number of BCI applications [85]. SVM inherently have slow speeds of execution due to its high complexity.

Neural Networks

A Neural Network (NN) is an assembly of several artificial neurons that are able to produce non-linear decision boundaries [86]. Multi Layer Perceptron (MLP) is the most widely used NN in BCI. An MLP is composed of several layers of neurons: an input layer, possibly one or several hidden layers, and an output layer [86]. However, MLP's are sensitive to overtraining. The problems are intensified with noisy and non-stationary EEG data. Therefore, careful selection of architecture and regularization is critical to avoid overtraining when using NN classifiers [83].

Other types of NN architectures are also used in the field of BCI. Learning Vector Quantization (LVQ) Neural Networks, Fuzzy ARTMAP Neural Network [88], Finite Impulse Response Neural Network (FIRNN) [89], the Time-Delay Neural Network (TDNN) or the Gamma Dynamic Neural Network (GDNN), Radial Basis Function (RBF) Neural Network , Bayesian Logistic Regression Neural Network (BLRNN), Adaptive Logic Network (ALN) and Probability estimating Guarded Neural Classifier (PeGNC) have also been attempted in the last decade for classification of EEG signals [90].

Bayesian Classifiers

Bayesian classifiers are an important class of classifiers used in BCI. The decision boundaries generated by Bayesian classifiers are non-linear. Two major classification algorithms can be found under this category: Bayes quadratic and Hidden Markov Model. In Bayes quadratic classification the Bayes rule is used to compute a posterior probability of a feature vector belonging to a given class [86]. Using the Maximum A Posteriori (MAP) rule and these probabilities, the class of this feature vector can be estimated. This has been applied with success to motor imagery and mental task classification [91, 92].

Hidden Markov Models

Hidden Markov Models (HMM) is a probabilistic automaton that can provide the probability of observing a given sequence of feature vectors [93, 94]. HMM are quite suitable for the classification of time series. As EEG components used to drive BCI have specific time courses, HMM have been applied to the classification of temporal sequences of BCI features [80] and even for classification of raw EEG [96].

k-Nearest Neighbour Classifier

The k-Nearest Neighbour (k-NN) is a classifier that assigns the class label for new data based on the class with the most occurrences in a set of k nearest training data points usually computed using a distance measure such as the Euclidean distance [136].

Classification and Regression Tree

Classification and Regression Tree (CART) is a classifier which uses symbolic tree like representations of finite sets of if-then-else questions that are natural, intuitive and interpretable. They are multi-stage decision systems in which classes are sequentially rejected until an acceptable class is found. The feature space is split into unique regions, corresponding to the classes, in a sequential manner. Upon the arrival of a feature vector, the searching of the region to which the feature vector will be assigned is achieved via a sequence of decisions along a path of nodes of an appropriately constructed tree. Such schemes are usually advantageous when a large number of classes are involved [83].

Recent Trends in Classification

Recent trends in BCI research is reaching for subject independent and co-adaptive classifiers [97]. Relatively simple linear classifiers are optimized adaptively for each user. Supervised and unsupervised adaptation of LDA classifier parameters has been attempted [97]. Another novel approach is to combine pre-processing, feature extraction, feature selection, feature combination and classification all into one regularized discriminative framework [98].

2.6 Adaptive BCI to Address Non-stationarity

Adaptive methods in BCI has been studied quite extensively in literature. The current trends in BCI is towards adaptations at all possible levels of a BCI system such as, feature extraction, feature translation, classification and user interfaces.

Schlogl et al [95] has identified adaptation as a method to overcome non-stationarity in EEG. Two types of non-stationarities have been identified in [95] as short-term changes and long-term changes. The short-term changes have been found to be related to different mental activities such as hand movements, mental arithmetic, etc. Long term changes have been described as related to fatigue, changes in the recording conditions, and effects of feedback training.

Non-stationarities arising from short-term changes can usually be addressed in the feature extraction step. Short-term changes that are unrelated to the motor imagery task could cause reduction in classification accuracies. These components are often mixed with white noise in the background. Therefore, these are not specifically addressed here. The non-stationarities caused by long-term changes such as feedback training effect, fatigue, changed recording conditions must be addressed in the classification step. Feedback training can modify the subject's EEG patterns, that would require an adaptation of the classifier, which might again cause the feedback to change. The possible difficulties of such a circular relation have been known as the "manmachine learning dilemma" [23, 56].

A few methods, such as, Bayesian transduction, active learning and distribution matching have been suggested to address the non-stationarity issue [106, 131–133]. Stationary Subspace

Analysis (SSA) [134] is another unsupervised learning method that finds subspaces in which data distributions stay invariant over time.

Segmentation-type approaches such as, extracting features from short data segments (e.g. FFT-based Bandpower [56,57,119], AR-based spectra in [120], slow cortical potentials by [121], or CSP combined with Bandpower [7,122–124] have also been used to address non-stationarities. Classifiers obtained and retrained from specific sessions or runs have also been attempted [7,56]. Modelling the non-stationarity of densities where the conditional probability $P(\omega|x)$ stays stable while the densities $P(x)$ exhibit variation has been successful in modelling the covariate shift [12].

Segmentation approaches can cause sudden changes from one segment to the next. Adaptive methods are able to avoid such sudden changes by continuously updating to the new situation. Therefore, adaptive methods can react faster, and have a smaller deviation from the true system state [95]. Sliding window approaches where segmentation is combined with overlapping windows also provide a similar advantage as adaptation. But, it has been shown that sliding window methods have much higher computational costs than adaptive methods in general [95].

Adaptive estimators for statistics such as mean, variance and covariance have been proposed in literature [95, 97]. Adaptive Inverse Covariance Matrix Estimation by adaptively estimating the inverse of the extended covariance matrix facilitates construction of adaptive LDA and Quadratic Discriminant Analysis (QDA) classifiers [95, 97].

To ensure the robustness of the system in the presence of co-adaptation of the user and the system, most adaptive methods use small update coefficients. The results from [97, 125–127] prove that adaptive methods lead to robust BCI systems. However, theoretical analyses are limited by the fact that the behaviour of the subject must also be considered. But since the BCI control is based on deliberate actions of the subject, the subject's behaviour cannot be easily described [95]. Therefore, it is difficult to analyse the stability of such adaptive systems theoretically.

2.7 Ensemble Classifiers in BCI

Many ensemble methods have been attempted for BCI with the objective of improving ITR and classification accuracy [143]. It is commonly accepted that classifier ensemble can outperform a single classifier under most conditions [144]. Here we briefly review the state of the art in ensemble classifiers.

Ensemble classifiers have been known by several names in literature such as: combination of multiple classifiers, classifier fusion, mixture of experts, consensus aggregation, voting pool of classifiers, divide-and-conquer classifiers, stacked generalization, collective recognition methods and composite classifier systems [168].

Stability of the classifier is an important factor in ensemble classifiers. Lotte [144] has defined a stable classifier as one presenting a high bias and a low variance. An unstable classifier usually results in a low bias and a high variance for training data [150].

Classifier ensembles have been described as being particularly efficient for synchronous BCI [144]. They are capable of decreasing the error variance [158, 175]. Lotte [144] shows that the classification error in BCI systems is formed by the three components, noise, bias, and variance. Since the variability of EEG signals is large in BCI systems, the main component of the error function is the variance. Therefore, decreasing the variance is very important for EEG signal classification [144,174]. However, the effective improvements in terms of error variance depends on the stability of the classifiers included in the ensemble. Therefore the combined classifiers must be unstable in the sense described in [83, 144] in order to successfully decrease the error variance. On the contrary if the combined methodologies are stable, i.e., they present a low variance, the resulting ensemble will probably present the same error, since the combination mainly targets the variance error.

Another positive feature of ensembles is their capability to cope with high-dimensional data with small training sets [185, 187]. Larger the dimensionality of the feature space, more samples have to be taken into account for training a classifier. This so-called "curse of dimensionality" is caused by the increase of complexity in high-dimensional spaces when estimating the decision surface, which is the surface in the feature space generated by training the classification procedure for discriminating among classes [83]. A rule of thumb even advises 5to10 training samples per class and per feature component [144, 162, 179]. The availability of high-dimensional data in EEG warrants the use of ensembles in BCI.

The advantage of ensembles can be attributed to the fact that they divide the complexity of the original decision surface estimation to simpler problems. This reduction even leads, in some cases, to a reduction in the dimensionality of the feature space, e.g., in ensembles based on bagging and feature sub-sampling. However, other re-sampling strategies like random sub-sampling without replacement reduce the training data sets even more. Consequently, they should not be applied on small training sets [143].

Another important issue in the application of ensemble classifiers is the number of components to be generated [180]. Salvaris [181] has evaluated performance variation with respect to the number of components with random sampling without replacement. In this case the optimal number is four and performance decreases when augmenting it. The degradation in performance is caused by the fact that, with each new classifier the number of samples to train is less because of the chosen re-sampling strategy. However, Sun has shown that classifier ensembles are able to make use of the time variability of EEG signals by partitioning data in the time domain [187]. Sun [187] advocates to increase the number of classifiers when data is partitioned in the time domain.

The first publication describing an ensemble classifier for BCI [177] has used a decision fusion framework to combine the classification decisions from different Linear Vector Quantization (LVQ) classifiers. Voting logic has been applied for fusing the decisions from each classifier in order to arrive at the final classification decision. Feature integration approaches, where different features are combined can be found in [153, 160]. Coyle [153] has carried out feature extraction in the temporal, spatial, and frequency domains and has sequentially combined the features for the ensemble. In [160], the features generated by setting up different configurations of a basic processing chain are concatenated. The result from feature concatenation is delivered to a final classifier, which compares the performance of a SVM and a logistic regression classifier [160].

Other interesting applications of ensembles for BCI include ensemble of SVM classifiers [85]. Rakotomamonjy has used each SVM classifier to classify a group of channels selected through accuracy analysis and has tuned it with a parameter set [85]. Ensemble of LDA's has been used in [181], where the feature extraction is carried out by wavelet coefficient computation for different types of wavelets. Johnson et al has used an ensemble of stepwise Linear Discriminant Analysis classifiers [164]. Different fusion operators are used in each approach and their performances are compared in [164].

Density estimation to learn class conditional distributions has been attempted by Hastie et al [205] for discriminant analysis of Gaussian mixtures. Using probability forecasting has been extensively studied by Dawid et al. in [206] for probabilistic expert systems. Bayesian combination of classifiers has been extensively studied by Ghahramani et al in [207]. Recent advances include a unifying framework for learning linear combiners for classifier ensembles [208] and Bayesian combination of multiple imperfect classifiers proposed by Simpson et al in [209]. An ensemble framework for constructing subject independent BCI classification has also been attempted by Fazli et al in [155]. For stationary data, the Bayesian optimal classifier combination has been proposed by Kuncheva [102]. We later present how ensemble classifiers can be used to improve ITR under non-stationary conditions.

This chapter presented an overview of EEG based BCI and reviewed related literature upon which the current study is based. The subsequent chapters present the proposed methods to increase ITR by improving classification accuracies.

Chapter 3

Joint Diagonalization for Multi Class Common Spatial Patterns

3.1 Introduction

Usability of BCI's in real world applications is hindered by the low ITR of BCI systems. Therefore it is vital to improve the pattern classification framework driving the BCI systems in order to achieve higher ITR that give more robust and reliable control. ITR can be increased by increasing the number of different classes as well as by improving the classification accuracy.

Since every EEG electrode only measures a superposition of signals derived from various sources in the brain, it is a difficult task to find the signal that originates at a specific scalp location. One of the main problems in this context is the low signal to noise ratio (SNR) of the recorded data. This has motivated research on spatial filters that are designed to extract those components of the EEG/MEG data that provide most information on the intention of the BCI user. Spatial Filters are tools for extracting specific sources, but they can also be used to alleviate the influence of non-cerebral signals such as eye blinks or head movements.

One algorithm that is very frequently used for this purpose is the common spatial patterns (CSP) algorithm. CSP is a technique to analyse multichannel data based on recordings from two classes (conditions). CSP was first proposed for the analysis of EEG/MEG in [66] and was applied for classification of motor imagery in [210].

The CSP algorithm calculates optimal features for binary classification. The CSP algorithm is capable of calculating spatial filters that maximize the ratio between the variances of data conditioned on two classes, when the EEG/MEG data of two different classes are provided [136]. An underlying limitation of CSP is that it can only handle two classes. This is because simultaneous diagonalization, upon which CSP is based, can be carried out only for two matrices. There is no canonical method for computing the relevant CSP patterns for multi-class classification [14].

A number of methods have been proposed to extend the CSP algorithm to multi-class paradigm [193]. Performing two-class CSP on different combinations of classes is one method of extending CSP to multi-class case (e.g., by computing CSPs for all combinations of classes or by computing CSP for one class versus all the other classes). An extension of CSP for multi-class case has been proposed in [193] where it is decomposed into a set of binary problems. Spatial patterns for each class against all others are calculated in this approach. Classification is then performed on the variances of the projections of the EEG signals on all these CSP patterns [102]. However, the performance of one versus rest CSP in general is still limited [194].

Joint approximate diagonalization (JAD) provides a more intuitive alternative for multi-class CSP (MCSP). Multiple matrices are simultaneously diagonalized using approximate optimization methods in JAD. A linear Least Squares algorithm for joint diagonalization has been attempted in [14]. CSP by joint approximate diagonalization has been shown to be equivalent to independent component analysis (ICA) in [194]. By improving the diagonalization step, better classifiers can be built, resulting in higher classification accuracies for multiple classes. Improved accuracies for multi-class motor imagery BCI will lead to increased ITR of BCI systems.

Two implementations of multi-class CSP (MSCP) are proposed in this chapter. First method is based on fast Frobenius algorithm and the second method utilizes Jacobi angles for joint diagonalization.

This chapter is organized as follows. Section 3.2 provides descriptions of the proposed JAD methods. In Section 3.3 the methodologies synthesizing MCSP and classifiers are described. The Data and experimental paradigm are presented in Section 3.4, followed by comparative results in Section 3.5. In Section 3.6, the conclusions are drawn up.

3.2 Methods

Proper preprocessing of data is vital for the ultimate success of a learning machine. Non-informative dimensions of the data can be discarded and the features of interest for classification can be selected through suitable preprocessing techniques. The figure (3.1) shows a schematic diagram for the proposed method. First, the training data and test data are subjected to bandpass filtering. Joint approximate diagonalization is applied on multiple covariance matrices for each class resulting in a single projection matrix that is used to extract the bandpower features from the data. The training data is projected and multiple discriminant analysis (MDA) is applied to select the features for training the classifier. Test data is also projected and best features are selected by MDA. Multi-class classifiers produce classification decisions on the test data.

In this section, two JAD methods are presented. The first method is based on Fast Frobenius Algorithm and the second method is based on Jacobi angles for simultaneous diagonalization.

3.2.1 Fast Frobenius Algorithm for Joint Diagonalization

The fast algorithm for joint diagonalization (FFDIAG) is founded on the Frobenius norm formulation. Frobenius norm formulation has been used in a few approaches for joint diagonalization in literature [195, 196].

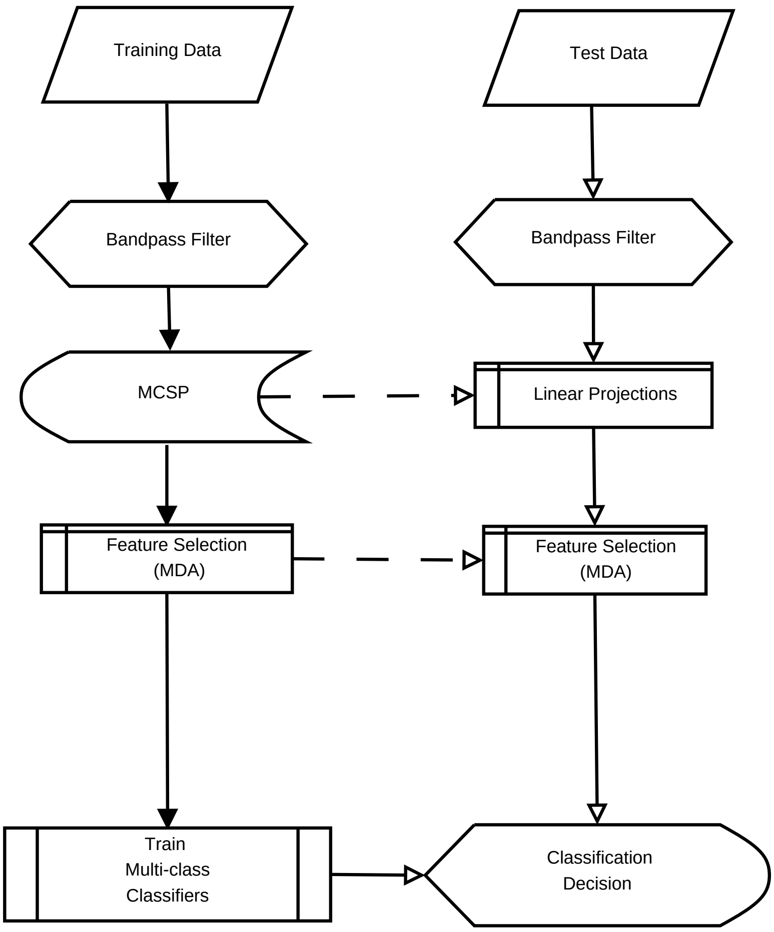


Figure 3.1: Schematic Diagram.

The training data and test data are subjected to bandpass filtering. JAD method is applied to obtain the projection matrix using the training data. The test data are projected using a single projection matrix to extract band power features. Multiple discriminant analysis is applied on the extracted features to select the most informative features. Selected features from training data are used to train multi-class classifiers. Multi-class classifiers produce classification decisions on the test data.

Let, $F^k = VC^kV^T$ denote the result of applying transformation V to matrix C_k. Joint diagonalization can be expressed as the following optimization problem:

$\min_{v\in\mathbb{R}^{\mathbb{N}\times\mathbb{N}}} \sum_{k=1}^{K} M_D(F^k)$, where the diagonality measure M_D is the Frobenius norm of the off-diagonal elements in F_k:

$$M_D(F^k) = off(F^k) = \sum_{i\neq j} \left(F^k_{ij}\right)^2 . \tag{3.1}$$

The FFDIAG proposed in [197] is an iterative scheme to approximate the solution of the following optimization problem:

$$\min_{v\in\mathbb{R}^{\mathbb{N}\times\mathbb{N}}} \sum_{k=1}^{K} \sum_{i\neq j} \left(\left(VC^kV^T\right)_{ij}\right)^2. \tag{3.2}$$

The invertibility of the matrix V is used as a constraint preventing convergence of the cost function to the trivial solution of $V = 0$. Invertibility is implicitly assumed in many applications of diagonalization algorithms, e.g. in blind source separation. Therefore making use of such a constraint is very natural and does not limit the generality from the practical point of view [197].

Invertibility can be guaranteed by carrying out the update of V in multiplicative form as, $V_{(n+1)} \to (I + W_{(n)})\, V_{(n)}$, where I denotes the identity matrix, the update matrix $W_{(n)}$ is constrained to have zeros on the main diagonal, and n is the iteration number. In order to maintain invertibility of V it is sufficient to enforce invertibility of $I + W_{(n)}$.

According to the Levi-Desplanques Theorem, if an $n \times n$ matrix A is strictly diagonally-dominant, then it is invertible [197]. An $n\times n$ matrix A is said to be strictly diagonally dominant if, $\|a_{ii}\| > \sum_{j\neq i} \|a_{ij}\|$, for all $i = 1, \ldots, n$, where a_{ij} are elements of matrix A.

The Levi-Desplanques theorem can be used to control invertibility of $I + W_{(n)}$. The diagonal entries in $I + W_{(n)}$ are all equal to 1. Therefore, it suffices to ensure that $\max_i \sum_{j\neq i} \|W_{ij}\| = \|W_{(n)}\|_\infty < 1$. This can be achieved by dividing $W_{(n)}$ by its infinity norm whenever the latter exceeds some fixed $\theta < 1$. An even stricter condition can be imposed by using a Frobenius norm in the same way as, $W_{(n)} \to \frac{\theta}{\|W_{(n)}\|_F} W_{(n)}$. To determine the optimal update $W_{(n)}$ at each iteration, first-order optimality constraints for the objective (3.2) are used. A special approximation of the objective function enables efficient computation of $W_{(n)}$.

Let $D^k_{(n)}$ and $E^k_{(n)}$ denote the diagonal and off-diagonal parts of $C^k_{(n)}$, respectively. In order to simplify the optimization problem we assume that the norms of $W_{(n)}$ and $E^k_{(n)}$ are small, i.e. quadratic terms in the expression for the new set of matrices can be ignored. $C^k_{(n+1)} = (I + W_{(n)})\left(D^k_{(n)} + E^k_{(n)}\right)(I + W_{(n)})^T$, $C^k_{(n+1)} \approx D^k_{(n)} + W_{(n)}D^k_{(n)} + D^k_{(n)}W^T_{(n)} + E^k_{(n)}$. With these simplifications, and ignoring the already diagonal terms D^k, the diagonality measure (3.1) can be computed using expressions linear in W,

$$F^k \approx \tilde{F}^k = WD^k + D^k W^T + E^k. \tag{3.3}$$

The linearity of terms in (3.3) allows to explicitly compute the optimal update matrix $W_{(n)}$ minimizing the approximated diagonality criterion, $\min_W \sum_{k=1}^{K} \sum_{ij} \left(\left(WD^k + D^k W^T + E^k \right)_{ij} \right)^2$.

The FFDIAG algorithm is able to approximate the joint diagonal matrix owing to the sparseness introduced by (3.3). If the $N(N-1)$ off-diagonal entries of the update matrix W are arranged as a vector $w = \left(W_{12}, W_{21}, \ldots W_{ij}, W_{ji}, \ldots \right)^T$, where the order of elements in w reflects the pairwise relationship of the elements in W. If the $KN(N1)$ off-diagonal entries of the matrices E^k are also arranged as, $e = \left(E^1_{12}, E^1_{21}, \ldots E^1_{ij}, E^1_{ji}, \ldots, E^k_{ij}, E^k_{ji} .. \right)$. A large but very sparse, $KN(N-1)N(N-1)$ matrix J is built in the following form
$J = \begin{pmatrix} j_1 \\ \vdots \\ j_k \end{pmatrix}$ with $J_k = \begin{pmatrix} D^k_{12} & & \\ & \ddots & \\ & & D^k_{ij} \end{pmatrix}$, where each J_k is block-diagonal, containing $\frac{N(N1)}{2}$ matrices of dimension 2×2. $D^k_{ij} = \begin{pmatrix} D^k_j & D^k_i \\ D^k_j & D^k_i \end{pmatrix}$, $i, j = 1, \ldots, N, i \neq j$, where D^k_i is a short-hand notation for the ii^{th} entry of the diagonal matrix D^k.

The approximate cost function can be re-written as the linear least-squares problem $L(w) = \sum_k \sum_{i \neq j} \left(\tilde{F}^k_{ij} \right)^2 = (jw + e)^T (jw + e)$.

The solution to this problem is,

$$w = -\left(J^T J \right)^{-1} J^T e. \tag{3.4}$$

Using the sparseness of J and e to enable the direct computation of the elements of w in (3.4), the matrix product $J^T J$ can be written as a block-diagonal matrix,
$J^T J = \begin{pmatrix} \sum_k \left(D^k_{12} \right)^T D^k_{12} & & \\ & \ddots & \\ & & \sum_k \left(D^k_{ij} \right)^T D^k_{ij} \end{pmatrix}$ whose blocks are 2×2 matrices.

Thus the system (3.4) actually consists of decoupled equations,
$\begin{pmatrix} W_{ij} \\ W_{ji} \end{pmatrix} = - \begin{pmatrix} z_{jj} & z_{ij} \\ z_{ij} & z_{ii} \end{pmatrix}^{-1} \begin{pmatrix} y_{ij} \\ y_{ji} \end{pmatrix}, i, j = 1, \ldots, N, i \neq j,$
where $z_{ij} = \sum_k D_i^k D_j^k$ and $y_{ij} = \sum_k D_j^k \frac{E_{ij}^k + E_{ji}^k}{2} = \sum_k D_j^k E_{ij}^k$.

The matrix inverse can be computed in closed form, leading to the following expressions for the update of the entries of W, $W_{ij} = \frac{z_{ij} y_{ji} - z_{ii} y_{ij}}{z_{jj} z_{ii} - z_{ij}^2}$ and $W_{ji} = \frac{z_{ij} y_{ij} - z_{jj} y_{ji}}{z_{jj} z_{ii} - z_{ij}^2}$. Therefore, only the off-diagonal elements ($i \neq j$) need to be computed and the diagonal terms of W are set to zero. This makes this algorithm faster than other JAD methods [197].

3.2.2 Jacobi Angles for Simultaneous Diagonalization

Another approach for joint approximate diagonalization (JAD) is known as Jacobi angles for joint diagonalization. This method is based on the Jacobi technique which is a joint diagonality criterion optimized iteratively under plane rotations [195].

Consider a set, $C = \{C_k | k = 1, \ldots, K\}$ of K, $N \times N$ matrices. The off-diagonal elements of C can be defined as,

$$off(C) = \sum_{1 \leq i \neq j \leq N} ||c_{ij}||^2 \tag{3.5}$$

where c_{ij} denotes the $(i, j)th$ entry of matrix C. Simultaneous diagonalization can be obtained by minimizing the composite objective ${}_{k=1}^{K} off\left(UC_kU^H\right)$, by a unitary matrix U where the superscript H denotes the Hermitian transpose. The extended Jacobi technique for simultaneous diagonalization constructs U as a product of plane rotations globally applied to all the matrices in C. A plane rotation in the (i, j)-plane is a unitary matrix $R = R(i, j, c, s)$ defined as $R = I + (c - 1)e_i e_i^T - s e_i e_j^T + s e_j e_i^T + (c - 1)e_j e_j^T$ where $c, s \in C$ and $||c||^2 + ||s||^2 = 1$. It is desired for each choice of $i \neq j$, finding complex angles c and s that minimize the following objective function: $O(c, s) = \sum_{k=1}^{K} off\left(R(i, j, c, s) C_k R^H(i, j, c, s)\right)$. For a given pair of indices (i, j), a 3×3 real symmetric matrix G is defined as $G = Real\left(\sum_{k=1}^{K} h^H(C_k) h(C_k)\right)$.

