Adaptation and Control State Detection Techniques for Brain-Computer Interfaces

RAJESH CHANDRASEKHARA PANICKER

(Bachelor of Technology, University of Kerala)

A THESIS SUBMITTED
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE
2011

Acknowledgements

This thesis is a product of time and effort invested by a number of people, though I am mentioned to be the author. A non-exhaustive list is given below.

- My supervisors Prof. Sadasivan Puthusserypady and Dr. Sun Ying, for providing me with all the necessary support and guidance, for being very patient with me, and offering me a helping hand every time I stumbled. Their wealth of experience and insight has help me tide through many difficult situations.
- All my teachers, past and present. If I have seen a little further, it is by standing on their shoulders.
- Ananda for help in setting up and programming the BCI system.
- My thesis committee members Dr. Yen and Prof. Dipti for their advice and encouragement.
- Dr. Akash, Prof. Ashraf, Dr. Sahoo, Prof. Loh whom I worked with for the modules I tutored, and who chipped in with help, advice and support.

- Dr. Guan Cuntai, Yasamin, Omer, Roger, Jit Hon and Roshan for helpful and encouraging discussions.

- Lab officers Mdm. Chia, Fook Mun, Victor, King Hock and Francis who never hesitated to help whenever a need arose.

- All my friends who not only supported me all the way, but also volunteered to be subjects, whenever the need arose. This thesis wouldn't have materialized without their help.

- My dearest friends Yen, Abhilash, Vineesh, Deepu, Krishna, Kalesh, Jing, Rahul, Huaien, Tianfang, Khanh and Vasanth.

- NUS for supporting for providing me the financial support through Research Scholarship and Teaching Assistantship.

- Dennis Ritchie, the genius who passed away this year, for creating the wonderful programming language C, the derivative of which (C++) was used in programming our BCI system.

- My grandmother, parents, sisters, brother-in-law and relatives for their unconditional love and support.

Summary

A brain computer interface (BCI) is an alternate channel of communication between the user and the computer, without having to go through the usual neuromuscular pathways. Using BCI, disabled patients can communicate with a computer or control a prosthetic device just by modulating his/her brain activity. This thesis focuses on two of the desirable capabilities of a usable and practical BCI system - adaptation and control state detection. Adaptation is the ability of the BCI system to adapt itself to incoming data to achieve goals such as higher information transfer rate and lower training data requirement as compared to a non-adaptive system. Control state detection refers to its ability to determine whether the user is actively giving input. Such systems eliminate the need to follow the cues issued by the computer, and allows the user to give input naturally (at will). However, adaptation and control state detection are challenging tasks, and require the BCI system to be able to extract more information from the data being classified.

A co-training based approach is introduced for constructing high-performance classifiers for BCIs based on the P300 event-related potential (ERP), which were trained from very little data. It uses two classifiers - Fisher's linear discriminant analysis (FLDA) and Bayesian linear discriminant analysis (BLDA), progressively

teaching each other to build a final classifier, which is robust and able to learn effectively from unlabeled data. Detailed analysis of the performance is carried out through extensive cross-validations, and it is shown that the proposed approach is able to build high-performance classifiers from just a few minutes of labeled data and by making efficient use of unlabeled data. The performance improvement is shown to be even more significant in cases where the training data as well as the number of trials that are averaged for detection of a character is low, both of which are desired operational characteristics of a practical BCI system. Moreover, the proposed method outperforms the self-training-based approaches where the confident predictions of a classifier is used to retrain itself.

An asynchronous BCI system combining P300 and steady-state visually evoked potentials (SSVEP) paradigms is also proposed. The information transfer is accomplished using P300 ERP and the control state detection is achieved using SSVEP, overlaid on the P300 base system. Offline and online experiments have been performed with ten subjects to validate the proposed system. It is shown to achieve fast and accurate control state detection without significantly compromising the performance. Techniques for improving the performance of the proposed techniques are also suggested.

Contents

List of Abbreviations

ADC	Analog to Digital Converter
ALS	Amyotrophic Lateral Sclerosis
AUC	Area Under Curve (ROC)
BCI	Brain-computer Interface
BCI	Brain Computer Interface
BLDA	Bayesian Linear Discriminant Analysis
BOLD	Blood Oxygenation Level Dependent
CA	Classification Accuracy
CBLDA	Co-training Bayesian Linear Discriminant Analysis
CCA	Canonical Correlation Analysis
CD	Control state Detection
CLDA	Co-training Linear Discriminant Analysis
CS	Control State
CSP	Common Spatial Patterns
ECoG	Electrocorticogram
EEG	Electroencephalogram
EOG	Electrooculogram

ERD	Event Related Desynchonization
ERP	Event Related Potential
ErrP	Error related Potential (or Error Potential)
FFT	Fast Fourier Transform
FLDA	Fisher's Linear Discriminant Analysis
fMRI	functional Magnetic Resonance Imaging
FPR	False Positive Rate
GA	Genetic Algorithm
GMM	Gaussian Mixture Model
HDR	HaemoDynamic Response
HMM	Hidden Markov Model
ICA	Independent Component Analysis
ICG	Inter-Character Gap
IP	Internet Protocol
ISI	Inter-Stimulus Interval
ITR	Information Transfer Rate
KNN	K-Nearest Neighbour
LDA	Linear Discriminant Analysis
LF-ASD	Low-Frequency Asynchronous Switch Design
MEG	Magnetoencephalogram
MI	Motor Imagery
ML	Maximum Likelihood
NCS	Non-Control State
NIRS	Near Infra Red Spectroscopy
PCA	Principal Component Analysis
PSD	Power Spectral Density
ROC	Receiver-Operating Characteristic

SBLDA	Self-training Bayesian Linear Discriminant Analysis
SCP	Slow Cortical Potentials
SFML	Simple and Fast Multimedia Library
SLDA	Self-training Linear Discriminant Analysis
S-LIC	Stimulus-Locked Inter-trace Correlation
SMA	Supplementary Motor Area
SMO	Sequential Minimal Optimization
SNR	Signal to Noise Ratio
SSVEP	Steady-State Visually Evoked Potential
SVM	Support Vector Machine
TCP	Transmission Control Protocol
TPR	True Positive Rate
VEP	Visually Evoked Potential
VMRP	Voluntary Movement-Related Potentials

List of Symbols

α	(8-12Hz) band in EEG
β	(12-30Hz) band in EEG
γ	(26-100Hz) band in EEG
δ	(0.5-4HZ) band in EEG
θ	(4-8Hz) band in EEG
μ	The rhythmic activity found in α, associated with motion
F_s	Sampling rate
$\mathbf{X^r}$	Raw pattern matrix
$\mathbf{W^r}$	Weight vector for feature reduction
$\mathbf{Y^r}$	Reduced pattern matrix
d	Dimensionality of the pattern matrix
m	Dimensionality of the reduced pattern matrix
n	Total data points (training + test)
$\mathbf{\Sigma}_{\mathbf{x}^r}$	Correlation matrix of $\mathbf{X^r}$
$\mathbf{S}$	Source matrix (signal produced by EEG sources)
$\mathbf{A}$	Mixing matrix
$\mathbf{w}^r$	A column of $\mathbf{W^r}$, or $\mathbf{W^r}$ when m=1

$\mathbf{Y^s}$	Reference signal for CCA
f_{st}	Stimulus frequency for SSVEP elicitation
n^h	Number of harmonics considered in CCA
$\mathbf{w}^s$	Projection vector for the reference signal
$\mathbf{X}$	Pattern matrix used for classification
$\mathbf{y}$	Label vector (first l elements)
$\mathbf{x}_i$s	Columns of $\mathbf{X}$, i^{th} feature vector
y_is	Elements of $\mathbf{y}$
l	Number of training data points
g	Dimensionality of the pattern matrix for classification
$\mathbf{w}$	Weight vector estimated by the classifier
m	Mean of all training vectors
m_k	Mean of all training vectors belonging to the k^{th} class
n^c	Number of classes (2 in all our experiments)
n^k	Number of elements belonging to the k^{th} class
w_is	Elements of $\mathbf{w}$
$\mathbf{S}_b$	Between class scatter matrix
$\mathbf{S}_w$	Within class scatter matrix
b	bias term for SVM
n^R	Rounds user for detection of a character
n^s	Number of equiprobable symbols detected by a BCI
CA	Classification accuracy (as fraction)
$B[bits]$	Effective number of bits detected per symbol
X	Set of all feature vectors
Y	Set of all labels
L	Set of labeled data
U	Set of unlabeled data

S	Set of all data (labeled + unlabeled)
H	Set of all mapping from $\mathbf{X}$ to $\mathbf{y}$
h^*	Mapping from $\mathbf{X}$ to $\mathbf{y}$
$\mathbf{n}$	Noise vector in BLDA
β'	Inverse variance of noise
D	The pair $(\mathbf{X}, \mathbf{y})$
α'	Hyper-parameters signifying the relevance of each feature
$\mathbf{I}'(\alpha')$	$g \times g$ matrix with α_i's along the diagonal
$\mathbf{C}$	Covariance of the posterior in BLDA
$\mathbf{m}'$	Mean of the posterior in BLDA
μ'	Mean of the predictive distribution
σ'	Standard deviation of the predictive distribution
c_{ii}	Diagonal elements of $\mathbf{C}$
m_i'	Elements of $\mathbf{m}'$
$tr(.)$	Trace of matrix
$\sigma_y i$	Standard deviation of averaged scores for i^{th} character detection
$sgn()$	Signum function
n_{iter}	Number of cross-validation iterations
n_{true}	Number of iterations for which the null-hypothesis is true
σ	Standard deviation of the results in cross-validation iterations
$[S(f)]_{f_{st}}$	Power spectral density at stimulus frequency
f_n	Narrow frequency range around f_{st}
f_w	Wider frequency range around f_{st}

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction to Brain Computer Interfaces

Severe neuromuscular disorders due to trauma, brain or spinal cord injury, brainstem stroke, muscular dystrophies, cerebral palsy, amyotrophic lateral sclerosis (ALS), multiple sclerosis etc. can result in peripheral motor neuron inactivity. Such patients typically experience a locked-in syndrome, rendering them unable to communicate their intentions or emotions in the usual manner, in spite of having a healthy brain. They require some device which has the ability to translate thoughts into actions without any muscular involvement - a device which, till recently, has always been themes of folklore and science fictions. Brain computer interface (BCI) is all about an alternate channel of communication between the user and the computer. A user can convey his intentions to the BCI by modulating his brain activity, which is translated to useful commands for the device to be controlled. Sitting in wheelchair, users might be able to browse the web, open e-mails, play games, switch on lights, move a robotic arm and so on; the technology has a list of applications which is virtually endless. They might even help the old and disabled to interact with robots, in a future scenario where robots will be

used as helpers to the old-aged and disabled. Such a device will help the patient have a better quality of life, and less dependent on a dedicated helper. Thus BCI hopes to provide a helping hand to patients who have permanent damage to the neuromuscular system, which no medicine, at least with the present state of the art, can hope to provide relief.

Apart from the utility in rehabilitation and assistive technologies, BCI also has applications in virtual reality, gaming etc. For example, a user with a head mounted device will be able to walk in a virtual environment by using his thought alone. BCI can also prove to be a peripheral for computer systems, taking the place of a conventional keyboard or a mouse. The user might be able to key in the alphabets from a keypad or dial a telephone number or move the cursor and thus browse the web. It can take the place of a joystick in gaming systems. The same BCI system can also be used to constantly monitor the well being of a person, thus adding utility with little or no extra cost.

Any device capable of recording the brain activity has the potential to be used in BCI. The most common and seemingly the only commercially viable system is the electrical activity of the brain, recorded by electrodes placed on the scalp, known as electroencephalogram (EEG). EEG acquisition requires only relatively simple and portable equipment, and does not require any invasive procedure. Various EEG activity patterns such as P300 (evoked by a surprise stimulus), steady state visually evoked potential[1] (SSVEP, evoked by repetitive visual stimuli), motor imagery (MI, associated with imagined limb movements) are used in BCIs.

The block diagram of a BCI system is shown in Fig. 1.1. The EEG is recorded using electrodes placed on the scalp. The signal is amplified using an amplifier, and then digitized using an analog to digital converter (ADC). The digitized signal is input to a computer, which processes the data to recognize activity patterns,

[1]the usage *steady state visual evoked potential* is also popular in the literature

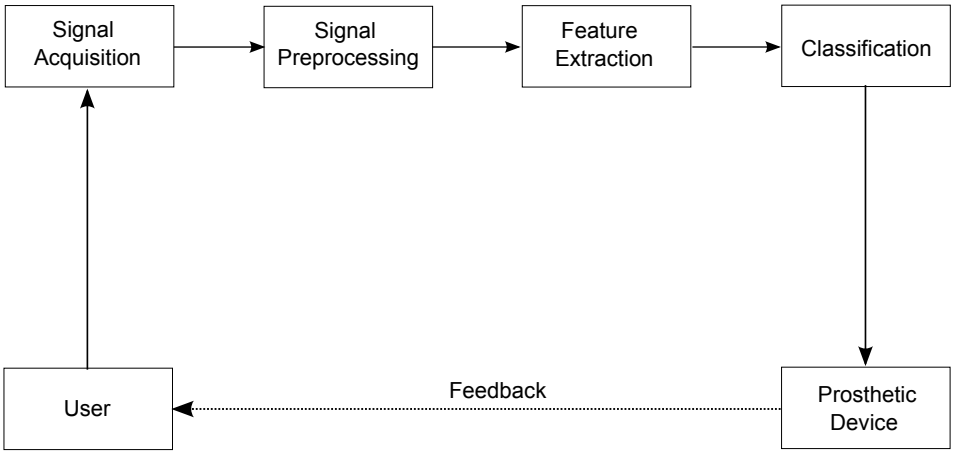


Figure 1.1: Block diagram of a BCI system.

which are interpreted as useful commands. The computer also produces required stimuli if the activity pattern needs to be evoked (which is the case for P300 and SSVEP), or cues suggesting the user to start giving an input if the activity pattern is spontaneous (such as motor imagery). The use of BCI system requires a training phase. During the training, the information is fed back to the user as a visual (e.g: movement of a cursor or bar on computer screen), auditory (a series of tones) or any other easily perceptible form, to help the user learn to modulate his brain activity so as to convey his intent. Training data is required for the computer as well, so that algorithms for processing and classification can be optimized for the user.

1.2 BCI Application Scenarios and State of the Art

Over the past decade, the BCI technology has grown leaps and bounds and thousands of BCI related publications have appeared in this period. Ultimately, the technologies have to be incorporated into usable products. Commercial products

from brands such as Emotiv, Neurosky etc. are available in the market. These companies provide their software development kits as a framework for developers to come up with interesting applications. There are a few free BCI frameworks such as BCI2000 and OpenVibe available in open domain mostly aimed at researchers in the field. To validate BCI feature extraction and classification methods, BCI competitions were held in 2002, 2003, 2005 and 2008, with winning entries published in special issues on IEEE Transactions on Biomedical Engineering and Transactions on Neural Systems and Rehabilitation Engineering.

The most common application of BCIs is the speller, usually based on either P300 or SSVEP. A speller usually has a virtual keyboard arranged as a matrix, with keys depending on the application (6×6 alphabetic virtual keyboard being the most popular). The rows and columns are highlighted in a pseudo-random sequence (for P300) or using different frequencies / phases (for SSVEP systems). For the P300 BCI, when the row or column containing character the user wants to input is highlighted, a P300 response is produced. This can be used to find the row and column containing the character, and hence the character itself. However, some groups have proved that flashing of individual buttons might be better than the row/column paradigm [3, 4]. For SSVEP, individual buttons have to flicker at different frequncies and hence the number of characters that can be used is limited.

Another variant of the row-column paradigm is the Hex-O-Speller [5] introduced by Blankertz *et al.* of the Berlin BCI group. The Hex-O-Speller selects the character the user desires as a two-step process. First, it selects one of the 6 hexagons, which contains the desired character. The second step is to select the desired character out of the 6 selected characters. This interface was presented at the world's largest IT fair - CeBIT 2006, achieves very good accuracy and can be implemented using any potential offering a 2-state control (such as motor imagery or P300).

Bayliss and Ballard [6] have created P300 based systems usable for navigating in a virtual world. Several other groups have published results on using BCI systems for virtual reality and gaming. Bin *et al.* have developed a BCI system which allows the user to control a virtual helicopter continuously in a 3-D world through intelligent control strategies using non-invasive BCI systems [7]. These show that non-invasive BCIs can achieve a level of control which was previously thought to be infeasible.

A high-performance 2-D cursor control combining the μ and β rhythms with P300 and motor imagery was demonstrated by Guan *et al.* [8]. Another popular application of BCI is in wheelchair, as it is likely that the main target beneficiaries of BCIs are wheelchair bound patients. There have been several studies focusing on the usability and performance of such BCIs [9,10]. Recently, a hybrid BCI with a lot of desirable features have been proposed by Allison *et al.* [11].

As the world is moving to an era of mobile and hand-held devices, and with technological convergence, there has been a recent interest in incorporating BCI systems into mobile/embedded platforms [12] [13] [14].

1.3 Motivation and Objectives

Since extracting useful information from EEG is difficult, EEG based BCI systems had not been getting much attention till the last decade. However, research in this topic has geared up during the past few years, and the technology has seen tremendous improvements. The development of a BCI system is highly multidisciplinary, requiring inputs from neurology, electrophysiology, psychology, instrumentation, signal processing, pattern recognition, and computer science.

The goal of all research is to devise faster, more accurate and easier to use BCI systems, with a variety of applications; symbolizing the victory of brain over muscles. The success of a BCI system depends on how effectively the EEG patterns

are recognized and classified so that they reflect the intentions of the user. However, achieving this goal while enabling the user to interact naturally with the computer is a challenging task. In particular, the user should be able to

- operate the system with minimal training / calibration. The system should be able to adapt to the user without requiring a large amount of intermittent training data, thus reducing the "warm-up" time required for the user to start operating the system. This requires devising efficient learning techniques so that the system can adapt to the user faster and more efficiently (*adaptation*).

- give input at will, i.e., without waiting for the computer to dictate when and whether an input can be/should be given. The system should be able to detect whether the user is intending to give an input at all, i.e., the system should be able to detect the control state (*control state detection*, and a system capable of control state detection is termed as an *asynchronous* system).

In this thesis, we propose techniques to achieve the two desirable characteristics mentioned above - adaptation and control state detection. We base our study on P300 and SSVEP based systems as they are easy to implement and requires relatively less amount of training for the user. Also, the relatively high amplitudes of P300 and SSVEP responses enable easy detection and thereby, high information transfer rates (ITRs). Moreover, P300 interfaces have the advantage of applicability to a wider range of patients.

1.4 Thesis Contributions and Organization

The main contributions in this thesis are :

- Development of a flexible BCI system in Visual C++ and Matlab. The system is capable of working as a usual P300 interface in offline or online

mode. An option for introducing a P300 flicker to induce SSVEP is also present. This system exploits the power, speed an multi-threading capabilities of C++, while the data processing algorithms are implemented in Matlab which makes prototyping easier. The processing can be done another computer to which the data is sent via transmission control protocol (TCP)/ internet protocol (IP). The system is well-integrated and works in real-time.

- A Co-training based technique for fast adaptation in a P300 BCI. The system uses two classifiers which can learn from each other progressively, and thus using unlabeled data efficiently to deliver high-performance classifiers from very little training data. A detailed statistical analysis of the performance of the proposed method is done on data from 5 subjects.

- A hybrid P300-SSVEP system is proposed where P300 is used for information transfer, and the control state information obtained from SSVEP. Results from offline and online data from 10 subjects show that this system is able to achieve good ITRs while having robust control state detection capability. Hence, we demonstrate that the use of hybrid systems is a promising alternative for implementing asynchronous systems.

The thesis is organized as follows :

Chapter 2 gives an overview of SSVEP and P300 BCIs, and the commonly used feature extraction and classification methods. A review of the control state detection and adaptation techniques reported in the literature, and the detailed motivation for the present study is also given therein. A flexible P300/SSVEP system developed, and its performance evaluation is presented in Chapter 3. Chapter 4 proposes and analyzes the performance of a co-training based method for delivering a fast adaptation in P300 BCI. Chapter 5 proposes a control state detection technique in a P300 BCI with SSVEP based control state detection. Chapter 6

concludes the thesis with discussions on implementation issues in real world applications, as well as numerous future directions / improvements.

Chapter 2

Brain Computer Interface : Overview

2.1 The Human Brain

The brain is arguably the most mysterious and complex organ in the human body. It is estimated to have 80-120 billion neurons and is composed of 3 main parts – Cerebrum, Cerebellum and Medualla. The *Cerebrum* is the largest and outermost most part of the brain, which accounts for two-third of the weight of the brain. It is composed of two hemispheres, and a thick band of nerve fibres known as the *Corpus callosum* connecting them. The outer part of the cerebrum is known as the cerebral cortex (grey matter), and is the place where the majority of the actual information processing takes place. The cerebral cortex is mainly divided into 4 lobes (Fig. 2.1). The *Frontal* lobe is the front-most portion of cerebrum, involved in decision making, problem solving, planning and motion. The *Parietal* lobe, which is located posterior to the frontal lobe, is responsible for cognition, information processing, pain, touch, etc. The *Occipital* lobe is the main visual processing part of the brain, and is located inferior to the parietal lobe. The *Temporal* lobe, which is found anterior to the occipital lobe handles auditory perception, language and speech production. The *Cerebellum* is the area involved in balance, equilibrium

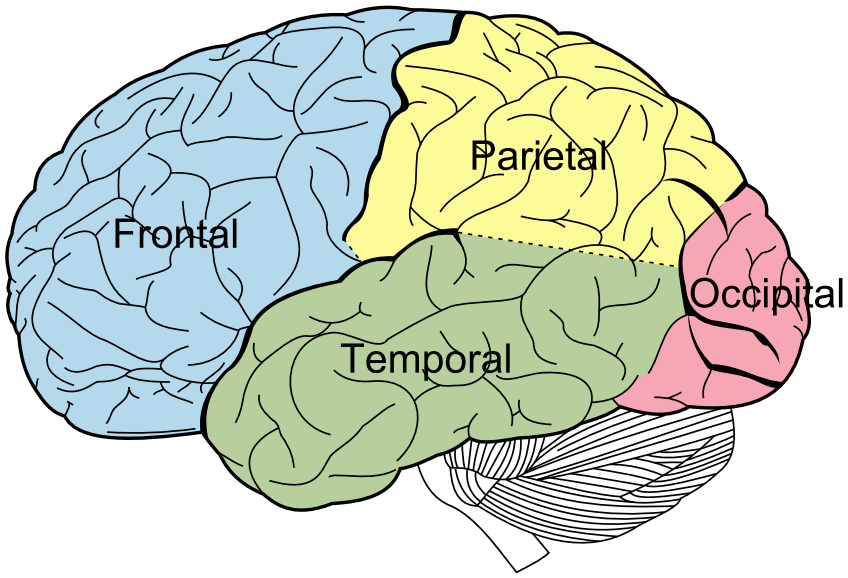


Figure 2.1: Lobes of human brain (Adapted from Fig.728, Gray's Anatomy [1]).

and movement co-ordination, relaying information between muscles and the area in the cerebral cortex involved in muscle control. The *Medulla Oblongata* is the portion of the brain controlling autonomic functions such as heart beat, digestion, blood vessel function etc. It is also responsible for transferring messages from various parts of the brain to the spinal cord.

2.1.1 Measuring brain activity

The firing of various neuronal groups in the brain causes measurable electric and magnetic signals. The variation in oxygenation level of the blood can also be detected using certain modalities. The activity thus measured provides insight into the working of the brain and cognition and helps explore various physiological /physophysical phenomena. Also, they give a means through which the user can convey his intentions directly to a computer, i.e., for BCIs. A brief description of the various techniques to explore the activity of the brain is given below, and their applicability and relevance to BCIs is explained.

EEG is the brain activity measured using electrodes placed on the scalp. It is the most popular modality used in BCIs, owing to the fact that it is non-invasive and recording systems are much cheaper. Also, the equipments involved are robust, portable and is less demanding on safety precautions and operator skills. Moreover, it has very good temporal resolution, and hence enables a faster detection of brain activities/responses. Owing to these advantages, most commercial BCI applications use EEG. Since our work is based on EEG, a detailed description of recording and analysis is given in the following sections.

Electrocorticogram (ECoG) is the electrical activity measured directly using electrode arrays placed surgically on the cortex surface. This is similar to EEG with respect to generation mechanism, but has much better spatial resolution due to the reduced volume conduction effects and attenuation by the skull. Also, ECoG is less prone to movement, muscle and eye artefacts. There have been a few studies using ECoG for BCI, and the accuracy of control achieved with ECoG is much better than BCIs. However, the obvious disadvantage of ECoG is that it is invasive andrequires the skull to be opened for installing the electrodes. Hence it is unlikely to get widespread acceptability except for a very small group of locked-in patients or those with Parkinson's disease.

Microelectrode Arrays measure the electrical activity from a single neuron or a small group of neurons. Similar to ECoG, the electrodes are surgically inserted in place. However, unlike ECoG, the needle electrodes are inserted *into* the cortex. Due to the complexity and risks involved, this procedure is done mostly in experiments on animals. A thorough exploration on human subjects is difficult and is unlikely to be popular in the near future.

Magnetoencephalogram (MEG) is the measurement of very small changes in magnetic field caused by intracellular currents of pyramidal neurons. The detection of such small changes in magnetic field is technically challenging, and hence MEG

equipment are generally expensive and bulky. An MEG based system is described in [15].

Functional Magnetic Resonance Imaging (fMRI) is a non-invasive modality which measures the blood oxygenation level dependent (BOLD) signal. It gives an indirect measure of neuronal activity through the so called haemodynamic response (HDR), which depends on the level of oxygenation of blood. It is a 4D imaging technique with a good spatial resolution. Though this is actually a desirable feature for a BCI system, its cost is too high and prohibitive for use in a consumer product. Moreover, its temporal resolution is a few 100s of milliseconds, which is quite unacceptable for a BCI system. However, fMRI based systems and combinations of EEG and fMRI based systems also have been reported in literature [16] [17].

Near Infra Red Spectroscopy (NIRS) is another non-invasive modality which measures the haemodynamic activity of the brain. In NIRS, sources emitting light in the near-infrared region is placed on the scalp. The intensity of the reflected light varies according to the level of oxygenation, which gives an indication of the brain activity. NIRS has a lower spatial and temporal resolution. Nevertheless, BCIs based on NIRS have appeared in the literature [18]

From the above discussions, it can be inferred that though any device capable of recording the brain activity has the potential to be used in BCI, the most practical and seemingly the only commercially viable modality is the EEG. In the next section, we describe EEG in detail.

2.2 Electroencephalogram (EEG)

EEG was discovered by Hans Berger in 1929. It is a widely used non-invasive technique for studying the brain activity. Main clinical uses of EEG are in epilepsy detection, sleep analysis, fatigue detection and in diagnosis of encephalopathies, coma, brain death etc. EEG is an especially valuable tool where sub-millisecond

temporal resolution is required, which is not possible with other techniques such as fMRI, in spite of having relatively poor spatial resolution.

The pyramidal neurons in the grey matter of the cortex are thought to be the principal source of electrical activity recorded by EEG. These neurons are well-aligned and fire in synchrony. This activity is transmitted to the surface of the scalp through volume conduction. The signals thus recorded is the spatial and temporal summation of the potentials produced by various groups of pyramidal neurons. The difference in electric potential caused across two electrodes can be measured / recorded using an appropriate device, and is what constitutes the EEG signal. EEG is usually recorded following the 10-20 system [19] of electrode placement. This system provides a set of standard electrode positions which are reasonably independent of various head geometries. The various electrode positions in the 10-20 system are shown in Fig. 2.2. The 10-20 system has been extended further to the 10-10 and 10-5 system for higher density recordings [20]. We use only one channel which is not present in the basic 10-20 system - Oz. Elastic electrode caps of various sizes are used for easy application of electrodes. The electrodes used are typically Silver/Silver Chloride, though Tin electrodes are used in cheaper recording setups. Some sort of conducting gel is typically used to reduce the resistance between the scalp and the electrodes. Since the EEG signal is usually of the order of a few micro volts, the signal is amplified using low noise instrumentation amplifiers. The signals are then sampled and quantized to digital form using sensitive analog-to-digital converters. The typical sampling rates used are 256 Hz, 512 Hz and 1024 Hz. Since most of the EEG activity is below 100 Hz, a sampling rate of 256 Hz might be sufficient for most practical applications, even without an anti-aliasing filter (it can optionally be used, though low noise filters tend to be costly). Each sample is usually represented using bits ranging from 8 to 32. A 3 second long raw EEG recorded at a sampling rate (F_s) of 256 Hz from

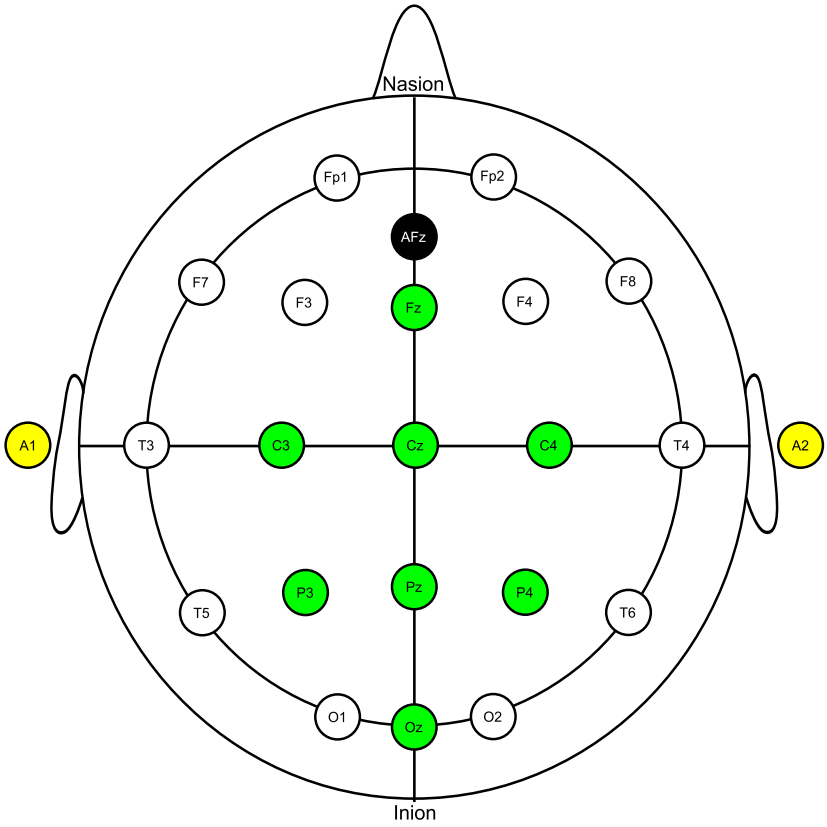
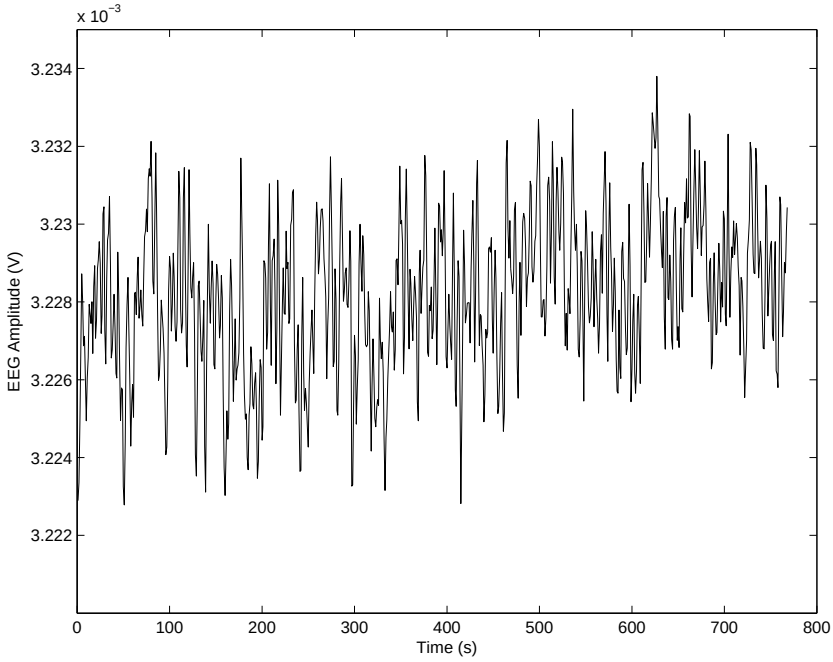


Figure 2.2: Electrode position in the 10-20 system of recording [adapted from http://www.beteredingen.nl (creative commons license)]). The channels used in our experiments are in green color. A1 and A2 (yellow) are the reference electrodes. AFz (in black color), is the ground.

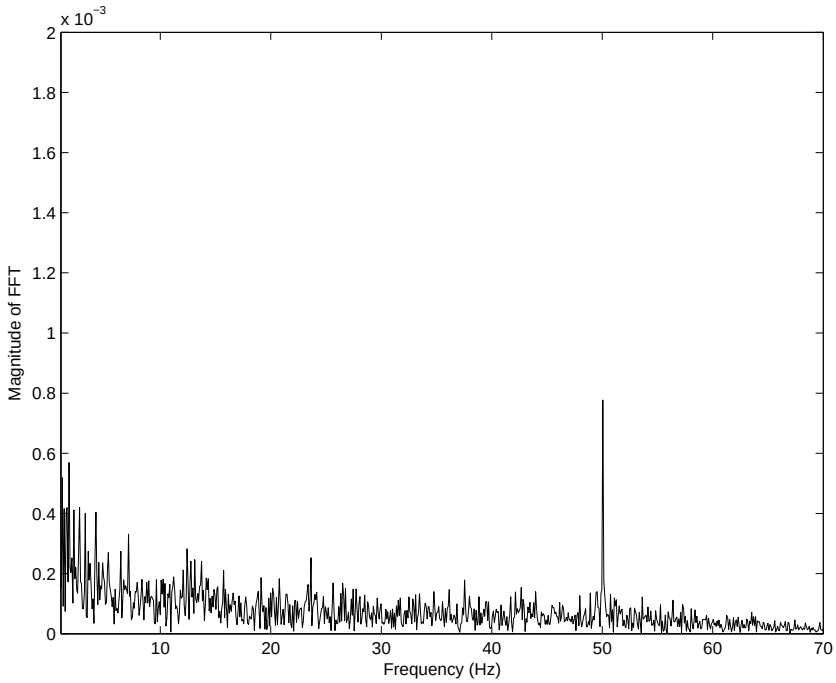
Cz electrode is shown in Fig. 2.3a and the corresponding Fourier spectrum in Fig. 2.3b.

2.2.1 Different types of EEG activities

The EEG activities can be broadly classified into two - rhythmic (spontaneous) and transient. Rythmic activities are due to synchronous oscillatory activities involving groups of neurons. The important bands in EEG based on their frequencies are δ (0.5-4 HZ), θ (4-8 Hz), α (8-12 Hz), β (12-30 Hz) and γ (26-100). The δ activity is mostly seen in infants and adults in deep sleep or meditation. θ activity can be



(a) A sample 3s long EEG signal recorded at $F_s = 256$ Hz at Oz



(b) Spectrum of the EEG signal, after removing the drift by high-pass filtering with a cut-off frequency of 0.5 Hz.

Figure 2.3: EEG signal and spectrum

seen in children and in adults while in normal sleep and drowsiness. α rhythms are more prominent in the occipital region in a state of relaxation with eyes closed, and without information processing involving concentration. β is associated with alertness and active concentration. γ reflects cross-modal sensory perception (involving fusion of information from different senses), higher-level cognition and short-term memory matching.

Event related potentials (ERPs) are the spatio-temporal patterns in EEG (the modality can be anything, though we are considering only ERPs in EEG in this context), formed in response to an event, and usually time-locked to the event (for example, surprise or initiation/imagination of movement). ERPs have historically had many clinical utilities, and are the most important brain activity pattern for BCIs. They are described in more detail in the following section.

2.2.2 EEG activities used in BCIs

Some well known and well studied EEG patterns used in BCI are described below.

P300: It is a positive deflection in EEG, peaking approximately 300 ms after the presentation of a rare, task-relevant stimuli (popularly known as the *oddball paradigm*) [21]. As this work mostly involves P300, more details about this ERP is given in Section 2.3.1.

SSVEPs: These are oscillations observed at occipital regions, induced by a periodic stimuli. The observed oscillations will have responses of the same frequency and its harmonics. This can be exploited in BCIs by requiring the user to concentrate on the desired input (which can, for example, be a number pad or a keyboard). By processing the EEG, the desired input can be found out [22].

ERPs based on MI: When a person is about to perform a motor function, the group of neurons in the contralateral hemisphere of the brain fires, and the amplitude of the measurable electrical activity reduces. This is called event related

desynchronization (ERD). They are most prominent in frontal and parietal locations and can be detected using μ[1] and β rhythms. This activity shows a strong contralateral dominance (i.e., if the movement of left limb is imagined, then the right part of the brain shows a larger amplitude for the rhythms than the left). This feature is exploited in BCIs [23]. Interestingly, it has been shown that this phenomenon is seen even when only an imagination of movement is done, and thus even disabled people can use BCI based on motor imagery.

Slow cortical potentials (SCP): These are slow, non-movement related potentials which reflect changes in cortical polarization of EEG. It has been shown that control over SCPs can be achieved by practice, and hence can be used in BCI [24].

The intent of the user can be conveyed by different means such as self-regulation of EEG (μ and β rhythms), oddball paradigm (P300), or evoked responses (e.g: visually evoked potentials, VEPs). The extracted information is then translated and used for the control of the target equipment according to the user's intent. Owing to factors described in Section 1.3, the systems described in this thesis utilizes P300 and SSVEP ERPs. Hence, we describe P300 and SSVEP phenomena and their detection in the following sections.

2.3 P300 and SSVEP based BCIs

2.3.1 P300 - Overview

P300 was first observed by Sutton in 1965 [25]. It is a positive deflection in EEG, observed about 300ms after the subject is present with an oddball paradigm, i.e., when the subject is experiencing a relatively rare stimulus among a sequence of more frequent stimuli. P300 is a result of conscious processing of stimuli, and hence

[1] An EEG rhythm in the α band, produced by motor cortex when when there is no hand/arm movement

is classified as an endogenous ERP. Various factors affecting the P300 have been extensively studied. P300 has a significant clinical utility, as the peak amplitudes and latencies (which is the time delay between the presentation of stimulus and the peak of P300 response) are found to be influenced by the mental state of the patient and presence of cognitive disorders like Schizophrenia and Alzheimer's disease. It is actually a composite signal, with 2 main subcomponents - P3a and P3b. The former is elicited when the subject pays little attention to the stimuli and is mostly observed in the fronto-central regions whereas P3b is elicited when the subject is given a task-relevant stimuli (for example, when the subject is required to count the number of occurrences of a particular type of stimulus). P3b is mostly observed around the centro-parietal region. P300 is believed to be due to the firing of neurons as a result of high-level, conscious information processing, though the exact cause is still debated.

2.3.2 P300 BCIs

The amplitude of the P300 is relatively higher than most other ERPs, and it is easier to elicit. Owing to these advantages, P300 based BCIs are getting increased attention. A number of BCI groups are using P300 based BCIs, and the results reported have been encouraging [3, 26, 27]. The most popular paradigm in P300 based BCIs is the speller [21]. Though P300 can be produced by auditory and tactile stimuli, visual P300 is, by far, the most popular in choice for BCI applications. Figure 2.4 shows the response for target (surprise) and non-target stimuli. A clear peak in amplitude can be seen at around 300 ms after the presentation of a target stimuli, whereas it is absent for a non-target stimuli.

A user of the P300 interface is typically required to concentrate on the object to be selected, and to silently count the number of times it blinks. This causes the P300 to be evoked at each blink of the desired object. In a speller paradigm, rows

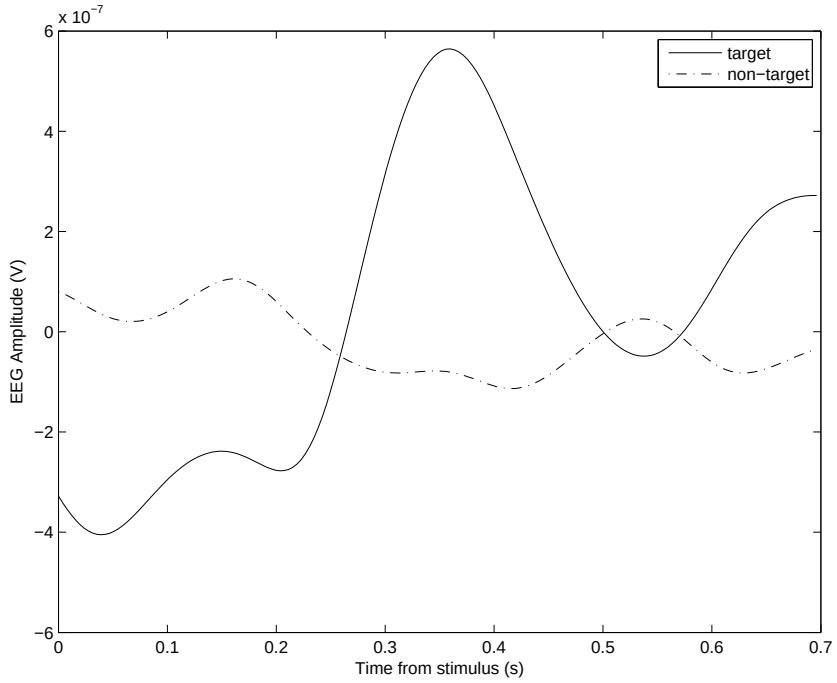


Figure 2.4: Response for target and non-target stimuli (low-pass filtered with a cut-off frequency of 12 Hz).

and columns of an on-screen keyboard (usually in the shape of a square matrix) are highlighted in a pseudo-random sequence such that each row and each column is highlighted once in every round (see Fig. 2.5). The diagram of a P300 speller (6×6) is given in Fig. 2.6. Once the row and column the user is concentrating on is accomplished, the character selection is complete (for example, character 'Y' is selected when a P300 is elicited after illumination of the 5th row and 1st column).

The P300 signal usually requires averaging of several trials to increase the signal to noise ratio (SNR) required for reliable detection. For selecting an object, the user will have to concentrate on it for several blinks. The EEG data associated with the flashing of one button, and that associated with one complete cycle of flashings are called *epoch* and *round*, respectively in this thesis. The lesser the number of rounds required to select a character, the more efficient the BCI is and better is the information transfer rate. The goal of all signal processing and classification

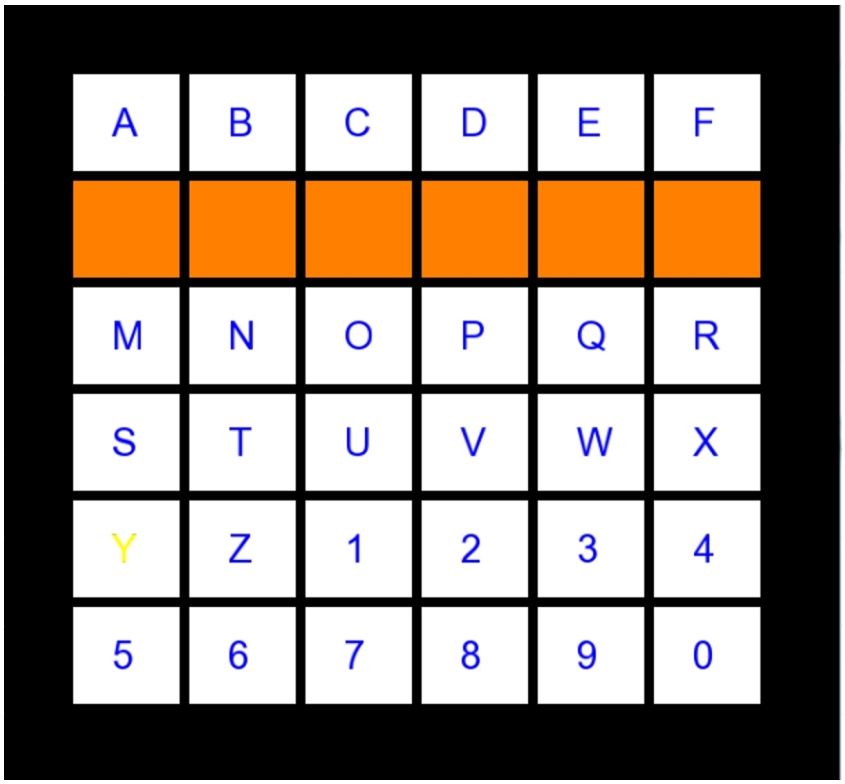


Figure 2.5: The P300 speller interface. The target character during the training phase is 'Y', which is yellow in color.

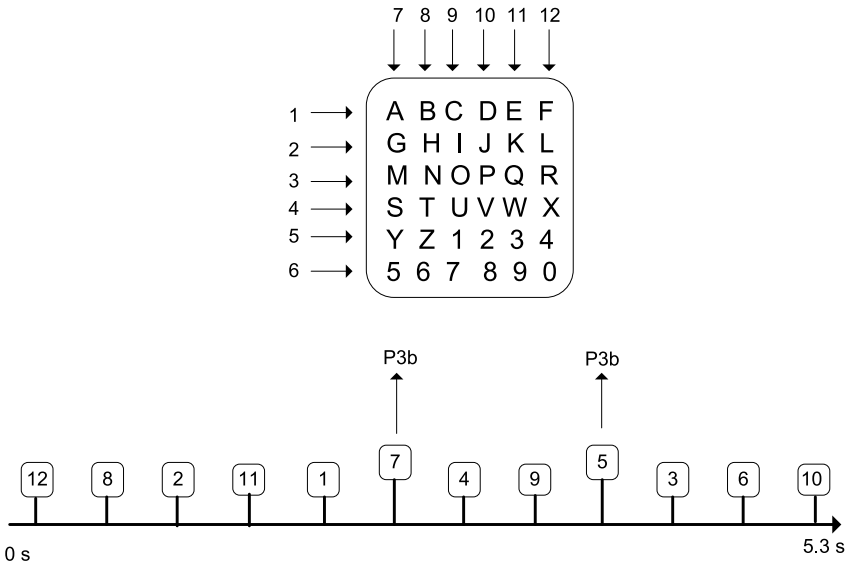


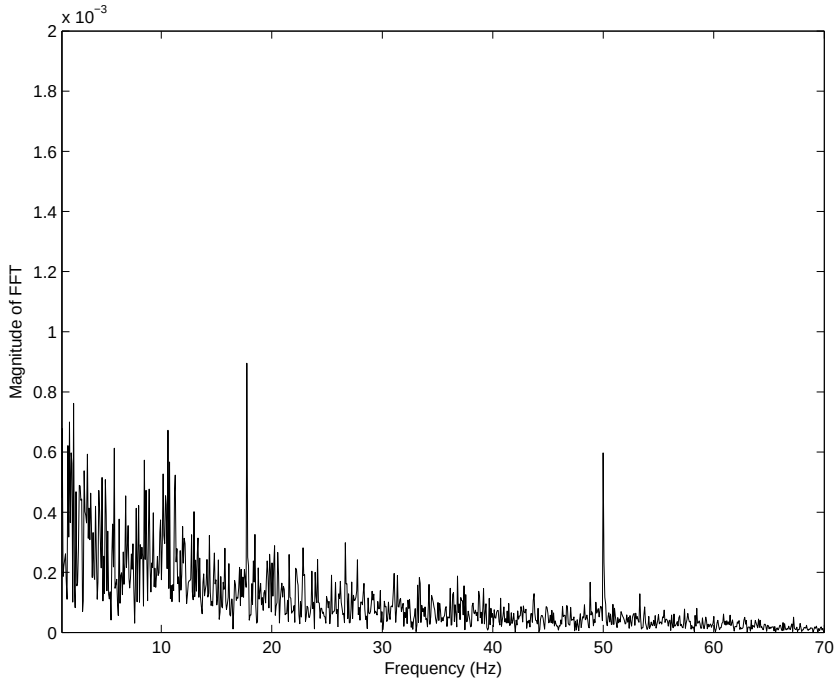
Figure 2.6: P300 speller operation (adapted from [2]).

algorithms are to enable faithful detection of P300 in a minimum number of rounds. See Section 2.6.1 for more details on the performance measures of BCIs.

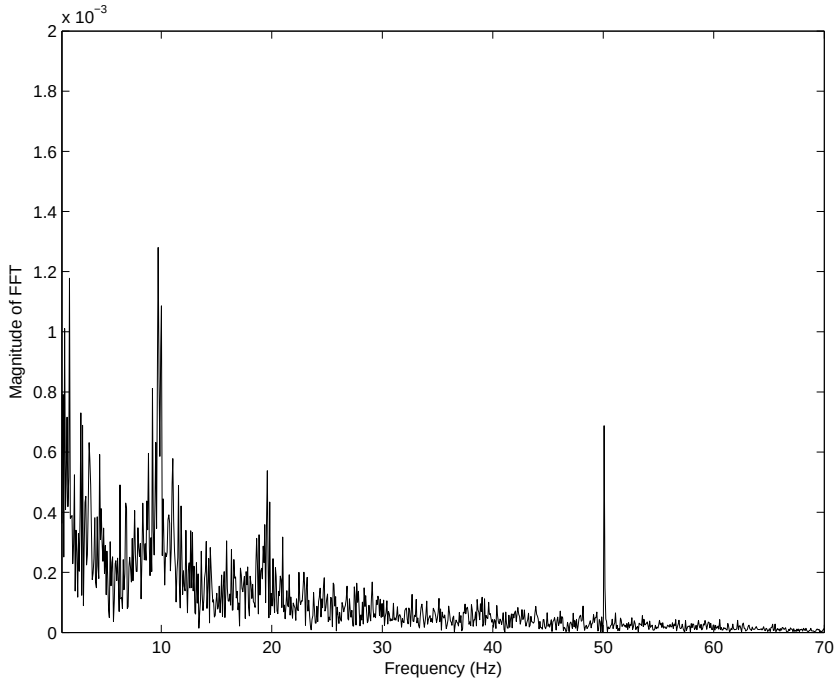
There were studies involving the effect of matrix sizes on BCIs and conclusion in the study by Allison *et al.* [28] was that the increase in matrix size increases the amplitude of the P300 signal. However, as the interval between successive stimuli increases, the communication rate will be reduced, though the number of bits conveyed per symbol is increased.

2.3.3 SSVEP - Overview

SSVEP is a potential produced by the brain in response to repetitive periodic visual stimulus. When the subject attends to a flickering stimulus of a certain frequency in the range of 3-75 Hz, a detectable signal of the same frequency and its harmonics are produced by the brain, predominantly at the occipital region [29]. Researchers have developed robust SSVEP-based BCI systems capable of reaching ITRs of up to 58±9.6 bits/minute using an interface with several stimuli, each flickering at a different frequency [30]. This is much faster than the rates reported for P300 based BCIs, which are usually less than 40 bits/minute [31–33]. SSVEP is usually very precise about the stimulus frequency. Gao *et al.* reported the possibility of distinguishing two stimuli with frequency difference of just 0.2 Hz [34]. Fig.2.7 shows the EEG spectrum with and without SSVEP, with the stimulus frequency being 17.7 Hz for eliciting SSVEP. A clear peak can be seen at the stimulus frequency in Fig.2.7b. The peak at 50 Hz is the power line noise and can be seen in both figures. The simplest method to detection SSVEP is thresholding of the amplitude of the signal's Fourier spectrum. Various techniques for enhanced detection of SSVEP can be found in [35–38].



(a) With SSVEP



(b) Without SSVEP

Figure 2.7: EEG spectrum with and without SSVEP. The stimulus frequency is 17.7 Hz. The higher amplitude at around 10 Hz in the absence of SSVEP is due to higher alpha activity with the subject having eyes closed. F_S=256 Hz, and drift is removed by high-pass filtering with 0.5 Hz cut-off.

2.3.4 Challenges in detection and classification of P300 and SSVEP

- *Low SNR*: The SNR of the P300 responses are usually very low, typically less than 0 dB. This makes the detection process very challenging. The noise overlaps with the desired signal significantly even within the same band. Moreover, the noise is not completely uncorrelated with the P300 or SSVEP, which makes the problem even more challenging.

- *Latency Jitter*: One of the major challenges in P300 detection is the problem of latency jitter. The latency of the P300 signals is again, dependent on a lot of factors. A classifier trained for one subject will not be optimum for another or even for the same subject, at a different point of time. Some approaches to adapt the classifier in an unsupervised or semi-supervised manner according to the training data has been suggested. Various approaches include use of Kalman filters or variants of it [39], semi-supervised updation of classifier etc [40].

- *Non-linearity and non-stationarity*: Many of the signal pre-processing and classification techniques require the signals to be stationary. Latency jitter, apart from other factors such as drying up of electrode gel, changes in attention and habituation of the stimulation paradigm causes differences in the nature of single trial responses, reducing the effectiveness of pre-processing and classification methods optimized at an earlier point in time. Many pre-processing techniques such as principal component analysis (PCA) and independent component analysis (ICA) makes the assumption that the EEG sources are point sources and that the scalp forms a linear conductor, which is not that accurate. The result is that the signals at the electrodes might be non-linearly related to the actual signal, which again reduces the effectiveness

of P300 extraction and classification.

- *BCI illiteracy*: It has been reported by almost all BCI groups that any particular type of BCI cannot be operated by a non-negligible fraction of the population. This is known to be independent of age, sex and other factors [41–43].