Jacobi angles can be computed for any set of $N \times N$ matrices using the theorem shown in equation (3.6). Under the constraint $||c||^2 + ||s||^2 = 1$, the objective function, $O(c, s)$ is minimized at,

$$c = \sqrt{\frac{x+r}{2r}},\ s = \frac{y - iz}{\sqrt{2r(x+r)}} \text{ and } r = \sqrt{x^2 + y^2 + z^2} \tag{3.6}$$

where $[x, y, z]^T$ is any eigenvector associated with the largest eigenvalue of G. Proof of this theorem can be found in [195].

Thus, the minimization of $O(c, s)$ under the constraint $||c||^2 + ||s||^2 = 1$ is equivalent to maximization of real 3×3 quadratic form under unit norm constraint. The solution is given by unit norm eigenvector of G associated with the maximum eigenvalue. More theoretical analysis of this method can be found in [199].

When C_k is a set of real symmetric matrices, the rotation parameters c and s become real. The last component of each vector $h(C_k)$ then is zero and G is reduced to a 2×2 matrix by deleting the last row and the last column.

3.3 Synthesized Methods

We investigated the use of the FFDIAG algorithm and Jacobi angles method for approximate diagonalization to develop multi-class common spatial patterns.

The first algorithm was implemented by utilizing the FFDIAG method to jointly diagonalize M number of covariance matrices. The Frobenius norm of covariance matrices C_k are calculated according to (3.1) and the minimization problem shown in (3.2) is iteratively deduced as explained in the section (3.2.1). The resulting eigenvectors are employed to spatially filter the covariance matrices.

The second method based on Jacobi angles also takes the multiple covariance matrices as inputs. This corresponds to the matrix C in equation (3.5). The real part of the resulting diagonalized matrix is used to spatially filter the covariance matrices.

Multiple discriminant analysis (MDA) is carried out in order to select the most discriminating features from the filtered covariance data. Thirteen features were selected in order to distinguish the four classes. These selected features were used to train the classification algorithms and the 10×10 cross-validation accuracies were calculated.

The performances of the implemented spatial filters were compared with one another and one versus rest multi-class CSP using three multi-class classifiers. K-Nearest Neighbour, Classification and Regression Trees, and Support Vector Machine classifiers were implemented and the performances were compared.

Classifier boosting with Adaboost and Stagewise Additive Modelling using a Multi-class exponential loss function (SAMME) algorithm was also investigated to analyse the effects of boosting to improve classification accuracy.

3.3.1 Adaboost

Adaboost algorithm stands for adaptive boosting. Boosting is related to the general problem of producing a very accurate prediction rule by combining rough moderately inaccurate rules of thumb [113]. The general idea of boosting is to develop a team of classifiers incrementally, adding one classifier at a time.

The classifier that joins the ensemble at a given step is trained on a data set selectively sampled from the training set. The sampling distribution begins with a uniform distribution giving all training data equal chance to be selected. In later steps, the training data points which are harder to classify are given higher likelihood to be chosen.

The Adaboost.M1 algorithm which is the multi-class extension of the Adaboost algorithm was implemented in this work. The base classifier in this implementation was SVM classifier, as other classifiers did not satisfy the minimum accuracy of 50% to be used as base classifier. 20

weak learners were combined in the implementation.

3.3.2 Stagewise Additive Modelling using a Multi-class exponential loss function

AdaBoost.M1 is a trivial extension of AdaBoost to the multi-class classification problem, in which the only modification is that the component classifiers must be capable of multi-class classification. However, the component classifiers are still required to have accuracies greater than 50%. This requirement places an undue constraint on the type of classifiers that can be boosted. Several approaches have been designed to lift this restriction [102].

Stagewise Additive Modelling using a Multi-class exponential loss function (SAMME) is a natural extension of AdaBoost to the multi-class case. A major difference is that component classifiers are no longer required to have accuracies greater than 50%. They are needed only to be better than random guessing. SAMME was proposed in [103]. Empirical tests conducted show that performance to be comparable, if not slightly better, than that of AdaBoost [103].

The SAMME algorithm was employed to boost the classifiers whose performances were not good enough to be boosted using Adaboost.M1.

3.4 Data and Experimental Procedure

The data set 2A of the BCI Competition IV [142] considered in this study, is comprised of EEG data collected from 9 subjects. The data has been recorded during two sessions on separate days for each subject. Four different motor imagery tasks: left hand (class 1), right hand (class 2), both feet (class 3), and tongue (class 4) has been considered in this dataset. Each session is comprised of 6 runs separated by short breaks, each run comprised 48 trials (12 for each class), amounting to a total of 288 trials per session.

The subjects have been seated on an armchair in front of a computer screen and at the beginning of a trial ($t = 0s$), a fixation cross has appeared on the black screen. Short acoustic

Figure 3.2: BCI Competition IV Data Set 2A: Timing Scheme

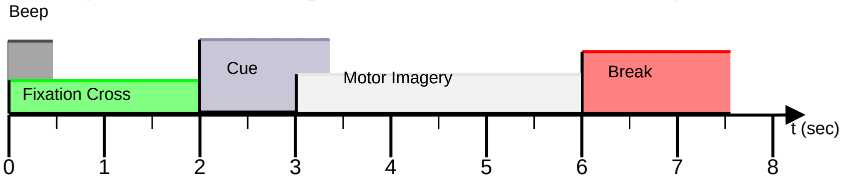


Table 3.1: Comparative classification accuracy: k-NN classifier

Subject	1	2	3	4	5	6	7	8	9	Avg.
FFDIAG	49.3	40.3	49.4	49.3	48.6	49.3	48.2	50.1	49.2	48.2
Jacobi	29.1	27.4	28.9	29.2	28.7	29.4	27.9	32.1	29.1	29.1
CSP (OVR)	26.3	25.1	26.2	25.1	26.9	24.3	26.1	27.0	25.8	25.9

warning tones have also been presented at the start of the trial. After two seconds ($t = 2sec$), a cue has been presented. This cue has been in the form of an arrow pointing either to the left, right, down or up (corresponding to one of the four classes left hand, right hand, foot or tongue). The cue has appeared and stayed on the screen for 1.25 seconds and the subjects were supposed to perform the corresponding motor imagery task. The subjects have been instructed to carry out the motor imagery tasks until the fixation cross disappeared from the screen at $t = 6sec.$, without any feedback on their performance. A short break had been given before the next trial. This procedure has been repeated for each of the 6 runs in a session. The timing scheme of this paradigm is depicted in figure (3.2). For more details on the protocol please refer to [142].

3.5 Results and Discussions

Cross-validation results obtained for the proposed methods of multi-class CSP based on FF-DIAG and Jacobi angles with k-NN classifier are depicted in table (3.1). One over rest application of the binary CSP is also presented in order to compare the performances.

Table (3.2) shows the cross validation results obtained for the same multi-class CSP methods where the classification is carried out by Classification and Regression Trees (CART) algorithm.

Table 3.2: Comparative classification accuracy: CART classifier

Subject	1	2	3	4	5	6	7	8	9	Avg.
FFDIAG	43.8	35.6	44.2	43.4	43.1	43.5	41.9	44.7	43.9	42.7
Jacobi	25.4	24.7	25.2	26.5	25.1	26.7	24.8	29.6	25.3	25.9
CSP (OVR)	26.1	24.8	25.9	24.5	26.3	24.1	25.4	26.8	25.3	25.5

Table 3.3: Comparative classification accuracy: SVM classifier

Subject	1	2	3	4	5	6	7	8	9	Avg.
FFDIAG	63.2	58.8	64.2	42.1	39.4	42.6	56.3	69.3	45.9	53.6
Jacobi	33.4	30.9	31.2	33.7	32.4	33.1	31.8	35.3	33.5	32.8
CSP (OVR)	26.9	23.3	28.9	27.6	27.8	28.1	28.9	29.5	29.8	27.8

Results obtained for the classification by Support Vector Machines (SVM) is presented in table (3.3).

Table (3.4) shows the results yielded for FFDIAG with the k-NN classifier boosted by SAMME algorithm. Table (3.5) and table (3.6) depict the results of FFDIAG method boosted by SAMME with CART algorithm and SVM classifiers as the base classifiers respectively.

The results of the Adaboost.M1 algorithm applied to the FFDIAG method with SVM as the base classifier is presented in table (3.7). The Adaboost.M1 cannot be applied to one versus rest CSP method because the base classifier does not meet the performance requirement of 50% for the considered data set.

The highest average classification accuracy of 54.1% is recorded by the JAD method based on FFDIAG when the classification is carried out by SVM classifier boosted by SAMME algorithm. SVM boosted by Adaboost.M1 yields an average accuracy of 53.8% for FFDIAG method. Average accuracies of 53.6%, 48.2% and 42.7% are yielded under multi-class SVM, k-NN and CART

Table 3.4: Comparative classification accuracy: k-NN classifier Boosted with SAMME

Subject	1	2	3	4	5	6	7	8	9	Avg.
FFDIAG	50.4	42.2	49.9	49.6	49.2	49.9	49.3	52.1	49.4	49.1
CSP (OVR)	29.1	26.6	27.9	26.3	27.2	26	27.8	28.7	26.9	27.3

Table 3.5: Comparative classification accuracy: CART classifier Boosted with SAMME

Subject	1	2	3	4	5	6	7	8	9	Avg.
FFDIAG	45.7	36.4	48.2	46.8	43.8	46.4	42.1	47.2	45.3	44.6
CSP (OVR)	28.7	26.7	28.8	27.2	29.3	26.5	25.9	30.4	26.5	27.7

Table 3.6: Comparative classification accuracy: SVM classifier Boosted with SAMME

Subject	1	2	3	4	5	6	7	8	9	Avg.
FFDIAG	63.6	58.5	61.2	46.3	38.8	42.7	58.8	66.6	50	54.1
CSP (OVR)	27.2	26.9	29.5	27.2	28	29.1	29.2	31.4	30.6	28.8

classification methods respectively for the same diagonalization method. FFDIAG method with the classifiers boosted using SAMME yield 49.1% and 44.6% under the k-NN and CART as base classifiers. The classification accuracies of 10×10-fold cross-validation indicate that the JAD method based on FFDIAG performs better than the one versus rest CSP. The Jacobi angles based method slightly outperforms the one versus rest binary CSP.

Support Vector Machines (SVM) outperforms the other two classifiers. The best average performance is produced by the SAMME boosting algorithm. The average classification accuracy for one versus rest CSP boosted by SAMME using the CART as the base classifier is almost the same as the accuracy yielded by SVM without boosting.

This observation can be attributed to the instability of the CART classifier in the presence of noise. Unstable base classifiers generate sufficiently different decision boundaries even for small perturbations in their training parameters [83]. Therefore it can be inferred that the performance of CART classifier is boosted more by the SAMME algorithm than k-NN classifier. However, the SVM classifier gives more robust classification results overall than the other classifiers considered.

Table 3.7: Comparative classification accuracy: SVM classifier Boosted with Adaboost.M1

Subject	1	2	3	4	5	6	7	8	9	Avg.
FFDIAG	60.4	58.1	62.2	46.9	43.3	44.3	54.5	59.8	54.7	53.8

3.6 Conclusion

In this chapter a blind source separation approach based on JAD methods was proposed for multi-class Common Spatial Patterns for processing EEG measurements in multi-class motor imagery-based BCI. MCSP extends the binary CSP technique to a truly multi-class paradigm and proves to be better than one versus rest application of the binary CSP.

The proposed JAD method was compared on the BCI Competition IV for dataset 2a. Experimental results showed that the proposed MCSP based on FFDIAG yields superior classification accuracy compared to the one versus rest CSP method.

In the analysis carried out on the three classification algorithms and the two boosting algorithms, it was identified that SVM algorithm consistently gives a higher accuracy than the other two classification methods. The SAMME algorithm for boosting slightly outperforms the Adaboost.M1 algorithm for multi-class boosting with SVM as the base classifier. However due to the complexity of the considered dataset none of the other classifiers reached the required performance to be boosted using Adaboost.M1. The results of k-NN and CART classifiers boosted using SAMME algorithm did not yield satisfactory results as the SVM classifier.

In the next chapter, we will present the adaptively weighted ensemble classification technique for addressing the non-stationarity in EEG.

Chapter 4

Adaptively Weighted Ensemble Classification

4.1 Introduction

A major challenge for BCI research is the non-stationarity in the brain activity occurring continuously in association with diverse behavioural and mental states [200]. Non-stationarity refers to a change in the class definitions over time, which therefore causes a change in the distributions from which the data are drawn [9]. Consider the Bayesian posterior probability of a class ω given instance x, $P(\omega|x) = \frac{P(x|\omega)\cdot P(\omega)}{P(x)}$, non-stationarity is defined as any scenario where the posterior probability changes over time, i.e., $P_{t+1}(\omega|x) \neq P_t(\omega|x)$, where ω is the class to which data instance x belongs.

The non-stationarity of EEG signals is caused by factors such as, changes in the physical properties of the sensors, variabilities in neurophysiological conditions, psychological parameters, ambient noise, and motion artefacts. Two main factors contributing to non-stationarity as reported in [10, 11] are: the differences between the samples extracted from a training session and the samples extracted during an online session, and the changes in the user's brain activity during online operation. As a result, the general hypothesis that the signals sampled in the training set follow a similar probability distribution to the signals sampled in the test set from a different session is violated [12].

Kaplan has studied fast dynamics of quasi-stationary episodes in EEG signals and has identified different operating modes in the EEG time series [201]. Several machine learning techniques have been attempted recently to address the non-stationarity issue in BCI [202–204]. Robust PCA has proposed to visualize spatial patterns with the most prominent variability in the data to automatically identify and reject outlying non-informative signals [202]. Stationary LDA attempts to find a direction in feature space which is both discriminative and stationary [203]. Stationary sub-space analysis is an unsupervised learning method that finds sub-spaces in which data distributions stay invariant over time [204]. Methods such as Bayesian transduction, transfer learning, active learning, and distribution matching has also been proposed to address the non-stationarity issue [106]. Even though it would be interesting to study the application of these methods, it exceeds the scope of the current study.

Density estimation to learn class conditional distributions has been attempted by Hastie et al. [205] for discriminant analysis of Gaussian mixtures. Using probability forecasting has been extensively studied by Dawid et al. in [206] for probabilistic expert systems. Bayesian combination of classifiers has been extensively studied by Ghahramani et al in [207]. Recent advances include a unifying framework for learning linear combiners for classifier ensembles [208] and Bayesian combination of multiple imperfect classifiers proposed by Simpson et al in [209].

In this chapter we propose an Adaptively Weighted Ensemble Classification (AWEC) framework to cluster features extracted using Common Spatial Patterns (CSP), and build an ensemble of multiple classifiers on the clustered features in order to address the session to session non-stationarity in the EEG data for the operation of a BCI. Clustering the features extracted after CSP filtering facilitates the identification of different modes in the EEG. Classifiers trained on the clustered features offer complimentary decisions. Improved accuracies can be achieved by appropriately combining the decisions from an ensemble of multiple classifiers. An ensemble framework for constructing subject independent BCI classification has also been attempted by Fazli et al in [155].

For stationary data, the Bayesian optimal classifier combination has been proposed by Kuncheva [102]. This work extends the concept of Bayesian optimal combination for non-stationary data. Since the underlying distribution of the test data is unknown, classification accuracies for each classifier need to be re-estimated. Particularly, we consider each test sample to adaptively estimate the classification accuracy based on the relative location of samples with respect to the clusters.

The remainder of this chapter is organized as follows: Section 4.2 provides the synthesized materials followed by methods in Section 4.3. Section 4.4 presents comparative results and discussion. Finally, section 4.5 concludes the chapter.

4.2 Materials

Two datasets were evaluated using the proposed method. Publicly available BCI Competition IV dataset 2A [142] and motor imagery dataset collected in-house from 12 healthy subjects.

The BCI Competition IV Dataset 2A is comprised of EEG data collected from 9 subjects that were recorded during two sessions on separate days for each subject. The data has been collected on four different motor imagery tasks: left hand (class 1), right hand (class 2), both feet (class 3), and tongue (class 4). Each session is comprised of 6 runs separated by short breaks, each run comprised 48 trials (12 for each class), amounting to a total of 288 trials per session. Only the two class classification between left hand and right hand motor imagery was considered for this study. For more details on the protocol please refer to [142]. The motor imagery data from the first session were used to train the classifiers, and motor imagery data from the second session were used as test data.

The EEG data collected in the laboratory in Institute for Infocomm Research was collected using a Nuamps EEG acquisition hardware (http://www.neuroscan.com) with unipolar Ag/AgCl electrodes, digitally sampled at 250 Hz with a resolution of 22 bits for voltage ranges of $\pm 130mV$. EEG signals from 22 scalp positions, mainly covering the primary motor cortices bilaterally were recorded. The sensitivity of the amplifier was set to $100\mu V$. 12 healthy subjects were recruited for the study. Two subjects chose to perform left hand motor imagery while the remaining 10 subjects chose to perform on the right hand. The subjects were instructed, in the form of visual cues displayed on the computer screen, to perform kinaesthetic motor imagery of the chosen hand, and rest during the background rest condition.

The EEG data were collected in two sessions for this study from each subject on two different days. In the first session, two runs of EEG data were collected from a subject while performing motor imagery of the chosen hand and background rest condition. In the second session on another day, three runs of EEG data were collected while performing motor imagery of the chosen hand and background rest condition. Each run lasted approximately 16 minutes that comprised of 40 trials of motor imagery and 40 trials of background rest condition. The motor imagery data collected during the first session were used to train the classifiers, and motor imagery data from the subsequent sessions were used as test data.

4.3 Methods

The proposed framework consists of two steps: training and testing. In the training step, the EEG data used for training were subjected to pre-processing and feature extraction. In this experiment, EEG data were bandpass filtered at 8-30Hz and spatially filtered using the CSP algorithm. The extracted features of each class were subjected to clustering separately. The clustered features were subsequently used to train an ensemble of multiple classifiers by combining all possible clusters from each class.

In the testing step, the EEG data used for testing were subjected to pre-processing and feature extraction similar to the training data. In this experiment the EEG data used for testing were bandpass filtered at 8-30Hz and spatially filtered using CSP filter trained during the training step. The extracted features were then evaluated by the ensemble of multiple classifiers. The decisions from the classifiers in the ensemble were adaptively combined using a weighted majority voting method based on a similarity measure computed from the distance of the test data to each cluster centre of each classifier.

The following subsections provide a more detailed description of the proposed framework. The figure (4.1) summarizes the processes involved in the proposed method.

4.3.1 Feature Extraction

EEG signals resulting from motor imagery have been found to contain specific temporal, frequential and spatial features, that enables them to be recognized automatically [17]. For example, imagining a left hand movement is known to trigger a decrease of power known as Event Related De-synchronisation (ERD) in the μ and β rhythms, over the right motor cortex [17]. Increase of band power that occurs after the motor imagery is known as Event Related Synchronisation (ERS) [17].

The Common Spatial Patterns (CSP) algorithm was used to extract the features from the EEG data, which is effective in computing spatial filters for detecting ERD/ERS effects [66,210]. It has been extended to multi-class problems in [211], and further extensions and robustifications using simultaneous optimization of spatial and frequency filters have been proposed in [123, 124, 138].

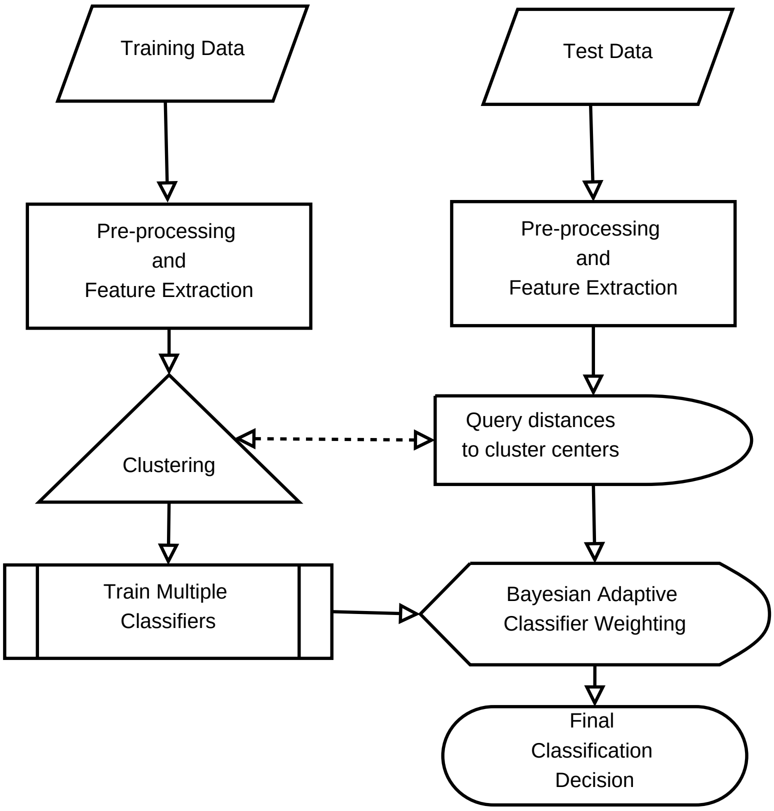


Figure 4.1: Schematic Diagram.

The training data and test data are pre-processed and features are extracted. Training data are clustered and multiple classifiers are trained on clustered features. The decisions from multiple classifiers are adaptively weighted to arrive at the final classification decision.

4.3.2 Clustering of EEG with Minimum Entropy Criterion

Since the features extracted using the CSP algorithm are the solutions of a generalized eigenvalue problem, a multiple of the extracted feature vector is again a solution to the eigenvalue problem. In order to compare the extracted features it should be noted that the feature space is inherently non-Euclidean. An appropriate comparison between two feature vectors x_1 and x_2 in this non-Euclidean space is the angle between these two vectors, given by the cosine distance, $d(x_1, x_2) = 1 - \left(\frac{x_1 \cdot x_2^T}{|x_1| \cdot |x_2|}\right)$. Clustering EEG data using the angle distance between the feature vectors extracted by CSP has been shown to yield correct source signals in high dimensional data [105].

In this work, the features extracted from the training data were initially clustered using k-means algorithm with cosine distance measure. The resulting initial clusters were optimized using minimum entropy criterion [115]. The normalized information distance measures were used to quantify the amount of information shared between clusters.

In the minimum entropy criterion, given a spatially filtered features set $\mathcal{X} = \{x_1, \ldots, x_T\}$ of T items in $\mathbb{R}^n$, a partitional clustering $C = \{c_1, \ldots, c_K\}$ is a way to divide $\mathcal{X}$ into K non-overlapped subsets. If C is the space of all possible $K-$cluster partitions of $\mathcal{X}$, the optimal clustering $C^* \in C$ would have maximum mutual information between the data and the clustering:

$$C^* = \arg\max_{c \in C} \{I(c; X)\}. \tag{4.1}$$

The entropy relation of (4.1) can be expressed as:

$C^* = \arg\min_{c \in C} \{H(X|c)\}$.

The minimum entropy criterion is based on the argument that the optimal clustering would maximize the information shared between the clustering and data. It has been shown that, by using Havrda-Charvats structural entropy measure the conditional entropy can be estimated without any assumptions about the distribution of the data. Havrda-Charvats structural entropy is defined as:

$$H_\alpha = \left(2^{1-\alpha} - 1\right)^{-1} \left[\sum_{k=1}^{K} p_k^\alpha - 1\right], \alpha > 0, \alpha \neq 1, \tag{4.2}$$

where α is the structural dimension, K is the number of partitions and p_k^α is the probability of a sample being included in kth partition in the α-dimension [212].

The equation (4.2) can be simplified by discarding the constant coefficient and with $\alpha = 2$ to give: $H_2 = 1 - \sum_{k=1}^{K} p_k^2$.

The conditional quadratic Havrda-Charvats entropy of $\mathcal{X}$ given C can be defined as:

$$H_2(\mathcal{X}|C) = \sum_{k=1}^{K} p(c_k) H_2(\mathcal{X}|C = c_k). \tag{4.3}$$

With the measure of conditional entropy (4.3) , the objective function (4.4) can be expressed as:

$$C^* = \arg\min_{c \in C} \left\{ \sum_{k=1}^{K} p(c_k) H_2(\mathcal{X}|c = c_k) \right\}. \tag{4.4}$$

Estimating the conditional entropy without information about the underlying probability distributions is difficult. A solution is to use Parzen window [213] method for density estimation as suggested in [214]. Principe et al have used Parzen window method in conjunction with quadratic Renyis entropy for density estimation [215]. In a similar manner we use the Parzen window [213] to estimate the conditional entropy. Given that a Gaussian kernel in n-dimensional space is

$$G\left(x - a, \sigma^2\right) = \frac{1}{(2\pi\sigma)^{\frac{n}{2}}} \exp\left\{ \frac{-\left\|x - a\right\|^2}{2\sigma^2} \right\}, \tag{4.5}$$

where σ is the kernel width parameter, and a is the center of the Gaussian window; the probability density estimation of $x \in \mathcal{X}$ can be expressed as

$$p(x) = \frac{1}{T} \sum_{i=1}^{T} G\left(x - x_i, \sigma^2\right). \tag{4.6}$$

The quadratic entropy of features $\mathcal{X}$ can then be estimated by

$$\begin{aligned} H_2(\mathcal{X}) &= 1 - \int_x p^2(x)\, dx \\ &= 1 - \frac{1}{T^2} \int_x \left(\sum_{i=1}^{T} G\left(x - x_i, \sigma^2\right) \right)^2 dx. \end{aligned} \tag{4.7}$$

Since convolving two Gaussians yield a Gaussian, equation (4.7) can be expressed as

$$H_2(\mathcal{X}) = 1 - \frac{1}{T^2}\sum_{i=1}^{T}\sum_{j=1}^{T} G\left(x_i - x_j, 2\sigma^2\right). \tag{4.8}$$

In a similar manner, the conditional quadratic entropy can be estimated as

$$H_2(\mathcal{X}|C = c_k) = 1 - \frac{1}{t_k^2}\sum_{x_i \in c_k}\sum_{x_j \in c_k} G\left(x_i - x_j, 2\sigma^2\right), \tag{4.9}$$

where t_k is the number of the data items in cluster c_k. Given the estimate in equation (4.9), the objective function (4.4) can be written as

$$C^* = \arg\max_{c \in C}\left\{\sum_{k=1}^{K} p(c_k)\frac{1}{t_k^2}\sum_{x_i \in c_k}\sum_{x_j \in c_k} G\left(x_i - x_j, 2\sigma^2\right)\right\}. \tag{4.10}$$

Here the probability of encountering the cluster c_k in C is $\frac{t_k}{T}$. Therefore the conditional entropy ε based objective function becomes

$$C^* = \arg\max_{c \in C} \varepsilon(C), \tag{4.11}$$

where,

$$\varepsilon(C) = \sum_{k=1}^{K}\frac{1}{t_k}\sum_{x_i, x_j \in c_k} \exp\left\{\frac{-\left\|x_i - x_j\right\|^2}{4\sigma^2}\right\}. \tag{4.12}$$

Therefore, by maximizing $\varepsilon(C)$, the conditional entropy criterion is minimized.