2.4 Preprocessing

The recorded EEG signal is usually corrupted by unwanted signals such as :

- *Artefacts*: Various non-cerebral sources which produce electrical activity such as eye-movements (electrooculogram, EOG), heart-beat (electrocardiogram, ECG) and muscle movements causes large deviations in the measured EEG from the original signal produced by the brain.

- *Measurement noise*: It is introduced by the recording equipment due to quantization, linearity imperfections, shot noise from the amplifier, power line noise, the ever-present Gaussian noise etc. The effect of power line noise (50 Hz) can be seen in Fig. 2.7.

Preprocessing of the signals is necessary to remove these artefacts before being used for feature extraction/classification. Since the signal from the preprocessing stage is used for feature extraction and classification, it has a profound effect on the effectiveness of the performance of the BCI system. Out of these artefacts, electrooculogram (EOG, caused due to eye movements) is the one which is more frequently present, and having amplitudes much larger than the signals of interest. A variety of methods have been proposed in the literature to detect and correct these artefacts or to reject portions of the data corrupted by the artefact. They can mainly be classified into 4 types :

- *Filtering based methods*: Filtering based methods work on the principle of adaptive noise cancelation. The recorded EOG signals (or the signals from the electrodes FP1 and FP2) are used as the desired signal for an adaptive filter. The scalp is considered to be a linear or non-linear filter, and we attempt to model this filter so that the component of EOG arriving at a particular electrode can be estimated and subtracted to remove it [44]. Filtering is also useful for removing band-specific noises such as the power line noise.

- *Component based methods*: Component based methods attempt to find out the spatial pattern of the corrupting artefact signal, and removes it. The popular component based methods are principal component analysis (PCA) and independent component analysis (ICA). The former finds the directions along which the variance is maximum, which usually corresponds to the artefact signal, which is extracted and removed. ICA attempts to find a linear mixing matrix, and removes the component which is likely to be the artefact signal (identified by some postprocessing technique, such as thresholding or based on spatial patterns of the component). These techniques are detailed in the next section.

- *Regression based techniques* : This is a simple technique in which we find the regression coefficients of the EOG signal on each electrodes, which is used to remove the EOG artefacts.

- *Thresholding based techniques*: Here, the artefacts are detected by a simple amplitude thresholding. The procedure is frequently referred to as windsorizing in the literature. The top and bottom 5 percentiles of the amplitudes (outliers) are assumed to be from artefacts and is clipped. Though this method is very simple, it has been shown to give reasonably good performance in BCI [2]. Hence, we used only thresholding in our analysis.

2.5 Feature extraction

In pattern recognition problems, there might be numerous features that can be used to separate the data into a fixed number of classes, and together they form the pattern vector. For a well performing pattern recognition system, the training data requirement increases with the dimensionality of the pattern vector (curse of dimensionality) [45]. Not all features may be relevant to the classification, and hence we can avoid using unwanted features. Removing features that does not contribute to the classification may, in fact improve the classification accuracy and reduce the training data requirement. Hence, feature extraction can be described as the process of extracting relevant features (from the recorded EEG signal, for BCI) to be fed into the classifier. A good overview of the signal processing/feature extraction methods employed in BCI can be found in [46]. For EEG, feature extraction can be spatial, temporal or spectral. Spatial feature extraction usually refers to selecting specific channels and/or finding linear combinations of channels which will enable an easy detection of signals of interest. In temporal feature extraction, we select specific time segments or parameters extracted from the time series as features. Spectral feature extraction refers to extracting features in the frequency domain such as band powers. It is not a popular method in P300 processing, as P300 has a high variability in its spectrum, whereas for SSVEP detection, spectral methods are very effective.

2.5.1 Spatial feature extraction

Projecting the vectors into lower dimensional spaces usually enables the implementation of the system using less training data. The generalized problem formulation for feature selection/dimensionality reduction is as follows - given an $n \times d$ pattern matrix $\mathbf{X}^r$, derive an $n \times m$ pattern matrix $\mathbf{Y^r}$, such that $m \leq d$, and $\mathbf{Y}^r = \mathbf{W}^r\mathbf{X}^r$, where $\mathbf{W}^r$ is a $d \times m$ transformation matrix. The main spatial (component based)

feature extraction methods commonly used in BCIs are :

(i) PCA: It decomposes the signals into mutually orthogonal components with an ordering dependent on their respective variances. Hence, the signals of interests need to be of relatively high variance to avoid loss of information relevant for classification. In PCA, the matrix $\mathbf{W}^r$ is a $d \times m$ transformation matrix, which has columns being eigenvectors corresponding to the m largest eigenvalues of the correlation matrix of $\mathbf{X}^r$, given by

$$\mathbf{\Sigma}_{\mathbf{x}}^r = \mathbf{X}^r \mathbf{X}^{rT} \tag{2.1}$$

The assumption that the P300 components have relatively high variance is not always valid, yet PCA is still used for dimensionality reduction before further feature extraction/classification [47]. PCA can also be calculated using Singular Value Decomposition (SVD) [45].

(ii) ICA: Compared to PCA which transforms the data to maximize the variance or signal energy, ICA tries to find the statistically independent components either in the spatial or in the temporal domain. This is a stronger constraint than decorrelation. The mixing process can be represented as $\mathbf{X^r} = \mathbf{AS}$, where $\mathbf{A}$ is the unknown mixing matrix to be estimated and $\mathbf{S}$ is the source matrix. We need to find an estimate of the mixing matrix, $\mathbf{W^r} \approx \mathbf{A^{-1}}$ and hence, the source matrix, $\mathbf{Y^r} \approx \mathbf{S}$.

The solution to this problem involves the measure of statistical independence, which determines a practical algorithm for the solution. The most commonly used methods are based on maximizing non-Gaussianity (kurtosis, negentropy) or maximizing the joint entropy of the output or by tensorial methods [48].

Both PCA and ICA can be implemented spatially as well as temporally, though spatial independence is a more popular criterion in BCI context. It is a common practice (and mandatory for fastICA) to reduce the dimensionality (for example,

by PCA) and to do whitening (making the components uncorrelated and their variances to be unity) and centering (making mean to be 0).

(iii) Common Spatial Patterns (CSP): Common spatial pattern algorithms are popular spatial feature extraction methods for MI based BCIs where the two classes can be discriminated based on difference in band powers in the two hemispheres of the brain (contra-lateral dominance). These methods aim to find a projection direction which maximizes the variance of signals from one class while minimizing the variance of those from the other class. This maximization is usually done through solving an eigenvalue problem. However, since P300 or SSVEP do not exhibit a definitive contra-lateral dominance, such techniques are not used in SSVEP or P300 BCIs and are not explained in detail in this thesis.

2.5.2 Temporal feature extraction

Temporal feature extraction is popularly used for potentials such as P300 which are more reliably detected using time domain feature [49]. Various feature extraction operations done in time domain are

(i) Filtering and downsampling: BCI systems focus on one or more of specific EEG activities, which are usually bandlimited. The first step in feature extraction is bandpass filtering of the signals. In addition to removal of out of band noises, drifts and trends, bandpass filtering also enables the downsampling of the signal, which reduces the computational burden and pattern size. This is especially advantageous if the feature vector is a time series (which is mostly the case with P300).

(ii) Time domain features for P300(peak picking, area picking): The easiest way to detect P300 is to find the peak amplitude within a trial or the area of time series corresponding to a trial and use it for classification. In area picking, the total area in the region around 300 ms after the stimuli presentation is used as

the feature. The disadvantage of these methods is that it is very prone to decision errors, especially if the data is very noisy.

(iii) Time Domain features for SSVEP: While most of the SSVEP detection techniques are frequency based, there have been time-domain approaches too which have yielded very good detection results. In [50], the authors have used a time domain method which they call stimulus-locked inter-trace correlation (SLIC) method. The time domain method is suitable for detection of VEPs which are evoked by random or irregular stimuli as well, provided there is a reasonable phase lock between the stimuli and the response. A prior knowledge of the stimuli pattern and precise knowledge of the experimental conditions are not required. The data recorded from the channel PO2 was segmented into multiple traces (stimuli-locked time-series), each starting with a stimulus. They were correlated in pairs, and the median correlation was used as a feature in classification.

2.5.3 Spatio-Spectral feature extraction

Canonical Correlation Analysis (CCA): A popular and arguably one of the best SSVEP feature extraction method is based on CCA as proposed by Gao *et al.* [34]. The CCA method tries to find projections where two multidimensional random variables $\mathbf{X^r}$ and $\mathbf{Y^s}$ such that the correlation $\rho(\mathbf{y}^r, \mathbf{y}^s)$ between the projected quantities $\mathbf{y}^r = \mathbf{w}^{rT}\mathbf{X^r}$ and $\mathbf{y}^s = \mathbf{w}^{sT}\mathbf{Y^s}$ is maximized.

The reference signal $\mathbf{Y^s} = [sin(2\pi f_{st}t)cos(2\pi f_{st}t) \ldots sin(2\pi N f_{st}t)cos(2\pi N f_{st}t)]^T$, where f_{st} is the stimulus frequency, n^h is the number of harmonics to be considered and t is the time. The optimum projection vectors can be found through eigenvalue based methods. The correlation coefficient can help identify the sinusoids present in the waveform, and hence we can detect the character user desires to input. The features extracted by CCA is proportional to the power of the signal obtained from spatio-temporal filtering.

2.5.4 Power spectral density (PSD) based techniques

PSD based techniques are the traditional feature extraction/detection techniques for SSVEP based BCI. The PSD is estimated using a windowed time series using fast Fourier transform (FFT). The FFT from the frequency ranges used in the SSVEP-BCI system is compared and the frequency with the largest PSD is selected. PSD based techniques are simple yet popular for SSVEP detection, as the stimulus frequency and their harmonics will have a noticeable increase in power when the user is focusing on the stimuli.

2.6 Classification algorithms

A comprehensive review of classification algorithms used in BCI is given in [49]. For P300 based BCIs, the problem reduces to a binary classification problem (P300 is present or not). The purpose of the classification algorithms are such that given the training samples $\{\mathbf{X}, \mathbf{y}\}$, where $\mathbf{X}$ is the feature vector having n samples (i.e., $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_l, \mathbf{x}_{l+1}, \ldots, \mathbf{x}_n\}$), each being a column vector of length g ($g = m$ if dimensionality reduction is done; $g = d$ and $\mathbf{X} = \mathbf{X}^r$ otherwise), and $\mathbf{y}$ (of the form $\mathbf{y} = \{y_1, y_2, \ldots, y_n\}$ is a label vector of length n, with only l values (labels) known. The classification problem can be defined as : Given $\{(\mathbf{x}_i, y_i)\}_{i=l+1}^{n}$, find the optimum values of the weight vector $\mathbf{w}$ such that the true labels of the test samples can be predicted from $y_i = \mathbf{w}^T\mathbf{x}_i$, by some simple operation. For P300 detection, y_i over a few rounds are summed, and the object corresponding to the maximum sum is chosen as the prediction.

(i) Fisher's linear discriminant analysis (FLDA): FLDA is a popular method for finding a linear boundary between classes. FLDA is actually a feature reduction technique, but in the context of BCI systems where the number of classes is 2, FLDA serves as a classification technique. In FLDA, the data is projected to a

lower dimension such that the projected means of classes are far apart, while the spread of the projected data is small (Fisher's criterion). This can be realized by optimizing a cost function related to the within-class scatter matrix ($\mathbf{S}_w$) and the between-class scatter matrix ($\mathbf{S}_b$), which are defined as

$$\mathbf{S}_w = \sum_{k=1}^{n^c} \sum_{\mathbf{x}_j \in c_k} (\mathbf{x}_j - \mathbf{m}_k)(\mathbf{x}_j - \mathbf{m}_k)^T \tag{2.2}$$

$$\mathbf{S}_b = \sum_{k=1}^{n^c} n^k (\mathbf{m}_k - \mathbf{m})(\mathbf{m}_k - \mathbf{m})^T \tag{2.3}$$

where $\mathbf{x}_j;\ j = 1, 2, \ldots, l$ are the training data vectors, c_k denotes the k^{th} class, $\mathbf{m}_k$ is the mean of samples belonging to the k^{th} class, $\mathbf{m}$ is the global mean, n^c is the number of classes (n^c=2 in our classification, denoting either the presence or the absence of P300), and n^k is the number of samples in the k^{th} class. In FLDA, the problem is to find a projection vector $\mathbf{w} = [w_1, w_2, \ldots, w_g]^T$ such that the projection

$$\mathbf{y} = \mathbf{w}^T \mathbf{X} \tag{2.4}$$

maximizes the criterion function $J_p(\mathbf{w})$ defined as

$$J_p(\mathbf{w}) = \frac{\det(\mathbf{w}^T \mathbf{S}_b \mathbf{w})}{\det(\mathbf{w}^T \mathbf{S}_w \mathbf{w})}. \tag{2.5}$$

The solution [45] is to choose $\mathbf{w}$ satisfying the eigen equation

$$\mathbf{S}_w^{-1} \mathbf{S}_b \mathbf{w} = \lambda \mathbf{w}, \tag{2.6}$$

if $\mathbf{S}_w^{-1}$ exists, λ being the only non-zero eigenvalue [45] of $\mathbf{S}_w^{-1}\mathbf{S}_b$. Once $\mathbf{w}$ is estimated, the classifier design is complete and the output for a single feature vector

$(\mathbf{x}_j)$ is the scalar

$$y_j = \mathbf{w}^T \mathbf{x}_j. \tag{2.7}$$

Reliable detection of the P300 usually requires several rounds (a *round* is the data associated with one complete cycle of row and column flashings [51]) of stimulus presentations. In each round, the scores for all rows and columns are calculated (using Eq.(2.7)). The scores are typically averaged over a fixed number of rounds (denoted by n^R). The symbol at the intersection of the row and the column having the maximum of the averaged scores is the predicted character. This scheme performs a multi-class classification, even though the underlying classifications are binary. This scheme is applicable for the methods described in the following sections too.

(ii) Support Vector Machines (SVM): SVMs try to build a linear classifier such that the margin of the classifier from the points nearest to the class boundary are maximally separated. These points acts as the support vectors to the classifier. These vectors can be obtained by solving a quadratic optimization problem. Given the training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$, find the optimum values of the weight vector $\mathbf{w}$, and bias b such that they satisfy the constraints

$$y_i \left(\mathbf{w}^T \mathbf{x}_i + b\right) \geq 1 \quad for \quad i = 1, 2, ...l \tag{2.8}$$

and the weight vector $\mathbf{w}$ minimizes the cost function:

$$\phi\left(\mathbf{w}\right) = \frac{1}{2}\mathbf{w}^T \mathbf{w} \tag{2.9}$$

which is usually done by a quadratic optimization procedure. Similar to the case with FLDA, the output is $y_i = \mathbf{w}^T \mathbf{x}_i + b;\ i = l + 1...n$. This value is summed over a number of trials and the symbol with maximum sum is selected. SVMs are widely used in P300 classification due to their simplicity in the classification phase

(training phase, however, is computationally intensive).

(iii) Kernel Machines : Kernel machines use a non-linear kernel to project the data non-linearly into a higher-dimensional space. This increases the discriminability of the data provided the kernel function and the hyperparameters are properly selected. The kernel trick can be applied to linear classifiers to accomplish non-linear classification [52]. SVMs are frequently used with kernels in many practical applications. However, due to the higher dimensionality of the feature space, BCI applications frequently use a linear kernel.

(iv) Neural Networks : Neural networks are a class of networks inspired from the working of biological neurons. A feed-forward neural network typically has an input layer, one or more hidden layers, and an output layer. Each neuron calculates the weighted sum of the inputs from previous stages, and calculates the output through an activation function from the output of the previous stages. They are trained either by back-propagation algorithm or with evolutionary algorithms. Radial basis function neural networks, on the other hand, calculates the output as a weighted sum of outputs of hidden layers, which are basically radial basis functions. They are usually trained by a least-square algorithm [52]. For reasons similar to that of kernel methods (high data dimensionality), neural networks are relatively less popular in the BCI literature.

(v) Ensemble of Classifiers : Usually, it is advantageous to train a number of classes, each to a specific set of data. The class prediction is made based on the outputs of a number of classifiers. This method is called classifier ensembles. As can be seen from [53], classifier ensembles give very good results.

(vi) Bayesian Classifiers : Bayesian classifiers are ideally the optimum classifiers, but the determination of the perfect model is infeasible in most applications. A convenient method is to use Bayesian learning to obtain a regularized linear classifier. More details about this method can be found in [2]. The effectiveness of

Bayesian classifiers for BCI is clearly demonstrated in [2] and [54].

2.6.1 Evaluation criteria for BCIs

Depending on the type of BCI, several evaluation criteria have been proposed. A detailed overview is given in [55]. However, since BCI is a communication system, the ITR is also an important figure of merit. Bit rate increases with the classification accuracy (ratio of number of correct classifications to the total number of classifications), and decreases with the number of rounds required for correct classification. Based on the suggestion of Wolpaw *et al.* [55,56], the formula for information per detected symbol is calculated as

$$\mathrm{B}[bits] = \log_2(n^s) + \mathrm{CA} \cdot \log_2(\mathrm{CA}) + (1-\mathrm{CA}) \cdot \log_2 \left[\frac{(1-\mathrm{CA})}{(\mathrm{n^s}-1)} \right], \qquad (2.10)$$

where n^s is the number of equiprobable symbols (36 in our speller paradigm; as there are 6 rows and 6 columns) and CA is the classification accuracy (the ratio of correct symbols detected to the total number of symbols being classified). It is assumed that CA is uniform among classes.

If the online system involves a provision to correct a wrong input, then the number of bit finally input is the correct number of bits, and ITR can be calculated by dividing it with the time taken. If error correction facility, such as a delete or a backspace button [57] is not available (which is the case with our system), the number of bits detected per detected symbol is calculated as Eq.(2.11). Another option is to find the number of characters communicated = (number of correct detections – number of wrong detections). This is to account for the extra character to be input to inform the system of a wrong entry. However, this method has not been used, as a wrong detection need not necessarily carry a completely negative information, and systems with correction using a dictionary can still make use

of the detection to guess which is the closest word (for example, in a T9 like system) [58, 59].

Given the inter-stimulus interval (ISI, the interval between 2 consecutive stimulus presentations, in seconds) and the inter-character gap (ICG, the time gap between 2 consecutive blocks, in seconds), ITR (in bits/min) is calculated as

$$\text{ITR} = \frac{\text{B}[bits]}{n^R \times \text{ISI} \times 12 + \text{ICG}} \times 60, \tag{2.11}$$

where n^R is the number of rounds required for the detection of a character. The CA by chance is 0.0278 (1/36, or 2.78%), and the corresponding ITR is 0, from the above equation.

2.7 Adaptation

The presence or absence of P300 is typically detected with the help of a classifier, which is trained using some data for which the labels are known [49] [46]. The labeled data for this purpose is obtained through a training process. Given the inter and intra-personal variations in EEG, the training time is several tens of minutes to obtain satisfactory performance [55]. Also, the classifier trained for a particular user may not be valid for him/her over time, and even less likely for other users. The requirement of long training time is tiring on the part of the user, and hence there is a strong motivation for developing BCI systems which require only little training data. The main reasons for non-stationarity in EEG potentials are :

Cognitive changes : Cognitive changes are those changes which occur in the human brain over time, not necessarily related to the task being performed. This can be due to fatigue, attention and motivation level changes etc.

Adaptation on the part of the user : In general, the operation of a BCI system is

the interaction between two systems which can learn : the computer/algorithm and the human. In the course of using the BCI system, the human subject might learn or adapt to the BCI system, and the features related to the potential of interest in his EEG might change. This can render the classifier incapable of detecting the potential/signal of interest effectively.

Recording related factors : The electrode gels typically used in EEG recording (which act as a conducting medium between the electrode and the scalp)are susceptible to drying which increases its resistivity. Also, electrodes other than those made of silver or gold are susceptible to polarization, which can also affect the signal quality adversely. This is especially the case for low-cost consumer grade systems. The drying / polarization can cause intra-session non-stationarity, whereas differences in the positioning of the electrodes / sensors used can cause inter-session non-linearity.

Hence, the fact is that the BCI system, even after the tiring training process, can be used only for short periods of time. For yielding good classification performance, the system typically requires re-training which adds to the user frustration. These long and intermittent training requirements reduce the attractiveness of BCI as an alternate channel of communication. Hence, building good classifiers from shorter training sessions has become a topic of great interest to the BCI research community. Such an adaptive BCI has become an active field of research and encouraging results have been reported by various groups [33, 40, 54, 60]. Various methods to deal with non-stationarity have been reported in the literature which includes retraining using a full set of new training data, adaptation using a small amount of training data, and semi-supervised learning techniques. The fundamental questions to be answered in this context are (i) what to adapt (ii) when to adapt and (iii) how to adapt [61], which are briefly discussed in the following sections.

2.7.1 What to adapt

Paradigm: One option is to adapt the key parameters of the paradigm itself, such as n^R, ISI etc. BCI systems where the n^R was modified adaptively to obtain increased ITRs are reported in [33] and [31]. In [28], Allison *et al.* demonstrated that the discriminability of the P300 signal (for a speller) varies with matrix sizes and ISIs. This shows that the interface itself can potentially be varied to achieve best performance for a specific user. It is difficult to statistically analyze the advantage brought in through paradigm adaptation, as it will require extensive experimentation with a number of subjects.

Classifier: There have been a number of studies proposing to adapt the classifier itself. This method has been the most popular in BCI community, mostly owing to the fact that such studies can be conducted offline though various validation schemes. Several such studies have been reported in [54, 61–64]. Due to practical reasons, our work also involves adapting the classifier, and a more detailed picture is presented in Section 2.7.3.

Features: Another option is to decide/adapt features to be used in classification intermittently. Various methods to adaptively extract relevant spatial patterns (for MI based BCI) have been reported in [65] and [66]. In [67], Li *et al.* proposes an algorithm which can do both feature and classifier adaptation in an adaptive MI based BCI.

2.7.2 When to adapt

Systems with a correction input : For certain systems with error correction, the criterion for whether to adapt can be made by analyzing the number of corrections performed by the patient. For example, in a speller with a backspace button, the number of times the backspace button is input could be a measure of the accuracy of the classifier. However, if the classification accuracy is very low, this might not

be an option as the backspace/delete character themselves might not be detected properly. In this case, the system might not be able to "suggest" a re-training, and will have to be manually initiated.

Another option to identify errors in detection is through error potentials (ErrP). ErrP is produced by the user as a reaction to his/her own mistakes. ErrP can be detected fairly well even with a single trial. The information from the ErrP can be used to decide the error rate of the interface, and can also provide correction information in some cases [68]. Buttfield *et al.* [69] demonstrated that the error made by the BCI interface can also cause elicitation of a version of ErrP. They have demonstrated the use of an adaptation scheme based on ErrP for a motor-imagery based BCI, with promising results. In a very recent article, Combaz *et al.* [70] has demonstrated improving the performance of a P300 speller by incorporating ErrP information. They achieved upto 15% improvement in the classification accuracy by selecting the symbol with the next best classifier score in event of a ErrP detection. However, their work did not explicitly adapt the classifier or the interface. They have suggested ways to adapt the paradigm by using a few extra rounds of EEG to update the classifier scores when an error is detected.

Using classifier confidence criterion : Classifier confidence criterion can tell us the confidence with which the classifier labels the inputs. The usual (and reasonable) assumption is that the higher the confidence, the higher the chance that the classification is correct. However, this confidence criterion can be classifier and paradigm dependent. For example, for a left-right classification problem, the confidence criterion might be the mean of differences in scores for left and right detections. For a speller paradigm, the confidence criterion should be a measure of prominence of the maximum score among the scores for rows and columns. While evaluating the confidence of a larger set of data, criterion such as Fisher scores [63] or Raleigh coefficients [67] might be more appropriate. However, they

are not meaningful for smaller data segments such as that for a single round. In the next chapter, we have introduced the z-score of the character with the maximum row/column score within a round as a criterion in P300.

2.7.3 How to adapt

One way to improve the performance is to have improved feature extraction and classification techniques which can reduce the effects of non-stationarity (by eliminating features which vary with time), and the other is to render the classifier able to adjust itself based on new labeled or unlabeled data. The various methods to adapt the classifier are given below.

Retraining : This is the simplest and most direct way of dealing with non-stationarity. Any time when it is deemed that the classification accuracy has fallen below a threshold, the complete training process can be repeated. While not efficient in most circumstances, this might be preferable if the current classifier has limited ability to classify the data reliably due to prolonged non-stationarity.

Adaptation based on intermittent labeled samples (active learning) : This is another option where we can feed in intermittent samples with known labels which can act as calibration inputs to the classifier. The classifier can correct/adapt itself in such a way that some cost function based on both the initial training samples as well as the calibration samples is minimized. However, this requires careful choice of calibration data, which is done through some efficient strategy to query for the labels. In [71], Zhao *et al.* adopt a strategy whereby the samples chosen for query will have maximum uncertainty (entropy) about its own label, while the certainty of labels for the rest of the test samples will be maximized once its true label is known. Their method achieves an accuracy comparable to fully supervised classifiers while requiring labels of only about half (70 out of 143) of the trials.

ErrP, as described in Section 2.7.2 [69], could also serve the purpose of intermittent labeled samples in 2-class classification problems. This has the added advantage that queries are not explicitly made, and hence generating additional labels does not add to the training overhead.

Semi-supervised learning : Another option to deal with non-stationary nature of EEG patterns useful in BCI is to use the information from unlabeled data to improve the classifier gradually. In semi-supervised learning, the initial classifier is trained using (limited) labeled data, which adapts itself efficiently based on the incoming (unlabeled) data. Semi-supervised techniques can be classified as *inductive* or *transductive* depending on whether the data being labeled affects the decision boundary or not. In recent years, many groups have published encouraging results on adaptation, though most of the studies have been on motor imagery based BCIs. Some of these works are reviewed below.

(a) Transductive methods : The labels of the new incoming data are not available for adaptation of the classifiers in BCIs, unlike in active learning scenarios [72]. This necessitates the classifier being able to adapt the classification boundary blindly from the incoming data. Transductive and semi-supervised algorithms have been recently used as alternatives to the strenuous training effort required on the part of the user. Transductive algorithms classify the unlabeled data by optimizing a joint function of labeled and unlabeled data. A transductive version of support vector machines (SVMs), which aligns the classification boundary maximally away from the unlabeled data has been proposed for use in BCI systems [62]. Unlike a standard SVM, the optimization problem for transductive SVM is non-convex. This requires complex numerical routines, and there is no guarantee of the solution being a global optimum. On the other hand, usual semi-supervised algorithms define a classifier as a function for which an unlabeled data is essentially test data - the posterior probabilities of data being labeled are independent.