4.3.3 Base Classifier

The class-wise training data partitioned to clusters were used to train the ensemble. Individual SVM classifiers that make up the ensemble were trained independently.

SVM has been found to yield highest classification accuracies for synchronous BCI experiments [216]. Dara et al [217] has shown that classification performance of a single SVM classifier can be surpassed by using an ensemble of SVM classifiers. It has also been shown that a combination of different SVM classifiers expands the regions of test samples resulting in improved classification. If there are L different SVM classifiers in an ensemble that has been trained independently on different training samples, then each SVM classifier would have different generalization performances [84].

The SVM classifier has been known to show good generalization performance with easy to learn exact parameters for the global optimum [84]. Considering all these factors, SVM classifiers with linear kernels were used as the base classifiers in the ensemble.

4.3.4 Adaptively Weighted Ensemble Classification (AWEC) Method for Non-stationary Data

A classifier is any function $\Lambda : \mathbb{R}^n \to \Omega$, that maps a given object $x \in \mathbb{R}^n$, where $\mathbb{R}^n$ is the feature space to a class label ω. Let the class label ω be a random variable that can take values in the set of class labels $\Omega = \{\omega_1, ..., \omega_\Gamma\}$, where Γ is the number of classes. The class with the highest posterior probability is the most natural choice for a given object $x \in \mathbb{R}^n$, where $\mathbb{R}^n$ is the feature space. In the canonical model of a classifier [83] a set of Γ discriminant functions $G = \{g_1(x), \ldots, g_\Gamma(x)\}$, $g_i : \mathbb{R}^n \to \mathbb{R}, i = 1, \ldots, c$, each yielding a score for the respective class. The final output class label of the classifier is determined according to the maximum membership rule. Maximum membership rule can be given as $\Lambda(x) = \omega_{i*} \in \Omega \leftrightarrow g_{i*}(x) = \max_{k=1,\ldots,\Gamma} \{g_i(x)\}$. In an ensemble consisting of L such classifiers where each classifier Λ_j, produces a class label $s_j \in \Omega$ where $j = 1, \ldots, L$. Thus for any object $x \in \mathbb{R}^n$ to be classified, the outputs from the L classifiers produce a vector $s = [s_1, \ldots, s_L]^T \in \Omega^L$.

The Bayesian optimal weighted majority voting for combining an ensemble of classifiers has been defined in [102]. The label outputs produced by each classifier in the ensemble are represented as degrees of support for each class in the following manner:

$$\lambda_{j,k} = \begin{cases} 1, & \text{if } \Lambda_j \text{ labels } x \text{ in class } \omega_k \\ 0, & \text{otherwise.} \end{cases}$$

The discriminant function for class ω_k obtained through weighted voting is $g_k(x) = \sum_{j=1}^{L} b_j \lambda_{j,k}$, where b_j is a coefficient for classifier Λ_j. Thus the value of the discriminant function would be the sum of the coefficients for these members of the ensemble whose outputs for x are ω_k. In this context, the optimal set of discriminant functions based on the outputs of the L classifiers is

$$g_k(x) = \log P(\omega_k) P(x|\omega_k), k = 1, \ldots, \Gamma.$$

Kuncheva [102] has shown that in an ensemble of L classifiers with individual training accuracies $p_1, \ldots, p_L$ the optimal set of discriminant functions can be achieved by weighted majority voting with individual weights

$$b_j \propto \log \frac{p_j}{1 - p_j}, \tag{4.13}$$

where p_j is the training accuracy of the jth classifier where $j = 1, \ldots, L$.

The equation (4.13) is applicable only for stationary data, where the distribution of the training data is similar to the distribution of test data. In the presence of non-stationarity, using equation (4.13) with training accuracies would not lead to the optimal set of discriminant functions. Therefore under non-stationarity, the accuracies for each test sample should be considered individually to reach the optimal set of discriminant functions.

Since the performances of classifiers are not known for the test samples, the weights b_j are actively calculated for each test sample based on estimated individual accuracies of classifiers in the ensemble in the proposed method. An estimate for classification accuracy of each classifier is adaptively calculated based on the distances from test sample to the centres of the clusters consisting of training data.

In the proposed method the training data is partitioned by clustering the features of the two classes separately. Let U and V be the number of clusters of class 1 and class 2 respectively. Let the clusters of class 1 be denoted by c_{1u}, where $u = 1, \ldots, U$ and clusters of class 2 be c_{2v}, where $v = 1, \ldots, V$. Then, the distances from the sample to the cluster center c_{1u} be d_u and distance to cluster center c_{2v} be d_v. Let the ratio between the two distance measures be denoted by d_{uv}, where, $d_{uv} = \frac{d_u}{d_v}$.

A function to estimate the probability of correct classification based on the distance measures to centres of clusters c_{1u} and c_{2v} consisting of training samples for the classifier is defined as

$$p_{uv}(\vec{x_t}) = 1 - \frac{1}{2}exp\left(-\frac{1}{\psi_{uv}^2}\left(log\left(d_{tu}\right) - log\left(d_{tv}\right)\right)^2\right) \tag{4.14}$$

where $t = 1, \ldots, T$ denotes the index of the training samples in the vector $\vec{x_t}$ and p_{uv} is the estimated accuracy of the classifier made from clusters c_{1u} and c_{2v}.

This function to estimate classification accuracy satisfies the following limits: $p_{uv} \to 1$, when $d_{uv} \to 0$ and $p_{uv} \to 0$, when $d_{uv} \to \infty$. It should also be noted that $p_{uv} \in [0.5, 1]$.ψ_{km} is a parameter whose optimal value should be found by optimizing the objective function given in equation (4.15) on the training data given by

$$f\left(\psi_{uv}\right) = \left[\frac{1}{T}\left[\sum_{t=1}^{T} p_{uv}\left(\vec{x_t}\right)\right] - p_j\right]^2 \tag{4.15}$$

where p_j is the training accuracy of jth classifier where $j = 1, \ldots, L$. In order to find an exact solution for the ψ_{uv} parameter by optimizing the objective function given in equation (4.15), it must be monotonically decreasing. It can be shown that

$$\frac{\partial p_{uv}}{\partial \psi_{uv}^2} = -\frac{1}{2}exp\left(\frac{1}{\psi_{uv}^2}\left(log\left(\frac{d_{tu}}{d_{tv}}\right)\right)^2\right)\left(log\left(\frac{d_{tu}}{d_{tv}}\right)\right)^2\left(\psi_{uv}^2\right)^{-2} \leqslant 0. \tag{4.16}$$

Equation (4.16) implies $\frac{\partial \frac{1}{T}\left[\sum_{t=1}^{T} p_{uv}(\vec{x_t})\right]}{\partial \psi_{uv}^2} \leqslant 0$. Therefore an exact solution for the ψ_{uv} parameter can be found by optimizing equation (4.15). After optimal ψ_{uv} parameter is found, the accuracy can be estimated by substituting the ψ_{uv} parameter value in equation (4.14). Next, the weights for the jth classifier can be calculated as, $b_j = \log \frac{p_{uv}(\vec{x_t})}{1 - p_{uv}(\vec{x_t})}$. Figure (4.2) summarizes the steps involved in the Adaptively Weighted Ensemble Classification Method.

Input

- Training data *Train* with correct labels $\omega_i \in \Omega = \{\omega_1, ..., \omega_\Gamma\}$ representing Γ classes.
- Integer K specifying number of clusters.
- Test data *Test*

Output

- Predicted class labels $s \in \Omega$ for *Test*.

Training

1. Cluster training data of each class into K clusters.
2. Train classifiers on clustered training data.

Test- Given a test instance $x \in Test$

1. Evaluate distances d_u and d_v from the cluster centres to sample x.
2. Optimize objective function based on distance ratio to find suitable ψ_{uv} parameter.
3. Calculate weights for each classifier in ensemble.
4. Choose the classifier decision that receives the highest weighted majority vote.

Figure 4.2: Adaptively Weighted Ensemble Classification Method.

The inputs to the algorithm are training data and the number of clusters to partition. Training step consists of clustering and training classifier ensemble. In the testing step a previously unseen instance is presented to the classifier ensemble.

4.4 Results & Discussions

The proposed AWEC method was tested on publicly available BCI Competition dataset 2A [7] and data collected from 12 healthy subjects. For both data sets single-trial EEG data were extracted for training the CSP algorithm. Three pairs of CSP features in the 8-30Hz band-pass filtered EEG measurements, extracted at the time segment of 0.5-2.5s after the onset of the visual cue were used.

The number of component classifiers in the ensemble depends on the number of clusters as too many clusters will result in smaller partitions leading to over fitting and lower generalization accuracies for unseen data. Therefore only two to seven clusters, resulting in four to forty nine individual classifiers respectively, were investigated.

4.4.1 Classification Accuracies

The proposed AWEC method was evaluated on the dataset 2A of BCI Competition IV. Six separate ensembles of classifiers were developed consisting of four to forty nine individual classifiers. Their performances were compared against a single SVM classifier. The empirical results for the dataset 2A of BCI Competition IV are shown in Table (4.1).

The highest classification accuracies for each subject are in boldface. A series of pairwise t-tests were carried out between the baseline results and each of the clustering approaches. It can be seen that the optimal number of clusters yielded a statistically significant improvement over the baseline result (p=0.048). However, the ensemble of classifiers resulting from 3 clusters yielded the best overall classification accuracy (81.5%). A t-test between the ensemble built with 3 clusters and the ensemble built with 7 clusters revealed that the two ensemble classifiers are not statistically different (p=0.93). This could be attributed to over-training of component classifiers and lack of sufficient training data as the sample numbers for training is reduced when more clusters are created.

The results obtained for the data collected from 12 healthy subjects are shown in Table (4.2). The training data was clustered only to 3 clusters based on the previous results. A pairwise t-test was carried out at a confidence level of 0.05 and the increase over the baseline results obtained with a single SVM classifier was found to be statistically significant (p=2.67×10^{-5}).

Table 4.1: Results of BCI Competition Dataset 2A.

The baseline results produced by a single SVM classifier are compared against ensembles created by combining multiple classifiers trained on clustered training data for the BCI Competition IV Dataset 2A. The two sample Student t-test is used to assess the statistical significance of the improvement at a confidence level of 0.05.

		Number of Clusters Training Data is Partitioned					
Subject	Baseline Acc.	2	3	4	5	6	7
A1	87.3	95.2	**95.4**	94.8	94.4	94.8	94.6
A2	56.8	63.8	**64.2**	64.1	62.5	63.9	63.4
A3	93.1	**96.9**	96.8	96.2	96.5	95.2	95.9
A4	63.6	66.7	**67.3**	66.7	66.8	66.4	65.5
A5	54.8	**75.9**	**75.9**	75.6	75.4	75.7	75.6
A6	62.6	64.9	65.2	63.6	**65.8**	63.8	64.5
A7	77.1	78.1	78.1	77.9	78.1	78.5	**78.7**
A8	94.2	96.1	96.1	**96.4**	95.2	95.7	95.6
A9	**93.8**	92.6	93.2	92.8	93.25	92.8	93.2
Mean	75.9	81.3	**81.5**	81.0	80.9	80.8	80.9
Std. Dev.	16.6	14.3	14.2	14.4	14.1	14.1	14.4
p value		0.039	0.032	0.047	0.047	0.059	0.048

Table 4.2: Results of Data Collected from 12 Healthy Subjects.

This Table compares the baseline accuracy given by a single SVM classifier against the ensemble classifier trained on 3 clusters of training data for the data collected from 12 healthy subjects. The two sample Student t-test is used to assess the statistical significance of the improvement at a confidence level of 0.05.

Subject	Baseline Acc.	Acc. from AWEC with 3 Clusters
1	60.7	65.0
2	62.1	65.2
3	52.7	57.5
4	69.4	70.7
5	67.2	69.3
6	82.2	87.9
7	81.1	84.3
8	95.2	97.5
9	73.0	75.0
10	57.2	61.9
11	49.4	56.6
12	82.7	84.7
Mean	69.4	73.0
T test (P value)		2.67×10^{-5}

4.4.2 Addressing Non-stationarity

The presence of non-stationarity in session to session data can be clearly identified by the clustering analysis. Figure (4.3) highlights the presence of non-stationarity in the data set 2A. A classifier trained on the first session will not be able to classify the data from subsequent sessions due to the presence of this non-stationarity.

Figure (4.4) shows two examples that are correctly classified only by the proposed method. Three base classifier hyperplanes are shown in the figure in dashed lines. The classifier L_{11} is trained on cluster 1 of class 1 and cluster 1 of class 2. L_{22} is trained on cluster 2 of class 1 and cluster 2 of class 2 and L_{33} is trained on cluster 3 of class 1 and cluster 3 of class 2. The baseline ensemble without adaptive weighting is also shown as a dashed line. The black dots represent features from the second session. Test sample x_1 belongs to class 1, but it is classified wrongly to class 2 by classifiers L_{22} and L_{33}, however L_{11} classifies it correctly and because the decision of L_{11} is magnified by the weighting method, the effective hyperplane of the ensemble for x_1 shown as $EL1$ correctly classifies the sample x_1 in class 1.

Test sample x_2 also belongs to class 1, but it is incorrectly classified to class 2 by classifiers L_{11} and L_{33}, however L_{22} classifies it correctly and because the decision of L_{22} is magnified by the weighting method, the effective hyperplane of the ensemble for x_2 shown as $EL2$ correctly classifies the sample x_2 in class 1.

A further analysis was carried out on the BCI Competition dataset 2A to ascertain whether the proposed AWEC method is capable of accounting for non-stationarity in EEG data. In this study, a part of the test data was also included in the training data. The hypothesis, that the clustering based classifier ensemble is capable of accounting for non-stationarity when there is more variability in the data was statistically analysed for significance. Table (4.3) summarizes the results of the analysis.

Table 4.3: Comparison of Effects of Including Data from Second Session.

Case 1: Train classifiers on all training data and test on half of test data, Case 2: Train with half of training data and half of test data and test on the other half of test data, Case 3: Train on all training data and test on half of test data, Case 4: Train with half of training data and half of test data and test on the other half of test data.
P1 compares the significance between baseline cases (test 1 and test 3) against the corresponding approaches with ensemble built by 3 clusters (Case 2 and Case 4). P2 statistic compares the case where half of the test samples were included for training without the proposed classifier combination method (Case 2) against the case where classifiers are trained with only the training data and tested on half of test data (Case 3).

	Methods			
	Baseline Without Clustering		AWEC With 3 Clusters	
Subject	Case 1	Case 2	Case 3	Case 4
A1	87.49	90.06	96.17	97.42
A2	56.85	60.24	66.08	68.51
A3	93.25	96.91	97.13	98.47
A4	63.64	64.99	68.72	70.34
A5	55.03	57.09	76.39	78.47
A6	64.75	64.87	68.87	69.17
A7	77.11	78.35	78.82	80.17
A8	94.27	96.34	97.95	98.11
A9	93.92	95.71	95.01	96.59
Mean	76.26	78.51	82.79	84.14
Std Dev	16.47	16.57	13.65	13.41
P1	-	-	0.013	0.031
P2	-	0.068		-

Two baseline cases were considered in the analysis (Case 1 and case 2). In the first case, the classifiers were trained with all the training data similar to the standard procedure and evaluated only on half of the randomly chosen test data. In the second set-up (case 2), half of the test data was randomly selected to be incorporated into the training data and was tested on the other half of test data. Clustering based ensemble was also trained in a similar manner and tested on randomly chosen half of the original test samples.

Two statistical tests were carried out to compare the mean results of this study. First, the baseline cases without ensemble classifiers were compared against the corresponding cases with the ensembles. The probability values of the pairwise t-tests are denoted as P1 in Table (4.3). The tests suggest that the proposed AWEC method results in statistically significant improvements over the respective baseline cases under both settings (P1=0.013 and 0.031).

The second comparison was carried out between the case where half of the test samples were included for training without the proposed classifier combination method against the case where classifier ensemble was trained with only the training data and tested on half of the test data. The test indicates that the mean accuracies resulting from the two cases are not different at a 0.05 level of significance (P2=0.068).

4.4.3 Complexity Analysis

The complexity of the proposed framework depends on the complexities of the main components: CSP algorithm, clustering mechanism, classifier ensemble and optimal weights calculation.

Pre-processing and feature extraction steps depend mostly on the complexity of the CSP algorithm. The CSP algorithm needs to compute covariance matrices, which is in the order $O(N*ch^3)$,where N is the dimensionality of data and ch is the number of components (channels).

The complexity of the clustering algorithm depends on the initialization step and the iterative

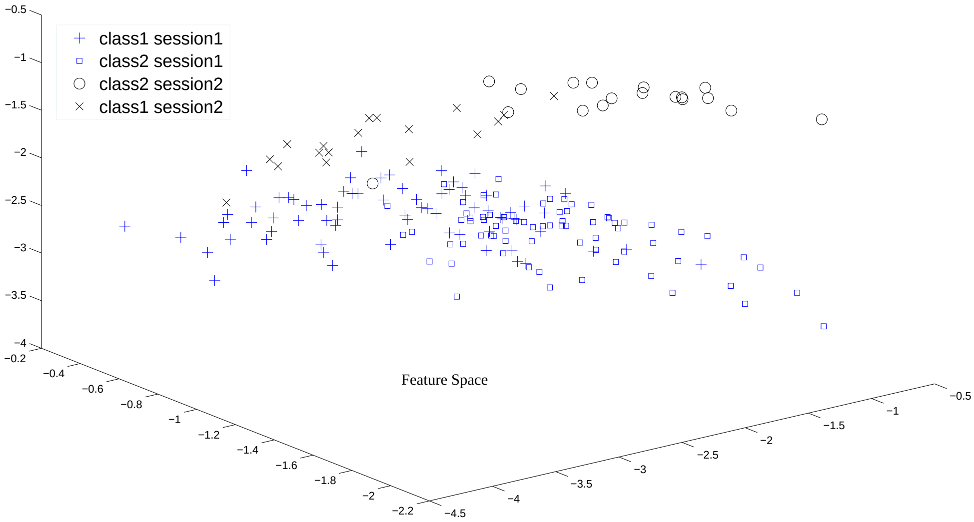


Figure 4.3: Session-to-session Non-stationarity in BCIC IV Data Set 2A Subject A1.

updates. The initialization step costs $O(T^2 * N)$ as the complete kernel matrix needs to be set up. Finding the best target cluster for each datum costs $O(K)$ time and the update procedure costs $O(T)$ time. K is the number of clusters and T is the number of data samples. The cost of the main loop of the algorithm is therefore $O(I * T(K + \mu * T))$ where I is the number of iterations and $0 < \mu < 1$ is the expected ratio of data items that change membership. The number of membership changes is large for the first few iterations, then quickly reduces as the algorithm converges. Overall, the time complexity of clustering is dominated by the quadratic cost of computing the kernel matrix. The maximum number of iterations was set to 50 to increase efficiency.

The complexity of the ensemble depends partly on the number of SVM classifiers and on the SVM classification algorithm. The complexity of one SVM classifier depends on the number of features and support vectors. When a linear kernel is used, the time complexity depends only on

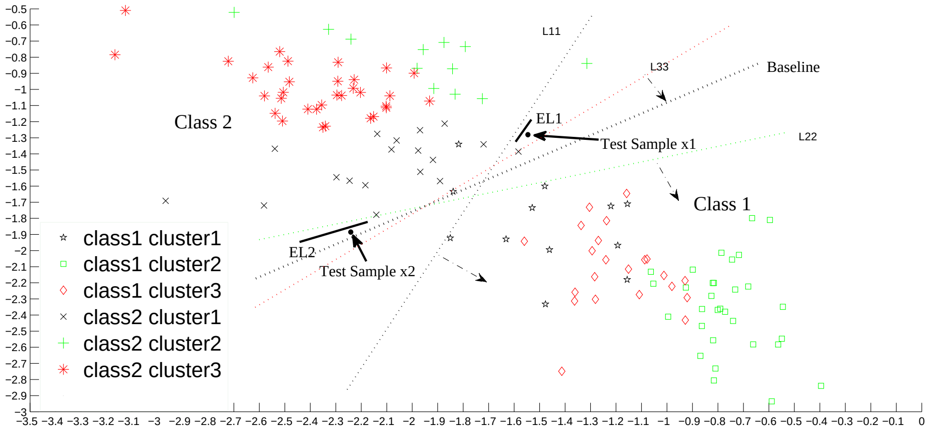


Figure 4.4: Examples of Two Test Samples from in-house dataset subject 3.

3 clusters of each class are combined resulting in 9 classifiers. Only three classifier hyperplanes L11, L22 and L33 are shown in the figure. The baseline classifier hyperplane is also shown in a dashed line. The chosen test samples are shown as black dots are correctly classified by the proposed method but misclassified by other combination methods. The effective hyperplanes, resulting from adaptive weighting, for each of the test samples are shown as solid lines EL1 and EL2. The dashed arrows perpendicular to the classifier hyperplanes indicate the direction of class 1 by each classifier.

the feature dimensionality [84]. Therefore, the complexity for one SVM classifier is in $O(N)$, where N is the dimensionality of data. The complexity of the whole ensemble is $O(N * K^2)$.

The calculation of optimal weights involves $O(K^2)$ distance measures and their optimization. The optimization function is smooth and convex with complexity of $O(K^2)$. Each gradient computation complexity is also $O(K^2)$, so if all of them have to be computed during an iteration that adds $O(K^4)$. if the total number of iterations for the optimization is I the complexity of optimization adds upto $O(I * K^4)$.

4.5 Conclusion

In this chapter, we proposed a novel method to partition EEG data using clustering, and multiple classifiers were trained using the partitioned datasets. The final decision of the classifier ensemble was then obtained by weighting the classification decisions of individual classifiers. A combination method based on the distances from the test sample to the constituent cluster centres that form the specific classifier was subsequently used to weigh the classifier decisions. The proposed AWEC method was applied on publicly available dataset 2A from BCI Competition IV and data set collected from 12 healthy subjects. Classification accuracies obtained showed that the proposed method yielded statistically significant improvements. The analysis carried out in section 4.4.2 showed that the proposed AWEC approach can be used to address non-stationarity in the EEG data.

Chapter 5

Error Entropy Based Kernel Adaptation for Adaptive Classifier Training

5.1 Introduction

Brain-Computer Interfaces (BCIs) are communication systems that enable subjects to send commands to computers using only their brain activity [121]. Non-stationarity arising from high variability of EEG signals is a major obstacle in EEG-based BCI systems. Non-stationarity has been found to be linked to various factors such as, changes in the physical properties of the sensors, variability in neurophysiological conditions, psychological parameters, ambient noise and motion artifacts [131, 132, 134, 244].

The importance of addressing session to session non-stationarity has been widely recognized in the BCI community. Various signal processing and learning methods such as, Bayesian transduction, active learning and distribution matching have been proposed [106, 131, 133, 134]. Stationary Subspace Analysis (SSA) [134] is another unsupervised learning method that finds subspaces in which data distributions stay invariant over time. Current research addressing non-stationarity also includes methods that adapt the classifiers using the knowledge from empirical data [15, 245, 246]. These methods include adaptation of LDA and SVM classifiers which are the commonly used classification methods in BCI [74]. Adaption of LDA involves updating the statistical parameters such as mean, covariance and bias [15]. Adaptive SVM methods include least square based methods with various penalty functions [245, 246].

All these adaptive methods use minimization of error, based on the classification output to optimize some parameter in the classifiers [15, 226, 245, 246]. In this type of adaptations, the error is under the control of the parameters of the adaptive system because the error depends on the true labels which is a function of the parameters that are adapted. Error entropy criterion takes into account the amount of information in the error distributions. Therefore, minimization of error entropy considers the error distributions rather than error values. Error entropy based adaptive systems have been applied in designing adaptive filters [215, 247, 248]. However, the use of the error entropy for the adaptation of kernel classifiers has not been attempted. In this work we propose to use the error entropy to adapt the width of the Gaussian kernel of the SVM classifier. A subset of data from the later session is used as adaptation data to estimate the error entropy based cost function which is minimized by adapting the kernel width. Positive results were obtained for the proposed method on motor imagery EEG data collected on different days.

5.2 Materials

Two datasets were evaluated using the proposed method. Publicly available BCI Competition IV dataset 2A [142] and motor imagery dataset collected in-house from 12 healthy subjects.