(b) Inductive methods : In inductive methods, the current data being labeled does not affect the classification boundary during the classification process (the unlabeled data will be used to adapt the classifier only after their labels are estimated). The main advantage of inductive methods over transductive methods is that the classification itself is relatively inexpensive with respect to computational complexity. This leads to BCI systems with a faster response. In a real-time BCI system, the time between classifications (i.e., while the data corresponding to the next character is recorded) can be used for adaptation. Some works reporting the use of inductive semi-supervised techniques are reviewed in the following paragraphs.

A widely used semi-supervised technique is self-training, which uses the most confident predictions from the classifier for additional labeled data [63]. In [64], authors present an SVM based adaptation technique which uses a Gaussian kernel and updates the classification boundary using an incremental model. Different Gaussian approximation methods to classify a data which outperforms SVM based and k-nearest neighbor (KNN) based methods are presented in [73]. The authors have used a Gaussian process classifier to adaptively classify the signals based on a Gaussian process model. The advantage of a Gaussian process classifier is that the tradeoff between the data-fit and penalty parameters is automatic. Hasan *et al.* [74] proposed a Gaussian mixture model based adaptation for BCI. In Gaussian mixture models, the data is assumed to be a multi-dimensional Gaussian, and an expectation maximization procedure is usually used to find out the means and the correlation matrix. Once these two are found for all the classes, finding the classification boundary is straight-forward.

In [61], a semi-supervised version of LDA is presented where the the Bayesian or LDA classifiers are adapted based on labels predicted from unlabeled data (for P300 interfaces). The means and standard deviations of the LDA classifier are

updated iteratively based on the predictions given by the boundary derived from the previous estimates. Another recent work on LDA based adaptation can be found in [75], using similar techniques for motor imagery based data. Skykasek *et al.* developed a BCI system using a variational Bayesian Kalman filtering. An a-posteriori estimate of labels from a Bayesian classifier is used in the adaptation of the classifier [54].

Another technique used in motor-imagery based BCIs is the covariate-shift adaptation [76]. Covariate shift adaptation works on the assumption that the distribution of inputs changes, while the conditional distribution of output given the input is unchanged. To reduce the instability of the technique due to large variances, the authors combine it with bootstrap aggregating (bagging) to reduce inconsistencies in view of covariate shift. Feature extraction was done by CSP. The new parameters of the classifier are found using importance-weighted LDA. Some works use a forgetting factor to limit the amount of data to be processed [54, 75] at each update.

In [63], Li *et al.* introduced a P300 BCI system using semi-supervised SVMs. They used the confident predictions of the classifier to update it. The disadvantage of this method is that SVMs are computationally complex since it involves solving a quadratic programming problem. This updation has to be performed in frequent intervals, which makes it progressively computationally intense. In a recent paper, Chen *et al.* used a boosting based scheme to achieve efficient semi-supervised learning [77], under the assumption that the feature distribution is smooth, clustered and lie in a manifold. They proved the superiority of their technique on several real-world datasets, including BCI.

In [78], an interesting method using inter-subject information and online adaptation is performed. The prior information regarding the characteristics of the P300 signal is obtained from the data from a vast pool of subjects. The average

classifier is used to build a preliminary subject specific classifier. This classifier was further adapted online using the incoming test data. The results from 10 subjects in online adaptation experiments showed that a subject can start using a P300 BCI with virtually no training.

All these studies show that good performance classifiers can be obtained even with little training data using efficient adaptation techniques. However, many of the methods are computationally complex, and uses predictions of the classifier to update itself. In Chapter 4, we propose a technique based on co-training, where we have two classifiers learning from each other, resulting in a faster adaptation through more efficient use of unlabeled data.

2.8 Control State Detection

Control state detection helps deal with the *Midas Touch* effect (BCI system getting activated even though the user is not intending it) associated with BCIs. In practical scenarios, a BCI system cannot expect the user to be giving input always - the BCI system should be able to detect if the user intents to issue a command, and should recognize that command. As it allows the user to give input at will, asynchronous BCIs are a more natural way of interacting with a machine. This increases the usability of BCI tremendously. However, asynchronous BCIs are much more demanding on signal processing and classification techniques. The state at which the user is actively giving input is called the *control state* whereas the state in which the user is idle is called *non-control state*. The challenge is in detecting if the user is in control state. Once the user is detected to be in control state, the system has to recognize the specific command the user is trying to input.

The performance of the system can be evaluated in terms of the false positive rate (FPR) and true positive rate (TPR). The FPR is the rate at which a non-control state is detected as a control state by the system whereas TPR is the

rate at which control state is detected as a control state. The goal of all research is to achieve a better tradeoff between these two quantities. The exact point of operation is application dependent. In applications such as a wheelchair where an unwanted triggering of the system can have undesired consequences, the FPRs have to be extremely low (<2%). However, for applications such a speller, a higher FPRs might be tolerable to have improved TPRs. This can also be made adaptive / user selectable / modifiable.

Asynchronous BCI has become an active field of research and encouraging results have been reported by various groups [79–81]. Recognizing the importance of asynchronous operation, one of the datasets in the BCI competition 2007 was based on self-paced motor imagery. The challenge was to identify ERPs from the continuous stream of data without cues (i.e., from a continuous stream of data [82]). Asynchronous BCIs were pioneered by Mason and Birch at Neil Squire foundation [83–85]. They have been involved in assistive technologies for the last 2 decades and have made numerous contributions to the terminology, standardization and evaluation of asynchronous BCIs. They developed several asynchronous BCIs [79, 80, 83, 84]. Their primary focus has been the development of a low-frequency asynchronous switch design (LF-ASD). Their initial design was based on voluntary movement-related potentials (VMRP). The relative power increase in the 1-4 Hz band over SMA and MI corticalareas was used to accomplish the switching action. They were able to achieve >94% accuracy with a false alarm rate of 20-30%, and a TPR of 60% at FPR of 2-3%. They reported improvements to the basic switch in a series of publications [79, 80, 86]. However, most of the systems/schemes they developed caters to only a switching action with negligible actual information transfer. Such a system, while being sufficient for applications like wheelchair control, is not good enough for computer interactive application such as a browser or a speller.

In [87], the authors describe an SSVEP system with control-state detection capability. They use a threshold of SSVEP band powers to detect if the user is gazing at one of the 4 possible stimuli. They use a sliding window of 2s length, with estimations done once every 250 ms to check for a valid detection. Frequency in the range 37-40 Hz were used. Control state classification accuracies of 65-100% were obtained for the 6 subjects who participated in this study, with ITRs ranging from 9.4 to 45 bits/min. [88] proposes an inhibitor that keeps the BCI system turned off till specific conditions are met. They detect the initiation of control state when certain brain activity conditions are met (for example, stability in β band). They show that it enhances the usability of the BCI system. An asynchronous speller based on imagined hand and foot movements was proposed in [89].

However, very few P300-based asynchronous systems have been reported. Zhang *et al.* developed an asynchronous P300 speller which is able to communicate at an average of 20 bits/min, and an FPR of 1 event/min [51]. They achieved asynchronous control by setting a threshold for the likelihood derived from a probabilistic model of P300 classifier scores. Another work on asynchronous P300 interfaces has been recently reported in [90]. They also use a threshold based system for distinguishing between control state and non-control state. The non-control state tasks include watching and listening to a movie, computation and fixating on a cross hair at the screen center. The methodology used for discrimination of control and non-control states is based on the statistical distribution of classifier scores, similar to that in [51], with the threshold determined from the analysis of offline data collected from subjects. Nevertheless, P300-only systems are still vulnerable to environmental factors. For example, a very loud sound can cause a strong P3a to be elicited, which might generate a classifier score well above usual thresholds. This might inadvertently activate the system, with undesirable consequences. To minimize such false activations, the detection thresholds will have to be kept very

high, which might make the system hard to activate.

In Chapter 5, a hybrid system has been proposed, which uses different potentials for control state detection (SSVEP) and data transfer (P300). Since two relatively independent responses are used, the control state detection is not affected by the statistics of the P300 signal. The system is likely to be less subject to non-stationarity effects and environmental conditions, and hence, provides a robust control state detection.

Chapter 3

BCI System Implementation

3.1 System Architecture

A custom, flexible BCI system developed for performing the experiments detailed in the subsequent chapters is described in this chapter. This system can perform as a canonical P300 speller in the offline as well as online modes, and can be hybridized with SSVEP to impart desirable features.

The BCI system makes use of a 24 channel EEG acquisition device from ANT-Neuro. The EEG signal is recorded using Ag/AgCl electrodes. The amplifier supports all the 19 channels in the standard 10-20 system, and 4 additional channels. The recorded signal is amplified by a factor of 1000 by the amplifier (common mode rejection ratio $>$ 110dB). The amplified signal is digitized by sampling at rates (F_s) of 256 Hz, 512 Hz or 1024 Hz, with a resolution of 22 bits (71.5 nV / bit). The device is connected to the USB port of the PC though an optical fibre interface. A photograph of a user operating our BCI system is shown in Fig.3.1.

The system is implemented as a multi-threaded program implemented in Visual C++. ANT's neurofeedback toolbox [91] provides an ActiveX control for communication with the acquisition device. Our program acts as the ActiveX client, using

Figure 3.1: A user operating the BCI system

the interface methods to control the acquisition device. Every 250ms, the ActiveX interface triggers a callback function, which is used to obtain a packet of data.

The interface which is the speller paradigm [21] is handled by a second thread. The interface is implemented using SFML (Simple and Fast Multimedia Library) - a multimedia library providing hardware accelerated graphics using OpenGL as the back-end. For the display, we used a 19-inch CRT monitor with a vertical refresh rate of 120 Hz. The speller consists of 36 characters, arranged as a 6×6 matrix with characters A-Z and 0-9. Rows and columns are highlighted in a random order such that all rows and columns are highlighted once in every round. An area of size 1024x768 pixels was used for the display. Once a character is recognized by the system, it is displayed on the bottom of the window (only in online experiments). There is also a provision to show the desired character to be input in a separate color so that the user does not have to remember it. After each round, the order (the order of highlighting of rows/columns) and timing (the precise times at which each blink took place) information is also stored.

A third thread sends the data along with a time-stamp to a data processing unit (running on the same/another computer) through TCP/IP. After the completion of each round, a packet containing the highlighting order/timing information is also sent. To ensure precise timing, all the time-stamps are recorded from the same timer. A fourth thread waits for decisions (again received through TCP/IP from the processing unit) and passes it to the display interface.

The data processing unit is also based on VC++, interacting with Matlab through an ActiveX interface. The program receives the data through TCP/IP, and passes it to Matlab, which stores the data and processes it in real-time using various pre-processing, feature extraction and classification algorithms. Matlab passes the class information (detected character) back to the C++ program, which relays it to the display program.

3.2 Performance Analysis of the Basic System

3.2.1 Experimental setup

EEG from 7 channels (Cz, C3, C4, Pz, P3, P4 and Oz) were recorded following the standard 10-20 system at a sampling rate of 256 Hz. Electrode AFz was used as the ground, and linked-ear was used as the reference. All electrodes used were passive and unshielded, and the impedances were kept below 10KΩ throughout the experiment. The experiment was conducted in a laboratory environment, with sound absorbent screens to enable the user to concentrate better, and without electro-magnetic shielding.

Off-line experiments were conducted on five healthy subjects aged 22-27; four males and one female. Subjects 1 and 4 had some prior experience with P300 BCIs whereas the other three were BCI-naive. Each subject performed an experiment of 72 characters, each repeated for 20 rounds, with an ISI of 175 ms. An ICG of 1 second was provided to enable the subject to shift his/her attention to the next character. In our experiments, the target character was highlighted so that the user does not have to memorize any character order. It also helps to minimize the possibility of character positional biases in the P300 signal by allowing the usage of random characters as targets.

3.2.2 Data Analysis

As most of the discriminant information in the P300 resides in lower-frequencies, the collected data is zero-phase (forward-backward) bandpass filtered between 0.5 Hz and 12 Hz using a Butterworth filter of order 3. To reduce the feature size, it is down-sampled to 32 Hz, and the data for a duration of 0.7 seconds (23 samples) from the start of the stimulus is considered to belong to that particular epoch. A 161-dimensional feature vector is constructed by the concatenation of the data thus

obtained, from all the 7 channels. For cross validation, we consider the 1440 rounds as composed of 144 characters with 10 rounds/character. The optimum number of rounds to be chosen is usually a trade-off between the classification accuracy and the ITR, and varies from person to person. Performance evaluations with $n^R = 1$ to 10 were done, and the corresponding bit rates were calculated.

Results for only 3 subjects using FLDA classifier are presented in this chapter. The 144 characters were split into 4 continuous sections of 36 characters each. The data from one section (360 rounds) was used for training the classifier. Data from the other three sections were used as the test data. This process was repeated for all the sections and averaged, thus performing a 4-fold cross-validation. For each iteration, classification accuracy was determined for n^R values from 1 to 10, and the corresponding ITRs were calculated as described using Eq. (2.10) and Eq.(2.11).

3.2.3 Results

Figures 3.2, 3.3 and 3.4 show the cross-validation accuracies and bit-rates for the 3 subjects. The subjectivity in trade-off between classification accuracy and ITRs can be clearly seen from these figures. When $n^R = 1$ (3.1s for detecting a character), Subject 2 is able to achieve an ITR of 48.38 bits/min at an accuracy of 66%. The mean ITR is 38.69 and the mean accuracy is 57%. A mean accuracy of 77% is achieved when $n^R = 2$, the corresponding ITR being 37.35 bits/min. The mean accuracy increases further to 86% when $n^R = 3$, at the expense of ITR, which drops to 32 bits/min. For computer-interaction applications, this accuracy should be good enough. For applications such as wheel chair direction control too, a reasonably fast response is desirable, and $n^R = 3$ might be a good compromise between speed and accuracy. However, for issuing a command such as the one to start a wheelchair, a higher accuracy is desirable, as a false start can have very

Table 3.1: State of the art P300 BCIs

ITR	Accuracy	Publication	Highlight
23.3	79.5%	Hilit *et al.*, 2004 [27]	Maximum likelihood (ML) classifier.
29.3	>95%	Hoffman *et al.*, 2007 [2]	Does not use row/column paradigm.
29.4	87.5%	Lenhardt *et al.*, 2008 [33]	Adaptive LDA with interface adaptation.
26.7	96%	Jin *et al.*, 2011 [92]	Hybrid interface - P300 + VEP.
46.4	90%	Frye *et al.*, 2011 [32]	Modified calibration technique.
37	98%	Jin *et al.*, 2011 [31]	Adaptive flashing patterns (rounds).
32	86%	Current basic system	

undesirable consequences. When $n^R = 10$, a very high accuracy of 98% could be achieved, though the ITR is only 13.5 bits/min. It should be noted that 22s is required to input a command (character) in this case. The results of our basic non-adaptive system is comparable to that of the state of the art BCI systems as shown in Table. 3.1.

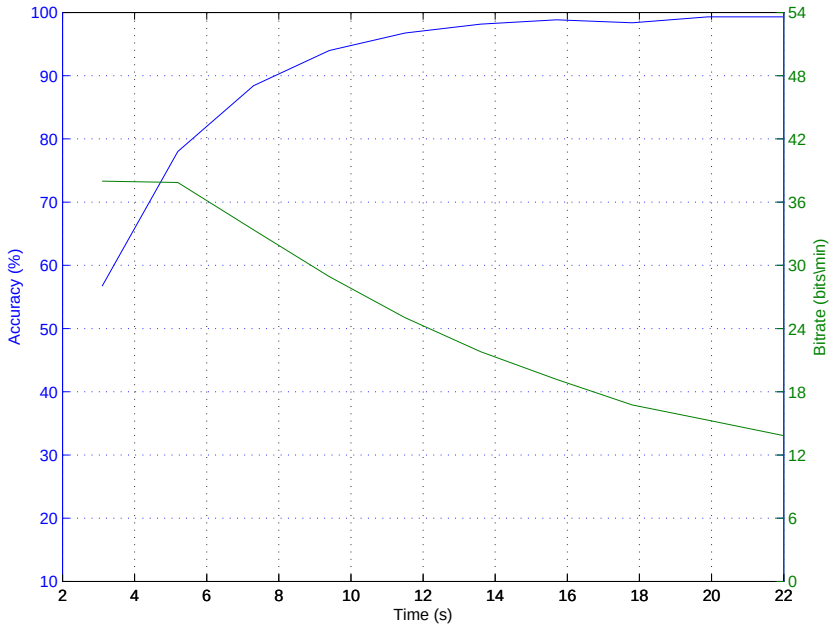


Figure 3.2: Cross-validation results for subject 1.

However, the base system described here does not have the capabilities for adaptation and control state detection. Techniques developed for imparting these to the system are described in the following chapters.

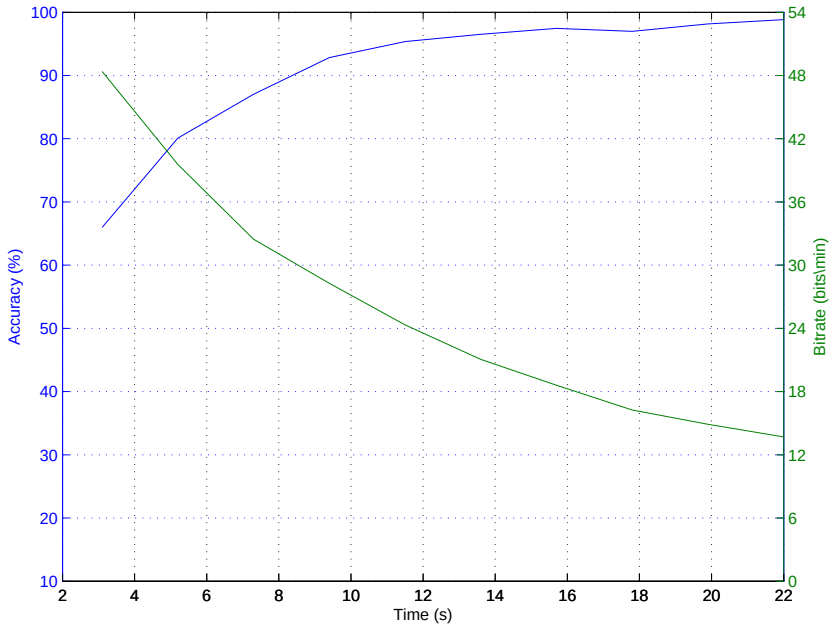


Figure 3.3: Cross-validation results for subject 2.

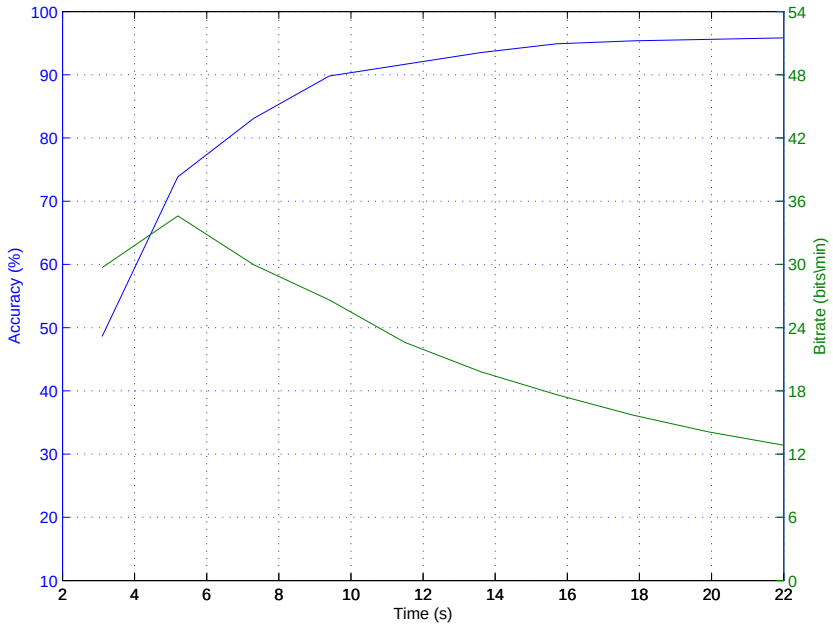


Figure 3.4: Cross-validation results for subject 3.

Chapter 4

A Two-Classifier Co-Training Approach for Adaptation in P300 BCIs

4.1 Introduction

A co-training based method for fast adaptation in a P300 BCI is introduced in this chapter. Classical co-training, pioneered by Blum and Mitchell [93] is a popular method for semi-supervised learning. This method requires two redundant and sufficient views of the data, i.e., two sets of independent features both of which have the classification information. In co-training, the most confident predictions by a classifier trained on one view (one set of features) is used to train the other, and vice versa. This dual-view requirements cannot be met in most practical scenarios, including BCI. Goldman and Zhou [94] later showed that labels generated through two different classification methods using same features can also be used to generate additional data, thus doing away with the multi-view requirement. Such a two-classifier co-training based approach is introduced here to reduce the training effort, combining FLDA and the Bayesian linear discriminant analysis (BLDA). The algorithm exploits the difference between the classifiers to generate

different labels for the data. Recently, a mathematical reasoning for the success of co-training style algorithms were given by Zhou *et al.* [95]. They proved that the success of co-training based algorithms is higher when the difference between the classifiers is maximized [96].

The proposed method is described in Section 4.2. Section 4.3 details the experiments and data analysis, followed by results and discussions in Section 4.4. The chapter is concluded with some remarks in Section 4.6.

4.2 Co-Training Method

Let the data from the j^{th} trial, $\mathbf{x}_j = [x_{1j}, x_{2j}, \ldots, x_{gj}]^T$ be the feature vector of length g for the classification problem, with x_{ij}'s, $i = 1, 2, \ldots, g$ denoting the individual features and let $y_j \in \{-1, 1\}$ be the corresponding labels. Let $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_l, \mathbf{x}_{l+1}, \ldots, \mathbf{x}_n\}$ be the set of all n data points in feature space, of which l points have known labels given by, $Y = \{y_1, y_2, \ldots, y_l\}$. The semi-supervised classification problem can be defined as follows: Given the data set $S = L \cup U$, where $L = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_l, y_l)\} \subset X \times Y$ is the labeled data set and $U = \{\mathbf{x}_{l+1}, \mathbf{x}_{l+2}, \ldots, \mathbf{x}_n\} \subset X$ is the unlabeled data set, find a mapping $h^* \in H$ which holds for the entire S and gives a perfect generalization, where H: $X \to Y$ denotes the set of all classifiers. This will be hard to realize in most practical applications where the data is generally noisy; also for small l, the mapping will be less accurate. Co-training method uses two initial classifiers, namely $h_1^0 \in H$ and $h_2^0 \in H$, trained on L, and iteratively updates them, with h_1^{i+1} and h_2^{i+1} hopefully providing a better mapping than h_1^i and h_2^i, where i is the iteration number.

The algorithm can be summarized as follows: Given the initial training data (L) and the unlabeled data (U),

1. Obtain the initial classifiers h_1^0 and h_2^0, using the training data $L_1^0 = L_2^0 = L$;

and set $i = 1$.

2. Take u number of unlabeled instances from U, and label them using h_1^{i-1} and h_2^{i-1}.

3. Construct a new labeled set L_1^i by combining L_1^{i-1} and the labeled data given by h_2^{i-1}, and L_2^i by combining L_2^{i-1} and the labeled data given by h_1^{i-1} in the previous step.

4. Obtain the updated classifiers h_1^i and h_2^i using L_1^i and L_2^i. In certain cases, only a fraction of the most confident among the u labels predicted by each classifier is used for updating.

5. Increment i and repeat steps 2 to 5 till stopping criterion is met.

6. Stop the training if all the unlabeled data has been classified or if the confidence improvement due to the addition of unlabeled data is minimal.

Several classifiers for classification of P300 have been reported in the literature which include FLDA [97], SVM [53], BLDA [2, 98] etc. The classifiers used (for implementing h_1^i's and h_2^i's, respectively) are BLDA and FLDA for the following reasons:

- In the preliminary experiments, BLDA and FLDA gave very good accuracies. Some studies have reported that the algorithms give accuracies comparable to that of SVMs [99].

- Both are computationally simple and do not require complex cross-validation procedures for tuning their hyper-parameters. Although BLDA uses a data dependant expectation-maximization type algorithm for hyper-parameter optimization, the empirical complexity was found to be very low, especially as compared to competing classifiers like SVMs.

- The two classifiers gave reasonably different separating planes, which is a crucial factor in co-training approaches. Since BLDA uses an entirely different optimization method, the biases of the two algorithms are different. Such a diversity is crucial from the point of co-training. In our experiments, it was observed that reasonable diversity was maintained, though it is inevitable that the classifiers produce closer and closer predictions as the co-training proceeds with more and more unlabeled samples.

A detailed description of FLDA can be found in Section 2.6. A brief description of the BLDA algorithm is given below.

4.2.1 BLDA

BLDA uses an entirely different approach for optimizing the weights. Instead of committing a particular value of the projection vector, it creates the posterior distribution using the Bayesian criterion. The BLDA implemented in this work is similar to the one described in [2]. More general descriptions of this method can be found in [100] and [101]. The basic assumption in BLDA is that the regression targets

$$\mathbf{y} = \mathbf{w}^T\mathbf{X} + \mathbf{n}, \tag{4.1}$$

where $\mathbf{n}$ is the noise vector. For simplicity and mathematical tractability, the noise is assumed to be Gaussian. Therefore, the likelihood function can be written as

$$p(D|\beta', \mathbf{w}) = \left(\frac{\beta'}{2\pi}\right)^{\frac{l}{2}} e^{-\frac{\beta'}{2}||\mathbf{w}^T\mathbf{X}-\mathbf{y}||^2}, \tag{4.2}$$

where D denotes the pair $(\mathbf{X}, \mathbf{y})$, β' denotes the inverse variance of noise, and l is the number of examples in the training set. For Bayesian inference, we specify a prior distribution for the weight vector $\mathbf{w}$. The expression for the prior distribution

(which is assumed to be Gaussian) is

$$p(\mathbf{w}/\alpha') = \prod_{i=1}^{g} \left(\frac{\alpha_i'}{2\pi}\right)^{\frac{1}{2}} e^{-\frac{1}{2}\left(\mathbf{w}^T \mathbf{I}'(\alpha')\mathbf{w}\right)}, \tag{4.3}$$

where α_i''s are the hyper-parameters (which signifies the inverse relevance of each feature), and $\mathbf{I}'(\alpha')$ is a $g \times g$ dimensional square matrix, with α_i''s along the diagonal.