The BCI Competition IV Dataset 2A is comprised of EEG data collected from 9 subjects that were recorded during two sessions on separate days for each subject. The data has been collected on four different motor imagery tasks: left hand (class 1), right hand (class 2), both feet (class

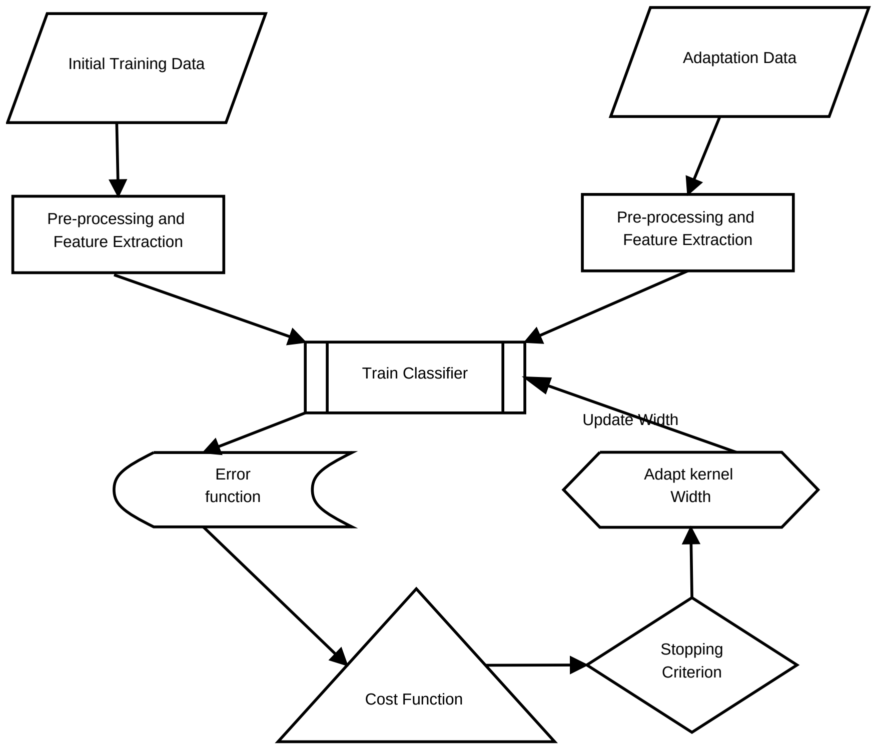


Figure 5.1: Block Diagram of Proposed Method

3), and tongue (class 4). Each session is comprised of 6 runs separated by short breaks, each run comprised 48 trials (12 for each class), amounting to a total of 288 trials per session. Only the two class classification between left hand and right hand motor imagery was considered for this study. The data from the first session were used as training data for learning CSP spatial filters and the initial classifier, and the first half of data from the later session was used as adaptation data. The second half of motor imagery data from the later session was used as the test data. For more details on the protocol please refer to [142].

The in-house motor imagery data used for the analysis were collected using a Nuamps EEG acquisition hardware (http://www.neuroscan.com) with unipolar Ag/AgCl electrodes, digitally sampled at 250 Hz with a resolution of 22 bits for voltage ranges of 130mV. EEG signals from 22 scalp positions, mainly covering the primary motor cortices bilaterally were recorded. The sensitivity of the amplifier has been set to 100μV.

A total of 12 healthy subjects were recruited for the study. Ethics approval and informed consent were obtained. Two subjects chose to perform left hand motor imagery while the remaining 10 subjects chose to perform on the right hand. The subjects were instructed, in the form of visual cues displayed on the computer screen, to perform kinaesthetic motor imagery of the chosen hand, and rest during the background rest condition.

EEG data were collected without feedback in two sessions from each subject on separate days. In the first session, two runs of EEG data were collected from a subject while performing motor imagery of the chosen hand and background rest condition. In the second session, three runs of EEG data were collected on another day while performing motor imagery of the chosen hand and background rest condition. Each run lasted approximately 16 minutes that comprised 40 trials of motor imagery and 40 trials of rest condition. The motor imagery data collected during first session were used as training data for learning CSP spatial filters and the initial classifier, and first half of motor imagery data from the later session was used as adaptation data. The second half of motor imagery data from the later session was used as test data.

5.3 Methods

The data from the initial session was used first to generate an initial model for the classifier after the basic preprocessing steps of bandpass and spectral filtering. Adaptation data from the subsequent session was used to optimize the kernel width parameter.

Figure (5.1) summarizes the proposed method. The pseudo-code of the proposed method is shown in Figure (5.2) for further clarification. The initial training data from the first session and the adaptation data from later session were subjected to pre-processing steps of bandpass filtering at 8-30Hz. Initial training data were spatially filtered by the Common Spatial Patterns (CSP) method [138, 210]. Adaptation data on the other hand, used the CSP projection matrix

Inputs

- Initial Training data *Train* with correct labels
- Adaptation data *Adapt* with correct labels

Output

- Classification model.

Algorithm

1. Band-pass filter Initial Training data
2. Spatial filter Initial training data
3. Extract initial features
4. Train initial classification model
5. Band-pass filter Adaptation data
6. Extract features from Adaptation data
7. Feed features to classifier
8. Calculate error entropy
9. Calculate cost function value
10. Adapt the Kernel width of Kernel
11. Repeat steps 7:10 until all adaptation data are used

Figure 5.2: Pseudo-code of the proposed method.

created on the initial data.

The initial classifier model was trained only on the training data from the first session. The adaptation data was used to iteratively update the classifier kernel based on the error function. The error function indicates the error margin of the SVM classifier.

The KL divergence based cost function measures the difference in the estimated error and the actual error. We studied the effect of adaptively training the classifier on the adaptation data from the second session by optimizing the kernel width of the parameter to minimize the KL divergence based cost function.

5.3.1 Error Entropy Criterion

The goal of adaptation using error entropy criterion (EEC) is to remove as much uncertainty as possible from the error signal [247]. This can be achieved by calculating the entropy of the error and minimizing it with respect to the free parameters. The error entropy minimization can be achieved using information theoretic estimators. Principe et al. [215] developed estimators of information theoretic quantities based on Information Potential (IP), which is the mean of the probability density function of data and happens to be the integrand of Renyi's quadratic entropy. Renyi's quadratic entropy of the error is defined as $H_2(e) = -logV(e)$, where $V(e) = E[p(e)]$ is the expected error. Hence, Renyi's quadratic entropy is a monotonic function of the negative of $V(e)$. The logarithm is dropped as it does not change the location of the stationary point of the cost function for optimization. The minimization of entropy corresponds to maximization of $V(e)$. An efficient method to maximize $V(e)$ is to use estimators of information theoretic quantities. Minimizing the Kullback-Leibler divergence (KL) between the true and estimated probability distribution functions of error, denoted $f(e)$ and $\hat{f}_\sigma(e)$, as a function of the kernel width σ [248].

5.3.2 Minimizing Kullback−Leibler Divergence for Kernel Width Adaptation

The estimators of information theoretic quantities like entropy are based on Parzen kernels. Therefore, a kernel needs to be selected to estimate the pairwise interactions between samples. In this criterion, kernel width controls the smoothing introduced by a kernel function used for non-parametric estimation of the probability density function from samples, as in Parzen windows [213]. The kernel width is considered as a parameter that can be adapted in a way that the discriminant information or the Kullback-Leibler loss between the estimated density (using the kernel) and the true density is minimized. In other words, the kernel width is adapted with its own cost function in a way that the estimated error distribution resembles the true error distribution as closely as possible, based on Kullback-Leibler divergence.

Singh et al [248] proposed to minimize the KL divergence between the true and estimated probability distribution functions of error, denoted $f(e)$ and $\hat{f}(e)$, as a function of the kernel width σ. The objective is to minimize

$$D_{KL}\left(f \parallel \hat{f}_\sigma\right) = \int f(e)\, log\left(\frac{f(e)}{(\hat{f}_\sigma(e))}\right) de, \tag{5.1}$$

where the subscript σ denotes the dependency of estimated probability distribution function $\hat{f}_\sigma$ on the kernel width. The equation (5.1) can be re-written as,

$$\begin{aligned} & D_{KL}(f \parallel \hat{f}_\sigma) \\ & = \int f(e) log(f(e)) de - \int log(\hat{f}_\sigma(e)) f(e) de \\ & = \int f(e) log(f(e)) de - E[log(\hat{f}_\sigma(e))], \end{aligned} \tag{5.2}$$

where $E[\cdot]$ is the expectation operator over the true distribution of errors e. The first term in equation 5.2 is independent of the kernel width. Therefore, minimizing $D_{KL}(f \parallel \hat{f}_\sigma)$ with respect to σ is equivalent to maximizing the second term $E[log(\hat{f}_\sigma(x))]$. Which can be interpreted as the cross-entropy of the estimated probability distribution function, and the true probability distribution function. Using the sample estimator for the expectation operator for a Gaussian Kernel the objective function becomes

$$\hat{J}_{KL}(\sigma) = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{1}{(N-1)} \sum_{j=1, j \neq i}^{N} \left(G_\sigma(e_i - e_j) \right) \right), \tag{5.3}$$

where N is the window of samples used to estimate density of the error, for a Gaussian kernel with width σ. Taking the derivative of objective function in equation 5.3 with respect to kernel width σ results in,

$$\left(\frac{\partial J_{KL}(\sigma)}{\partial\sigma}\right)$$
$$= E\left[\frac{(\frac{\partial\hat{f}_\sigma(e)}{\partial\sigma})}{\hat{f}_\sigma(e)}\right]$$
$$= E\left[\frac{\sum_{i=n-L}^{n-1}\exp\left(-\frac{(e-e_i)^2}{2\sigma^2}\right)\left(\frac{(e-e_i)^2}{\sigma^3-\frac{1}{\sigma}}\right)}{\sum_{i=n-L}^{n-1}\exp\left(-\frac{(e-e_i)^2}{2\sigma^2}\right)}\right]. \tag{5.4}$$

Using the equation (5.4) the update rule for kernel size can be formulated as,

$$\sigma_{n+1}$$
$$= \sigma_n + \frac{\gamma\,(\partial J_{KL}(\sigma))}{\partial\sigma}$$
$$= \sigma_n + \gamma E\left[\frac{\sum_{i=n-L}^{n-1}\exp\left(-\frac{(e-e_i)^2}{2\sigma^2}\right)\left(\frac{(e-e_i)^2}{\sigma^3-\frac{1}{\sigma}}\right)}{\sum_{i=n-L}^{n-1}\exp\left(-\frac{(e-e_i)^2}{2\sigma^2}\right)}\right].$$

By evaluating the operand at the current sample of the error while dropping the expectation operator results in an approximation of the gradient which can be used as an efficient update rule,

$$\sigma_{n+1} = \sigma_n + \gamma E\left[\frac{\sum_{i=n-L}^{n-1}\exp\left(-\frac{(e_n-e_i)^2}{2[\sigma_n]^2}\right)\left(\frac{(e_n-e_i)^2}{[\sigma_n]^3} - \frac{1}{\sigma_n}\right)}{\sum_{i=n-L}^{n-1}\exp\left(-\frac{(e_n-e_i)^2}{(2[\sigma_n]^2)}\right)}\right]. \tag{5.5}$$

The update rule in equation (5.5) is iteratively applied until all adaptation samples are considered. The updated kernel is applied for classification of test samples.

5.4 Results & Discussions

The results obtained for the data collected from 12 healthy subjects are shown in Table (5.1). The twelve subjects are denoted as A1 to A12. The mean accuracies and standard deviations calculated for all the subjects are denoted as mean and S.D. in the table (5.1). The baseline classification used an SVM classifier with a static Kernel. Half of the training data and the adaptation data for was used for training the classifier. In the proposed method, half of the data

Table 5.1: Comparative Classification Accuracy on the Data Collected from 12 Healthy Subjects.

P-value denotes the result of pairwise t-test against the baseline.

Subject	Baseline	Proposed Method	Increment
A1	60.5	68.3	7.8
A2	58.3	67.5	9.1
A3	51.1	55.9	4.7
A4	63.9	79.4	15.5
A5	64.2	74.3	10.1
A6	83.3	88.7	5.4
A7	79.4	79.4	4.5
A8	93.6	93.6	0.0
A9	65.5	79.6	14.1
A10	54.7	61.9	7.1
A11	50.5	65.9	15.3
A12	79.7	85.0	5.3
Statistics			
Mean	67.07	75.0	
S.D.	13.82	11.33	
P		0.00029	

collected during the first session and the adaptation data were used to train the classifiers. The second half of motor imagery data from the later session were used as test data.

The observed mean baseline accuracy was 67%. The baseline result was compared against the results obtained using the proposed Kernel width adaptation method. Pairwise t-test was carried out between the baseline results and the proposed method. The mean accuracies from the proposed Kernel width adaptation method were found to be significantly higher than the baseline at a confidence level of 0.05.

The increments made by the proposed adaptive method over the baseline are shown in the fourth column of Table (5.1). Only one subject did not show any improvement in accuracy. All the other subjects showed substantial increments in accuracy.

The results obtained for the BCI Competition IV data set 2A are shown in Table (5.2). The mean accuracies and standard deviations calculated for all the subjects are denoted as mean and S.D. in the table (5.2). The baseline classification used an SVM classifier with a static Kernel and used half of the training data and the adaptation data for training the classifier. In the proposed method half of the data from the first session and the adaptation data were used to train the classifiers. The second half of data from the later session was used as test data. The results were compared with the results obtained for the AWEC method proposed in chapter 4 (see table (4.3) for details). The AWEC method with 3 clusters also used half of the training data from session 1 and the adaptation data for training a classifier ensemble. The second half of data from the second session was used as the test data. The proposed method was compared against the baseline as well as against the AWEC method in table (5.2).

The observed mean baseline accuracy was 78.51%. The baseline result was compared against the results obtained using the proposed Kernel width adaptation method and that from the AWEC method with 3 clusters. Pairwise t-tests were carried out between the baseline results and the proposed methods. The mean accuracies from the proposed Kernel width adaptation method were found to be significantly higher than the baseline at a confidence level of 0.05. However, the AWEC method with 3 clusters produced much higher overall mean accuracy of 84.14%, compared to the 82.73% resulting from the proposed Kernel width adaptation method. AWEC based ensemble classifier showed increments over the baseline in all nine subjects, whereas the proposed Kernel adaptation method showed increments only in 6 subjects.

5.5 Conclusion

In this study, a novel algorithm to adapt the Kernel width parameter of SVM classifier to improve classification of non-stationary EEG data was proposed. In the proposed algorithm, the width parameter of the Kernel of the classifier was iteratively adapted based on Information

Table 5.2: Comparative Classification Accuracy on the BCI Competition Data Set 2A

P-value denotes the result of pairwise t-test against the baseline.

Subject	Baseline	AWEC with 3 clusters	Proposed Method	Increment over Baseline
1	90.06	97.42	92.87	2.81
2	60.24	68.51	69.35	9.11
3	96.91	98.47	96.91	0.0
4	64.99	70.34	74.28	9.29
5	57.09	78.47	64.47	7.38
6	64.87	69.17	72.62	7.75
7	78.35	80.17	82.04	3.69
8	96.34	98.11	96.34	0.0
9	95.71	96.59	95.71	0.0
Statistics				
Mean	78.51	84.14	82.73	
S.D.	16.57	13.41	12.96	
P		0.031	0.011	

theoretic cost function to minimize the KL divergence between the estimated and the actual error distributions.

The proposed method was applied on publicly available BCI Competition dataset 2A and data collected without feedback from 12 healthy subjects in two sessions on separate days. The results using the proposed method yielded statistically significant improvements in classification accuracies on non-stationary EEG data across sessions compared to the baseline without kernel adaptation.

Future work based on this approach would include adaptation of Kernel mean and other parameters to optimize the adaptation.

Chapter 6

Learning from Feedback Training Data in Self-paced BCI

6.1 Introduction

Inherent changes in brain signals pose a critical challenge to EEG-based brain-computer interface (BCI) research [1, 56, 218], and has recently attracted a surge of attention in the field [11, 15, 125, 126, 219–223]. There has been a lot of interest in motor imagery (MI) based BCI [56, 136, 224] which are driven by the imagination or mental rehearsal of a motor action without any real motor output.

The underlying non-stationarity of EEG signals cause the distribution of electrical fields on the scalp to large variations over time. This non-stationarity, as outlined in chapter 2, can be caused by shifts in background brain activities, varying mental states, and individual users changing their strategy for BCI control [220]. In feedback BCI applications this is further complicated by activation of additional brain functions. Complex EEG phenomena such as error potentials [225] and rhythmic power shifts over the scalp [11] have been observed in some feedback BCI studies.

The feature extraction and prediction models (e.g. a classifier) built on data from past BCI sessions may become ineffective as a result of non-stationarity. Therefore, there is a strong need for new mathematical models capable of accurately predicting a user's intentions from his/her brain signals in session to session transfer. Adaptive BCI that can learn from new data, in supervised, semi-supervised or unsupervised manner is a viable approach to solve this problem.

Most research on adaptive BCI have been focused on adapting the classifiers. Three supervised adaptation methods using labelled data has been investigated in [11]. These included a simple bias adjustment technique, a linear discriminant analysis (LDA) retraining technique, and a technique which retrains both LDA and common spatial pattern (CSP)-based feature extraction [210]. It has been reported that LDA-retraining approach has yielded the lowest error rate. A covariance shift algorithm has been introduced for unsupervised adaptation of a linear classifier in [12]. Li et al. [226], have combined a method for adaptation with a bagging approach which has resulted in improved stability. Different adaptation methods have been extensively studied using multiple BCI data sets in [15]. In these studies, bias adjustment methods have been more promising than the generic covariance shift adaptation methods.

Online adaptation of Quadratic Discriminative Analysis (QDA) classifier after each trial in a cue-based BCI setting has been presented in [125]. It has been demonstrated that the distribution of EEG features significantly shift from one session to another. Further studies using adaptive autoregressive features, band powers, and the combination of the two have been reported in [126]. In [221], a classifier with band power features as input has been updated continuously, where only non-feedback (i.e. calibration) sessions have been used for offline study.

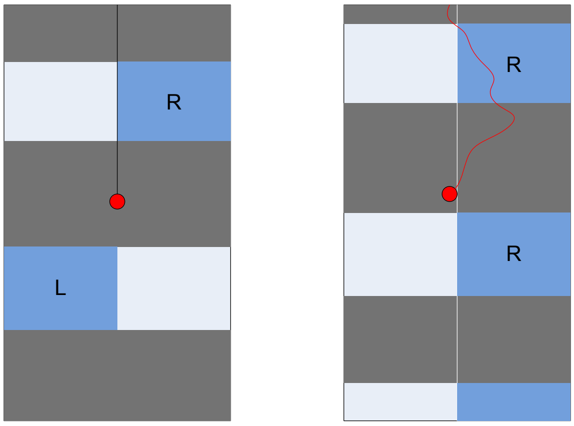
However, little work has been carried out on adaptation of feature extraction models for exploring feedback training data including idle state. There is evidence that the non-stationarity may not be solved by adapting classifiers alone as indicated by experimental results in [125] and [15]. Significant changes in brain signals, from calibration to feedback training sessions can make the feature space derived from calibration data ineffective, where little discriminative information can be extracted.

The primary purpose of this work was to validate the feasibility and the importance of adapting feature extraction models, especially for self-paced MI BCI that allows continuous feedback control [61,227–230,234]. Adapting feature extraction models has been found to be challenging according to the unsatisfactory performance of retrained CSP models in [11].

A new self-paced BCI with idle class was developed and the performance of calibration and feedback training was tested on three able-bodied, naïve subjects. The empirical results demonstrated the limitations of applying the feature space derived from calibration data to feedback sessions. Hence, a novel supervised method that learns from feedback sessions to construct a more appropriate feature space was proposed. Particularly, the method attempts to account for the underlying complex relationships between feedback signal, target signal and EEG, using a mutual information formulation. The learning objective was formulated as maximizing kernel-based mutual information estimation with respect to the spatial-spectral filters. A gradient-based optimization algorithm was derived to solve the learning task.

An experimental study was conducted using offline simulations. The results indicate that the proposed method is capable of constructing effective feature spaces that capture more discriminative information in the feedback sessions. Consequently, the classification accuracies can also be significantly increased by using the new features.

The rest of the chapter is organized as follows. Section 6.2 describes the data collection with a self-paced BCI, and the results of online training. Section 6.3 elaborates the new method for learning effective spatial and spectral features from feedback session data. Section 6.4 presents an extensive analysis, followed by discussions in Section 6.5. Section 6.6 finally concludes the chapter.



Calibration *Feedback Training*

Figure 6.1: The Graphical User Interface for Calibration and Feed-back

GUI on left panel is for calibration and right panel for self-paced feedback training. The grey and blue color block scrolls smoothly upwards in the background, and the red circle in the center serves as the eye-fixation point. During feedback training, the horizontal position of the red circle serves as the feedback signal that updates every 40 milliseconds, while its trajectory over the background blocks is depicted by a red curve.

6.2 Materials

6.2.1 Feedback training data collection

Three BCI-naïve adults were recruited as subjects for the data collection. Informed consent was obtained according to criteria approved by the Institutional Review Board of the National University of Singapore. The subjects were seated comfortably in an armed chair, with their hands rested on the chair arms or on the table in front of them. A 20-inch widescreen LCD monitor was placed on the table at a distance of approximately 1 meter from the subject. Subjects were asked to remain comfortably but motionless to minimize motion artifacts.

EEG was recorded using Neuroscan NuAmps 40-channel data acquisition system, with electrodes placed according to an extended international 10-20 system and a sampling frequency of 500Hz. A total of 30 channels were used, including F7, F3, Fz, F4, F8, FT7, FC3, FC4, FT8, T7, C3, Cz, C4, T8, TP7, CP3, CPz, CP4, TP8, P7, P3, Pz, P4, P8, O1, Oz, O2, PO1, PO2. The reference electrode was attached to the right ear. A high-pass filter at 0.05Hz was applied in the

Neuroscan's data acquisition setting.

The subjects faced a graphic user interface displayed on the LCD monitor as illustrated in Figure (6.1), which guided them through the following sessions.

- **Calibration session**. The calibration session consisted of 40 MI tasks; each was 4-seconds long and followed by a 6-second idle state. The MI tasks were evenly and pseudo-randomly distributed into left and right hand MI tasks. A graphical user interface, as illustrated in the left panel of figure (6.1), guided the subjects through the session. The red circle in the middle served as the eye fixation point. In the background, a sequence of rectangular shapes were scrolling upwards, representing left/right hand MI tasks by blue color boxes on the left/right side, and idle state tasks by grey bars. When the red circle was in a grey-color bar, the subject should relax while minimizing physical movements; when a blue-color box was on the left/right side of the red circle the subject was supposed to imagine left/right hand movement.

  The filter-bank CSP (FBCSP) [104, 231, 232] method was employed to build subject-specific MI detection models. The method learnt two separate models from the calibration data. One model was for differentiating between left-hand MI and idle state (hereafter referred to as L-model), and the other model for differentiating between right-hand MI versus idle state (hereafter R-model). For the L-model (or the R-model), each 2.5-seconds long shift window of EEG with a step of 0.5 seconds was mapped to the label of the data: 0 if the time window ends in an idle state time period, 1 (or -1) if in a left-hand (or right hand) MI period. The mapping parameters were obtained using the linear least-mean-square method.

  Since a user's mental state could be uncertain and varying during the transition period from one state to another, a grey region was defined as [-1 1] seconds with respect to the boundary of each idle/MI task. All EEG segments with centers in this grey region were excluded from FBCSP learning.

- **Feedback training sessions**. After calibration, each subject participated in 4 sessions of feedback training, i.e. 2 sessions of left-hand MI BCI training using the L-model and 2 sessions of right-hand MI training using the R-model. This arrangement allowed a subject to concentrate on one particular MI task in each session. A training session consisted of 20 MI tasks, each lasting 5-seconds, followed by a 6-seconds idle state. A graphical user interface, as illustrated on the right panel of Figure (6.1) guided the user through the session. The indications of the symbols were similar to that for calibration, except that the red circle was moving horizontally as a feedback signal. The horizontal position of the red circle was determined by the output of FBCSP output updated every 40-milliseconds.
  During the feedback training sessions, the subjects attempted to move the red circle to the left/right side as much as possible during left-hand/right-hand MI tasks. The subjects were requested not to voluntarily control the feedback signal by any means during periods of idle state as it would spoil EEG data corresponding to the idle state.

Short breaks were taken in-between the feedback training sessions. The first feedback training session started within 5 minutes after finishing the calibration session. The intervals between consecutive feedback sessions were limited to 1 to 5 minutes. Special try-out sessions were carried out right after the calibration, where each subject tried online feedback for a short while in order to familiarize with the feedback signals and also to prepare for the actual training sessions. These try-out sessions were not included in the analysis.

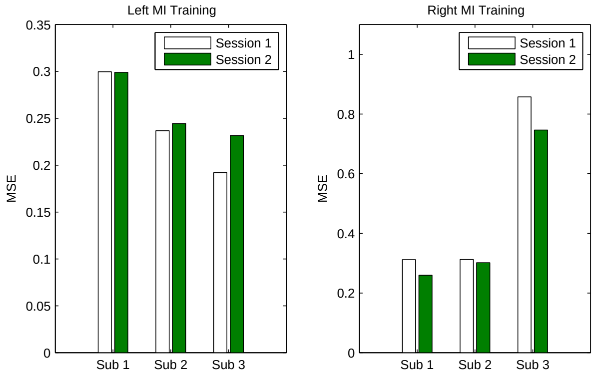


Figure 6.2: Online performance of subjects in terms of mean square error between feedback signal and target.

There is a strong bias shift (from calibration to feedback) in right motor imagery (MI) sessions in Subject 3, which explains his particularly large error.

6.2.2 Data screening

The EEG data recorded during feedback training sessions were inspected visually using MATLAB. Any EEG segments identified to contain EOG and EMG contaminations [233] were rejected and excluded from the analysis. The grey regions were defined in a similar manner as in the calibration sessions described above. Therefore, all EEG segments that were within [-1 1] seconds window with respect to any MI task boundary were excluded from the analysis.

6.2.3 Online performance and initial data analysis

Online performance was assessed using the mean-square-error (MSE) measure between the feedback signal and the target signal. Figure (6.2) shows a bar graph of MSE in each feedback training session. The errors were apparently not significantly different between the first training session and the second session in most cases. This actually indicates that online feedback training in BCI can be a difficult task, since it was expected that the subjects would have gained better control of the BCI over training sessions. This further substantiates the necessity of adapting models during session to session transfers.

The distribution of EEG feature vector samples produced by FBCSP are shown in Figure (6.3), to further understand the feedback training data. Evenly re-sampled feature vector samples were used for clarity, because the original samples amount to thousands. The MI class samples and the idle class samples were easily separable in the calibration data as anticipated. However, the discriminative information had disappeared in the same feature space in most of feedback training sessions. This indicates that, either there was no effective separation between the two classes, or the separation hyper-plane was severely altered (similar to some cases in [15, 125]).

Therefore, it was decided to first investigate the issue of ineffective feature space before trying to adapt a classifier/regressor. A novel method to learn an effective feature space from feedback data was proposed to address this issue. It should also be noted that compared to the calibration data, online feedback training data poses more challenges to effective feature extraction, because the feedback can involve more brain functions and produce more complex EEG phenomena [11, 225].

6.3 The New Learning Method

6.3.1 Spatio-Spectral Features

The primary phenomenon of MI EEG is event-related desynchronization(ERD) or event-related synchronization(ERS) [56, 136], where the rhythmic activity over the sensorimotor cortex, generally in the μ (8-14 Hz) and β (14-30 Hz) rhythms either attenuates or increases, respectively . The ERD/ERS can be induced by both imagined movements in healthy people or intended movements in paralyzed patients [14, 194, 234].

Feature extraction of ERD/ERS is a challenging task due to its poor signal to noise ratio. Therefore, spatial filtering in conjunction with frequency selection (via processing in either tem-

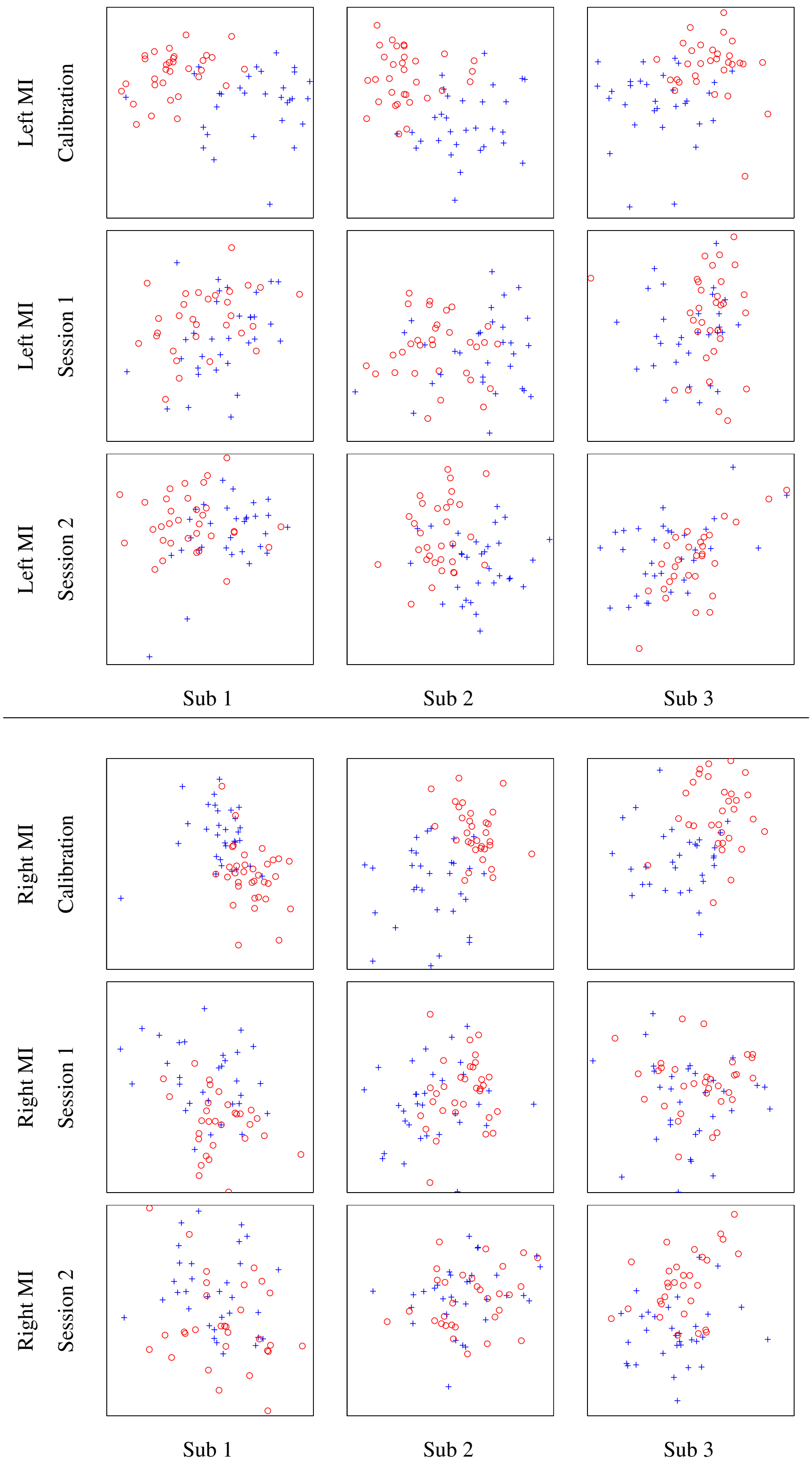


Figure 6.3: Feature distributions during motor imagery (MI) calibration and feedback training sessions

Left MI in the upper three rows and right MI in the lower three rows. The horizontal axis and the vertical axis are the first and the second FBCSP features. Red circles represent motor imagery samples, while black crosses denote idle state samples.

poral domain or spectral domain) in multi-channel EEG is essential for increasing the signal to noise ratio [7, 123, 124, 210, 232].

The spatial-spectral filtering in the spectral domain, for a n_c-channel EEG segment with a sampling rate of F_s-Hz can be described by an $n_c \times n_f$ matrix

$$\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{1n_f} \\ \vdots & \ddots & \vdots \\ x_{n_c1} & \cdots & x_{n_cn_f} \end{bmatrix}, \tag{6.1}$$

where x_{ij} denotes the discrete Fourier transform of the i-th channel at frequency $\omega_j = \frac{j-1}{2n_f}F_s$.

A joint spatial-spectral filter on $\mathbf{X}$ can be essentially represented by a spatial filtering vector $\mathbf{w} \in \mathbb{R}^{n_c \times 1}$ and a spectral filter vector $\mathbf{f} \in \mathbb{R}^{n_f \times 1}$. The feature y_0 is the energy of the EEG segment after filtering:

$$y_0 = \text{diag}\left\{\widetilde{\mathbf{w}^T\mathbf{X}}\left(\mathbf{w}^T\mathbf{X}\right)\right\}\mathbf{f}, \tag{6.2}$$

where the wave line $\widetilde{}$ on the right side of the equation denotes the conjugate of a complex value, and the diag() function stands for the diagonal vector of a matrix.

A general case in which multiple spatial filters are associated with one particular spectral filter was considered in this study. Therefore, the feature extraction model was determined by the matrix $\mathbf{f}$ and a vector $\mathbf{W}$, the latter being the collection of spatial filters in columns:

$$\mathbf{W} = [\mathbf{w}_1 \ \ldots \ \mathbf{w}_{n_w}] \tag{6.3}$$

If the spectral filters in $\mathbf{F}$ are given (see the last paragraph of Section. 6.3.3 for details), the following shorthand notation for the auto-correlation matrix of EEG processed by the k-th spectral filter

$$\hat{\mathbf{X}}_k = \sum_{i}^{n_f} f_i \mathbf{X}\widetilde{\mathbf{X}}, \tag{6.4}$$

can be used. The logarithmic feature vector can be represented as,

$$\mathbf{y} = \left[\log(\mathbf{w}_1\hat{\mathbf{X}}_1\mathbf{w}_1^T), \ldots, \log(\mathbf{w}_{n_w}\hat{\mathbf{X}}_{n_w}\mathbf{w}_{n_w}^T)\right]^T. \tag{6.5}$$

6.3.2 Formulation of the objective function for learning

A mutual information based objective function for learning **W** and **F** was formulated to capture the underlying complex structure of spatio-spectral data in ERD/ERS. Mutual information [235], which stemmed from information theory, basically measures the reduction of uncertainty about class labels due to the knowledge of the features [236–241].

A mutual information measure $\hat{I}$ between the class labels and the EEG features as well as the feedback signal was considered for feedback training data. The mutual information is measured between the class label (i.e. the variable to be predicted) and the observations including both the feedback signal and the EEG feature vector. Let the random variables of the label, the EEG feature vector, and the feedback signal be C, $\mathcal{Y}$ and $\mathcal{Z}$, respectively. The mutual information measure can be expressed as

$$\hat{I}(\{\mathcal{Y}, \mathcal{Z}\}, C) = \hat{H}(\mathcal{Y}, \mathcal{Z}) - \sum_{c} P(c)\hat{H}(\mathcal{Y}, \mathcal{Z}|c), \tag{6.6}$$

where $\hat{H}$ denotes the entropy measure of a random variable.

A non-parametric approach for mutual information estimation was employed as in [239,241], since it does not rely on the underlying distributions. Suppose the feedback training data was comprised of l samples of EEG to be represented by the feature vectors $\mathbf{y}_i$s and the concurrent feedback signal z_is ($i \in [1, \dots, l]$). The non-parametric approach computes each entropy in Eq. 6.6 separately, e.g. $\hat{H}(\mathcal{Y}, \mathcal{Z})$ by

$$\hat{H}(\mathcal{Y}, \mathcal{Z}) = -\frac{1}{l} \sum_{i=1}^{l} \log \left\{ \frac{1}{l} \sum_{j=1}^{l} \varphi_y(\mathbf{y}_i, \mathbf{y}_j)\varphi_z(z_i, z_j) \right\}, \tag{6.7}$$

where φ_y and φ_z are kernel functions and usually take a Gaussian form. For example,

$$\varphi(\mathbf{y}, \mathbf{y}_i) = \alpha \exp\left(-\frac{1}{2}(\mathbf{y} - \mathbf{y}_i)^T \mathbf{\Psi}^{-1} (\mathbf{y} - \mathbf{y}_i) \right). \tag{6.8}$$

The coefficient α is discarded hereafter because it is cancelled out when Eq. 6.8 is substituted in Eq. 6.7 and then substituted in Eq. 6.6. It should also be noted that the kernel size matrix $\mathbf{\Psi}$ is diagonal, and each diagonal element is determined by

$$\psi_{k,k} = \zeta \frac{1}{l-1} \sum_{i=1}^{l} (y_{ik} - \bar{y_k})^2 . \tag{6.9}$$

where $\bar{\mathbf{y}}_k$ is the empirical mean of $\mathbf{y}_k$, and we set the coefficient $\zeta = \left(\frac{4}{3l}\right)^{0.1}$ according to the normal optimal smoothing strategy [242].

The conditional entropy $\hat{H}(\mathcal{Y}|c)$ in Eq. 6.6 can also be estimated similar to Eq. 6.7, but using samples from class-c only. Using the maximum mutual information principle [236], the learning task can be formulated as searching for the optimum spatial and spectral filters $\mathbf{W}$ and $\mathbf{F}$ that satisfies

$$\{\mathbf{W}, \mathbf{F}\}_{\text{opt}} = \underset{\{\mathbf{W},\mathbf{F}\}}{\operatorname{argmax}} \hat{I}(\{\mathcal{Y}, \mathcal{Z}\}, C). \tag{6.10}$$

The above formulation describes the inter-dependency between the target signal, the feedback signal and the EEG signal as a function over the feature extraction parameters in spatial-spectral filters. It basically aims to maximize the information about the target signal to be predicted, contained in the extracted features in conjunction with feedback.

6.3.3 Gradient-based solution to the learning problem

A numerical solution to Eq. 6.10 was proposed by devising a gradient-based optimization algorithm. A spatial filter vector $\mathbf{w}_k$ was considered, where the gradient of the objective function $\hat{I}$ with respect to $\mathbf{w}_k$ is

$$\nabla_{\mathbf{w}_k} \hat{I}(\{\mathcal{Y}, \mathcal{Z}\}, C) = \nabla_{\mathbf{w}_k} \hat{H}(\mathcal{Y}, \mathcal{Z}) - \sum_{c \in C} P(c) \nabla_{\mathbf{w}_k} \hat{H}(\mathcal{Y}, \mathcal{Z}|c). \tag{6.11}$$

Using Eq. 6.7, this can be simplified to

$$\nabla_{\mathbf{w}_k} \hat{H}(\mathcal{Y}, \mathcal{Z}) = -\frac{1}{l} \sum_{i=1}^{l} \beta_i \frac{1}{l} \sum_{j=1}^{l} \varphi_z(z_i, z_j) \frac{\partial \varphi_y(\mathbf{y}_i, \mathbf{y}_j)}{\partial \mathbf{w}_k}, \tag{6.12}$$

where

$$\beta_i = \left(\frac{1}{l} \sum_{j=1}^{l} \varphi_z(z_i, z_j) \varphi_y(\mathbf{y}_i, \mathbf{y}_j) \right)^{-1} . \tag{6.13}$$

Using Eq. 6.8,

$$\frac{\partial \varphi_y(\mathbf{y}_i, \mathbf{y}_j)}{\partial \mathbf{w}_k} = -\frac{1}{2}\varphi_y(\mathbf{y}_i, \mathbf{y}_j)\frac{\partial (\mathbf{y}_i - \mathbf{y}_j)^T \Psi^{-1}(\mathbf{y}_i - \mathbf{y}_j)}{\partial \mathbf{w}_k}. \tag{6.14}$$

Let the quadratic function $(\mathbf{y}_i - \mathbf{y}_j)^T \Psi^{-1}(\mathbf{y}_i - \mathbf{y}_j)$ be denoted by ϑ_{ij}, which can be further decomposed to,

$$\vartheta_{ij} = \sum_{k_1=1}^{d_o}\sum_{k_2=1}^{d_o} \psi^{-1}_{k_1k_2}(y_{ik_1} - y_{jk_1})(y_{ik_2} - y_{jk_2}). \tag{6.15}$$

Hence, the gradient of ϑ_{ij} is

$$\begin{aligned} \frac{\partial \vartheta_{ij}}{\partial \mathbf{w}_k} \quad = \quad & \sum_{k_1=1}^{d_o}\sum_{k_2=1}^{d_o}\left[\frac{\partial \psi^{-1}_{k_1k_2}}{\partial \mathbf{w}_k}(y_{ik_1} - y_{jk_1})(y_{ik_2} - y_{jk_2}) \right. \\ & \left. +\psi^{-1}_{k_1k_2}\frac{\partial (y_{ik_1} - y_{jk_1})(y_{ik_2} - y_{jk_2})}{\partial \mathbf{w}_k}\right]. \end{aligned} \tag{6.16}$$

Consider that $(y_{ik_1} - y_{jk_2})^2$ is a function of $\mathbf{w}_k$ if and only if $k_1 = k$ and/or $k_2 = k$, and $\psi^{-1}_{k_1k_2}$ is a function of $\mathbf{w}_k$ if and only if $k_1 = k_2 = k$. Furthermore, $\psi^{-1}_{k_1k_2} = 0$ only if $k_1 \neq k$ or $k_2 \neq k$. The expression of the gradient above can be written as

$$\frac{\partial \vartheta_{ij}}{\partial \mathbf{w}_k} = \frac{\partial \psi^{-1}_{kk}}{\partial \mathbf{w}_k}(y_{ik} - y_{jk})^2 + \psi^{-1}_{kk}\frac{\partial (y_{ik} - y_{jk})^2}{\partial \mathbf{w}_k} \tag{6.17}$$

From Eq. 6.9, we have

$$\frac{\partial \psi^{-1}_{k,k}}{\partial \mathbf{w}_k} = -\frac{2\zeta}{\psi^2_{k,k}(l-1)}\sum_{i'=1}^{l}(y_{i'k} - \bar{y}_k)\frac{\partial\,(y_{i'k} - \bar{y}_k)}{\partial \mathbf{w}_k} \tag{6.18}$$

where $\bar{y}_k$ denotes the mean value of $y_{i'k}$s, and its partial derivative w.r.t. $\mathbf{w}_k$ can be expressed by

$$\frac{\partial \bar{y}_k}{\partial \mathbf{w}_k} = \frac{1}{l}\sum_{i''}^{l}\frac{\partial y_{i''k}}{\partial \mathbf{w}_k} \tag{6.19}$$

It should also be noted that, $\hat{\mathbf{X}}_{ki}$ (the auto-correlation matrix for the i-th EEG sample processed by the k-th spectral filter, see Eq. 6.4)) is conjugate symmetric, and

$$\frac{\partial y_{ik}}{\partial \mathbf{w}_k} = \frac{(\hat{\mathbf{X}}_{ki} + \hat{\mathbf{X}}^T_{ki})\mathbf{w}_k}{y_{ik}} = \frac{2\mathrm{Re}(\hat{\mathbf{X}}_{ki})\mathbf{w}_k}{y_{ik}} \tag{6.20}$$

where Re() denotes the real part of a complex matrix. The derivatives of $y_{i'k}$ and y_{jk} can be computed the same way as above.

The above steps can be summarized as follows.

$$\nabla_{\mathbf{w}_k}\hat{H}(\mathcal{Y}) = \mathbf{A}\mathbf{w}_k, \tag{6.21}$$

where

$$\begin{aligned}
\mathbf{A} \;=\; & \frac{2}{l^2}\sum_{i=1}^{l}\beta_i\sum_{j=1}^{l}\varphi_z(z_i,z_j)\varphi_y(\mathbf{y}_i,\mathbf{y}_j)\left[\frac{-\zeta(y_{ik}-y_{jk})^2}{\psi_{k,k}^2(l-1)}\cdot\right.\\
& \sum_{i'=1}^{l}(y_{i'k}-\bar{y_k})\left(\frac{\mathrm{Re}(\hat{\mathbf{X}}_{ki'})}{y_{i'k}}-\frac{1}{l}\sum_{i''}^{l}\frac{\mathrm{Re}(\hat{\mathbf{X}}_{ki''})}{y_{i''k}}\right)+\\
& \left.\psi_{kk}^{-1}(y_{ik}-y_{jk})\left(\frac{\mathrm{Re}(\hat{\mathbf{X}}_{ki})}{y_{ik}}-\frac{\mathrm{Re}(\hat{\mathbf{X}}_{kj})}{y_{jk}}\right)\right].
\end{aligned} \tag{6.22}$$

For each conditional entropy $\hat{H}(\mathcal{Y}|c)$, there is an equation similar to Eq. 6.21. The gradient of the objective function $\mathbf{I}$ with respect to the spatial filter $\mathbf{w}_k$ then becomes

$$\nabla_{\mathbf{w}_k}\hat{I}(\{\mathcal{Y},\mathcal{Z}\},C) = \left(\mathbf{A}-\sum_c P(c)\mathbf{A}_c\right)\mathbf{w}_k. \tag{6.23}$$

However, the above equation does not suggest that the gradient is a linear function over $\mathbf{w}_k$, since the multiplier term $(\mathbf{A}-\sum_c P(c)\mathbf{A}_c)$ itself is a rather complicated function over $\{\mathbf{y}_i\}$ which in turn is a function of $\mathbf{W}$.

The iterative optimization algorithm updates a spatial filter with the gradient information by

$$\mathbf{w}_k^{(iter+1)} = \mathbf{w}_k^{(iter)} + \lambda\nabla_{\mathbf{w}_k}\hat{I}(\{\mathcal{Y}^{(iter)},\mathcal{Z}\},C), \tag{6.24}$$

where λ is the step size. A line search procedure was used to determine the step size in each of the iteration. It should be noted that all spatial filter vectors in $\mathbf{W}$ are updated together.

The implemented line search procedure tested a number of (tentatively 16) λ values in the range of [-0.05 0.10]$\times\xi$, and decreased ξ in a logarithmic scale until a local maximum of $\mathbf{I}$ was found except for at $\lambda = 0$. The λ for the local maximum was then used to update all the spatial filters $\mathbf{w}_k$s in Eq. 6.24, and then the optimization procedure proceeded to the next iteration.

Mutual information gain was used as the termination criterion. When mutual information gain was less than 1e-5 the iterations were terminated.

The initial values for $\mathbf{w}_k$ can be learnt by the CSP method [210] that maximizes the Rayleigh coefficient

$$\frac{\mathbf{w}_k \sum_{i=1}^{l_1} \hat{\mathbf{X}}_{ki} \mathbf{w}_k}{\mathbf{w}_k \sum_{j=1}^{l_0} \hat{\mathbf{X}}_{kj} \mathbf{w}_k}, \tag{6.25}$$

where $\hat{\mathbf{X}}_{ki}$ denotes the i-th sample of motor imagery EEG while $\hat{\mathbf{X}}_{kj}$ the j-th sample of idle state EEG.

A set of candidate spectral filters consisting of band-pass filters that cover the motor imagery EEG spectrum was created for the selection of spectral filters for $\mathbf{F}$ similar to the filter banks configuration used in [231]. In the experimental study introduced in the next section, the filter banks configuration from [231] was implemented with 8 band-pass filters with central frequency ranging from 4 to 32 Hz. After band-pass filtering in spectral domain, CSP was trained according to Eq. 6.25 to extract discriminative energy features. Next, the optimum n_w features were selected from all the features using the robust mutual informatin based feature selection method proposed in [231]. The spectral filters associated with the optimum features then comprised the matrix $\mathbf{F}$.

6.4 Results

An offline simulation of the self-paced BCI using the online feedback training data was conducted. The simulation was run in MATLAB, and the proposed method was implemented in hybrid MATLAB and C code so as to improve computation and programming efficiency. The EEG features together with the feedback signal z served as the inputs to a regressor, in order to predict the target value of 0 (idle state), -1 (right-hand MI) or 1 (left-hand MI). A linear

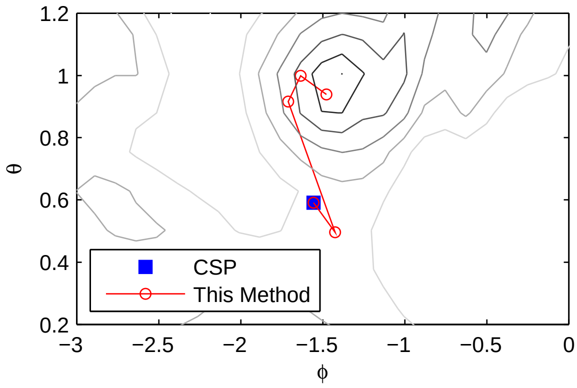


Figure 6.4: Optimization on the mutual information surface

An example with a spatial filter vector for three-channel EEG. See Section 6.4.1 for details.

support vector regression using the LibSVM toolbox [243] was employed. Other regression methods such as Gaussian-kernel non-linear support vector regression, linear mean-square-error regression were also attempted. However, no significant difference was found in the results. Therefore, only the linear support vector regression results were considered for analysis.

Similar to the online feedback training described in Section 6.2, the offline simulation tested left-hand MI BCI and right-hand MI BCI separately. For example, for the left-hand MI BCI, the first left-hand MI training session was used to learn the optimum spatial-spectral filtering and then the linear support vector regressor was trained. Next, the feature extraction and regression was tested on the second left-hand MI training session. The simulation used a 2-second long shift window with a step of 0.4 seconds.

6.4.1 Convergence of the Optimization Algorithm

The convergence of the optimization algorithm was analysed with a simple scenario which included only three EEG channels (CP3,CPz,CP4) and one spatial filter. Since the mutual information measure is always invariant to non-zero norm of the spatial filter, the norm of the spatial filter was set to 1 without loss of generality. Therefore, the spatial filter can be represented by two variables in the spherical coordinate system: $\theta = \text{acos}(w_3)$ and $\phi = \text{atan}(\frac{w_2}{w_1})$. This should not be confused with the Euclidean space where the actual optimization takes place. The two-variable spherical coordinate representation was used only for visualization purposes.

Figure (6.4) shows a typical example from the left-hand MI learning of Subject 2. The spatial filter solution migrated in 4 steps from the initial point (generated by CSP) to approximately a local maximum where the iteration converged (mutual information gain <1e-5).

The proposed algorithm was initialized using the method described in the previous section, and then in most cases the proposed optimization algorithm converged within 7 iterations. Random initialization of spatial filters was also considered and the iteration procedure generally became longer but converged within 50 iterations in all 100 test runs.

6.4.2 Feature Distributions

The first feedback training session was used to learn 2 spatial-spectral filters by the proposed method, and EEG features from the second feedback session were extracted. Figure (6.5) plots the distribution of the features (as the original samples amount to thousands, evenly re-sampled feature vector samples were used for clarity).

The new features appear to be more separable between the MI classes and the idle states when comparing with the features produced by calibration models in Figure (6.3) (especially in the bottom row for the same training session). The separability in terms of classification accuracy was assessed by a linear support vector machine (using the same LibSVM toolbox from [243]). The comparison of results on the original features and the new features are shown in Table (6.1).

The table (6.1) clearly indicates that the proposed method, which adapted both the classifier and the feature extraction model, produced significantly better performance in terms of class separability, than when only the classifier was adapted. This finding substantiates the argument that the non-stationarity in EEG may not be solved only by adapting classifiers. Rather, it is

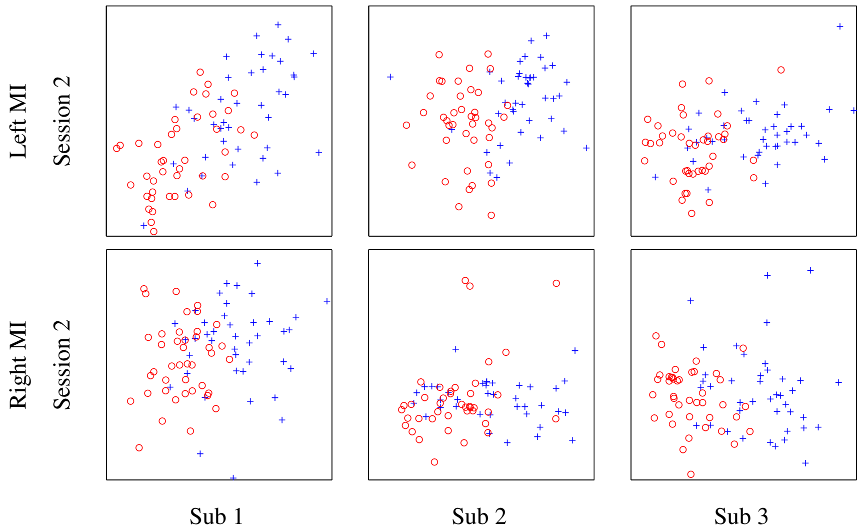


Figure 6.5: Feature distributions by the proposed learning method for the left/right motor imagery (MI) feedback training session 2.