Using Bayes theorem, it can be shown [2] that the posterior is also a Gaussian, with mean ($\mathbf{m}'$) and the covariance ($\mathbf{C}$) given by

$$\mathbf{C} = \beta' \left(\beta' \mathbf{X}\mathbf{X}^T + \mathbf{I}'(\alpha')\right)^{-1}, \tag{4.4a}$$

$$\mathbf{m}' = \beta' \mathbf{C}\mathbf{X}\mathbf{y}. \tag{4.4b}$$

The predictive distribution of the target y' for previously unseen $\mathbf{x}'$ is also Gaussian, the mean (μ') and variance (σ'^2) of which is given by

$$\mu' = \mathbf{m}'^T \mathbf{x}', \tag{4.5a}$$

$$\sigma'^2 = \frac{1}{\beta'} + \mathbf{x}'^T \mathbf{C}\mathbf{x}', \tag{4.5b}$$

of which only the mean is being used for the class predictions. The likelihood, $p(D|\beta', \alpha')$ is given by marginalizing Eq.(4.2) as

$$p(D|\beta', \alpha') = \int p(D|\beta', \mathbf{w}) p(\mathbf{w}|\alpha') d\mathbf{w}. \tag{4.6}$$

The update equations for α' and β' are obtained by maximizing the log likelihood, and setting the partial derivatives with respect to α' and β' to 0 as

$$\alpha'_i = \frac{1}{c_{ii} + m'^2_i}, \tag{4.7a}$$

$$\beta' = \frac{g}{\mathrm{tr}(\mathbf{X}\mathbf{X}^T)\mathbf{C} + ||\mathbf{m}'^T\mathbf{X} - \mathbf{y}||^2}, \tag{4.7b}$$

where c_{ii}'s are the diagonal elements of $\mathbf{C}$, m'_i's are the elements of $\mathbf{m}'$, and $\mathrm{tr}(\cdot)$ denotes the trace of matrix. Eqs. (4.4a), (4.4b) and Eqs. (4.7a), (4.7b) form a set of coupled equations, which can be iterated to optimize the values of α' and β'. Once the optimization is complete, mean of the posterior ($\mathbf{m}'$) given by Eq.(4.4b) is taken as the optimum value of $\mathbf{w}$. The character detection then proceeds in a manner similar to that with FLDA.

Depending on the semi-supervised strategy employed and the classifier giving the final output, we have the following four different classifiers:

1. self-training BLDA (SBLDA)
2. self-training FLDA (SLDA)
3. co-training BLDA (CBLDA) - in which the output is taken from BLDA classifier which is co-trained with FLDA
4. co-training FLDA (CLDA) - in which the output is taken from FLDA classifier which is co-trained with BLDA

Performance analysis of these four algorithms are given in Section 4.4.

4.2.2 Confidence Criterion

The co-training process is repeated once 100 rounds of fresh unlabeled data is made available to the classifier, as it was empirically found to give reasonably

good performances while avoiding frequent updating of the classifier. The fraction of the classified data that is being added to the training data in each iteration of the algorithm is determined by a confidence criterion. In our work, the z-score defined in Eq.(4.8) is the metric used for calculating the confidence of predictions. The z-score corresponding to the i^{th} character is given as

$$z_i = \frac{y_{\max,i} - y_{\text{mean},i}}{\sigma_{y_i}}, \tag{4.8}$$

where $y_{\max,i}$, $y_{\text{mean},i}$ and σ_{y_i} are the maximum, mean and the standard deviation respectively of the averaged scores corresponding to the rows/columns associated with the i^{th} character detection.

4.2.3 Evaluation Criteria

The BCI performance evaluation was done as described in Section 2.6.1. The effective number of bits detected is calculated using Eq. (2.10). The information transfer rate is calculated using Eq. (2.11). Statistical significance tests were conducted to facilitate a conclusive interpretation of the results obtained. A lilliefors test on random ensembles of classification accuracies revealed that its distribution is not always Gaussian, and hence t-test is not appropriate [102]. Hence, for all the comparisons on classification accuracies done in this chapter, the sign test, which makes little assumptions regarding the data distribution, is used. Sign test is performed as follows :

$$n_{true} = \frac{1}{2}\left(n_{iter} - \sum_{i=1}^{n_{iter}} sgn\left(\mathrm{CA}_1^i - \mathrm{CA}_2^i\right)\right), \tag{4.9}$$

where CA_1^i and CA_2^i are the classification accuracies for method 1 and method 2 respectively and n_{iter} is the number of cross-validation iterations. The one-tailed p-value [102] (for the null hypothesis that method 1 does not yield a better median

accuracy than method 2) is calculated using the binomial cumulative distribution function (CDF), with n_{true} trials turning out to be true, out of the n_{iter} binomial trials.

4.3 Data Recording and Analysis

4.3.1 Off-line Experiments

Details of experiments and pre-processing are described in Section. 3.2. Each of the 5 subjects underwent a session of 72 characters. An ISI of 175 ms was used, and each character was repeated for 20 rounds. Having 20 rounds/character enables analysis using n^R = 1,2,4,5,10 and 20. However, results for analysis using $n^R = 1$ and 2 only are presented in this chapter.

4.3.2 Cross-Validation

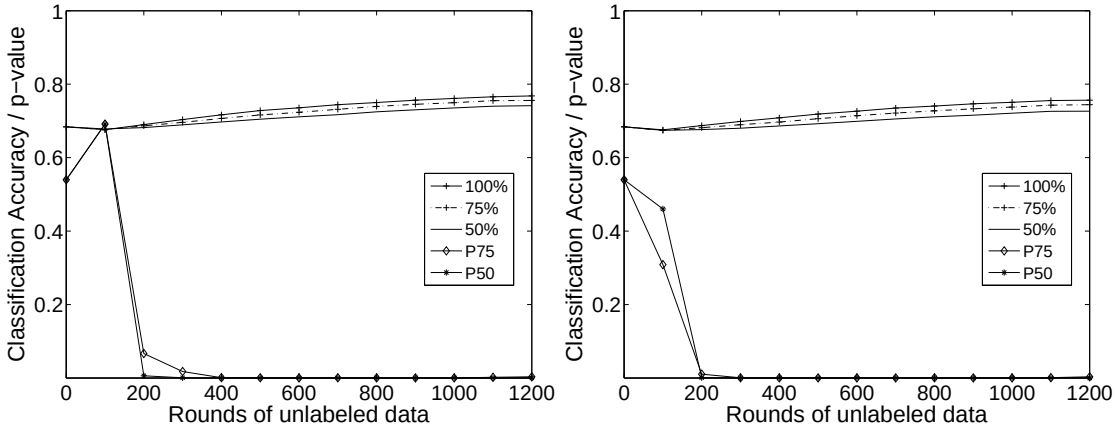
To evaluate the performance of co-training, an extensive cross-validation analysis is carried out. First, the data is shuffled 100 (n_{iter}) times randomly with the constraint that the data for any one character is kept together, to obtain 100 different data ensembles. For each ensemble, first l rounds of training data is used to train an initial classifier. The rest of the data is treated as unlabeled data and is progressively added in batches of 100, and the self/co-training algorithms are applied. The means, standard deviations and p-values are calculated as appropriate using the results from the 100 ensembles. This scheme gives us a realistic measure of the performance of the algorithm and has been used in many previous works involving semi-supervised learning [96]. The disadvantage of such a scheme is that the scrambling of data forces the algorithm to ignore any adaptation effects. However, taking it into consideration would make the data requirement for performance analysis impractically huge. The validation scheme is applied for all

(l, n^R) combinations, maintaining the same orders of permutations of the data in each case.

4.4 Results and Discussion

The algorithms were run for eight different configurations of the initial training data (l) and the number of rounds (n^R) used for the detection of each character. They are (l, n^R) - (40, 1), (60, 1), (60, 2), and (300, 2). The measure that is used to determine the most confident instances for re-training the classifier is the z-score, given in Eq.(4.8). Analysis of self-training and co-training using 50%, 75% and 100% of the most confident predictions for updating the classifier was done. A sample result when $l = 60$ and $n^R = 2$, averaged over all the subjects is given in Fig. 4.1. The results clearly demonstrate a better performance when all the labels are used for re-training the classifiers. The statistical significance of this conclusion is established from the low p-values for the null hypothesis that using all the classifier predictions for self/co-training is not beneficial. A similar trend was observed for other configurations of (l, n^R) as well, for both self and co-training. Hence, all the results presented henceforth uses 100% of the classifier predictions for self/co-training.

The cross-validated results for the 4 algorithms (SBLDA, SLDA, CBLDA, CLDA) are summarized in Figs. 4.2 - 4.8. In most of the discussions that follows, only SBLDA and CBLDA are included, as these almost always gave better results than their FLDA based counterparts, the SLDA and CLDA. Also, since our work is meant to highlight semi-supervised learning in general, and co-training in particular, such a comparison would be more appropriate.

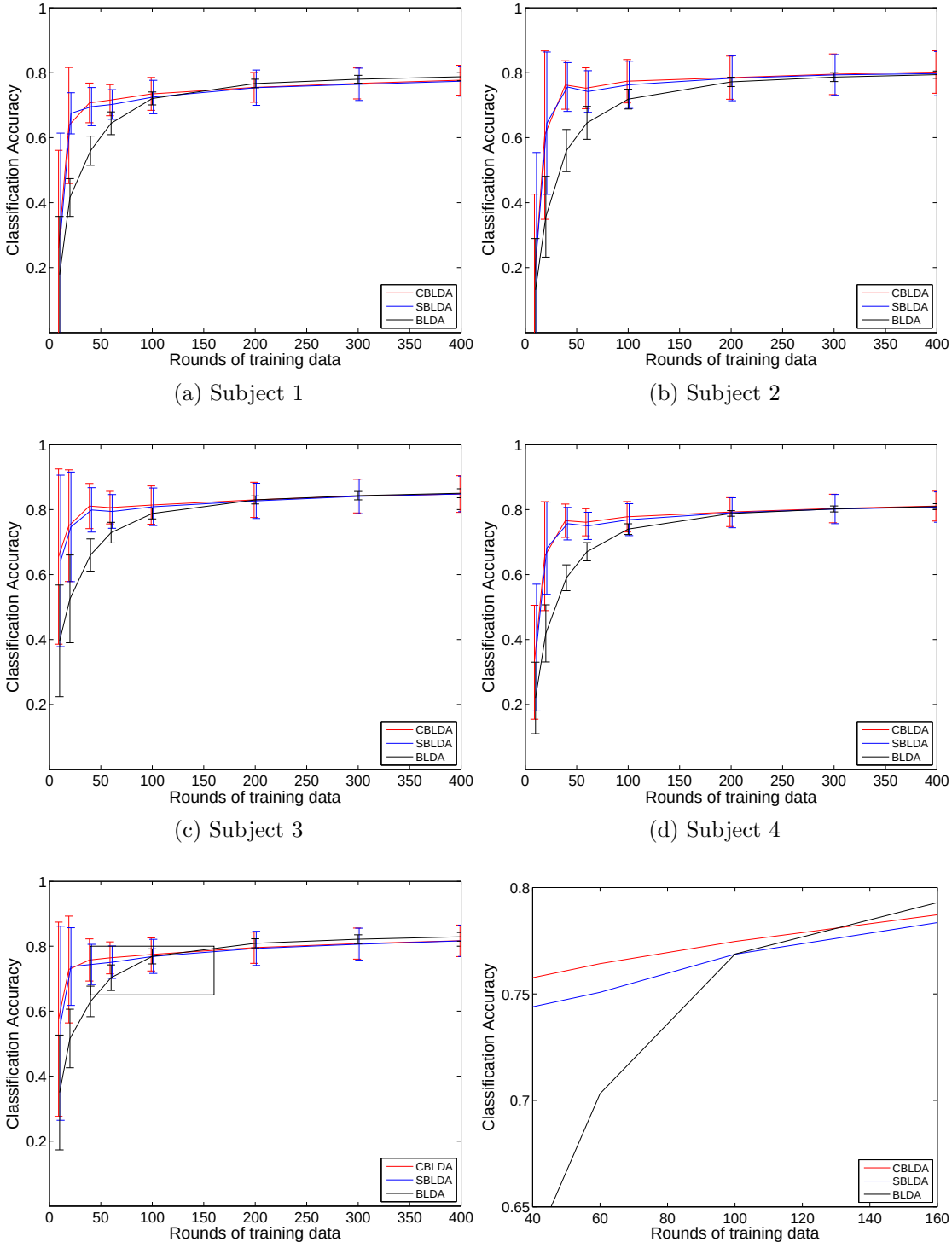


(a) Co-training (b) Self-training

Figure 4.1: Classification accuracy vs. rounds of unlabeled data for different percentages of classifier predictions used in self/co-training, for l=60 and $n^R = 2$. P75 and P50 denotes the p-values for similar performance of 75% and 50% of most confident classifier predictions as compared to using 100%.

4.4.1 Effect of Training Data

It can be seen from the results that the increase in proportion of unlabeled data leads to a significant increase in the classification accuracy, especially in situations where relatively low amount of labeled data is available. This could be expected, as empirical studies have shown that the classification accuracy increases exponentially with labeled data and linearly with unlabeled data [103]. These results can be observed in Fig. 4.2. For all the five subjects, it can be observed that when the labeled data is sufficient, addition of unlabeled data does not improve the classifier performance. For all the subjects, approximately 200 rounds of data was enough to learn a classifier which could not be improved further by semi-supervised learning, and the classification accuracy using CBLDA bettered that of SBLDA in most cases.



(a) Subject 1

(b) Subject 2

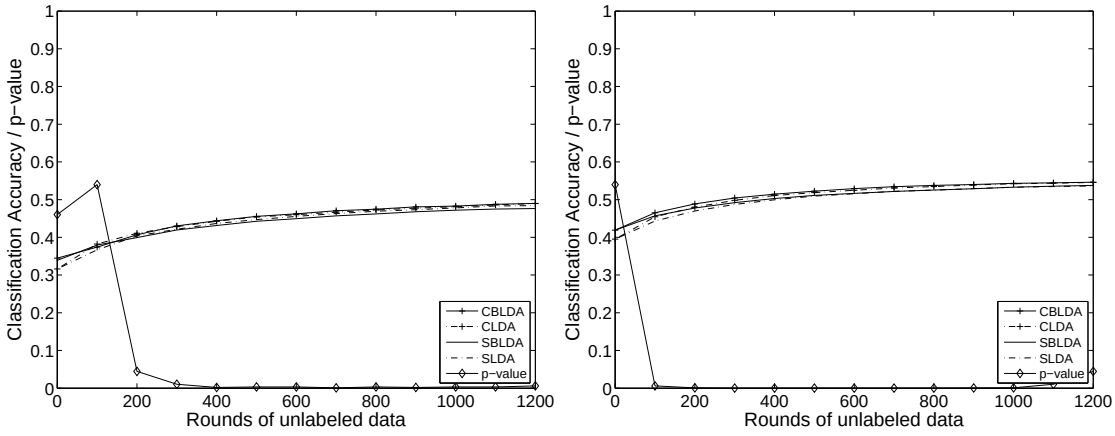
(c) Subject 3

(d) Subject 4

(e) Subject 5

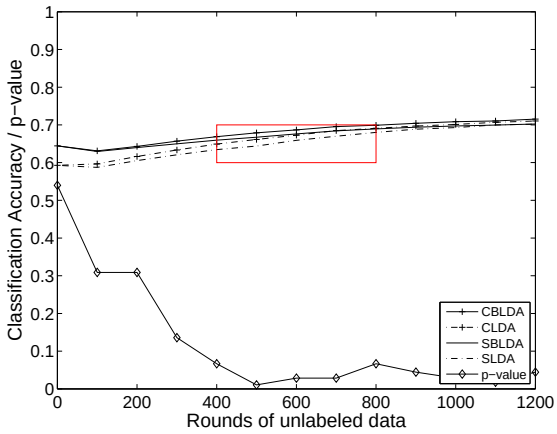
(f) Magnified version of data in the rectangle in Fig. 4.2(e)

Figure 4.2: Classification accuracy of CBLDA, SBLDA and fully supervised BLDA for various l (for $n^R = 2$), along with the bars for $\pm\sigma$ (population standard deviations, standard error of mean is $\pm 0.1 \times \sigma$).

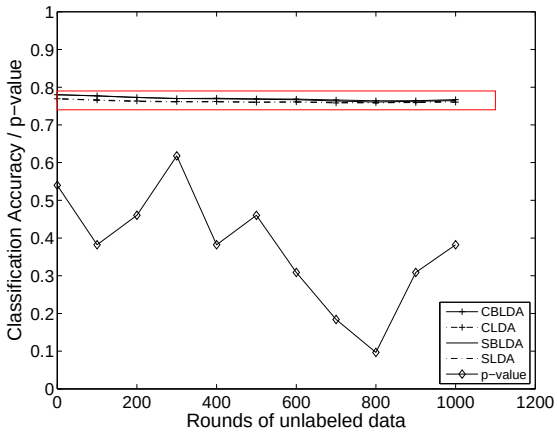


(a) $l = 40$, n^R=1

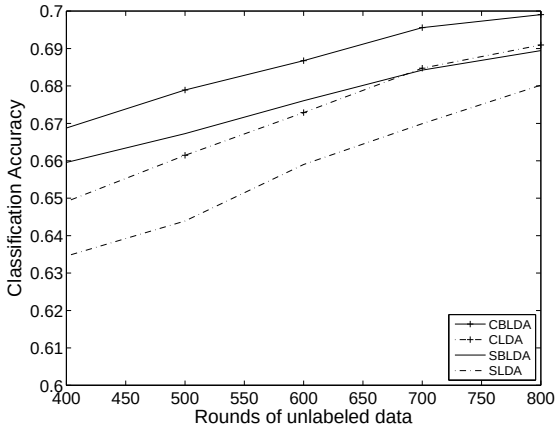
(b) $l = 60$, n^R=1



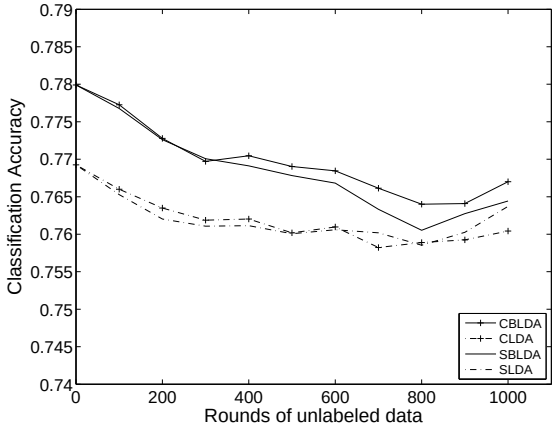
(c) $l = 60$, n^R=2



(d) $l = 300$, n^R=2

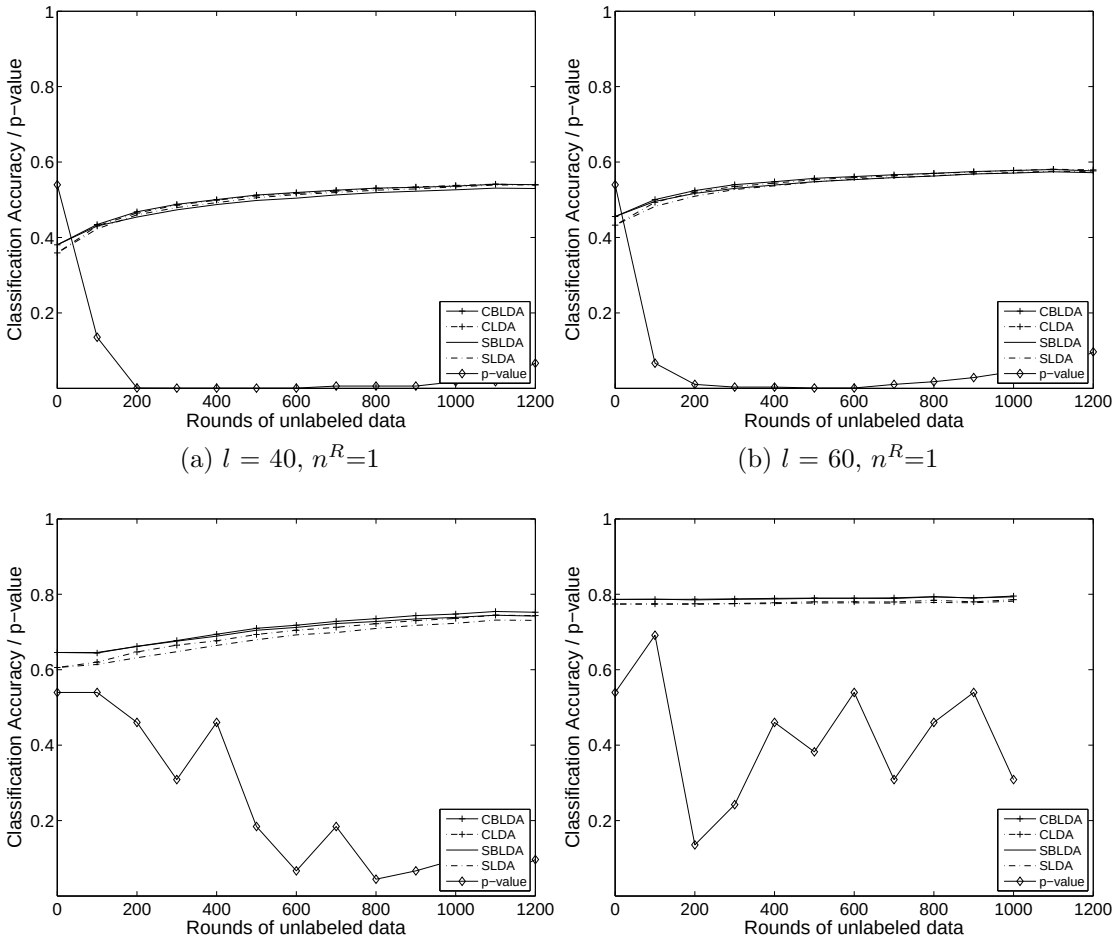


(e) Magnified version of data in the rectangle in Fig. 4.3(c)



(f) Magnified version of data in the rectangle in Fig. 4.3(d)

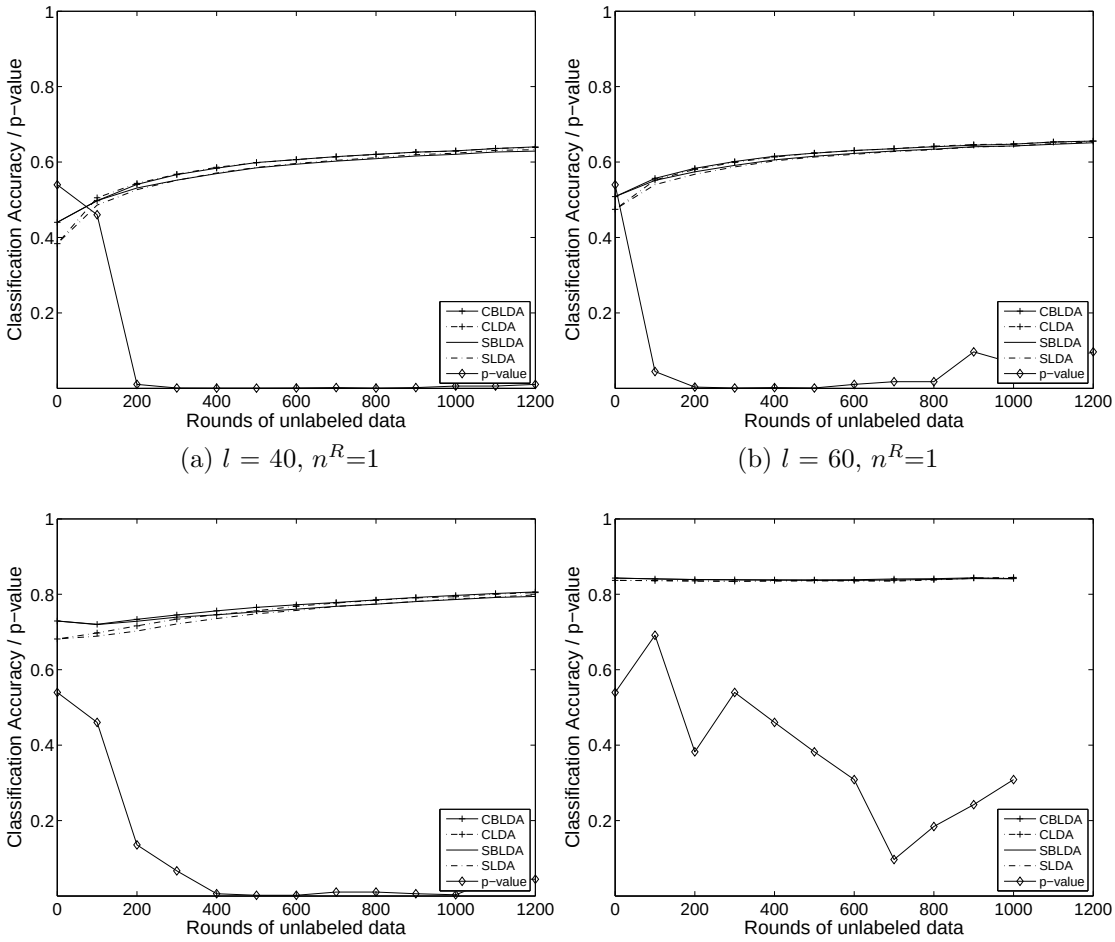
Figure 4.3: Classification accuracy vs. rounds of unlabeled data for subject 1 for various l and n^R.



(a) $l = 40$, n^R=1 (b) $l = 60$, n^R=1

(c) $l = 60$, n^R=2 (d) $l = 300$, n^R=2

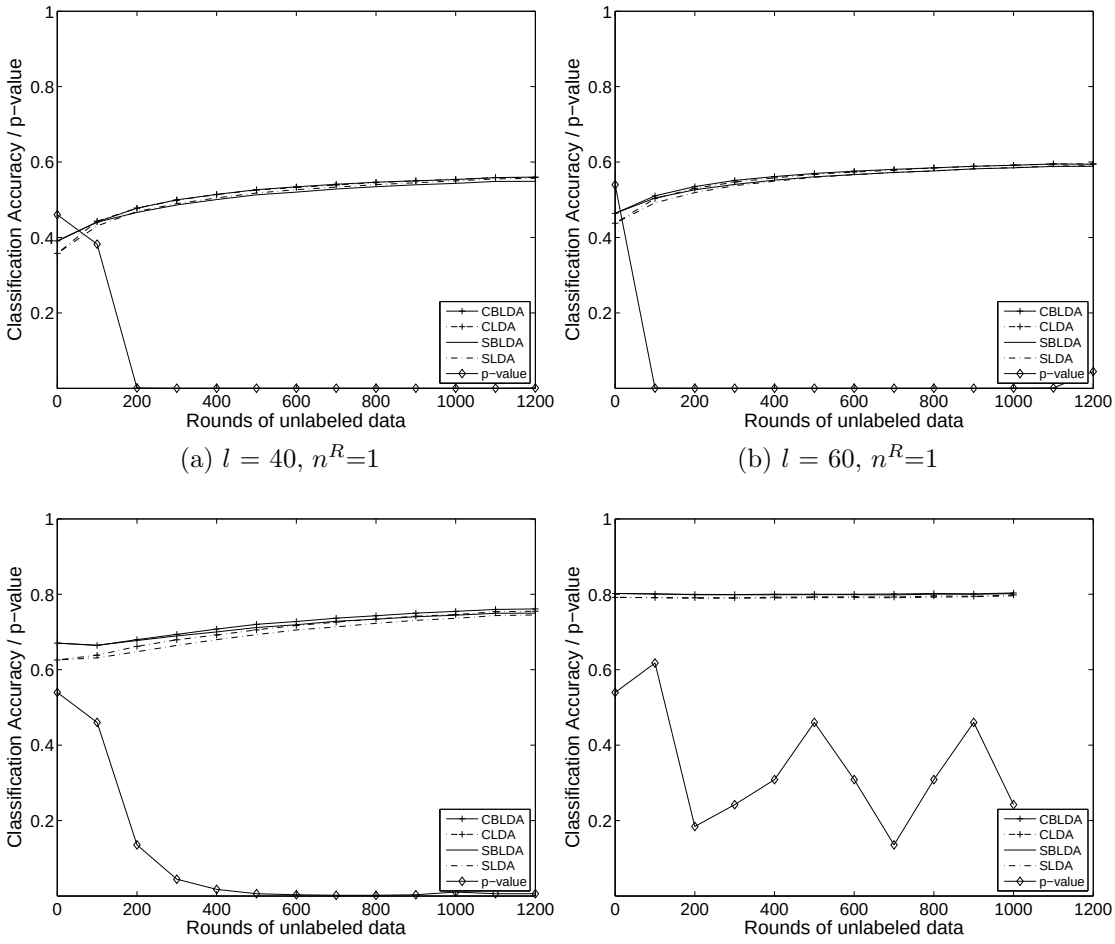
Figure 4.4: Classification accuracy vs. rounds of unlabeled data for subject 2 for various l and n^R.