The horizontal axis and the vertical axis are respectively the first and the second features learnt by the learning method. The graphs in the upper row are generated from left MI training data, while the lower row are from right MI training data. Red circles represent motor imagery samples, while black crosses denote idle state samples.

advisable to adapt both the feature extraction model and the classifier so as to accurately capture the variations of EEG over time.

6.4.3 Accuracy of Feedback Control Prediction

It was investigated whether the new features can generate better prediction of user state. Since the classification hyperplane may have shifted from the first feedback session to the second, the adaptation of the regressor was also tested. A supervised adaptation of the regressor was carried out using a portion of data from second feedback session (adaptation data). The regressor was re-trained using both the adaptation data and data from first feedback session, and the models were tested on the remainder of the second feedback session (excluding adaptation data). Different sizes for the adaptation data in terms of percentage of the whole session, ranging from 0 (i.e. no adaptation) to 0.45 was investigated.

Filter bank CSP (FBCSP) was also evaluated using the same method for comparison. The

	Features	Sub 1	Sub 2	Sub 3
Left MI	Original	73.7%	79.0%	66.9%
	This Method	**85.0%**	**84.8%**	**81.0%**
Right MI	Original	67.9%	59.7%	78.1%
	This Method	**80.0%**	**69.6%**	**84.0%**

Table 6.1: Class separability: new feature space ("This method") versus original feature space ("Original").

Class separability is measured as the classification accuracy by a linear support vector machine that is adapted to the data (feedback training session 2). Note "Original" uses adaptation of classifier only, while "This method" adapts both the classifier and the feature extraction model. The higher accuracy rates between the two feature spaces are shown in bold style.

comparative results are illustrated in figure (6.6). Apparently, both FBCSP and the proposed method can learn a more accurate predictor from the first feedback session than the original BCI that used only the calibration data. Furthermore, the prediction error was also effectively reduced by the supervised adaptation. But, this improvement is not as significant as the improvement observed from the original BCI to the proposed method. Furthermore, the proposed method also consistently outperformed FBCSP, significantly in most cases.

The impact of the new method on the feedback signal curves was also examined. Figure (6.7) illustrates a graph comparing the new feedback signal to the original feedback signal, for Subject 2. The new feedback signal curve followed the target curve much more accurately than the original feedback signal.

It was also investigated whether the proposed method works with a reduced set of channels. Particularly, 15, 9 and 6 channels were tested for the proposed method and FBCSP, using the same method described above (see Figure 6.6), and performed t-test to check whether the pro-

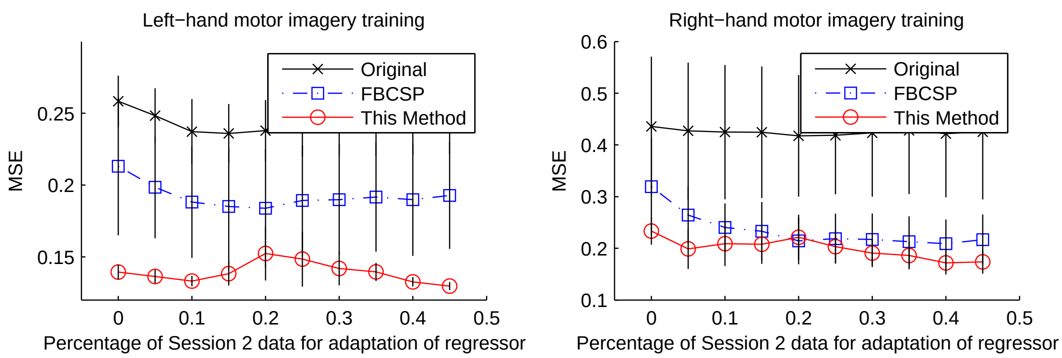


Figure 6.6: Comparison of prediction error in terms of mean-square-error (MSE) by different methods.

The horizontal axis denotes the percentage of the second feedback session being used for re-training the support vector regression machine that maps EEG features to the target signal. The curves plot the average of MSE over the three subjects, while the vertical line centered at the each point represents the standard deviation by its length.

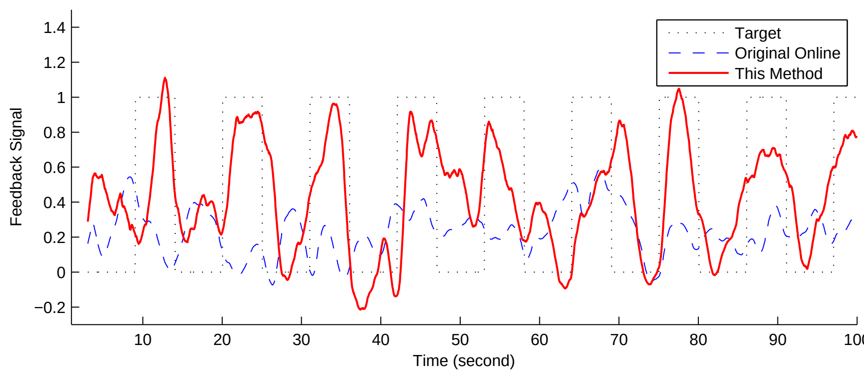


Figure 6.7: Comparison between target, original feedback signal and the new prediction by the proposed method.

An example from Subject-2's left motor imagery training session. The timing is in alternation between approximately 5-second motor imagery (*target*=1) and 6-second idle state (*target*=0) except the first idle state period which is slightly longer.

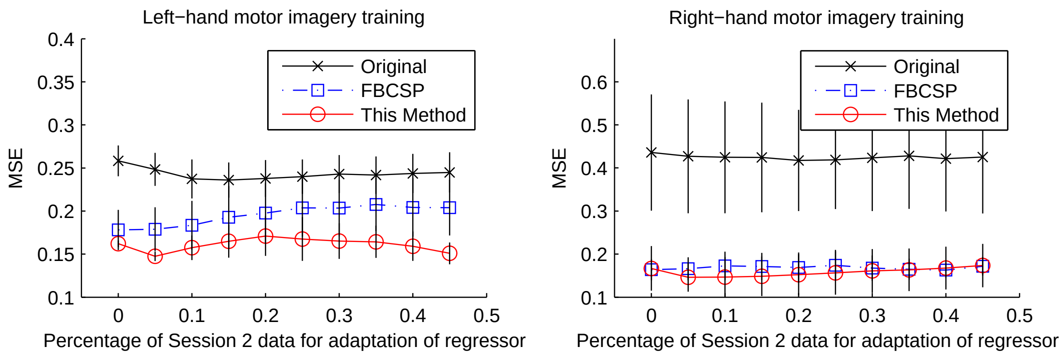


Figure 6.8: Comparison of prediction error in mean-square-error (MSE) by different methods using 9 EEG channels only.

#Ch	Data	p-value This vs FBCSP	p-value This vs Original	Channel Names
All	Left MI	<**0.01**	<**0.01**	All 30 Channels (See Section 6.2).
	Right MI	<**0.04**	<**0.01**	
15	Left MI	<**0.01**	<**0.01**	F3,F4,FC3,FCz,FC4,T3,Cz,
	Right MI	0.09	<**0.01**	C4,T4,CP3,CPz,CP4,P3,P4
9	Left MI	<**0.01**	<**0.01**	FC3,FCz,FC4,C3,Cz,C4,CP3,
	Right MI	0.86	<**0.01**	CPz,CP4
6	Left MI	0.48	<**0.01**	FC3,FC4,C3,C4,CP3,CP4
	Right MI	0.93	<**0.01**	

Table 6.2: Statistical paired t-test comparing the proposed method with FBCSP and the original feedback training results, using different number of channels.

Significant results with p-value <0.05 are shown in bold.

posed method produced lower MSE with statistical significance compared to FBCSP and the original feedback training result.

The results indicate that the proposed method significantly improved the performance in terms of MSE in all the channel sets that were tested. The proposed method yielded significantly lower MSE than FBCSP also with as few as 9 channels. In the case of 6 channels, the proposed method and FBCSP produced comparable results, while both significantly outperformed the original model constructed from calibration only.

6.5 Discussions

The figure (6.6) gives clear evidence that the proposed method of using the new spatial-spectral learning algorithm can significantly increase the prediction accuracy. The mean MSE for left (or right) MI feedback training was effectively reduced from approximately 0.3(or 0.5) to a slightly lesser value of 0.2 (or 0.25). The improved accuracy can also be seen in the prediction curves in the example case shown in Figure (6.7), which actually showcases a reduction of MSE from 0.24 to 0.13.

The increased accuracy can be largely attributed to the improved feature space shown in figure (6.5) in contrast to the original feature spaces in figure (6.3). The original feature space that was used in feedback training was built using the calibration data. The changes of feature distributions in the original feature space have highlighted the effect of session-to-session transfer, which is generally consistent with prior studies on adaptive BCI. Thus, during the feedback sessions, the motor imagery EEG and idle-state EEG was predominantly non-separable. Even if they were separable it was subjected to distribution shift. On the other hand, the new feature space was learnt from the feedback training data comprised of three sources of information, namely, EEG, the target signal and the feedback signal. Therefore, it has been able to capture essential information for user state prediction during online feedback training.

The new model uses a non-parametric formulation for learning, which aims to account for arbitrary dependencies among EEG, target and feedback signals. It was shown in section 6.4.1 that the proposed optimization algorithm, derived through the new formulation has good convergence properties. Figure (6.4) showed that the objective function surface for the 3-channel EEG data is smooth, which is a favourable condition for a greedy algorithm. However, the mutual information surface can become far more complicated, especially for EEG data with a large number of channels. Therefore, future research may investigate more advanced optimization techniques.

However, such techniques would usually incur much heavier computational costs.

This work focused on the development and validation of a new learning method for adaptive BCI, it would be interesting to investigate its performance during online training. Generally, a large number of subjects would be required in order to draw statistically significant comparisons between adaptive and non-adaptive BCI systems.

It would also be interesting to look into the formulation of objective formulation in Section 6.3.2. As stated earlier, the goal is to maximize the information about the target signal to be predicted, contained in the EEG features in conjunction with the feedback. Therefore, it is advisable to include both the new EEG features and the prediction outputs of the current model as inputs to the classifier or regression machine in the new model. Importantly, the feedback serves two purposes: not only does it serve as a visual "stimulus" to the subject, but it also represents the current prediction model that contains essential information extracted from earlier calibration/feedback sessions. The first rationale is that, feedback and its relative position to the target signal may have an effect on brain activations to complicate motor imagery EEG. The second function gives rise to multiple implications as explained below. First, the formulation considers only the output of the current BCI model but not the internal mechanism of the model. Thus, it can work with any BCI model and adapt them during new feedback training sessions. Secondly, if a user with a prediction model can control the feedback signal to match the target signal satisfactorily during a feedback session, further re-adaptation of the prediction model might be unnecessary as co-adaptation of user and machine has already been achieved. This can also be viewed as a special case of the objective function Eq. 6.10: if the feedback variable $\mathcal{Z}$ in the objective function already carries essential information about the target signal C, re-adaptation of BCI by including new EEG features would produce no significant gain in the objective function.

The proposed method works in a supervised learning fashion where it requires the data labels for adaptive learning. Unlike in unsupervised or semi-supervised online learning approaches, this enables the learning system to measure the compliance of a subject to the BCI tasks, so as to ensure the stability of the adaptation process.

The proposed method with the current solution may be more suited for offline adaptation than for online adaptation. In online adaptation, both user training and machine adaptation take place at the same time. While in offline adaptation, machine adaptation is performed after the user finishes a training session. Although this method is applicable to online adaptation, the expensive computation can be a serious concern for practical online use. The computational complexity of computing the gradient by Eq. 6.23 and Eq. 6.22 was estimated to be on the order of $O(l^2 n_c^2)$ and that of evaluating the objective function by Eq. 6.7 and Eq. 6.6 is $O(l^2 n_c)$. Here l denotes the number of samples and n_c the number of channels. In the experimental setup for the results presented in Section 6.4, a learning code using hybrid MATLAB and C coding without multi-threading was implemented. The code took approximately 130 seconds to complete one iteration for $n_c = 30$-channel EEG data, or 18 seconds for $n_c = 6$-channel EEG data, both of $l = 2230$ time segment samples on our test computer with a Xeon CPU at 2.93GHz. The primary cause for high computational complexity is the non-parametric (kernel-based) nature of the method that requires computations for each pair of samples. Therefore, a possible solution to this problem would be to reduce the number of samples used for adaptation.

6.6 Conclusion

In this chapter the critical issue of session to session transfer in brain-computer interface (BCI) was studied. While previous studies have often focused on adaptation of classifiers, the importance and the feasibility of adapting feature extraction models within a self-paced BCI paradigm was demonstrated. First, calibration and feedback training on able-bodied naïve subjects using a new self-paced motor imagery BCI including idle state was conducted. The online results suggest that the feature extraction models built from calibration data may not generalize well to feedback sessions. Hence, a new supervised adaptation method that learns from feedback data was proposed to construct a more accurate model for feedback training. The learning objective was formulated as maximization of kernel-based mutual information estimation with respect to spatial-spectral filters, and derived a gradient-based optimization algorithm for the learning task. An experimental study through offline simulations were conducted and the results suggest that the proposed method can significantly increase prediction accuracies for feedback training sessions.

Chapter 7

Conclusion and Future Work

In this chapter the results from the four methods that were proposed are summarized. An overview of possible future work based on the presented methods are discussed at the end of the chapter.

7.1 Summary of Results

This thesis presented multiple methods to improve the information transfer rate of current brain computer interfaces. Information transfer rate can be improved by increasing the classification performance and by increasing the number of classes that are effectively classified. However, even in multiclass classification the ITR is directly dependent on the performance of classifiers. Increasing the classification accuracies is further complicated by the non-stationarity of the EEG signals. Therefore, to address this issue, novel feature extraction and signal classification methods were explored.

In Chapters 3, joint approximate diagonalization (JAD) for multiclass CSP was considered to overcome the limitation of CSP algorithm for feature extraction. The current CSP algorithm can only consider two classes for simultaneous diagonalization. Multiple covariance matrices from different motor imagery signals from four classes can be simultaneously diagonalized with the proposed joint approximate diagonalization method.

Specifically, a fast Frobenius diagonalization (FFDIAG) based multiclass CSP was proposed to deal with the limitation of current CSP algorithm. Several classifiers, k-NN, CART and SVM were employed with the FFDIAG method for feature extraction. The results were compared against the baseline of one versus rest CSP method and Jacobi angle based simultaneous diagonalization method. The effects of boosting the classifiers were also analyzed with the implementations of Adaboost.M1 and SAMME algorithm for multiclass classifier boosting. The results showed significant improvements over the baseline classifiers. SVM classifier consistently gave the highest classification accuracies. SAMME algorithm was more practical in boosting the weak classifiers in the multiclass classification scenario as Adaboost algorithm needs a minimum performance of 50% from each weak classifier. Results showed that the proposed FFDIAG method effective in simultaneously diagonalizing more than two covariance matrices.

As another theoretical development, in Chapter 4, we developed an Adaptively Weighted Classifier Ensemble with clustering. The underlying idea of this new approach was to weigh the decisions from a classifier ensemble based on the closest cluster to a given test sample. The clusters are found by clustering the training data with minimum Havrda-Charvat structural entropy and cosine distance based clustering method.

The novelty of this approach is adaptively weighting the decisions from the component classifiers in the ensemble based on the measurement of distance from a given test to the clusters. The classifiers that are trained on data nearer to the given test sample get higher weight under this method. The proposed method is able to exploit the structural information contained in the training data by the distance metric on the clusters. Results show that the proposed method is able to handle session to session non-stationarity of EEG data. Another major advantage of the proposed method is its low complexity, making it more efficient than other complex methods such as EM algorithm and Bayesian methods.

In Chapter 5, an algorithm for adaptive training of a SVM classifier was proposed to improve classification accuracies under non-stationarity in EEG data. The proposed method adapts the SVM kernel to training data from subsequent sessions. The kernel width parameter of the kernel function of the SVM classifier was adapted using an information theoretic cost function based on minimum error entropy (MEE). The novelty of the method is, using the distribution of the error function rather than the error values to adapt the kernel width parameter to adaptively train the classifier. Experiments were performed using the proposed method on EEG data collected from 12 healthy subjects in two sessions on separate days. The results using the proposed method yielded significantly better classification accuracies compared to the baseline.

In Chapter 6, we applied the central idea of learning from feedback training data to a self-paced BCI scenario. The feasibility and the effectiveness of adaptive feature extraction was analysed by conducting calibration and feedback training on able-bodied naïve subjects. A novel self-paced motor imagery BCI including idle state was used in the experiments. The online results suggest that the feature space constructed from calibration data may become ineffective during feedback sessions due to non-stationarity issues. Therefore, a novel supervised method that learns from feedback data was used to construct a more appropriate feature space, on the basis of maximum mutual information principle between feedback signal, target signal and EEG.

Specifically, we formulated the learning objective as maximizing a kernel-based mutual information estimate with respect to the spatial-spectral filtering parameters. A gradient-based optimization algorithm was then derived for the learning task. An experimental study was conducted with offline simulations. The results suggest that the proposed method is able to construct effective feature spaces to capture the discriminative information in feedback training data. Results indicate that classification accuracies can be significantly improved using these new fea-

Table 7.1: Comparison of ITR of Implemented Methods

Dataset 2A	Baseline		FFDIAG	AWEC	MEE
	Two Class	Four Class			
Accuracy	75.9	28.8	54.1	81.5	82.7
Duration (min.)	10.03	17.25	29.68	12.87	13.64
Number of Decisions	1296	2592	2592	1296	1296
ITR	26.25	0.81	24.22	31.11	31.93

tures. By improving the classification accuracies we have been able to improve the overall information transfer rate of the BCI system. The Table (7.1) summarizes the ITR for the three synchronous BCI methods proposed. The computation times incurred for classifying 72 trials from each class on an Intel Core i5 CPU with 3.2GHz and 4GB RAM running on 32-bit Windows platform are shown. The baseline performances for two class and four class classification are compared with the corresponding proposed methods. FFDIAG method for joint diagonalization was tested on four class classification problem. The AWEC method for adaptive ensemble weighting and Minimum Error Entropy Kernel Adaptation (MEE) was tested on two class classification of BCI Competition IV data set 2A.

The FFDIAG method for multi-class classification has improved the ITR to 24.22 compared to the corresponding baseline ITR of 0.81 for four-class classification. The baseline for two-class classification was calculated to be 26.25. Both AWEC and MEE methods has improved ITR for the two-class classification at 31.11 and 31.93 respectively.

7.2 Real-time Implementation of Proposed Methods

Practical applications of BCI may be broadly classified into real-time and non-real-time implementations. Non-real time applications can be identified as systems that process the collected brain signals offline and output/use the results at a later time. Non-real time implementations are mostly found under laboratory conditions when testing new BCIs. Non-real time implementations are also found in a few gaming and disease diagnosis applications [249]. Systems that help diagnose diseases such as Epilepsy, Sleep disorders, Brain tumors, Autism and Alzheimer [250] by monitoring brain signals can also be categorized as non-real time implementations as process the signals offline. However, if the BCI systems assist the management of the disease, such as with cortical surface electrodes in the case of Epilepsy [251], then they are considered as real-time BCI's. BCI's of this type can predict an oncoming seizure in real time and in some cases prevent it by stimulating appropriate brain areas [252].

The associated computational issues for real-time BCIs can be identified at several stages of the BCI system. The low-signal to noise ratio remains a major challenge for substantial improvements in performance. Noise may include brain signals that are not associated with brain patterns generated by the user's intent, or signals added by the hardware used to acquire the brain signals. The first two methods proposed in this thesis can be applied in real-time BCI implementations. The computational costs of the methods are not very high as shown in Table (7.1).

The FFDIAG method for joint diagonalization and AWEC method for adaptive ensemble weighting can be easily implemented on a real-time BCI without much effect on run times. The Kernel adaptation method (MEE) need some past test samples for adaptation. In a real-time implementation stage-wise online adaptation of the Kernel width can be carried out after a specific number of test samples. The method of learning from feedback training data in self-spaced BCI was implemented in non-real time offline laboratory conditions because the method uses supervised learning for adaptation. Real time extension is possible with unsupervised/semi-supervised learning instead of the supervised learning mechanism. On the other hand, all of the proposed methods can be adapted for non-real time offline analysis of EEG data.

7.3 Suggestions for Future Work

Future research can transfer the methods proposed in this thesis to other similar scenarios, such as the transfer of classifier parameters from subject to subject based on the methods developed for session to session non-stationarity. Although the variability across subjects can easily be regarded within the same framework as the variability from session to session, it is out of the scope of this work. However, with this approach, BCI research can be conceivable for a wider range of applications, by reducing the calibration time for naive subjects.

Furthermore, it is a possible to apply these methods to other neurophysiological paradigms and multi-class applications. Future research should strive for robustification of the non-stationary EEG signal using machine learning methods to make BCI applications more usable.

The approximate joint daigonalization method proposed here can be extended to simultaneous diagonalization of more than two covariance matrices leading to more separable band power features. This would require further mathematical research which is outside the scope of current study.

The cluster based classifier ensemble framework can be extended to an adaptive classifier ensemble which adds and removes clusters automatically according to incoming test data in an online scenario. Such a system would be beneficial to long term users of a BCI system.

The idea of using error distribution parameters such as kernel width for adaptive training can be extended to other classifiers with suitable parameters that can be manipulated based on the error distribution.

Finally, the use of feedback training data in self-paced BCI can be extended to include error potential signals also. Adaptation of the feature selection methods can be extended to unsupervised and semi-supervised online adaptation. Online adaptation in the self-paced paradigm would be immensely valuable to overcome practical limitations of current BCI's.

Bibliography

[1] J.R. Wolpaw, N. Birbaumer, D.J. McFarland, G. Pfurtscheller, and T.M. Vaughan. “Brain computer interfaces for communication and control”. *Clinical Neurophysiology*, 113(6):767-791, 2002.

[2] J.R.Wolpaw and E.W.Wolpaw, “Brain Computer Interfaces: Something New Under the Sun”, *In: J.R.Wolpaw and E.W.Wolpaw (eds.), Brain-Computer Interfaces Principles and Practice*, Oxford University Press, New York, 227-240, 2012.

[3] E. Lalor, S. P. Kelly, C. Finucane, R. Burke, R. Smith, R. Reilly, and G. Mc-Darby. “Steady-state VEP-based brain-computer interface control in an immersive 3-D gaming environment”. *EURASIP journal on applied signal processing*, 2005.

[4] B. Allison, B. Graimann, and A. Grser. “Why use a BCI if you are healthy?” In proceedings of *BRAINPLAY 2007, Playing with Your Brain*, Austria, 2007.

[5] B.Z. Allison, E.W.Wolpaw, and J.R.Wolpaw. “Brain-computer interface systems: progress and prospects”. *Expert Review of Medical Devices*, 4(4):463-474, 2007.

[6] J.R. Wolpaw, “Brain Computer Interface Research Comes of Age: Traditional Assumptions Meet Emerging Realities”, *Journal of Motor Behavior*, 42:6, 2010.

[7] Benjamin Blankertz and Klaus-R. Mueller, “Computational Challenges for Noninvasive Brain Computer Interfaces”, *IEEE Intelligent Systems*, 23(3),78-79, 2008.

[8] Eilon Vaadia and Niels Birbaumer, "Grand challenges of brain computer interfaces in the years to come", *Frontiers in Neuroscience*, 3(2), 2009.

[9] P. Sykacek, S. Roberts, and M. Stokes, "Adaptive BCI based on variational Bayesian Kalman filtering: an empirical evaluation", In *IEEE Transactions on Biomedical Engineering*, 51(5), 719-729, 2004.

[10] C. Guger, H. Ramoser, and G. Pfurtscheller, "Real-time EEG analysis with subject-specific spatial patterns for a brain-computer interface", *IEEE Tran. on Rehab. Eng.*, 8(4), 447-456, 2000.

[11] P. Shenoy, M. Krauledat, B. Blankertz, R.P. Rao, and K. R. Mueller. "Towards adaptive classification for BCI", *Journal of Neural Engineering*, 3(1), 13-23, 2006.

[12] M. Sugiyama, M. Krauledat and K.-R. Mueller, "Covariate shift adaptation by importance weighted cross validation", *J Mach Learning Res.* 8, 1027-1061, 2007.

[13] J. Kronegg, G. Chanel, S. Voloshynovskiy, and T. Pun, "EEG-based synchronized brain-computer interfaces: A model for optimizing the number of mental tasks", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(1):50-58, 2007.

[14] G. Dornhege, B. Blankertz, G. Curio, and K.R. Mueller, "Boosting bit rates in non-invasive EEG single-trial classifications by feature combination and multi-class paradigms", *IEEE Transactions on Biomedical Engineering*, 51(6):993-1002, 2004.

[15] C. Vidaurre, C. Sannelli, K-R. Mller, B. Blankertz, "Machine-Learning-Based Coadaptive Calibration for Brain-Computer Interfaces", *Neural Computation*, 23(3): 791-816, 2011.

[16] M.A. Lebedev and M.A.L. Nicolelis. "Brain-machine interfaces: past, present and future". *Trends in Neurosciences*, 29(9):536-546, 2006.

[17] G. Pfurtscheller, B. Graimann, and C. Neuper, chapter "EEG-Based Brain-Computer Interface System". *Wiley Encyclopedia of Biomedical Engineering*, John Wiley & Sons, Inc., 2006.