(a) $l = 40$, n^R=1 (b) $l = 60$, n^R=1

(c) $l = 60$, n^R=2 (d) $l = 300$, n^R=2

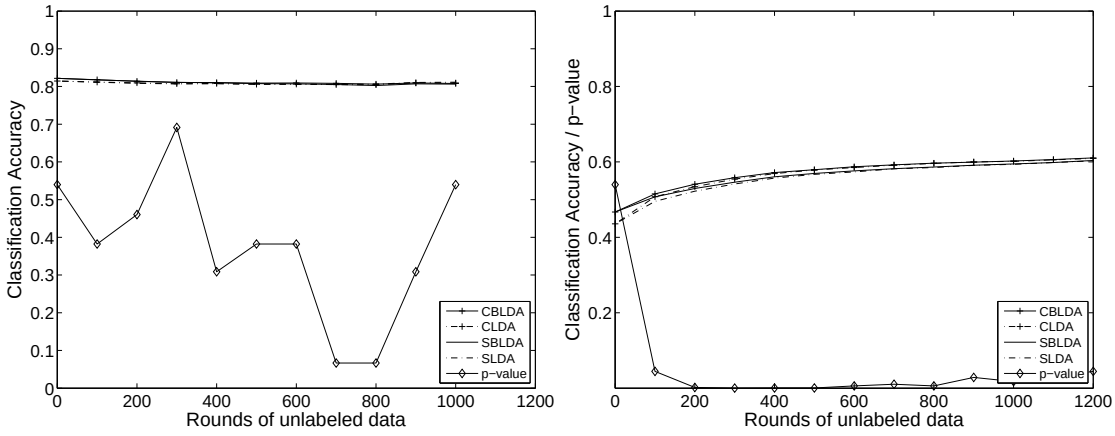
Figure 4.5: Classification accuracy vs. rounds of unlabeled data for subject 3 for various l and n^R.



(a) $l = 40$, n^R=1 (b) $l = 60$, n^R=1

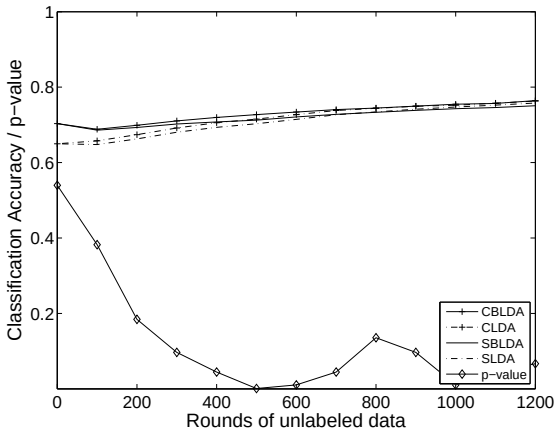
(c) $l = 60$, n^R=2 (d) $l = 300$, n^R=2

Figure 4.6: Classification accuracy vs. rounds of unlabeled data for subject 4 for various l and n^R.

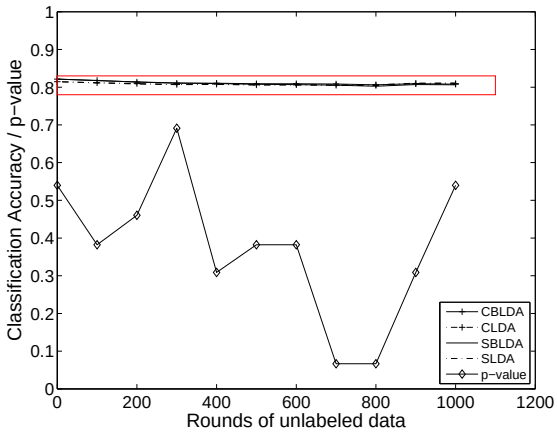


(a) $l = 40$, n^R=1

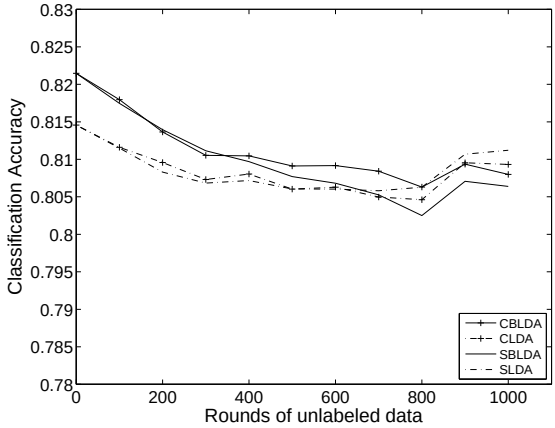
(b) $l = 60$, n^R=1



(c) $l = 60$, n^R=2



(d) $l = 300$, n^R=2



(e) Magnified version of data in the rectangle in Fig. 4.7(d)

Figure 4.7: Classification accuracy vs. rounds of unlabeled data for subject 5 for various l and n^R.

4.4.2 Effect of Unlabeled Data

The effect of unlabeled data can be seen from Figs. 4.3-4.7 for various configurations of l and n^R. It can be seen that in most cases, the addition of unlabeled data helps increase the accuracy. However, as the ratio of unlabeled data to labeled data increases, the performance improvement decreases and gradually becomes minimal. A mathematical reasoning for this effect can be found in [95]. It can be seen in Figs. 4.3d, 4.4d, 4.5d, 4.6d and 4.7d that the addition of unlabeled data when sufficient training data is available does not improve the classification performance of the system. For subject 1 and 5, when l= 300 (Figs. 4.3f and 4.7e), addition of unlabeled data in fact degrades the performance. This effect has been reported on semi-supervised learning on different datasets by Cohen *et al.* [104].

In cases where there is an improvement, CBLDA almost always gives a better improvement over SBLDA. It can be seen from Figs. 4.3 - 4.7 that the performance of CBLDA is significantly ($P<0.05$) better than SBLDA for all subjects except when l=300, in which case semi-supervised learning offered no significant improvement over fully supervised classification. For the configuration (l = 40, n^R = 1), CBLDA gives a performance improvement of 13.2 bits/min more than supervised classifiers, and 1.7 bits/min more than SBLDA for subject 1; and 16.4 bits/min more than supervised classifiers, and 1.2 bits/min more than SBLDA for subject 2. The improvement is 21.0 and 17.2 bits/min over supervised classifiers, and 1.5 bits/min and 1.4 bits/min over SBLDA for subject 3 and 4 respectively. For subject 5, the algorithm achieved an increase of 18.5 bits/min over supervised classifiers and 1.6 bits/min over SBLDA. From these results, we can see that CBLDA outperforms SBLDA in most situations, though the actual amount of increase is not large. The final bit rates averaged over all the subjects is approximately 37 bits/min; which is 17 bits/min more than the initial classification accuracy. This was achieved with just 40 rounds of labeled data, which corresponds to a training

Subject	p-Mean	p-Fin
Sub 1	**0.005**	0.210
Sub 2	0.109	0.207 (sign)
Sub 3	**0.018**	0.260 (sign)
Sub 4	**0.002**	0.163
Sub 5	**0.003**	0.448

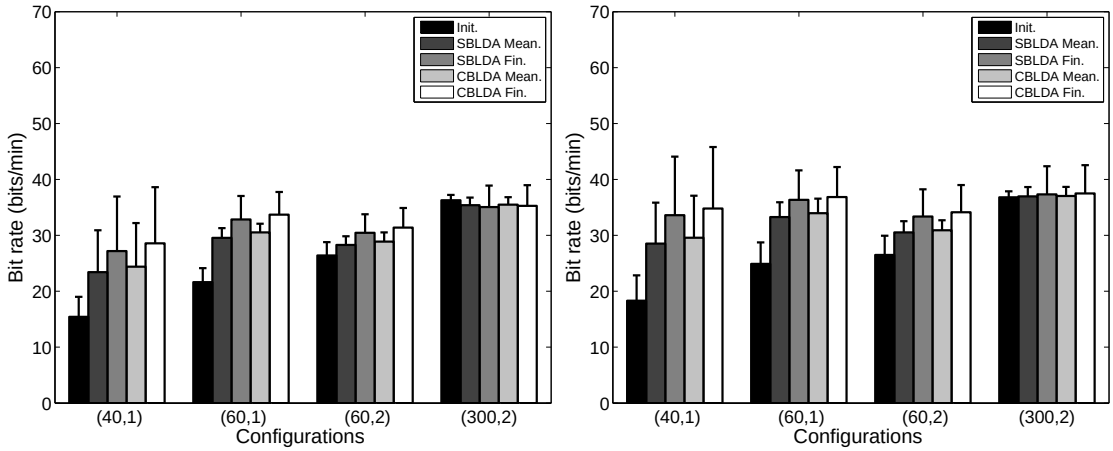
Table 4.1: Table showing p-values for CBLDA vs SBLDA for (300,2). p(Mean) and p(Fin) are the p-values given by t-test (and sign test for cases where distributions are found to be non-Gaussian through lilliefors test) for the comparison of mean and final values respectively for CBLDA vs SBLDA. Cases where CBLDA is significantly better than SBLDA are highlighted.

time of about 90 seconds. This compares favorably with most state of the art BCI systems, where average bit rates of 30-40 bits/min are achieved with several minutes of training data (please refer to Table. 3.1).

If unlabeled data was detrimental to classification performance (4^{th} group in Fig. 4.8 for all subjects except subject 2), CBLDA reduced the initial accuracy only slightly as compared to SBLDA in most cases. Such a degradation can be observed in subjects 1 and 5 with the addition of unlabeled data when $l = 300$. Table. 4.1 shows the t-test derived p-values (and sign test for cases where the distribution is non-Gaussian) obtained for comparison of CBLDA vs SBLDA for the mean and final ITRs achieved. It can be seen that CBLDA delivers mean ITRs which are significantly better for all except Subject 2. However, the final ITRs achieved by both algorithms are comparable, which could be due to the degradation of co-training to self-training.

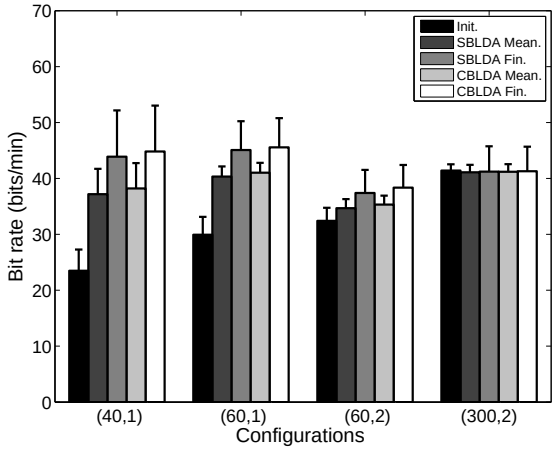
4.4.3 Stability

The stability of SBLDA and CBLDA are dependent on the initial training data used. As it follows a clustering-like approach where the previous results themselves update the distribution estimates / model parameters, the analysis is not mathematically tractable. Empirical studies have shown that if the model assumptions

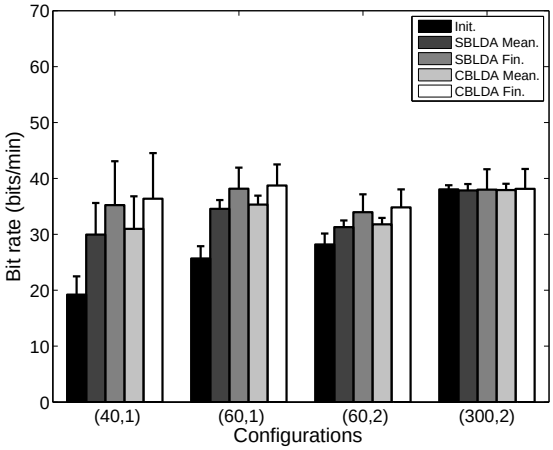


(a) Subject 1

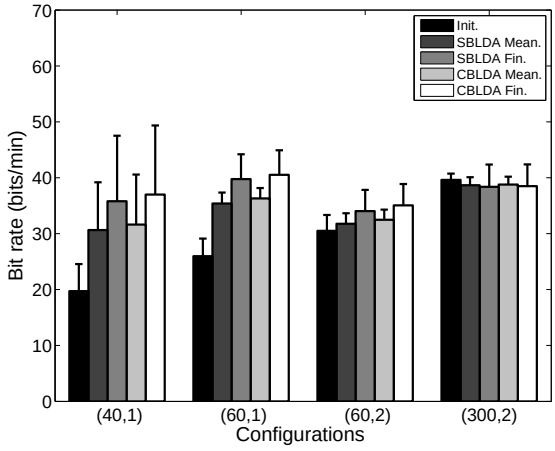
(b) Subject 2



(c) Subject 3



(d) Subject 4



(e) Subject 5

Figure 4.8: Bar chart showing the bit rates for various configurations of l and n^R. The initial bit rate (Init.); as well as the mean (Mean.) and final bit rates (Fin.) achieved are shown for each (l, n^R) configuration and for each subject. Please note that the error bars represent population standard deviations ($\pm\sigma$), standard error of mean is $\pm 0.1 \times \sigma$.

are correct and the information given by the initial training data is representative of the population, semi-supervised methods are likely to improve performance [105]. However, if the initial training data used is insufficient to train an informative classifier, divergence is likely. This could especially be the case for subjects who are not able-bodied, where the initial training data might give a less reliable classifier.

4.4.4 Subjectivity

The algorithm is found to have effective performance enhancement in all the five subjects. We can observe from the results that the algorithm is especially effective for subject 3, even when l is 40 and n^R is 1, which corresponds to a training time of less than 2 minutes. This could be due to the fact that subject 3 produces stronger P300 (which can be inferred from the fact that subject 3 gives the best performance with supervised classifiers, as can be seen from Fig.4.2, and the prediction by both the classifiers are reasonably good. Consequently, the possibility of errors reinforcing themselves catastrophically is minimized, especially in cases where the training data is low.

For subject 2, the performance of CBLDA was not significantly better than that of SBLDA (Fig. 4.4c) when $l = 60$, $n^R = 2$. For subject 1, when the training data is sufficient, degradation of accuracy with addition of unlabeled data is even more prominent (Fig. 4.3d), as compared to subjects 2, 3 and 4 (Figs. 4.4d, 4.5d and 4.6d). Subject 2 always had an increase in performance even when training data was abundant (Fig. 4.4d). For other configurations, the enhancement of accuracy in subjects 2, 3 and 4 are found to be consistently better than subject 1. These are reflected in the ITRs as well. From these results, it can be concluded that the semi-supervised learning is even more effective for subjects with a stronger P300, especially when there is a scarcity of training data.

4.4.5 Computational Complexity

Both self-training and co-training are generalized methods, and the exact complexity depends on the particular classification algorithms used in their realization. For self-training, the complexity is related to the complexity of the specific classifier used, whereas for co-training, it is determined by the sum of the complexities of the two classifiers used. FLDA and LDA mostly involves covariance matrix and eigenvalue calculations, which have a complexity of $O(l \times g^2)$, where l is the number of labelled data points and g is the length of the feature vector. FLDA takes a variable number of steps to converge (typically 15-50), and is data dependent. In both self and co-training, as more and more unlabeled data is added, complexity increases, as the classifier has to be re-trained from a bigger pool of data. One FLDA and LDA operation is required every 100 rounds (3-5 mins depending on n^R), when l gets incremented by 1200. CBLDA and SBLDA have comparable complexities due to the fact that the overall time taken for execution is dominated by BLDA, as it takes much more iterations than FLDA for convergence. One complete run of 1440 rounds (corresponding to iterative classification of approximately 1 hour of pre-processed data) using Matlab code running on a Windows Vista desktop computer with a 2.8 GHz dual-core processor and 4GB of RAM takes approximately 13.2 and 12.7 seconds respectively.

4.5 Limitations and Implementation Issues

In practical situations, there may be gradual changes in the data with factors such as gel drying, changes in cognitive state of patient, and adding unlabeled data might help in gradual adaptation of the classifier. In the cross validations used in this chapter, such adaptation effects are ignored. For getting a clearer picture on the performance of the proposed scheme in practical situations, extensive experiments

have to be done on a wider user base without having to rely on cross-validation schemes.

When the training data is low and based on a few characters, the classifier might have been trained on data which is subject to visual attention and spatial gradient effects [106], which can introduce a bias in the result. Also, as mentioned in Section. 4.4.3, very limited training data might not be sufficient to obtain a reliable initial classifier. With a carefully chosen initial training pattern, the initial classifier prediction will be sufficiently accurate and less subject to biases, and the performance is likely to improve.

The computational power required for classifying an hour of data adaptively is only a few seconds. However, the computational complexity keeps increasing exponentially as more data is added. Some forgetting factor could be added to keep the amount of data to be processed within limits, and thereby preventing an exponential increase in computational complexity. This could be crucial in low-power and mobile devices.

While it is generally accepted that P300 response has relatively relaxed requirement of visual attention, paying overt attention will increase the ITR of a P300 BCI [107]. Data from only healthy subjects are used in the current study. It could be expected that the disabled patients will be less effective in producing reliable training data, and hence the effectiveness of semi-supervised learning will be lower for a given amount of initial training data. Even in this case, it is plausible that the proposed methods could still yield the same classification accuracy with a smaller amount of training data than full supervised techniques. However, this would require extensive analysis to obtain a statistically valid confirmation.

Also, semi-supervised adaptation has the disadvantage that the initial classification accuracy might be relatively low, and the final classification accuracy is

achieved only after several minutes of adaptation. The low classification accuracy during this adaptation phase can still be annoying for many users, and might not be acceptable for many applications. For example, for an application such as wheelchair control, a very high classification accuracy is required before the system can be operated without risk. However, for an application such as a speller, this lower initial accuracy might be tolerable.

4.6 Conclusions

A two-classifier co-training based approach is proposed to train robust classifiers using both labeled and unlabeled data. The difference between the two classifiers is exploited for delivering a performance which is superior to that of single classifier systems. The algorithm is able to utilize unlabeled data effectively to improve the performance of the classifier. This leads to a reduction in the user effort, and consequently, results in a more convenient BCI system. Also, the proposed method is shown to outperform the self-training based approaches in most situations. The addition of unlabeled data was found to increase the classification accuracy to a limit, beyond which the improvement was minimal. Also, if sufficient training data is available, the performance improvement due to the algorithm is minimal, or even negative.

Introducing artificial training examples [108] to preserve diversity might reduce the tendency of co-training algorithms to degenerate to self-training with the addition of more and more unlabeled data. Another option is to use an ensemble of classifiers, to combine the advantages of co-training and ensemble based methods.

Chapter 5

Asynchronous P300 BCI : SSVEP-Based Control State Detection

5.1 Introduction

An alternate method to develop an asynchronous system is proposed in this chapter, using different potentials for control state detection and information transfer. The P300 based system is augmented with SSVEP providing control state information. Use of two different paradigms in a complementary fashion to achieve improved performance was recently reported by Pfurtscheller *et al.* [11, 109, 110]. In [109], they describe an SSVEP based system where ERD is used for control state detection.

The base system described here utilizes the P300 ERP. The exact level of visual attention required, and the suitability of P300 and SSVEP for various categories of subjects are still being studied by various groups [41, 42, 107]. However, there are several studies reporting severely paralyzed or disabled patients being able to use P300 BCIs, though not to the same extent as healthy subjects [111]. Moreover, it is relatively easy to detect, even with minimal training, and gives reasonably good

ITRs. Although SSVEP based systems are generally faster than P300 based systems, they suffer from several drawbacks such as the requirement of more accurate control of eye muscles [112, 113], precise and fast hardware, and unsuitability for people with epilepsy. Moreover, if low frequency stimulus is used, prolonged use of the system is very tiring whereas high frequency SSVEP response is weaker and harder to detect accurately. For a detailed description of various considerations in the operation of an SSVEP based BCI, see [11,41]. In our approach, instead of basing the complete system on SSVEP, we utilize it just for control state detection, with P300 as the main BCI paradigm. Unlike P300 signals, SSVEP is unlikely to produced accidentally in usual operational environments, which makes SSVEP based control state detection more robust and reliable than a system purely based on P300. While such a system is still dependent on control of eye muscles or other motor control for switching between control states, the information transfer itself is using P300 which has a relatively relaxed requirement as compared to SSVEP.

The rest of this chapter is organized as follows - the proposed method is described in Section 5.2. Section 5.3 describes the offline and online experiments and Section 5.4 details the data analysis. The results for the experiments are given in Section 5.5. The chapter is concluded with some remarks in Section 5.7.

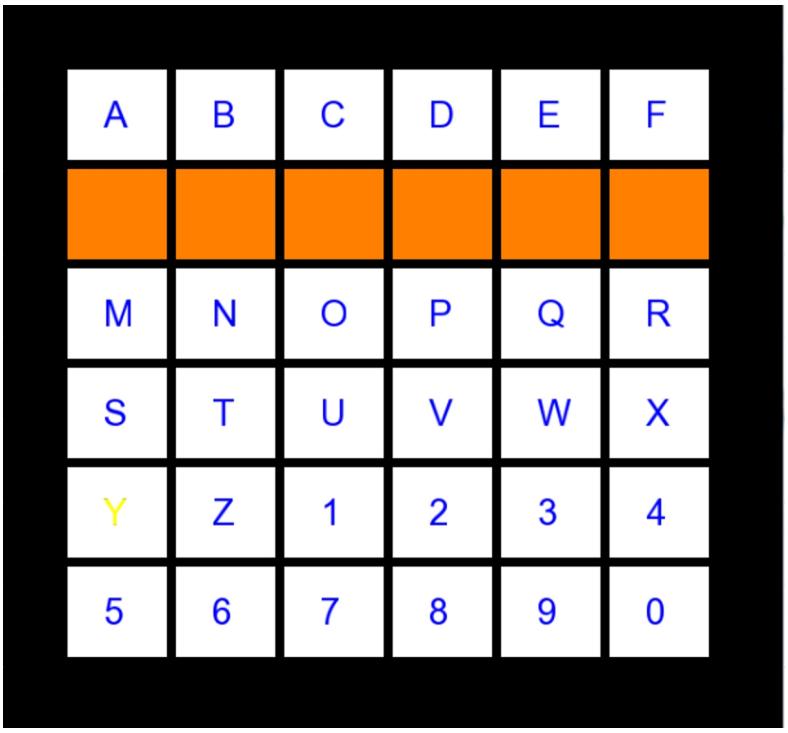
5.2 P300-SSVEP system

SSVEP is an ideal candidate to be used in conjunction with the P300 ERP, for several reasons. Both are well documented to be reliably evoked in most humans without prior training. The visual stimulus required to elicit SSVEP can be added to the existing P300 stimuli with relative ease, as both are usually evoked by a visual stimuli (P300 can also be evoked by other stimuli, but visual P300 is faster, and hence dominant in BCI research). Our experiments show that both signals can be elicited at the same time in an individual, without greatly compromising

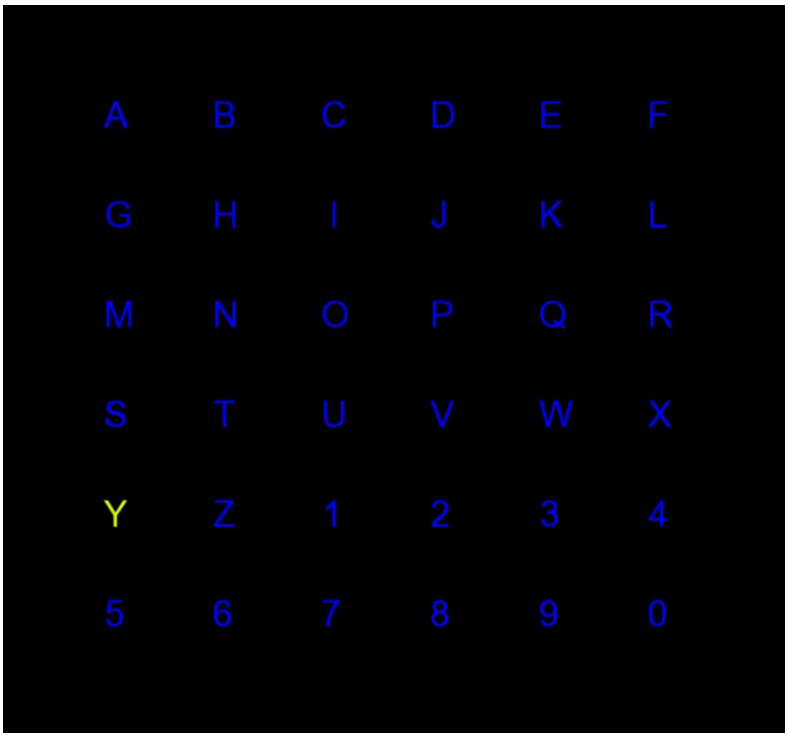
the detection accuracy of either.

The experimental setup makes use of a 24 channel EEG amplifier from ANT-Neuro, with a sampling rate of 256 Hz. EEG from 9 unshielded channels of the standard 10-20 system [114] - Cz, C1, C2, Pz, P1, P2, Oz, O1 and O2, were recorded. The data recording is controlled by the multi-threaded program implemented in Visual C++ described in Section. 3.1. As with a usual P300 speller, rows and columns are highlighted in a random order such that all rows and columns are highlighted once in every round. When the character of interest is highlighted, P300 is elicited. Data was sent to a second computer for processing in real-time, and the results are sent back to the first computer, where it is optionally displayed as a feedback. For eliciting SSVEP, all buttons are set to flicker at the desired frequency (alternating between black and white with 50% duty cycle). This stimulation paradigm of alternating a single graphic was reported to give a performance comparable to the pattern reversal paradigms [115]. Highlighting of rows and columns are done as usual for a P300 based interface, in a pseudo-random sequence such that all the rows and columns are highlighted once every round. Either red or orange color was used for row/column highlighting, completely occluding the character buttons. Figure 5.1 shows the two alternating states. When the user is gazing at the screen, it can be assumed that he/she wishes to input a command, which will manifest as the elicitation of SSVEP. With such an interface, the user would be able to naturally elicit both potentials without requiring a divided attention. Thus, the system can detect the target character and control state simultaneously.