[18] S. Mason, J. Kronegg, J. Huggins, M. Fatourechi, and A. Schloegl, "Evaluating the performance of self-paced BCI technology", *Neil Squire Soc., Vancouver, BC, Canada, Tech. Rep.*, 2006.

[19] J. Kalcher, D. Flotzinger, C. Neuper, S. Golly, and G. Pfurtscheller, "Graz Brain-computer interface II: towards communication between humans and computers based on online classification of three different EEG patterns", *Medical and Biological Engineering and Computing*, 34:383-388, 1996.

[20] G. Pfurtscheller, C. Neuper, G.R. Mueller, B. Obermaier, G. Krausz, A. Schlogl, R. Scherer, B. Graimann, C. Keinrath, D. Skliris, M. Wortz, G. Supp, and C. Schrank. "Graz-BCI: state of the art and clinical applications", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):1-4, 2003.

[21] A. Bashashati, R.K. Ward, and G.E. Birch. "Towards development of a 3- state self-paced brain-computer interface", *Computational Intelligence and Neuroscience*, 2007.

[22] R. Scherer, A. Schloegl, F. Lee, H. Bischof, J. Jansa, and G. Pfurtscheller, "The self-paced Graz brain-computer interface: Methods and applications", *Computational Intelligence and Neuroscience*, 2007.

[23] J. del R. Millan and J. Mourino, "Asynchronous BCI and local neural classifiers: An overview of the Adaptive Brain Interface project", *IEEE Transactions on Neural Systems and Rehabilitation Engineering, Special Issue on Brain-Computer Interface Technology*, 11(2): 159-161, 2003.

[24] G. Pfurtscheller, C. Neuper, and N. Birbaumer, “Motor cortex in voluntary movements”, *Human Brain-computer Interface*, CRC Press, 367-401, 2005.

[25] J.R.Wolpaw, G.E. Loeb, B.Z. Allison, E. Donchin, O.F. do Nascimento, W.J. Heetderks, F. Nijboer, W.G. Shain, and J. N. Turner, “BCI meeting 2005- workshop on signals and recording methods”, *IEEE Transaction on Neural Systems and Rehabilitation Engr.*, 14(2):138-141, 2006.

[26] J. W. de Moor, “Building a brain interface”, www.vf.bio.uu.nl/LAB/NE/scripties/Building_a_Brain_Interface 2009.

[27] J. Mellinger, G. Schalk, C. Braun, H. Preissl, W. Rosenstiel, N. Birbaumer, and A. Kubler, “ An MEG-based brain-computer interface (BCI)”. *Neuroimage*, 36(3):581-593, 2007.

[28] M. Besserve, K. Jerbi, F. Laurent, S. Baillet, J. Martinerie, and L. Garnero, “Classification methods for ongoing EEG and MEG signals.” *Biol. Res.*, 40(4):415-437, 2008.

[29] N. Weiskopf, K. Mathiak, S.W. Bock, F. Scharnowski, R. Veit, W. Grodd, R. Goebel, and N. Birbaumer, “Principles of a brain-computer interface (BCI) based on real-time functional magnetic resonance imaging (fMRI)”, *IEEE Transactions on Biomedical Engineering*, 51(6):966-970, 2004.

[30] S.M. Coyle, T.E.Ward, and C.M. Markham, “Brain-computer interface using a simplified functional near-infra red spectroscopy system”, *Journal of Neural Engineering*, 4:219-226, 2007.

[31] E.C. Leuthardt, K.J. Miller, G. Schalk, R.P.N. Rao, and J.G. Ojemann, “Electrocorticography-based brain computer interface-the Seattle experience”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2):194-198, 2006.

[32] E. Niedermeyer and F. Lopes da Silva. *Electroencephalography: basic principles, clinical applications, and related fields*, Lippincott Williams & Wilkins, 5th edition, 2005.

[33] H. Jasper, "Ten-twenty electrode system of the international federation", *Electroenceph. Clin. Neurophysiol*, 10:371-375, 1958.

[34] American electroencephalographic society, "Guidelines for standard electrode position nomenclature", *J Clin Neurophysiol.*, 8(2):200-202, 1991.

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

[36] E. A. Curran and M. J. B. Stokes, "Learning to control brain activity: a review of the production and control of EEG components for driving brain-computer interface (BCI) systems", *Brain and Cognition*, 326-336, 2003.

[37] C. Gouy-Pailler, S. Achard, B. Rivet, C. Jutten, E.Maby, A. Souloumiac, and M. Congedo, "Topographical dynamics of brain connections for the design of asynchronous brain-computer interfaces", *In Proc. Int. Conf. IEEE Engineering in Medicine and Biology Society (IEEE EMBC)* , 2520-2523, 2007.

[38] G. Mueller-Putz, R. Scherer, C. Neuper, and G. Pfurtscheller, "Steady-state somatosensory evoked potentials: suitable brain signals for brain-computer interfaces", *IEEE transactions on neural systems and rehabilitation engineering*, 14(1):30-37, 2006.

[39] G.R. McMillan, G.L. Calhoun, M.S. Middendorf, J.H Schuner, D.F. Ingle, and V.T. Nashman, "Direct brain interface utilizing self-regulation of steady state visual evoked response", *In Proceedings of RESNA*, 693-695, 1995.

[40] H. Touyama and M. Hirose, “Steady-state VEPs in CAVE for walking around the virtual world”, *Universal Access in Human-Computer Interaction*, 6, 715-717, 2007.

[41] T. Solis-Escalante, G.G. Gentiletti, and O. Yanez-Suarez, “Detection of steady-state visual evoked potentials based on the multisignal classification algorithm”, *In 3rd International IEEE/EMBS Conference on Neural Engineering*, 184-187, 2007.

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

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

[44] M. Cheng, X. Gao, S. Gao, and D. Xu, “Design and implementation of a brain-computer interface with high transfer rates”, *IEEE Transactions on Biomedical Engineering*, 49(10):1181-1186, 2002.

[45] L.J. Trejo, R. Rosipal, and B. Matthews, “Brain-computer interfaces for 1-D and 2-D cursor control: designs using volitional control of the EEG spectrum or steady-state visual evoked potentials”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2):225-229, 2006.

[46] K.D. Nielsen, A.F. Cabrera, and O.F. do Nascimento, “EEG based BCI towards a better control, Brain-computer interface research at Aalborg university”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2):202-204, 2006.

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

[48] L.A. Farwell and E. Donchin, “Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials”, *Electroencephalography and Clinical Neurophysiology*, 70:510-523, 1988.

[49] B. Rivet and A. Souloumiac, “Subspace estimation approach to P300 detection and application to brain-computer interface”, *In Proc. Int. Conf. IEEE Engineering in Medicine and Biology Society (EMBC)*, 5071-5074, 2007.

[50] D.J. Krusienski, E.W. Sellers, F. Cabestaing, S. Bayoudh, D.J. McFarland, T.M. Vaughan, and J.R. Wolpaw, “A comparison of classification techniques for the P300 speller”, *Journal of Neural Engineering*, 3:299-305, 2006.

[51] E.W. Sellers and E. Donchin, “A P300-based brain-computer interface: initial tests by ALS patients”, *Clin Neurophysiol.*, 117(3):538-548, 2006.

[52] F. Piccione, F. Giorgi, P. Tonin, K. Priftis, S. Giove, S. Silvoni, G. Palmas, and F. Beverina, “P300-based brain computer interface: reliability and performance in healthy and paralysed participants”, *Clin Neurophysiol.*, 117(3):531-537, 2006.

[53] J.R. Wolpaw and D.J. McFarland, “Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans”, *Proc. Natl. Acad. Sci. USA*, 101(51):49-54, 2004.

[54] J.R. Wolpaw, “Brain-computer interfaces as new brain output pathways”. *J Physiol*, 579:613-619, 2007.

[55] G. Pfurtscheller, C. Brunner, A. Schlogl, and F.H. Lopes da Silva, “Mu rhythm (de)synchronization and EEG single-trial classification of different motor imagery tasks.”, *NeuroImage*, 31(1):153-159, 2006.

[56] G. Pfurtscheller, C. Neuper, D. Flotzinger, and M. Pregenzer, “EEG-based discrimination between imagination of right and left hand movement”, *Electroencephalography and Clinical Neurophysiology*, 103:642-651, 1997.

[57] G. Pfurtscheller, C. Neuper, A. Schlogl, and K. Lugger, “Separability of EEG signals recorded during right and left motor imagery using adaptive autoregressive parameters”, *IEEE Transactions on Rehabilitation Engineering*, 6(3):316-325, 1998.

[58] G. Pfurtscheller and F. H. Lopes da Silva, “Event-related EEG/MEG synchronization and desynchronization: basic principles”, *Clinical Neurophysiology*, 110(11):1842-1857, 1999.

[59] R. Scherer, G. R. Mueller, C. Neuper, B. Graimann, and G. Pfurtscheller, “An asynchronously controlled EEG-based virtual keyboard: Improvement of the spelling rate”, *IEEE Transactions on Biomedical Engineering*, 51(6):979-984, 2004.

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

[61] B. Blankertz, G. Dornhege, M. Krauledat, G. Curio, and K.-R. Mller, “The non-invasive Berlin brain-computer interface: Fast acquisition of effective performance in untrained subjects.” *NeuroImage*, 37(2):539-550, 2007.

[62] B. Blankertz, G. Dornhege, M. Krauledat, K.-R. Mueller, V. Kunzmann, F. Losch, and G. Curio, “The Berlin brain-computer interface: EEG-based communication without subject training”, *IEEE Trans. Neural Sys. Rehab. Eng.*, 14(2):147-152, 2006.

[63] G. Pfurtscheller, C. Neuper, C. Guger, W. Harkam, H. Ramoser, A. Schlogl, B. Obermaier, and M. Pregenzer, “Current trends in Graz brain-computer interface (BCI) research”, *IEEE Transactions on Rehabilitation Engineering*, 8(2):216-219, 2000.

[64] S.W. Smith, *The Scientist & Engineer’s Guide to Digital Signal Processing*, ISBN 0-9660176-3-3, 1997.

[65] D. J.McFarland, L.M.McCane, S. V. David, and J. R.Wolpaw, “Spatial filter selection for EEG-based communication”, *Electroencephalographic Clinical Neurophysiology*, 103(3):386-394, 1997.

[66] Z.J. Koles and A. C. K. Soong, “EEG source localization: implementing the spatio-temporal decomposition approach”, *Electroencephalogr. Clin Neurophysiol*, 107:343-352, 1998.

[67] B. Blankertz, R. Tomioka, S. Lemm,M. Kawanabe, and K.-R.Mueller, “Optimizing spatial filters for robust EEG single-trial analysis”, *IEEE Signal Proc Magazine*, 25(1):41-56, 2008.

[68] A. Bashashati, M. Fatourechi, R. K. Ward, and G. E. Birch, “A survey of signal processing algorithms in brain-computer interfaces based on electrical brain signals”, *Journal of Neural engineering*, 4(2):35-57, 2007.

[69] D. J. McFarland, C. W. Anderson, K.-R. Mueller, A. Schlogl, and D. J. Krusienski, “BCI meeting 2005-workshop on BCI signal processing: feature extraction and translation”,

IEEE Transactions on Neural Systems and Rehabilitation Engineering, 14(2):135 - 138, 2006.

[70] A. Schlogl, K. Lugger, and G. Pfurtscheller, “Using adaptive autoregressive parameters for a brain-computer-interface experiment”, *In Proceedings 19th International Conference (EMBS)*, 1533-1535, 1997.

[71] C. W. Anderson, E. A. Stolz, and S. Shamsunder, “Multivariate autoregressive models for classification of spontaneous electroencephalographic signals during mental tasks”, *IEEE Transactions on Biomedical Engineering*, 45(3), 277-286, 1998.

[72] D.J.Krusienski, D.J.McFarland and J.C.Principe, “BCI Signal Processing: Feature extraction”, *In: J.R.Wolpaw and E.W.Wolpaw (eds.), Brain-Computer Interfaces Principles and Practice*, Oxford University Press, New York, 123-144, 2012.

[73] D.J.McFarland and D.J.Krusienski, “BCI Signal Processing: Feature translation”, In: J.R.Wolpaw and E.W.Wolpaw (eds.), *Brain-Computer Interfaces Principles and Practice*, Oxford University Press, New York, 123-144, 2012.

[74] Y.Li, K.K.Ang, C.T.Guan, “Digital Signal Processing and Machine Learning”, *In: B.Graimann, B.Allison, G. Pfurtscheller, Brain-Computer Interfaces Revolutionizing Human-Computer Interaction*, Springer, New York, 2010, 305-329.

[75] M. Kaper, P. Meinicke, U. Grossekathoefer, T. Lingner, and H. Ritter, “BCI competition 2003-data set IIb: support vector machines for the P300 speller paradigm”, *IEEE Transactions on Biomedical Engineering*, 51(6):1073- 1076, 2004.

[76] S. Rezaei, K. Tavakolian, A. M. Nasrabadi, and S. K. Setarehdan, “Different classification techniques considering brain computer interface applications”, *Journal of Neural Engineering*, 3:139-144, 2006.

[77] M. Besserve, L. Garnero, and J.Martinerie, “Cross-spectral discriminant analysis (CSDA) for the classification of brain computer interfaces”, *In 3rd International IEEE/EMBS Conference on Neural Engineering*, 375-378, 2007.

[78] V. Bostanov, “BCI competition 2003-data sets Ib and IIb: feature extraction from event-related brain potentials with the continuous wavelet transform and the t-value scalogram”, *IEEE Transactions on Biomedical Engineering*,51(6):1057-1061, 2004.

[79] T. Wang, J. Deng, and B. He, “Classifying EEG-based motor imagery tasks by means of time-frequency synthesized spatial patterns”, *Clinical Neurophysiology*, 115(12):2744-2753, 2004.

[80] B. Obermeier, C. Guger, C. Neuper, and G. Pfurtscheller, “Hidden markov models for online classification of single trial EEG”, *Pattern recognition letters*, 1299-1309, 2001.

[81] D. Coyle, G. Prasad, and T. M. McGinnity, “A time-frequency approach to feature extraction for a brain-computer interface with a comparative analysis of performance measures”, *EURASIP J. Appl. Signal Process.*, 2005(1):3141- 3151, 2005.

[82] V.J. Samar, A. Bopardikar, R. Rao, and K. Swartz, “Wavelet analysis of neuroelectric waveforms: A conceptual tutorial”, *Brain and Language*, 66(1):7-60, 1999.

[83] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed., John Wiley, New York, 2001.

[84] C. J. C. Burges, “A tutorial on support vector machines for pattern recognition”, *Knowledge Discovery and Data Mining*, 2:121-167, 1998.

[85] A. Rakotomamonjy and V. Guigue, “BCI competition III: Dataset II - ensemble of SVMs for BCI P300 speller”, *IEEE Trans. Biomedical Engineering*, 55(3):1147-1154, 2008.

[86] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1996.

[87] T. Kohonen, “The self-organizing map”, *In Proceedings of the IEEE*, 78, 1464-1480, 1990.

[88] R. Palaniappan, R. Paramesran, S. Nishida, and N. Saiwaki, “A new brain computer interface design using fuzzy ARTMAP”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 10:140-148, 2002.

[89] E. Haselsteiner and G. Pfurtscheller, “Using time-dependant neural networks for EEG classification”, *IEEE transactions on rehabilitation engineering*, 8:457-463, 2000.

[90] T. Felzer and B. Freisieben, “Analyzing EEG signals using the probability estimating guarded neural classifier”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(4):361- 371, 2003.

[91] G. Matthews, D.R. Davies, S.J. Westerman, and R.B.Stammers, “Human Performance: Cognition”, *Stress and individual differences*, Psychology Press, 2000.

[92] R. Fredlund, R. M. Everson, & J. E. Fieldsend, “A Bayesian Framework for Active Learning”, *IEEE World Congress on Computational Intelligence*, 2010.

[93] L. R. Rabiner, “A tutorial on hidden markov models and selected applications in speech recognition”, *In Proceedings of the IEEE*,77, 257- 286, 1989.

[94] R. Sitaram, H. Zhang, C. Guan, M. Thulasidas, Y. Hoshi, A. Ishikawa, K. Shimizu, and N. Birbaumer, “Temporal classification of multichannel near-infrared spectroscopy signals of

motor imagery for developing a brain-computer interface", *NeuroImage*, 34(4), 1416-1427, 2007.

[95] A. Schlogl, C. Vidaurre, and K.-R. Mller, "Adaptive Methods in BCI Research - An Introductory Tutorial", *In: B.Graimann, B.Allison, G. Pfurtscheller, Brain-Computer Interfaces Revolutionizing Human-Computer Interaction*, Springer, 331-355, 2010.

[96] S. Solhjoo, A. M. Nasrabadi, and M. R. H. Golpayegani, "Classification of chaotic signals using HMM classifiers: EEG-based mental task classification", *In Proceedings of the European Signal Processing Conference*, 2005.

[97] C. Vidaurre and B. Blankertz, "Towards a cure for BCI illiteracy", *Brain Topogr.*, 23:194-198, 2010.

[98] R. Tomioka, K-R. Mueller , "A regularized discriminative framework for EEG analysis with application to brain-computer interface", *NeuroImage*, 2009.

[99] S.Haykin, *Neural Networks: A Comprehensive Foundation*, Ed.2, Prentice Hall, 1998.

[100] J.Kennedy, R.C. Eberhart , "Particle swarm optimization", *In: Proceedings of the IEEE international conference on neural networks*, New Jersey, 1942-1948, 1995.

[101] P. Sajda, A. Gerson, K.-R. Muller, B. Blankertz, and L. Parra, "A data analysis competition to evaluate machine learning algorithms for use in brain-computer interfaces", *IEEE Trans. Neural Syst. Rehab. Eng.*, vol. 11, pp. 184-185, June 2003.

[102] L.I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*, Hoboken, NJ, Wiley, 2004.

[103] J. Zhu, S. Rosset, H. Zou, and T. Hastie, "Multi-class Adaboost", *Ann Arbor*, 1001(48109), 1612, 2006.

[104] K. K. Ang, Z. Y. Chin, and C. Guan, “Filter Bank Common Spatial Pattern (FBCSP) in Brain-Computer Interface”, *in Proceedings of the International Joint Conference on Neural Networks, (IJCNN 2008)*, 2391-2398,2008.

[105] M. Krauledat, M. Schrder, B. Blankertz, K-R.Mueller, “Reducing Calibration Time For Brain-Computer Interfaces: A Clustering Approach” , In B. Schlkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pp. 753-760, Cambridge, MA: MIT Press,2007.

[106] J. Quinonera-Candela, M. Sugiyama, A. Schwaighofer, N.D. Lawrence, , *Dataset Shift in Machine Learning*, MIT Press 2009.

[107] L. Kuncheva, “Using Diversity Measures for Generating Error Correcting Output Codes in Classifier Ensembles”, *Pattern Recognition Letters*, 26(1), 83-90, 2005.

[108] L.Kuncheva, “Diversity in Multiple Classifier Systems” *Information Fusion*, 6(1), 3-4, 2005.

[109] N. Chawla, T. Moore, L. Hall, L. Bowyer, P. Kegelmeyer, and C. Springer, “Distributed Learning with Bagging-like Performance”, *Pattern Recognition Letters*, 24, 455-471, 2003.

[110] K. Woods, W. Kegelmeyer, and K. Bowyer, “Combination of Multiple Classifiers Using Local Accuracy Estimates”, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(4), 405-410, 1997.

[111] T. Dietterich, “An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization”, *Machine Learning*, 40(2), 139-157, 2000.

[112] Y. Freund and R. Schapire, “Experiments with a New Boosting Algorithm”, *Proc. 13th Intl Conf. Machine Learning*, 148-156, 1996.

[113] Y. Freund and R.E.Schapire, “A decision-theoretic generalization of on-line learning and application to boosting”, *Journal of Computer and Systems Sciences*, 55(1):119-139,1997.

[114] D. Frosyniotis, A. Stafylopatis, and A. Likas, “A Divide and Conquer Method for Multi Net Classifiers”, *Pattern Analysis and Applications*, 6, 32-40, 2002.

[115] N. X. Vinh, J. Epps, and J. Bailey, “Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance” , *Journal of Machine Learning Research* 11, 2837-2854, 2010.

[116] N. X. Vinh, J. Epps, and J. Bailey, “Information theoretic measures for clusterings comparison: Is a correction for chance necessary”, *In ICML*, 2009.

[117] A. Arnold, R. Nallapati, and W.W. Cohen, “A Comparative Study of Methods for Transductive Transfer Learning”, *Proc. Seventh IEEE Intl Conf. Data Mining Workshops*, 77-82, 2007.

[118] W. Tu, S. Sun, “A subject transfer framework for EEG classification”, *Neurocomputing*, 1-11, 2011.

[119] G Pfurtscheller, J Kalcher, C. Neuper D. Flotzinger, and M. Pregenzer, “On-line EEG classification during externally-paced hand movements using a neural network-based classifier”, *Electroencephalogr Clin Neurophysiol*, 99(5), 416-25, 1996.

[120] J. McFarland, A.T. Lefkowicz, and J.R. Wolpaw, “Design and operation of an EEG-based brain-computer interface with digital signal processing technology”, *Behav. Res. Methods*, 29, 337-345, 1997.

[121] N. Birbaumer, A. Kubler, N. Ghanayim, T. Hinterberger, J. Perelmouter, J. Kaiser, I. Iversen, B. Kotchoubey, N. Neumann, and H. Flor, The thought-translation device (TTD): Neurobehavioral mechanisms and clinical outcome. IEEE Trans Neural Syst Rehabil Eng, 11(2), 120-123, 2003.

[122] G. Dornhege, B. Blankertz, G. Curio, and K.-R. Mueller, "Combining features for BCI", *In S. Becker, S. Thrun, and K. Obermayer (Eds.), Advances in neural information processing systems*, MIT Press, Cambridge, MA, 1115-1122, 2003.

[123] G. Dornhege, B. Blankertz, M. Krauledat, F. Losch, G. Curio, and K.-R. Mueller, "Combined optimization of spatial and temporal filters for improving braincomputer interfacing", *IEEE Trans Biomed Eng*, 53(11), 2274-2281, 2006.

[124] S. Lemm, B. Blankertz, G. Curio, and K.-R. Mueller, "Spatio-spectral filters for robust classification of single-trial EEG", *IEEE Trans Biomed Eng*, 52(9), 1541-48, 2005.

[125] C. Vidaurre, A. Schlogl, R. Cabeza, R. Scherer, and G. Pfurtscheller, "A fully online adaptive BCI", *IEEE Trans Biomed Eng*, 53, 1214-1219, 2006.

[126] C. Vidaurre, A. Schlogl, R. Cabeza, R. Scherer, and G. Pfurtscheller , "Study of online adaptive discriminant analysis for EEG-based brain computer interfaces", *IEEE Trans Biomed Eng.*, 54, 550-556, 2007.

[127] J.Wackermann, "Towards a quantitative characterization of functional states of the brain: from the non-linear methodology to the global linear descriptor", *Int J Psychophysiol*, 34, 65-80, 1999.

[128] J.R.Wolpaw, N.Birbaumer, W.J.Heetderks, D.J. McFarland, P.H. Peckham, G.Schalk, E. Donchin, L.a. Quatrano, C.J.Robinson, and T.M.Vaughan, "Brain-computer interface tech-

nology: A review of the first international meeting", *IEEE Transactions on Rehabilitation Engineering*, 8, 2000, 164-173.

[129] C.Neuper and G. Pfurtscheller, "Motor Imagery and ERD", *In: G.Pfurtscheller and F.H.Lopes da Silva (eds.), Event-Related Desynchronization. Handbook of Electreencephalography and Clinical Neurophysiology*, Amsterdam:Elsevier, 303-325, 1999 .

[130] G. Pfurtscheller and D.J. McFarland, "BCIs that use Sensorimotor Rhythms", *In: J.R.Wolpaw and E.W.Wolpaw (eds.), Brain-Computer Interfaces Principles and Practice*, Oxford University Press, New York, 227-240, 2012.

[131] J. Pascual, C. Vidaurre, and M. Kawanabe, "Investigating EEG nonstationarities with robust PCA and its application to improve BCI performance", *International Journal of Bioelectromagnetism*, 13, 2011, 50-51.

[132] J. Pascual, M. Kawanabe, and C. Vidaurre, "Modelling non-stationarities in EEG data with Robust Principal Component Analysis", *Hybrid Artificial Intelligent Systems*, 6679, Springer Berlin / Heidelberg, 51-58, 2011.

[133] M. Kawanabe, W. Samek, P. von Bunau, and F. Meinecke, "An Information Geometrical View of Stationary Subspace Analysis", *In T. Honkela, W. Duch, M.Girolami, and S. Kaski, (eds), Artificial Neural Networks and Machine Learning*, Springer Berlin / Heidelberg, 397-404, 2011.

[134] P. von Bunau, F.C. Meinecke, S. Scholler, K.-R.Mueller, "Finding stationary brain sources in EEG data", *In Proceedings of the 32nd Annual Conference of the IEEE EMBS*, 2810-2813, 2010.

[135] S.J. Pan, I.W. Tsang, J.T. Kwok, Q.Yang, "Domain Adaptation via Transfer Component Analysis. IEEE Transactions on Neural Networks", 22(2), 199-210, 2011.

[136] J. Mueller-Gerking, G. Pfurtscheller, and H. Flyvbjerg, "Designing optimal spatial filters for single-trial EEG classification in a movement task", *Clinical Neurophysiology*, 110(5), 787-798, 1999.

[137] Z. J. Koles, "The quantitative extraction and topographic mapping of the abnormal components in the clinical EEG", *Electroencephalogr. Clin. Neurophysiol.*, 79, 440-447, 1991.

[138] K. K. Ang, Z.Y. Chin, C.Wang, C.T.Guan and H.H. Zhang, "Filter Bank Common Spatial Pattern algorithm on BCI Competition IV Datasets 2a and 2b", *Frontiers in Neuroscience*, 6, 2012.

[139] I. Steinwart, "On the influence of the kernel on the consistency of support vector machines", *Journal of Machine Learning Research*, 2, 67-93, 2001.

[140] S. J. Pan, J. T. Kwok, and Q. Yang, "Transfer learning via dimensionality reduction", *In Proceedings of AAAI*, Illinois, USA, 677-682, 2008.

[141] M. Tangermann, K.R. Mueller, , A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K.J. Miller, G. Mller-Putz, G. Nolte, G. Pfurtscheller, H. Preissl, G. Schalk, A. Schlgl, C. Vidaurre, S. Waldert, and B. Blankertz, "Review of the BCI competition IV", *Frontiers in Neuroprosthetics*, 6:55, 2012.

[142] B. Blankertz, "BCI Competition IV", Fraunhofer FIRST.IDA, http://ida.first.fraunhofer.de/projects/bci/competition_iv/.

[143] A. Soria-Frisch, "A Critical Review on the Usage of Ensembles for BCI." *Towards Practical Brain-Computer Interfaces*, 41-65, 2012.

[144] F.P. Lotte, M. Congedo, A. Lecuyer, F. Lamarche, B. Arnaldi, “A review of classification algorithms for EEG-based brain-computer interfaces”, *J. Neural Eng.*, 4(2), R1-R13, 2007.

[145] M.A. Abidi, R.C. Gonzalez, *Data fusion in robotics and machine intelligence*, Academic Press, San Diego, CA, USA, 1992.

[146] O. Alzoubi, I. Koprinska, R.A. Calvo, “Classification of Brain-Computer Interface Data”, *In: Proc. 7th Australasian Data Mining Conference*, 123-132, 2008.

[147] A. Barbosa, D. Diaz, M. Vellasco, M.Meggiolaro, R. Tanscheit, “Mental Tasks Classification for a Noninvasive BCI Application”, In: Alippi, C., Polycarpou M., Panayiotou, C., Ellinas G (eds) *Artificial Neural Networks , Lecture Notes in Computer Science*, 5769(50), 495-504, Springer, Berlin/Heidelberg, 2009.

[148] G. Beliakov, A. Pradera, T. Calvo, “Aggregation Functions: A Guide for Practitioners”, *Studies in Fuzziness and Soft Computing*, 1st edn., Springer, Berlin/Heidelberg, 2008.

[149] A.V.Bogdanov, “Neuroinspired architecture for robust classifier fusion of multisensor imagery”, *IEEE Trans. Geosci. Remote Sens.*, 46(5), 1467-1487, 2008.

[150] L. Breiman, “Arcing classifiers”, *Ann. Stat.*, 26(3), 801-849, 1998.

[151] L.Breiman, “Random forests”, *Mach. Learn.*, 45(1), 5-32, 2001.

[152] I. Cester, A. Soria-Frisch, “Comparison of Feature Stages in a multi-classifier BCI”, *In Proc. 5th International Brain-Computer Interface Conference*, Graz, 2011.

[153] D. Coyle, “Neural network based auto association and time-series prediction for biosignal processing in brain-computer interfaces”, *IEEE Comput. Intell. Mag.*, 4(4), 47-59, 2009.

[154] B. Dasarathy, B. Sheela, “A composite classifier system design: Concepts and methodology”, *Proc. IEEE*, 67(5), 708-713, 1979.

[155] S. Fazli, C. Grozea, M. Danoczy, B. Blankertz, K-R. Mueller, F. Popescu, "Ensembles of temporal filters enhance classification performance for ERD-based BCI systems", *In: Proc. 4rd International Brain-Computer Interface Workshop and Training Course*, 247-253, 2008.

[156] S. Fazli, C. Grozea, M. Danoczy, B. Blankertz, F. Popescu, K-R. Mueller, "Subject independent EEG-based BCI decoding", *Advances in Neural Information Processing Systems*, 22, 513-521, 2009.

[157] S. Fazli, F. Popescu, M. Danoczy, B. Blankertz, K.-R. Mueller, C. Grozea, , "Subject independent mental state classification in single trials", *Neural Netw.*, 22(9), 1305-1312, 2009.

[158] P. Geurts, *Contributions to decision tree induction: bias/variance trade off and time series classification*, PhD thesis, University of Liege 2002.

[159] M. Grabisch, H.T. Nguyen, E.A. Walker, *Fundamentals of Uncertainty Calculi with Applications to Fuzzy Inference*, 1st edn., Kluwer Academic Publishers, Dordrecht, 1994.

[160] P.S. Hammon and V.R.de Sa, "Preprocessing and meta-classification for brain-computer interfaces", *IEEE Trans. Biomed. Eng.*, 54(3), 518-525 2007.

[161] P.S.Hammon, S. Makeig, H. Poizner, E.Todorov, and V.R.de Sa, "Predicting reaching targets from human EEG", *IEEE Signal Process. Mag.*, 25(1), 69-77, 2008.

[162] A.K. Jain, B. Chandrasekaran, "39 Dimensionality and sample size considerations in pattern recognition practice", *Handbook Stat.2*, 835-855, 1982.

[163] A.K. Jain, R.P.W. Duin, and J. Mao, "Statistical pattern recognition: a review", *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(1), 4-37, 2000.

[164] G.D. Johnson, D.J. Krusienski, "Ensemble SWLDA Classifiers for the P300 Speller", *In: Proc. 13th International Conference on Human-Computer Interaction. Part II: Novel Interaction Methods and Techniques*, 551-557. Springer, Berlin, Heidelberg, 2009.

[165] A. Kachenoura, L. Albera, L. Senhadji, P. Comon, "ICA: a potential tool for BCI systems", *IEEE Signal Process. Mag.*, 25(1), 57-68, 2008.

[166] J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas, "On combining classifiers", *IEEE Trans. Pattern Anal.Mach. Intell.*, 20(3), 226-239 1998.

[167] L.I. Kuncheva, "Fuzzy versus nonfuzzy in combining classifiers designed by boosting", *IEEE Trans. Fuzzy Syst.*, 11(6), 729-741, 2003.

[168] L.I. Kuncheva, "Classifier Ensembles: Facts, Fiction, Faults and Future", *In: Proc. 19th International Conference Pattern Recognition (ICPR)*, 2008.

[169] L.I. Kuncheva, J.C. Bezdek, and R.P.W. Duin, "Decision templates for multiple classifier fusion: an experimental comparison", *Pattern Recogn.*, 34, 299-314, 2001.

[170] L.I. Kuncheva, T. Christy, I. Pierce, and S.P. Mansoor, "Multi-modal Biometric Emotion Recognition using Classifier Ensembles", *In Proc 24th International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems*, 2011.

[171] X. Lei, P. Yang, P. Xu, T.J. Liu, and D.Z. Yao, "Common spatial pattern ensemble classifier and its application in brain-computer interface", *J. Electron. Sci. Tech. China*, 7(1), 17-21, 2009.

[172] R.C. Luo, M.G. Kay, *Multisensor integration and fusion for intelligent machines and systems*, Ablex Publishing Corp., Norwood, NJ, USA, 1995.

[173] J.D. Millan, R. Rupp, G.R. Mueller-Putz, R. Murray-Smith, C. Giugliemma, M. Tangermann, C. Vidaurre, F.Cincotti, A. Kubler, R. Leeb, C. Neuper, K.-R. Mueller, and D. Mattia, “Combining Brain-Computer Interfaces and Assistive Technologies: State-of-the-Art and Challenges”, *Front. Neurosci.*, 4:161, R1-R33, 2010.

[174] K.-R. Mueller, M. Tangermann, G. Dornhege, M. Krauledat, G. Curio, B. Blankertz, “Machine learning for real-time single-trial EEG-analysis: From brain-computer interfacing to mental state monitoring”, *J. Neurosci. Methods*, 167(1), 82-90, 2008.

[175] N. Oza, K. Tumer, “Classifier ensembles: Select real-world applications”, *Inf. Fusion*, 9(1), 4-20, 2008.

[176] L. Parra, C. Christoforou, A. Gerson, M. Dyrholm, A. Luo, M. Wagner, M. Philiastides, and P. Sajda, “Spatiotemporal Linear Decoding of Brain State”, *IEEE Signal. Process. Mag.*, 25(1), 107-115, 2008.

[177] G. Pfurtscheller, D. Flotzinger, J. Kalcher, “Brain-Computer Interface-a new communication device for handicapped persons”, *J.Microcomputer Appl.*, 16(3), 293-299, 1993.

[178] R. Polikar, “Ensemble based systems in decision making”, *IEEE Circuits Syst.Mag.*, 6(3), 21-45, 2006.

[179] S.Raudys and A. Jain, “Small sample size effects in statistical pattern recognition: recommendations for practitioners”, *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(3), 252-264, 1991.

[180] L. Rokach, “Ensemble-based classifiers”, *Artif. Intell. Rev.*, 33(1), 1-39, 2010.

[181] M. Salvaris, and F. Sepulveda, “Wavelets and ensemble of FLDs for P300 classification”, *In: Proc. 4th International IEEE/EMBS Conference on Neural Engineering*, 339-342, 2009.

[182] C. Sannelli, C. Vidaurre, K.-R. Mueller, B. Blankertz, "CSP patches: an ensemble of optimized spatial filters. An evaluation study", *J. Neural Eng.*, 8(2), 025-37, 2011.

[183] A.J. Sharkey, *Combining Artificial Neural Nets: Ensemble and Modular Multi-Net Systems*, 1st edn., Springer, New York, 1999.

[184] Z.S. Shirehjini, S. Bagheri Shouraki, M. Esmailee, "Variant Combination of Multiple Classifiers Methods for Classifying the EEG Signals in Brain-Computer Interface", In: H. Sarbazi-Azad, B. Parhami, S.G. Miremadi, S. Hessabi, (eds.), *Advances in Computer Science and Engineering, Communications in Computer and Information Science*, 6(59), Springer, Berlin, Heidelberg, 477-484, 2009.

[185] M. Skurichina, and R.P.W. Duin, "Bagging, boosting and the random subspace method for linear classifiers", *Pattern Anal. Appl.*, 5(2), 121-135, 2002.

[186] A. Soria-Frisch, A. Riera, S. Dunne, "Fusion operators for multi-modal biometric authentication based on physiological signals", *In: Proc. 2010 IEEE International Conference on Fuzzy Systems*, 1-7, 2010.

[187] S. Sun, C. Zhang, and D. Zhang, "An experimental evaluation of ensemble methods for EEG signal classification", *Pattern Recog. Lett.*, 28(15), 2157-2163, 2007.

[188] S. Sun, C. Zhang, Y. Lu, "The random electrode selection ensemble for EEG signal classification", *Pattern Recogn.*, 41, 1680-1692, 2008.

[189] A. Vallabhaneni, T. Wang, and B. He, "Brain-computer interface", In: He, B., He, B. (eds.) *Neural Engineering, Bioelectric Engineering*, Springer, US, 85-121, 2005.

[190] J.R. White, T. Levy, W. Bishop, J.D. Beaty, "Real-time decision fusion for multimodal neural prosthetic devices", *PLoS ONE*, 5(3), 2010.

[191] D.H.Wolpert, “Stacked generalization”, *Neural Netw.*, 5, 241-259, 1992.

[192] W. Wu, X. Gao, S. Gao, “One-Versus-the-Rest(OVR) Algorithm: An Extension of Common Spatial Patterns(CSP) Algorithm to Multi-class Case”, *In: Proc. 27th Annual International Conference of the EMBS, IEEE*, 2387-2390, 2005.

[193] K. Fukunaga, *Introduction to statistical pattern recognition*, 2nd ed., Academic Press, San Diego, CA, USA, 1990.

[194] M. Grosse-Wentrup and M. Buss, “Multiclass Common Spatial Patterns and Information Theoretic Feature Extraction”, *IEEE Transactions on Biomedical Engineering*, 55(8), 1991-2000, 2008.

[195] J.-F. Cardoso and A. Souloumiac. “Jacobi angles for simultaneous diagonalization”, *SIAM journal on matrix analysis and applications*, 17(1):161-164, 1996.

[196] M. Joho and K. Rahbar, “Joint diagonalization of correlation matrices by using Newton methods with application to blind signal separation”, *In Proceedings of IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*, 403-407, 2002.

[197] A. Ziehe, P. Laskov, G. Nolte, and K-R. Mueller , “A Fast Algorithm for Joint Diagonalization with Non-orthogonal Transformations and its Application to Blind Source Separation”, *Journal of Machine Learning Research*, 5, 777-800, 2004.

[198] G. Pfurtscheller,C. Guger,G.Mueller,G. Krausz, and C. Neuper, “Brain oscillations control hand orthosis in a tetraplegic”, *Neuroscience Letters*, 292(3), 211-214, 2000.

[199] A.Bunse-Gerstner, R. Byers, and V. Mehrmann, “Numerical methods for simultaneous diagonalization”, *SIAM journal on matrix analysis and applications* , 14, 927-949,1993.

[200] D. Gribkov and V. Gribkova, "Learning Dynamics from Non-stationary Time Series: Analysis of Electroencephalograms", *Physical Review E*, 61(6), 6538-6545, 2000.

[201] A.Y. Kaplan, A.A. Fingelkurts, S.V. Borisov, and B.S. Darkhovsky, "Nonstationary nature of the brain activity as revealed by EEG/MEG: methodological, practical and conceptual challenges", *Signal Process.*, 85(11), 2190-2212, 2005.

[202] J. Pascual, C. Vidaurre, and M. Kawanabe, "Investigating EEG non- stationarities with robust PCA and its application to improve BCI performance", *International Journal of Bioelectromagnetism*, 13, 50-51, 2011.

[203] J. Pascual, M. Kawanabe, and C. Vidaurre, "Modelling non-stationarities in EEG data with Robust Principal Component Analysis", *Hybrid Artificial Intelligent Systems*, 6679/2011, Springer Berlin / Heidelberg, 51-58, 2011.

[204] M. Kawanabe, W. Samek,P. von Bünau, and F. Meinecke, "An Information Geometrical View of Stationary Subspace Analysis", In T. Honkela, W. Duch, M.Girolami, and S. Kaski, editors, *Artificial Neural Networks and Machine Learning - ICANN 2011*, 397-404, Springer Berlin / Heidelberg, 2011.

[205] T. Hastie and R. Tibshirani, "Discriminant Adaptive Nearest Neighbour Classification" *Journal of the Royal Statistical Society. Series B*, 58(1) , 155-176, 1996.

[206] A.P. Dawid, "Applications of a general propagation algorithm for probabilistic expert systems", *Statistics and Computing* 2, 25-36, 1992.

[207] Z. Ghahramani, and H.C. Kim, *Bayesian classifier combination*, University College London, 2003.

[208] H.Erdogan, and M.U. Sen, "A unifying framework for learning the linear combiners for classifier ensembles" , *Proceeding of 20th International Conference on Pattern Recognition (ICPR)* , Istanbul, Turkey, 2010.

[209] E. Simpson, S.J. Roberts, A. Smith and C. Lintott, "Bayesian Combination of Multiple, Imperfect Classifiers", In *Neural Information Processing Systems Foundation (NIPS)* , Spain, 2011.

[210] H. Ramoser, J. Mueller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial EEG during imagined hand movement", *IEEE Trans. Rehab. Eng.*, 8(4), 441-446, 2000.

[211] G.Dornhege, J. del R. Millan, T. Hinterberger, D.McFarland, K.-R. Mueller, *Towards Brain-Computer Interfacing*, Cambridge,MA: MIT Press, 2007.

[212] J. Havrda and F. Charvat, "Quantification method of classification processes. concept of structural entropy", *Kybernetika*, 3, 30-35, 1967.

[213] E. Parzen, "On estimation of a probability density function and mode", *Annals of Mathematical Statistics*, 33(3), 1065-1076, 1962.

[214] M. Rosenblatt, "Remarks on some non-parametric estimates of a density function", *Annals of Mathematical Statistics*, 27(3), 832-837, 1956.

[215] J. C. Principe, D. Xu, and J. Fisher, "Information theoretic learning", *Unsupervised adaptive filtering*,1, 265-319, 2000.

[216] F. Lotte, M. Congedo, A. Lcuyer, F. Lamarche, and B. Arnaldi, "A Review of Classification Algorithms for EEG-based Brain-Computer Interfaces", *Journal of Neural Engineering*, 4, R1-R13, 2007.

[217] R.A.Dara, M. Makrehchi,and M.S. Kamel, “Filter-Based Data Partitioning for Training Multiple Classifier Systems”, *IEEE Trans. Knowledge and Data Eng.*, 22(4), 2010.

[218] Nijholt, A. and Tan, D. Brain-computer interfacing for intelligent systems IEEE Intelligent Systems, 2008, 23, 72-79

[219] Y. Li, and C.T. Guan, ”An Extended EM Algorithm for Joint Feature Extraction and Classification in Brain-Computer Interfaces”, *Neural Computation*, vol. 18, no. 11, pp. 2730-2761, 2006.

[220] J. d. R. Millan, A.Buttfield, C. Vidaurre, M.Krauledat, A. Schlogl, P.Shenoy, B. Blankertz, R.Rao, R.Cabeza, G.Pfurtscheller and K. R.Mueller, Dornhege, G.; Millan, J. d. R.; Hinterberger, T.; McFarland, D. and Mueller, K. R. (Eds.) Adaptation in Brain-Computer Interfaces Towards Brain-Computer Interfacing, The MIT Press, 2007

[221] A.Buttfield, P.Ferrez, and J. d. R. Millan, Towards a robust BCI: error recognition and online learning IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2006, 14, 164-168

[222] A. Lenhardt, M. Kaper, and H. Ritter, “An adaptive P300-based online brain-computer interface”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16, 1-11, 2008.

[223] B. Blankertz, M. Kawanabe, R. Tomioka, F. Hohlefeld, V. Nikulin, and K.-R. Mueller, “Invariant common spatial patterns: Alleviating nonstationarities in brain-computer interfacing”, *Advances in Neural Information Processing Systems*, MIT Press, Cambridge, MA, 113-120, 2008.

[224] B. Blankertz, G. Dornhege, C. Schafer, R. Krepki, J. Kohlmorgen, K.-R. Mueller, V. Kunzmann, F. Losch, and G. Curio, "Boosting Bit Rates and Error Detection for the Classification of Fast-Paced Motor Commands Based on Single-Trial EEG Analysis", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11, 127-131, 2003.

[225] G. Schalk, J. Wolpaw, D. McFarland, and G. Pfurtscheller, "EEG-based communication: presence of an error potential Clinical Neurophysiology", 111, 2138-2144, 2000.

[226] Y.Li, H. Kambara, Y. Koike, and M. Sugiyama, "Application of Covariate Shift Adaptation Techniques in Brain-Computer Interfaces", *IEEE Transactions on Biomedical Engineering*, 2010, 57, 1318-1324.

[227] H. Zhang, and C. Guan, "A maximum mutual information approach for constructing a 1D continuous control signal at a self-paced brain computer interface", *Journal of Neural Engineering*, 7, 056009, 2010.

[228] S.G. Mason, and G.E. Birch, "A brain-controlled switch for asynchronous control applications", *IEEE Transactions on Rehabilitation Engineering*, 47, 1297-1307, 2000.

[229] H. Zhang, C. Guan, and C. Wang, "Asynchronous P300-based brain-computer Interfaces: A Computational Approach with Statistical Models", *IEEE Transactions on Biomedical Engineering*, 55, 1754-1763, 2008.

[230] F. Galan, M. Nuttin, E. Lew, P. Ferrez, G. Vanacker, J. Philips, and J.R. Milln, "A Brain-Actuated Wheelchair: Asynchronous and Non-Invasive brain-computer Interfaces for Continuous Control of Robots", *Clinical Neurophysiology*, 119, 2159-2169, 2008.

[231] H. Zhang, C. Guan, and C. Wang, “Spatio-spectral feature selection based on robust mutual information estimate for brain-computer interfaces”, *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2391-2398, 2009.

[232] H. Zhang, Z.Y. Chin, K.K. Ang, C. Guan, and C. Wang, “Optimum Spatio-Spectral Filtering Network for Brain”, *Computer Interface IEEE Transactions on Neural Networks*, 22, 52-63, 2011.

[233] M. Fatourechi, A. Bashashati, R. K. Ward, and G. E. Birch. “EMG and EOG artifacts in brain computer interface systems: A survey”, *Clinical neurophysiology*, 118(3), 480-494, 2007.

[234] A. Kubler, F. Nijboer, J. Mellinger, T.M. Vaughan, H. Pawelzik, G. Schalk, D.J. McFarland, N. Birbaumer, and J.R. Wolpaw, “Patients with ALS can use sensorimotor rhythms to operate a brain-computer interface”, *Neurology*, 64, 1775-1777, 2005.

[235] T.M. Cover, and J.A. Thomas, *Elements of Information Theory*, Wiley, New York, 2006

[236] S. Petridis, and S. Perantonis,“On the relation between discriminant analysis and mutual information for supervised linear feature extraction”, *Pattern Recognition*, 37, 857-874, 2004.

[237] M. Ben-Bassat, P. Krishnaiah, and L. Kanal, “Uses of distance measures, information measures and error bounds in feature evaluation”, *Handbook of Statistics*, North-Holland, Amsterdam, 773-791, 1982.

[238] M. Last, A. Kander, and O. Maimon, “Information-theoretic algorithm for feature selection”, *Pattern Recognition Letters*, 22, 799-811, 2001.

[239] J. Sotoca, and F. Pla, "Supervised feature selection by clustering using conditional mutual information-based distances", Pattern Recognition, 43, 2068-2081, 2010.

[240] P. Estevez, M. Tesmer, C. Perez, and J. Zurada, "Normalized Mutual Information Feature Selection", *IEEE Transactions on Neural Networks*, 20, 189-201, 2009.

[241] N. Kwak, and C.-H. Choi, "Input feature selection by mutual information based on Parzen window", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, 1667 - 1671, 2002.

[242] A.W. Bowman, and A. Azzalini, *Applied Smoothing Techniques for Data Analysis: The Kernel Approach with S-Plus Illustrations*, Oxford University Press, 1997.

[243] C.-C. Chang, and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001.

[244] A.Y. Kaplan, S.L. Shishkin "Application of the change-point analysis to the investigation of the brain?s electrical activity", *Nonparametric statistical diagnosis: Problems and methods*, 333-388, 2000.

[245] G.L. Grinblat, L. C. Uzal, H. A. Ceccatto, and P. M. Granitto, "Solving nonstationary classification problems with coupled support vector machines", *IEEE Transactions on Neural Networks* 22(1), 37-51,2011.

[246] J. Liu, J. Li, W. Xu, and Y. Shi. "A weighted i Lq adaptive least squares support vector machine classifiers?Robust and sparse approximation." Expert Systems with Applications 38(3) 2253-2259, 2011.

[247] D.Erdogmus, and W. Liu. "Adaptive Information Filtering with Error Entropy and Error Correntropy Criteria", *Information Theoretic Learning* 103-140,2010.

[248] A. Singh and J. C. Principe, “Information theoretic learning with adaptive kernels”, *Signal Proc.* , 91, 203-213, 2011.

[249] Z. Minchev, G. Dukov, and S. Georgiev. ”EEG Spectral Analysis in Serious Gaming: An Ad Hoc Experimental Application.” International Journal BioAutomation 14.4, 79-88. 2009.

[250] L.R.Hochberg and K.D.Anderson, ?BCI Users and Their Needs?, In: J.R.Wolpaw and E.W.Wolpaw (eds.), Brain-Computer Interfaces Principles and Practice, Oxford University Press, New York, 227-240, 2012.

[251] F.T. Sun, M.J. Morrell, R.E.Wharen. Responsive cortical stimulation for the treatment of epilepsy. Neurotherapeutics 5:68?74, 2008.

[252] G.P.Topulos, R.W. Lansing, R.B. Banzett. The experience of complete neuromuscular blockade in awake humans. J Clin Anesth 5:369?374. 1993.

Author’s Publications

Journal Papers

- S.R.Liyanage, C.T. Guan, H.H. Zhang, K.K.Ang, J-X. Xu and T.H.Lee “Dynamically Weighted Ensemble Classification with Clustering for Non-Stationary EEG Processing”, *J. Neural Eng.*, vol.10, no.3, 036007, 2013.
- H.H. Zhang, S.R.Liyanage, C.C.Wang and C.T. Guan, “Learning from feedback training data at a self-paced braincomputer interface”, *J. Neural Eng.*, vol.8,no.4, 046035, 2011.

Conference Papers

- S. R. Liyanage, C. T. Guan, H.H.Zhang, K. K. Ang, J. -X. Xu, and T. H. Lee, “Error Entropy based Adaptive Kernel Classification for Non-stationary EEG Analysis”, *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, 26-31 May 2013.
- S. R. Liyanage, J.S.Pan, H.H.Zhang, C. T. Guan, K. K. Ang, J. -X. Xu, and T. H. Lee, “Stationary Transfer Component Analysis for Brain Computer Interfacing”, *IASTED Conference on Engineering and Applied Science*, December 2012, Colombo, Sri Lanka.

  (Best Student Paper Award Winner at IASTED EAS 2012, Colombo)

- S. R. Liyanage, C. T. Guan, H.H.Zhang, K. K. Ang, J. -X. Xu, and T. H. Lee, "Dynamically Weighted Classification with Clustering to Tackle Non-stationarity in Brain Computer Interfacing", *International Joint Conference on Neural Networks (IJCNN)*, Brisbane, 2012.

- S. R. Liyanage, J. -X. Xu, C. T. Guan, K. K. Ang, and T. H. Lee, "Multi-Class Motor Motion Imagery Using Common Spatial Patterns Based On Joint Approximate Diagonalization", *12th IASTED Conference on Control and Automation* , July 2010, Banff, Canada.

- S. R. Liyanage, J. -X. Xu, C. T. Guan, K. K. Ang, and T. H. Lee, "EEG Signal Separation for Multi-Class Motor Imagery using Common Spatial Patterns Based on Joint Approximate Diagonalization", *International Joint Conference on Neural Networks (IJCNN)*, Barcelona, 2010.

- S. R. Liyanage, J. -X. Xu, C. T. Guan, K. K. Ang, and T. H. Lee,"Classification of Self-paced Finger Movements with EEG Signals Using Neural Network and Evolutionary Approaches", *IEEE International Conference on Control and Automation (ICCA)*, Christchurch, New Zealand, 2009.