Since only one frequency is used, the task is thus reduced to the detection of any SSVEP near the frequency of interest, as opposed to detecting precisely one among several frequencies. Thus, the need for a dedicated hardware capable of creating very precise stimuli of various frequencies is also eliminated; inexpensive displays



(a)



(b)

Figure 5.1: Figures (a) and (b) show the two alternating states during flickering. Rows and columns are highlighted in a pseudo-random sequence such that each row and each column is highlighted once in every round, as with the case of a standard P300 speller. Here, the target character during the training phase is 'Y', which is yellow in color.

would be sufficient. By choosing the frequency to be outside of the P300 operating range (i.e., above 12 Hz), the two signals could be separated by simple bandpass filtering and thus there would be no reduction of accuracy in the classification process.

Preliminary experiments were conducted to explore the best flicker frequency to be used in the subsequent experiments and in the online experiments. A stimulus frequency (f_{st}) slightly less than 18 Hz was chosen due to the following considerations: (i) to keep the frequency at least 2 harmonics above the frequency corresponding to the P300 highlighting (reciprocal of the inter-stimulus interval, ISI which is 225 ms in our experiments), so that the user can distinguish between the two stimuli comfortably; (ii) to synchronize the P300 highlighting and SSVEP flicker, it is desirable to have a flicker frequency which is a multiple of P300 highlighting frequency; (iii) to avoid overlap with the alpha band (8-12 Hz); (iv) lower frequencies are easier to elicit but less comfortable for the user; (v) higher frequencies are harder to elicit and demanding on hardware.

5.3 Experiments

5.3.1 Offline Experiments

To evaluate the performance of the proposed scheme, offline experiments were conducted on ten healthy subjects aged 19-28; seven males and three females. All subjects were undergraduate or postgraduate students who had no known physical or mental disabilities, were volunteers and received no compensation for their participation. Subjects 1-3, 6 and 10 had prior experience with BCIs, whereas the others were BCI naive. The data for two subjects who could not concentrate on the task, and another two subjects for whom the complete set of experiments were

not done are excluded form the subject count and analysis[1]. First, for training a P300 classifier, EEG for 360 rounds of stimuli were recorded (only 300 rounds for subjects 1-3), with the target character shown with a separate color during the session, and an ISI of 225 ms. 360 rounds of stimuli without SSVEP (i.e., the standard speller) were also recorded for subjects 6-10 to study the effect of introduction of SSVEP on P300 classification accuracy. Subjects 6,7 and 10 underwent the session with SSVEP first, whereas subjects 8 and 9 had the session without SSVEP first, to balance any effects due to habituation. During these experiments, subjects were instructed to count the number of times the target character is highlighted. The system shows 5 rounds per character, regardless of the state of the user. Each subject performed an experiment of 40 characters for offline analysis. Subjects are in control state for the first 10 characters, in non-control state for the next 10 characters and so on. There is no gap between rounds belonging to the same character, but a 1 second gap was provided between the two characters (i.e., 5 rounds) to allow the user to shift his/her attention to the next character. In control state, the subject focusses on the target character. The subject is instructed to do a mental task (multiplication of two random numbers of their choice) and to relax with eyes closed for alternate non-control states. A break was provided after every 10 characters (i.e., during the transition of the control state) and the system resumed the presentation of stimuli after the user has pressed a keyboard button to indicate his/her readiness to continue. An auditory cue was provided to alert the subject that the non-control session is over.

[1]A total of 16 subjects participated in the experiments, data from 4 subjects were discarded. Data from 5 subjects were used for analysis of methods presented in Chapter 4 and 10 Subjects for this Chapter, with 3 subjects participating in both

5.3.2 Online Experiments

To highlight the capability of the control state detection of the BCI and for easier evaluation of its performance, the online experiment is implemented as semi-asynchronous. The BCI system is still operated in a discrete, predefined blocks of rounds. In this experiment, 5 rounds per block and an ISI of 225 ms were used. Once stimuli for one block is finished, the system will halt until a decision has been made, and a new block will start. In addition to detecting target character using P300 in each round, the presence of SSVEP is detected as well to validate it. As long as SSVEP is detected in at least 3 out of 5 rounds, the subject is deemed to be in control state. P300 classification is employed only when control state has been established. In this scheme, the user will have to wait for the beginning of a new block to change to control state and to start giving the input. The subjects underwent three experimental sessions, with 18 characters for each session. A character is identified and displayed on the screen once 5 rounds have been presented, and the character is determined to be null (shown on the screen using an '=') if control state is not detected. While in control state, the subjects are required to input a string of characters (either the sentence A QUICK BROWN FOX JUMPS OVER A LAZY DOG or a random string). In each run, the subject focused on the first 6 characters, gazed away for the next 6, and focused again on the last 6. Thus, there would be 54 blocks of 5 rounds each, 36 of which are in control state. For every 6 characters, the subjects were provided a break from which he/she can resume based on his/her readiness; and the end of a non-control session was indicated through an auditory cue similar to that of offline experiments.

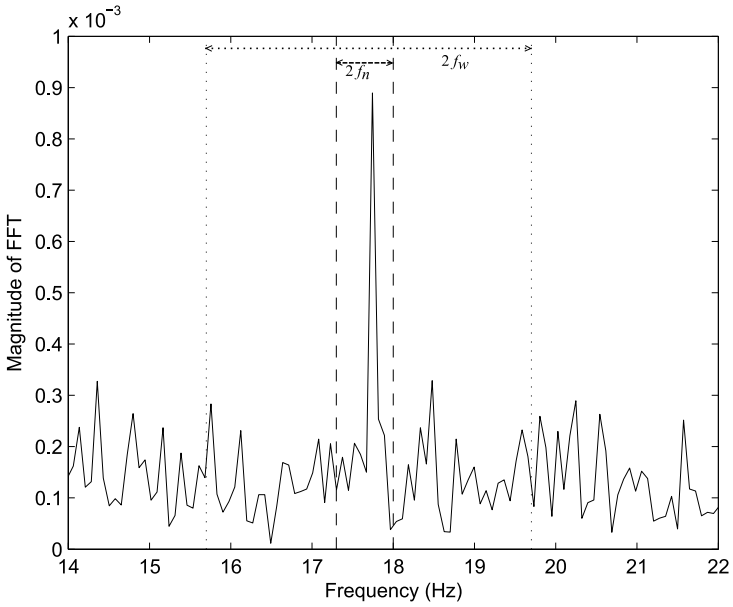


Figure 5.2: The peak picking algorithm. The objective function is the peak PSD in the band enclosed by the thick lines, relative to the mean PSD in the band enclosed by the thin lines.

5.4 Data Analysis

5.4.1 SSVEP Detection

SSVEP is usually very precise about the stimulus frequency. Gao *et al.* reported the possibility of distinguishing two stimuli with frequency difference of just 0.2 Hz [34]. Detection of SSVEP is usually done by a simple thresholding of the amplitude of the signal's Fourier spectrum. Various techniques for enhanced detection of SSVEP can be found in [35–38].

Unlike the techniques mentioned above, the detection task in our system is less demanding on frequency precision, as the presence or absence of SSVEP is all that is required to be estimated. Therefore, in this control state detection scheme, all other peaks not located around the target frequency can be assumed to be due to noise and were ignored. Simple thresholding of band-power would not work due to

high variability of EEG signals and the presence of a peak at the target frequency needs to be ascertained. Hence, the mean power in a wider range of frequencies is used as the benchmark for comparison. The window length of the FFT used is 1228, corresponding to approximately 4.8 seconds of data at a sampling frequency of 256 Hz. Figure 5.2 shows a sample FFT result of a round in which the user used the system with the screen flickering at around 17.7 Hz (a precise frequency is hard to maintain, owing to operating system scheduling related unpredictabilities [115], and lack of synchronization with vertical refresh of the monitor [116]). The relative mean PSD of frequency bins in the narrow range $f_{st} \pm f_n$ Hz as compared to the mean PSD in the wider range $f_{st} \pm f_w$ Hz is the metric chosen for detection. Thus, a simple objective function can be defined as

$$J(f_{st}) = \frac{[S(f)]_{f_{st}\pm f_n} - [S(f)]_{f_{st}\pm f_w}}{[S(f)]_{f_{st}\pm f_w}} \tag{5.1}$$

where $[S(f)]_{f_{st}\pm f_n}$ is the mean PSD in the narrow range and $[S(f)]_{f_{st}\pm f_w}$ is the mean PSD in the wider range. In our experiments, f_n is chosen to be 0.3 Hz, and f_w is chosen to be 2 Hz. The value of $J(f_{st})$ could then be compared with a threshold to detect the SSVEP, which in turn indicates user's desire to input a command. The frequency sensitivity of the algorithm could be tuned by setting the ranges. The threshold controls balance between TPR and FPR, the setting of which depends on the requirements of the specific application. Bashashati *et al.* have reported that false positive rates in asynchronous BCI applications above 1% - 2% cause frustration and distraction [86]. In most control applications of BCIs (e.g. wheelchairs), false positives should be minimized; whereas applications such as spellers are likely to be more tolerant to false positives. To an extent, this depends on user preferences as well. If the goal is to maximize the classification accuracy, the threshold can be found using an exhaustive search of a training data set. A value of 0.5, which was found to be giving a reasonable trade-off between

selectivity and sensitivity for most of the subjects was used in all results reported in this thesis. Channel selection was done based on inspection of power spectral densities of data from various channels at the frequency of interest. For SSVEP detection, Pz was used for subject 2, whereas for all others, Oz was chosen. From the pilot experiments, it was found that the data from just one round is not always sufficient for reliable control state detection. However, it is not necessary to follow P300's trial demarcation rigidly in this case, and it is possible to obtain more data per round without lengthening the ISI and sacrificing bit rates by allowing some overlap of data between rounds. It is justified if we assume that the user would have been focusing on the screen for at least a few seconds before the onset of the stimulus. Hence, for all the analysis as well as for online experiments, we included a 2 second overlap for SSVEP detection.

5.4.2 P300 Classification

As most of the information for detection of P300 lies in lower frequencies, the collected data is filtered with 0.5 Hz and 12 Hz as the lower and upper cut-off frequencies, zero-phase filtered using a Butterworth filter of order 3. To reduce the feature size, it is down-sampled to 32 Hz, and the data for a duration of 0.7 seconds from the start of the stimulus is considered to belong to that particular epoch. Due to the high amount of noise and background activity present in EEG, several rounds of data are required to get a reliable estimate of the P300 potential. The optimum number of rounds to be chosen for detection is a trade-off between accuracy and ITR, and varies from person to person. The number of rounds used in our experiments is fixed to be 5, as it was found to be giving a near-perfect accuracy in our preliminary experiments as well as in [117] and [118]. Several classifiers for classification of P300 have been reported in the literature including FLDA [97], support vector machines (SVM) [118], BLDA [2] etc. In our system, either FLDA or

BLDA is used for classification (FLDA for subjects 1-3 and BLDA for the rest of the subjects) as they are simple and computationally efficient, yet in our experiments, gave results comparable to other methods when there is reasonable amount of training data. A detailed description of FLDA can be found in Section 2.6. For BLDA, the criterion is the maximization of a log likelihood function involving $\mathbf{X}$ and $\mathbf{y}$, the details of which can be found in Section 4.2.1. As mentioned therein, the classifier scores calculated as $y_j = \mathbf{w}^T\mathbf{x}_j$ are summed over a number of rounds and the symbol corresponding to the maximum score is selected. The criterion for evaluating the BCI performance is described in Section 2.6.1. The effective number of bits is calculated using Eq. (2.10) and the information transfer rate is calculated using Eq. (2.11).

5.5 Results and Discussions

5.5.1 Effect of SSVEP Addition

To investigate the effect of addition of the flickering stimuli for eliciting SSVEP, a 4-fold cross-validation of 360-round data with and without SSVEP was done. The 360 rounds were split into 4 continuous sections of 90 rounds each. The classifier was trained on one of the sections and used to predict the class labels for the rest of the 3 sections. This was done for all the sections and the results were averaged. Table 5.1 shows the classification accuracies obtained for subjects 6-10 with and without SSVEP stimuli. For subject 10, slightly better accuracy was obtained when SSVEP was absent, whereas for subjects 6-8, introduction of SSVEP resulted in a slight improvement in accuracy. Results show that the accuracies are comparable and the introduction of SSVEP might not be detrimental to P300 detection.

Table 5.1: Table showing P300 detection accuracies with and without SSVEP stimuli.

Subject	CA (SSVEP stimuli present)	CA (SSVEP stimuli absent)
Sub 6	97.69	94.91
Sub 7	83.80	81.48
Sub 8	99.07	95.37
Sub 9	99.07	98.61
Sub 10	72.69	75.00

5.5.2 Results for Offline Analysis

The P300 detection accuracy was found to be very good - all the 20 characters were correctly detected for all subjects except Subjects 1, 5, 7 and 10, for whom the correct detections were 19,19,19 and 16 respectively. A sample spectrum of the first 20 characters is shown in Fig.5.3, which shows that with a full block of data, distinguishing between control and non-control states could be done.

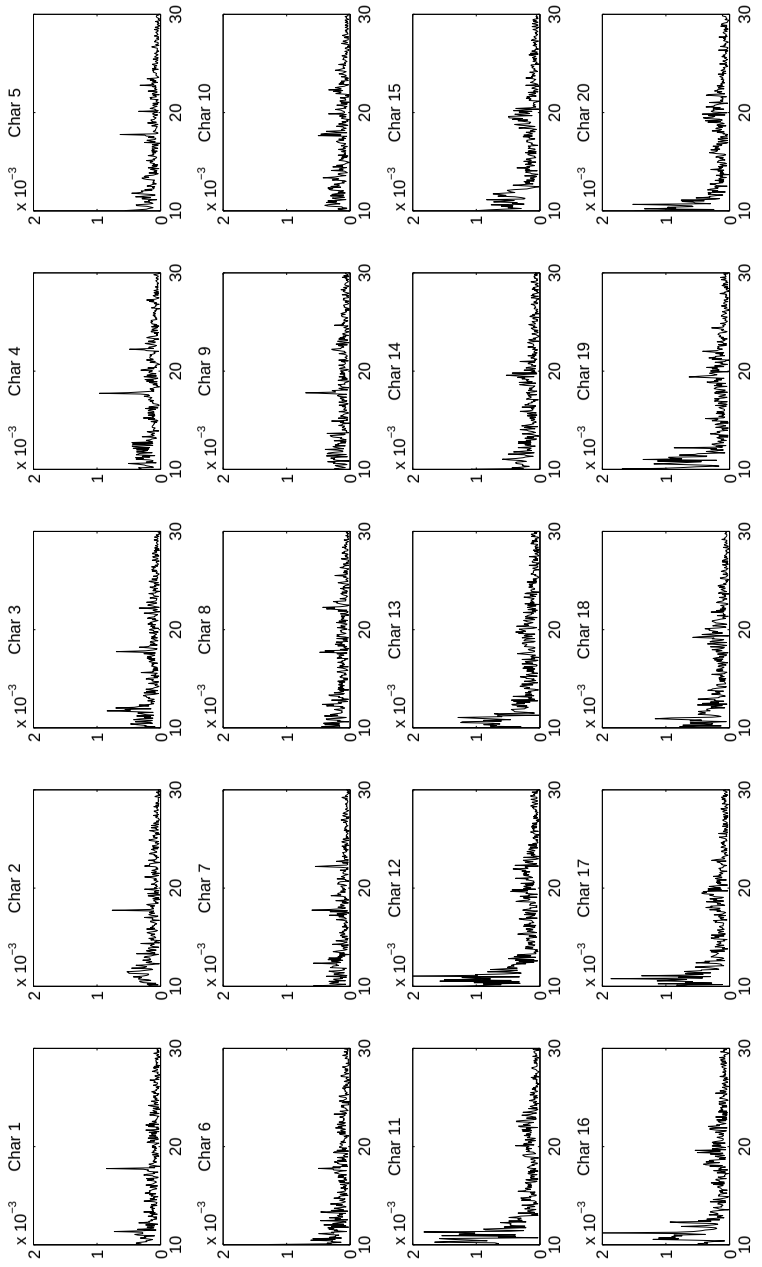


Figure 5.3: FFT of the first 20 characters for Subject 1. Characters 1-10 are in control state.

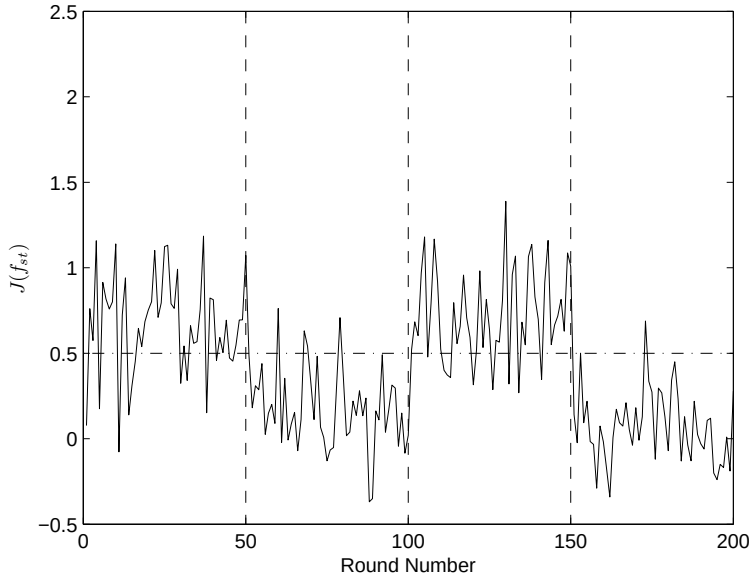
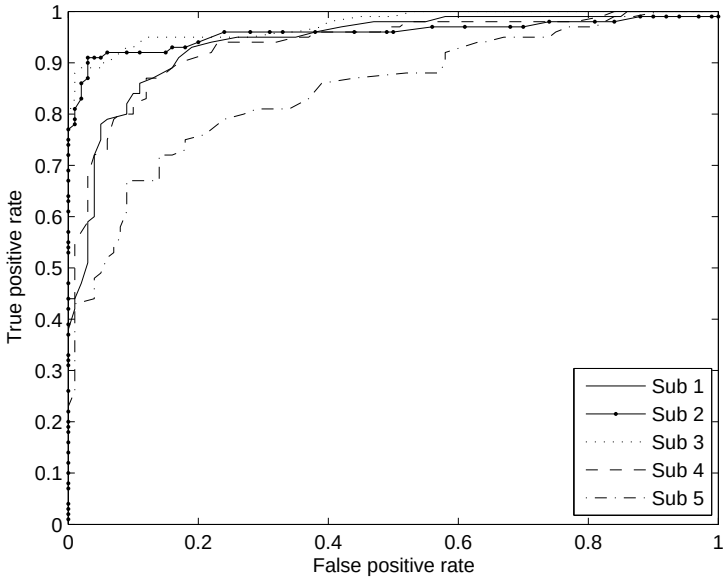


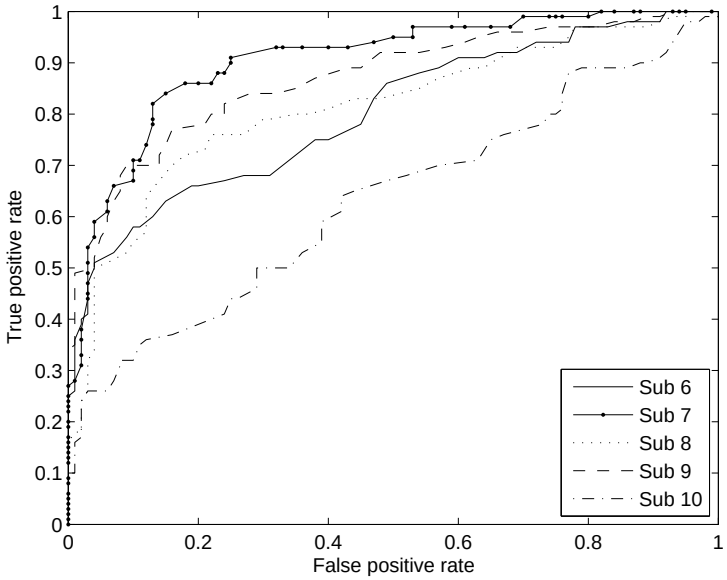
Figure 5.4: $J(f_{st})$ for Subject 1.

The values of the objective function ($J(f_{st})$), evaluated using Eq.(5.1) for 200 rounds of data are shown in Fig. 5.4. The vertical dashed lines indicate a change in control state (as the control state in the experiment changed every 50 rounds), and as expected, the non-control rounds have values around zero. The horizontal dashed line is the detection threshold which is set to 0.5. Those points below the threshold in control state are false negatives, and those in non-control state above the threshold are false positives. If SSVEP is absent, the mean PSD in the band of interest ($f_{st} \pm f_n$) is close to the mean PSD over $f_{st} \pm f_w$, and the value of the objective function would be nearly zero; which can be observed in Fig. 5.4.

To evaluate the performance of the system for various thresholds, the receiver-operating characteristic (ROC) was plotted for the first five subjects as shown in Fig. 5.5a and for the rest of the five subjects in Fig.5.5b. ROC plots the TPR (sensitivity) against FPR (selectivity). Area under curve (AUC) of the ROC, which is a performance measure of the system, is given in Table 5.2. Classification accuracies (CA) of the speller, the corresponding ITRs as well as control state



(a) Subjects 1-5



(b) Subjects 6-10

Figure 5.5: ROC for the Subjects.

detection accuracies (CD) for the subjects are also given therein. Average AUC over all subjects is 0.859, and Subject 10 has the minimum AUC at 0.632. For most subjects, the AUC is much higher than that for a random classification (0.5), which shows that the system has good control-state detection capability. For subject 10, the discrimination is very low, though it is significantly better than chance (a bootstrap analysis [102] reveals that the 95% confidence interval for the AUC does not include 0.5). Voting of classifier labels within a block was used for calculation of CD. For example, if blocks of 5 rounds are considered for the detection of one character; as long as at least 3 rounds are determined to be in the control state, the character is deemed valid. This will ensure that the false positives within a round does not necessarily cause an error in control state detection. However, using more rounds will inevitably reduce the ITRs achievable, and increases the waiting time for the user to start operating the system (user will not be able to start operating the system till the end of the number of rounds used in detection). It can be seen from Table. 5.2 that most subjects were able to achieve very good data transfer rates as well as control state detection accuracies. The control state detection accuracy for Subject 10 is very low as compared to other subjects. This could be expected, as a recent survey by Allison *et al.* [41, 119] on BCI demographics shows that there are BCI "illiterate" people who shows poor performance for certain BCI paradigms. The P300 classification accuracy for Subject 10 was also less than others. However, it is not possible to derive statistical conclusions on the relation between P300 and SSVEP detection accuracies based on the data - it could be due to attentional / motivational factors. Overall, the system achieved an average bit rate of 20 bits/min at an accuracy of 96.5%, and a control state detection accuracy of 88%.

Table 5.2: Detection results for the offline experiment. The classification accuracy for P300 (CA), the corresponding ITR, and the control state detection accuracies (CD) for various number of rounds used for the detection of a character.

Rounds/Char.		2			3			5		
Subject	AUC	CA (%)	ITR (bits/min)	CD (%)	CA (%)	ITR (bits/min)	CD (%)	CA (%)	ITR (bits/min)	CD (%)
Sub 1	0.928	90.00	39.26	72.50	95.00	30.51	97.50	95.00	19.15	95.00
Sub 2	0.958	65.00	22.88	85.00	95.00	30.51	95.00	100.00	21.39	97.50
Sub 3	0.964	85.00	35.54	82.50	95.00	30.51	95.00	100.00	21.39	97.50
Sub 4	0.925	85.00	35.54	72.50	95.00	30.51	92.50	100.00	21.39	97.50
Sub 5	0.836	75.00	28.84	55.00	85.00	24.99	85.00	95.00	19.15	87.50
Sub 6	0.791	90.00	39.26	55.00	95.00	30.51	72.50	100.00	21.39	77.50
Sub 7	0.894	65.00	22.88	72.50	85.00	24.99	85.00	95.00	19.15	92.50
Sub 8	0.802	80.00	32.08	55.00	95.00	30.51	77.50	100.00	21.39	85.00
Sub 9	0.859	95.00	43.38	52.50	100.00	34.09	82.50	100.00	21.39	92.50
Sub 10	0.632	35.00	8.45	50.00	50.00	10.58	62.50	80.00	14.16	57.50
Average	0.859	76.50	30.81	65.25	89.00	27.77	84.50	96.50	20.00	88.00

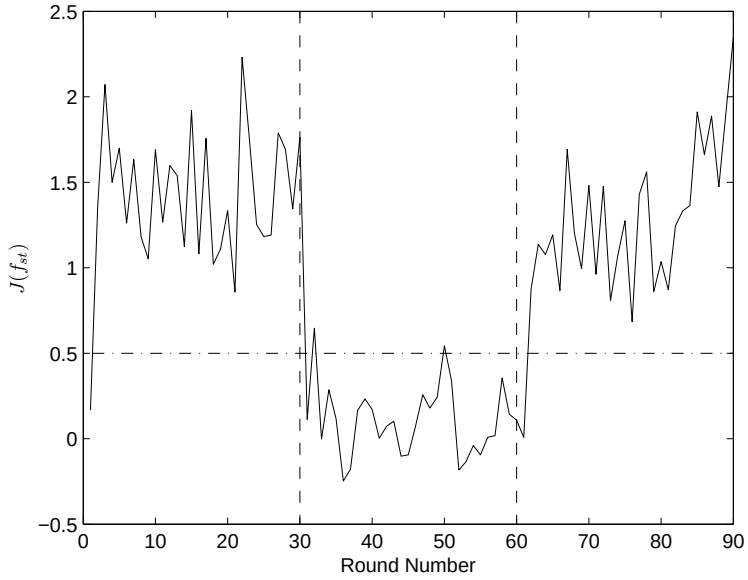


Figure 5.6: $J(f_{st})$ for Subject 1 in the online experiment.

5.5.3 Online Results

For online experiments, the threshold value was set to be 0.5. A plot of objective function for one session is given in Fig. 5.6. For blocks of 5 rounds (one character), the mean detections for control states (CS) and non-control states (NCS) are given in Table. 5.3. More than 4 out of 5 rounds gave correct detections in control state for Subjects 1-7. The mean of the number of rounds in a block where the system gives a positive detection during non-control state is 0.21 on an average, which corresponds to a round-wise false alarm rate of 4.2%. The corresponding control state detection accuracy is 88.15%. The control state detection accuracy was more than 94% for Subjects 1-7. For Subjects 8 and 10, these were less than 70%, and little discrimination could be achieved for Subject 10 (detection accuracy by chance is 50%). It was noted that accuracy is lower when focusing on the last column of the display, likely due to reduced visual attention to SSVEP; however more experiments are required to ascertain the statistical validity of this observation. Based on the

Table 5.3: Detection results for the online experiment. CS and NCS are the mean SSVEP detections for blocks of 5 rounds, when the subject is in control state and non-control state respectively. CD is the block-wise detection accuracy of control state.

Subject	CS	NCS	CD	CA	ITR (bits/min)
Sub 1	4.78	0.15	98.15	97.22	20.04
Sub 2	4.88	0.12	96.30	86.11	16.04
Sub 3	4.69	0.09	98.15	100.00	21.39
Sub 4	4.67	0.26	97.22	97.22	20.05
Sub 5	4.52	0.26	94.44	88.89	16.95
Sub 6	4.48	0.17	95.83	94.44	18.93
Sub 7	4.51	0.26	94.44	88.89	16.95
Sub 8	3.50	0.39	68.05	97.22	20.04
Sub 9	3.70	0.22	83.33	97.22	20.04
Sub 10	2.59	0.19	55.56	97.22	20.04
Average	4.23	0.21	88.15	94.44	19.05

P300 detection accuracy, the system is capable of information transfer at an average of 19.05 bits/min while maintaining very good classification accuracies when the subjects are in control state. The accuracy is more than 94% except for Subjects 2, 5 and 7, who also achieved more than 86% accuracy.

5.6 Limitations and Implementation Issues

The current system is not completely synchronous - the user still needs to wait for the gap between characters to start giving an input. Since we use 5 rounds/character, the maximum waiting time can be up to about 14 seconds, which can be frustrating for the user. A possible enhancement to mitigate this issue is to render the system more asynchronous by the use of a sliding window instead of discrete blocks for the detection of a character. The window should consists of 6 rounds, the control state is determined as follows: the first round has to contain SSVEP, and at most two of the other 5 can have a negative detection. If the current window fulfills that criteria, a decision is made through P300 classification of the last 5 rounds and the

queue is emptied. Otherwise, the window is moved forward by 1 round. The extra round at the beginning is not evaluated for P300 as the user might start focusing mid-round, producing the requisite SSVEP and without the proper P300 signal. This extra round can work as an alternative to the break enforced between blocks, and hence does not amount to a loss of efficiency.

Higher frequency flicker could be used to increase users' comfort while using the system. Various works have reported reliable detection of frequencies up to 40 Hz, which corresponds to a flicker that is barely perceptible [120, 121]. However, this would necessitate very sophisticated hardware and detection algorithms. Choosing frequencies which can synchronize with the vertical refresh of the monitor might reduce the BCI illiteracy [116].

To evaluate the advantages and limitations of the proposed method in practical scenarios, it needs to be tested for different BCI applications. Also, the performance of the system should be tested on a larger and more varied subject pool to get unbiased results on the usability and generality. This should also enable a better understanding on the effect of inter- and intra-subject variabilities on the combined elicitation of P300 and SSVEP. Techniques for adapting the SSVEP detection threshold needs to be explored for subjects with high intra-subject variability. A detailed study of factors affecting user comfort, and effects of habituation would enable necessary modifications to improve the usability and practicality of the system.

5.7 Conclusions

An asynchronous BCI system combining two different paradigms has been realized. This system takes advantage of the ease of elicitation of the SSVEP, and flexibility of P300 such that the system has efficient operation and reliable control state detection. In the online experiments, the system achieved an ITR of 19.05 bits/min,

with a control state classification accuracy of 88.15%. The system demonstrated very good control state detection accuracies for 8 of the 10 subjects, and had more than 94% discrimination for 7 subjects.

Since the purpose of this work is to validate the proposed technique of combining P300 and SSVEP paradigms, the system uses relatively simple hardware and detection algorithms. More responsive (faster) hardware devices such as LED/plasma displays might enable more accurate stimuli presentation, and consequently better performance. Pre-processing techniques such as ICA [48] could be employed to clean up the recorded signal by removing artefacts and spatially localized noise. This will enable the classifier to learn and classify the P300 response better, resulting in a higher ITR. More advanced SSVEP detection techniques such as CCA [34] could be used to improve the control state detection accuracy of the system. Also, combining the statistical information from the P300 classifier scores [51] with the information from SSVEP could improve the control state detection performance further.

Chapter 6

Conclusions and Future Directions

A BCI system can be used as a channel of communication between the user and a human-computer interaction system. This can be used by a disabled person to communicate with a computer through his "thought" alone. The applications of such a system are ever increasing and virtually endless. Basic commercial systems have already hit the market, and more sophisticated and cheaper systems are bound to enter the market soon. This chapter summarizes the contributions in this thesis toward the goal of a better P300/hybrid P300 interface, focusing on adaptation and control state detection capabilities.

A flexible BCI system was realized which can work as a stand-alone P300 speller or a hybrid system, or which can elicit P300 along with SSVEP. The baseline performance is evaluated thoroughly, and is able to achieve a classification accuracy of >85% with a bit rate of about 40bits/min. This is comparable to the state of the art BCI systems.

6.1 Adaptation

Fast adaptation schemes are necessary for realizing BCI systems which can operate with very little training data. This can result in a more usable BCI which will enhance the attractiveness and encourage their commercial viability and widespread adoption. A co-training approach, which uses two classifiers - FLDA and BLDA is used for fast adaptation in a P300 BCI. The resulting classifier predicted labels by one classifier is used for generating training data for the other classifier and vice versa. The method is shown to achieve a good and gradual increase in the classification accuracy, and consequently, bit rates. Thus, the user will be able to communicate faster, and the need for intermittent training sessions will be minimized. The method is found to be yielding a superior performance as compared to self-training where the classifier is re-trained using its own predictions. The performance improvement was found to be even more significant for the cases where the training data and the number of rounds used for detection of a character are low, both desirable qualities for a BCI.

The difference between the two classifiers add extra information to improve the adaptation performance. This is unlike in the case of self training where only one classifier is used. In usual self-training or co-training, only certain fraction constituting the most confident predictions of the classifier is used for adaptation. In this thesis, we have shown that adding all the predictions yield better adaptation performance for the P300 classifier. This is likely due to the fact that the data points which are nearer to the classification boundary are more important in determining the updates to the boundary as compared to those far away from the it (whose prediction confidence will be high anyway). Though this adds a risk of mislabeling, the information provided by the weak learners still add positively to the adaptation.

The co-training method yielded better results for subjects who produced a

stronger P300 (as could be inferred from the supervised classification rate). The final classification accuracy attained was, however, strongly dependent on the initial labelled data. However, the absolute increase in classification accuracy, desirably, is better when the training data is low. The method is able to achieve very good accuracy and data transfer rate even with very little (a few seconds of) training data. The accuracy was found to increase till a point, beyond which the addition of unlabeled data has little effect on accuracy. In some cases, addition of unlabeled data can be detrimental to performance.

The present study uses cross-validations to perform statistical analysis of classifier performances. Hence, it does not evaluate the performance when there are gradual variation in the characteristics of the recorded signal. Such adaptation effects are important in practical scenarios, though it is very hard to statistically analyse the relative performances of various algorithms without conducting very long experiments on a number of patients. For example, we might need to conduct experiments of several minutes duration with a gap of several hours in between to analyse the effect of adaptation in realistic BCI usage scenarios.

Though not fully studied yet, it is possible that the P300 characteristics may be dependent on the location of the character in the matrix producing P300. For a non-speller based interface too, the exact way in which P300 is elicited might have a bearing on its characteristics. In adaptive BCIs where the training data is low, the initial labeled data is likely to have a strong bearing on the final classification accuracy achieved. Hence, it is desirable to have a carefully chosen pattern which is less subject to visual attention and spatial effects to avoid such biases.

In this thesis, we have not attempted to derive techniques which can assist the system in determining when and whether to adapt, which are also equally important problems as how to adapt. As mentioned in Chapter 2, such techniques could focus on statistical analysis of classifier scores. For example, we can keep

adapting while the z-score of the classifier scores does not decrease. However, we should not stop adapting completely at any point, as the positive effect of adaptation might become relevant/significant at some point in the BCI operation. A detailed analysis of such schemes are left as future work.

With the addition of more unlabeled data, the difference in predictions of the two classifiers in co-training algorithm reduces, and co-training degenerates to self-training. One option to minimize this possibility and to preserve diversity is to introduce artificial training examples [108]. Another possible enhancement is to use an ensemble of classifiers instead of just two. This will help combine the advantages of both ensemble learning as well as co-training. The ensembles could be derived from different types of classifiers, or using the same classifier model, trained to different subsets of data.

Also, adaptation is intensive on memory and computational power. Future BCI systems are embedded / mobile applications. In such systems, resources come at a premium, and existing transductive / semi-supervised techniques have heavy requirements on these two resources. As such, these techniques might be difficult to implement on mobile or embedded platforms. Hence, there is a need for exploring incremental versions of the semi-supervised techniques. One method would be to carefully choose and retain only a limited set of confident labels, discarding the least confident ones in each round of prediction. Alternatively, the predicted data could be used to update a fixed set of training points based on some statistical criterion (such as a weighted addition to the point nearest, according to a distance metric such as Mahalanobis distance). Another promising method is to use a SVM trained using sequential minimal optimization (SMO), with the new training data derived from confident predictions by the existing classifier.

The prominence of P300 peak and its duration depends on a lot of factors, including, but not limited to the subject, his/her level of alertness, the novelty

of the event etc. Selecting the right features for a subject might help in better predictions from initial data, and consequently in better performance from semi-supervised learning techniques. Identifying the best features requires an exhaustive search, and genetic algorithms (GA), which tend to give solutions which are globally optimal are well suited for such tasks. The common use of feature selection in context of BCI is in the selection of electrodes in a multi-channel EEG recording arrangement, as can be seen in [30]. A spatio-temporal feature selection could be done with having starting and end point markers on the time series, and one bit per channel used as the mask vector. The feature selection algorithm could be run periodically and the feature sets could be adapted based on the statistics of classifier confidence (such as the average z-scores of predicted labels).

6.2 Control State Detection

Almost all of the present day BCI systems work in synchronous mode, i.e., the knowledge of precisely when the user will give an input to the computer is known. An ideal BCI system should be able to work in an asynchronous mode, i.e., the user should be able to give input at will. The challenge in developing an asynchronous BCI system is the control state detection. The system should be able to detect whether the user is trying to communicate, and detect the input. Otherwise, it should remain inactive. It might be required that some other criterion other than P300 needs to be used in such a system to initiate the system into active state. In this thesis, we propose an asynchronous BCI system combining two different paradigms. The system relies on P300 for information transfer. P300 can be elicited in a wider range of patients, has relaxed visual attention requirements, and the required stimuli paradigm can be realized using simpler hardware. This base system is augmented by the introduction of a constant frequency flicker. This elicits a relatively easy to detect SSVEP, which is utilized for control state detection. The

system is shown to achieve good control state detection and data transfer rates for most of the subjects.

The proposed system clearly shows the advantages of hybrid techniques. Hybrid techniques could be used for achieving desirable goals such as enhancing information transfer and as activation mechanisms for certain infrequently used inputs. It is known that BCI illiteracy is usually specific to a particular EEG pattern. Thus, a system which can utilize different potentials for the same goal could reduce the effects of BCI illiteracy. For example, in a system which combines P300 and SSVEP in a speller paradigm, if the user is unable to produce P300 signals which can be reliably detected, the SSVEP could provide additional information to the classifier, and hence target character could be detected more reliably. Due to all these advantages, hybrid BCIs hold a prominent place in the future of BCIs, and research in this direction has geared up recently. However, it should be noted that BCI illiteracy posses a greater challenge in the system described in this thesis. This is due to the fact that the system uses P300 and SSVEP for separate goals, and its proper operation requires the user to be able to produce both responses reliably.

Frequencies which are synchronized to the vertical refresh of the monitor could provide a more precise frequency stimuli, and consequently, better control state detection accuracy. This could reduce the BCI illiteracy due to SSVEP. Also, the statistical information from P300 classifier scores could be combined to increase the robustness of control state detection, and thereby reducing the need to detect both P300 and SSVEP reliably. Another disadvantage of the system is that the system uses flickering stimuli. This can be uncomfortable for certain users, and also has the potential to trigger epileptic seizures. These negative aspects can be mitigated to a great extent through the use of high-frequency SSVEP. This would make the system more comfortable for the user as compared to a sub-20 Hz flicker used in

the present study. However, elicitation and detection of high-frequency SSVEP is very demanding on the hardware and detection algorithms. The system described in this thesis has been kept fairly basic as the purpose was to validate the use of a hybrid technique for control state detection. More sophisticated hardware and detection algorithms (for example, CCA based or phase rectified methods) could yield a better performance. Also, the system was tested only on a relatively small pool of homogenous subjects. To evaluate the performance and usability of the system in a completely unbiased manner, it should be tested on a large subject pool similar to the demographics tests described in [41], and for various applications.

Bibliography

[1] P. Williams *et al.*, *Gray's anatomy.* Churchill Livingstone, London, 1989, vol. 378.

[2] U. Hoffmann, "Bayesian machine learning applied in a brain-computer interface for disabled users," Ph.D. dissertation, Ecole Polytechnique Federale de Lausanne, Switzerland, 2007.

[3] C. Guan, M. Thulasidas, and J. Wu, "High performance P300 speller for brain-computer interface," *IEEE Workshop on Biomed. Circuit and Sys.*, pp. 13–16, 2004.

[4] M. Thulasidas, C. Guan, and J. Wu, "Robust classification of eeg signal for brain-computer interface," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 14, no. 1, pp. 24–29, 2006.

[5] B. Blankertz, G. Dornhege, M. Krauledat, M. Schröder, J. Williamson, R. Murray-Smith, and K. Müller, "The Berlin brain-computer interface presents the novel mental typewriter hex-o-spell," in *Proc. of the 3rd Intl.*

Brain-Computer Interface Workshop and Training Course. Citeseer, 2006, pp. 108–109.

[6] J. D. Bayliss, "Use of the evoked potential P3 component for control in a virtual apartment," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 11, no. 2, pp. 113–116, 2003.

[7] A. Royer, A. Doud, M. Rose, and B. He, "EEG control of a virtual helicopter in 3-dimensional space using intelligent control strategies," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 18, no. 6, pp. 581–589, 2010.

[8] Y. Li, J. Long, T. Yu, Z. Yu, C. Wang, H. Zhang, and C. Guan, "An EEG-based BCI system for 2-D cursor control by combining mu/beta rhythm and P300 potential," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 10, pp. 2495–2505, 2010.

[9] S. Muller, T. Bastos-Filho, and M. Sarcinelli-Filho, "Using a SSVEP-BCI to command a robotic wheelchair," in *2011 IEEE Intl. Symp. Industrial Electronics (ISIE)*. IEEE, 2011, pp. 957–962.

[10] B. Rebsamen, C. Guan, H. Zhang, C. Wang, C. Teo, M. Ang, and E. Burdet, "A brain controlled wheelchair to navigate in familiar environments," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 18, no. 6, pp. 590–598, 2010.

[11] B. Allison, C. Brunner, V. Kaiser, G. Muller-Putz, C. Neuper, and G. Pfurtscheller, "Toward a hybrid brain–computer interface based on imagined movement and visual attention," *J. Neur. Eng.*, vol. 7, p. 026007, 2010.

[12] Y. T. Wang, Y. Wang, and T. Jung, "A cell-phone-based brain–computer interface for communication in daily life," *J. Neur. Eng.*, vol. 8, p. 025018, 2011.

[13] K. Shyu, P. Lee, M. Lee, M. Lin, R. Lai, and Y. Chiu, "Development of a low-cost FPGA-based SSVEP BCI multimedia control system," *IEEE Trans. Biomed. Circuits and Systems*, vol. 4, no. 2, pp. 125–132, 2010.

[14] A. Palumbo, B. Calabrese, G. Cocorullo, M. Lanuzza, P. Veltri, P. Vizza, A. Gambardella, and M. Sturniolo, "A novel ICA-based hardware system for reconfigurable and portable BCI," in *IEEE Intl. Workshop on Medical Measurements and Applications, MeMeA 2009.* IEEE, 2009, pp. 95–98.

[15] L. Kauhanen, T. Nykopp, and M. Sams, "Classification of single MEG trials related to left and right index finger movements," *Clinical Neurophysiology*, vol. 117, no. 2, pp. 430–439, 2006.

[16] N. Weiskopf, K. Mathiak, S. 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 Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 966–970, June 2004.

[17] T. Hinterberger, N. Weiskopf, R. Veit, B. Wilhelm, E. Betta, and N. Birbaumer, "An EEG-driven brain-computer interface combined with functional magnetic resonance imaging (fMRI)," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 971–974, June 2004.

[18] S. M. Coyle, T. E. Ward, and C. M. Markham, "Brain-computer interface using a simplified functional near-infrared spectroscopy system," *J Neur. Eng.*, vol. 4, pp. 219–226, 2007.

[19] H. Jasper, "The ten-twenty system of the international federation," *Electroenceph. Clin. Neurophysiol.*, vol. 10, pp. 371–375, 1958.

[20] R. Oostenveld and P. Praamstra, "The five percent electrode system for high-resolution EEG and ERP measurements," *Clinical Neurophysiology*, vol. 112, no. 4, pp. 713–719, 2001.

[21] E. Donchin, K. Spencer, and R. Wijesinghe, "The mental prosthesis: assessing the speed of a P300-based brain-computer interface," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 8, no. 2, pp. 174–179, Jun 2000.

[22] Y. Wang, R. Wang, X. Gao, and S. Gao, "Brain-computer interface based on the high-frequency steady-state visual evoked potential," *Proc. First Intl. Conf. Neural Interface and Control, 2005*, pp. 37–39, 26-28 May 2005.

[23] Y. Li, X. Gao, H. Liu, and S. Gao, "Classification of single-trial electroencephalogram during finger movement," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1019–1025, June 2004.

[24] T. Hinterberger, S. Schmidt, N. Neumann, J. Mellinger, B. Blankertz, G. Curio, and N. Birbaumer, "Brain-computer communication and slow cortical potentials," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1011–1018, June 2004.

[25] S. Sutton, M. Braren, J. Zubin, and E. R. John, "Evoked-potential correlates of stimulus uncertainty," *Science*, vol. 150, pp. 1187 – 1188, 1965.

[26] E. Donchin and Y. Arbel, "P300 based brain computer interfaces: A progress report," in *FAC '09: Proc. 5th Int. Conf. Foundations of Augmented Cognition. Neuroergonomics and Operational Neuroscience.* Berlin, Heidelberg: Springer-Verlag, 2009, pp. 724–731.

[27] H. Serby, E. Yom-Tov, and G. Inbar, "An improved P300-based brain-computer interface," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 13, no. 1, pp. 89–98, March 2005.

[28] B. Allison and J. Pineda, "ERPs evoked by different matrix sizes: implications for a brain computer interface (BCI) system," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 11, no. 2, pp. 110–113, June 2003.

[29] F. Beverina, G. Palmas, S. Silvoni, F. Piccione, and S. Giove, "User adaptive BCIs: SSVEP and P300 based interfaces," *PsychNology Journal*, vol. 1, no. 4, pp. 331–354, 2003.

[30] G. Bin, X. Gao, Z. Yan, B. Hong, and S. Gao, "An online multi-channel SSVEP-based brain–computer interface using a canonical correlation analysis method," *J. Neur. Eng.*, vol. 6, p. 046002, 2009.

[31] J. Jin, B. Allison, E. Sellers, C. Brunner, P. Horki, X. Wang, and C. Neuper, "An adaptive P300-based control system," *J. Neur. Eng.*, vol. 8, p. 036006, 2011.

[32] G. Frye, C. Hauser, G. Townsend, and E. Sellers, "Suppressing flashes of items surrounding targets during calibration of a P300-based brain–computer interface improves performance," *J. Neur. Eng.*, vol. 8, p. 025024, 2011.

[33] A. Lenhardt, M. Kaper, and H. Ritter, "An adaptive P300-based online Brain-Computer Interface," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 16, no. 2, pp. 121–130, April 2008.

[34] X. Gao, D. Xu, M. Cheng, and S. Gao, "A BCI-based environmental controller for the motion-disabled," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 11, no. 2, pp. 137–140, 2003.

[35] C. E. Davila and R. Srebro, "Subspace averaging of steady-state visual evoked potentials," *IEEE Trans. Biomed. Eng.*, vol. 47, no. 6, pp. 720–728, 2000.

[36] M. Middendorf, G. McMillan, G. Calhoun, and K. Jones, "Brain-computer interfaces based on the steady-state visual-evoked response," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 8, no. 2, pp. 211–214, 2000.

[37] A. Bauer, J. W. Kantelhardt, A. Bunde, P. Barthel, R. Schneider, M. Malik, and G. Schmidt, "Phase-rectified signal averaging detects quasi-periodicities in non-stationary data," *Physica A*, vol. 364, pp. 423–434, 2006.

[38] Z. Lin, C. Zhang, W. Wu, and X. Gao, "Frequency recognition based on canonical correlation analysis for SSVEP-based BCIs," *IEEE Tran. Biomed. Eng.*, vol. 54, no. 6, pp. 1172–1176, June 2007.

[39] J. D. Bayliss and D. H. Ballard, "Single trial P3 epoch recognition in a virtual environment," *Neurocomputing*, vol. 32, pp. 637–642, 2000.

[40] M. Thulasidas, C. Guan, and J. Wu, "Robust classification of EEG signal for brain-computer interface," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 14, no. 1, pp. 24–29, March 2006.

[41] B. Allison, T. Luth, D. Valbuena, A. Teymourian, I. Volosyak, and A. Graser, "BCI demographics: How many (and what kinds of) people can use an SSVEP BCI?" *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 18, no. 2, pp. 107 –116, april 2010.

[42] C. Guger, S. Daban, E. Sellers, C. Holzner, G. Krausz, R. Carabalona, F. Gramatica, and G. Edlinger, "How many people are able to control a P300-based brain-computer interface (BCI)?" *Neurosci. Let.*, vol. 462, no. 1, pp. 94–98, 2009.

[43] C. Vidaurre and B. Blankertz, "Towards a cure for BCI illiteracy," *Brain topography*, vol. 23, no. 2, pp. 194–198, 2010.

[44] S. Puthusserypady and T. Ratnarajah, "H-∞ adaptive filters for eye blink artifact minimization from electroencephalogram," *Signal Processing Letters, IEEE*, vol. 12, no. 12, pp. 816–819, 2005.

[45] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification II edition.* John-Wiley and Sons, 2001.

[46] 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," *J. Neur. Eng.*, vol. 4, pp. R32–R57, 2007.

[47] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Tran. on Pattern Anal. Mach. Intel.*, vol. 22, no. 1, pp. 4–37, 2000.

[48] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis.* John Wiley & Sons, 2001.

[49] F. Lotte, M. Congedo, A. Lecuyer, F. Lamarche, and B. Arnaldi1, "A review of classification algorithms for EEG-based brain-computer interfaces," *J. Neur. Eng.*, vol. 4, pp. R1–R13, 2007.

[50] A. Luo and T. Sullivan, "A user-friendly ssvep-based brain–computer interface using a time-domain classifier," *J. Neur. Eng.*, vol. 7, p. 026010, 2010.

[51] H. Zhang, C. Guan, and C. Wang, "Asynchronous P300-based brain-computer interfaces: A computational approach with statistical models," *IEEE Trans. Biomed. Eng.*, vol. 55, no. 6, pp. 1754–1763, 2008.

[52] S. Haykin, *Neural Networks, A comprehensive foundation.* Prentice Hall, 1999.

[53] A. Rakotomamonjy and V. Guigue, "BCI competition III: dataset II- ensemble of SVMs for BCI P300 speller," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 55, no. 3, pp. 1147–1154, 2008.

[54] P. Sykacek, S. Roberts, and M. Stokes, "Adaptive BCI based on variational Bayesian Kalman filtering: an empirical evaluation," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 5, pp. 719–727, May 2004.

[55] G. Dornhege, J. D. R. Millan, T. Hinterberger, D. J. Mcfarland, K.-R. Muller, and T. J. Sejnowski, *Toward Brain-Computer Interfacing.* MIT Press, 2007.

[56] J. Wolpaw, N. Birbaumer, W. Heetderks, D. McFarland, P. Peckham, G. Schalk, E. Donchin, L. Quatrano, C. Robinson, and T. Vaughan, "Brain-Computer Interface technology: a review of the first international meeting," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 8, no. 2, pp. 164–173, Jun 2000.

[57] H. Cecotti, "Spelling with non-invasive brain-computer interfaces-current and future trends," *Journal of Physiology-Paris*, 2011.

[58] S. T. Ahi, H. Kambara, and Y. Koike, "A dictionary-driven P300 speller with a modified interface," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 19, no. 1, pp. 6–14, 2011.

[59] J. Hohne, M. Schreuder, B. Blankertz, and M. Tangermann, "Two-dimensional auditory p300 speller with predictive text system," in *2010 Annual Conf. IEEE Engineering in Medicine and Biology Society (EMBC),*. IEEE, 2010, pp. 4185–4188.

[60] J. Blumberg, J. Rickert, S. Waldert, A. Schulze-Bonhage, A. Aertsen, and C. Mehring, "Adaptive classification for Brain-Computer Interfaces," in *29th Annual Int. Conf. IEEE Engineering in Medicine and Biology Society*, Aug. 2007, pp. 2536–2539.

[61] J. Q. Gan, "Self-adapting BCI based on unsupervised learning," in *3rd International Workshop on Brain-Computer Interfaces*, 2006, pp. 50–51.

[62] X. Liao, D. Yao, and C. Li, "Transductive SVM for reducing the training effort in BCI," *J. Neural Eng.*, vol. 4, no. 3, pp. 246–254, 2007.

[63] Y. Li, H. Li, C. Guan, and Z. Chin, "A self-training semi-supervised support vector machine algorithm and its applications in brain computer interface," *IEEE Intl. Conf. Acoustics, Speech and Signal Processing, 2007. ICASSP 2007.*, vol. 1, pp. 385–388, 15-20 April 2007.

[64] G. Molina, "BCI adaptation using incremental-SVM learning," in *3rd Intl. IEEE/EMBS Conf. Neur. Eng, 2007. CNE'07.* IEEE, May 2007, pp. 337–341.

[65] C. Chen, W. Song, J. Zhang, Z. Hu, and H. Xu, "An adaptive feature extraction method for motor-imagery BCI systems," in *International Conference on Computational Intelligence and Security (CIS) 2010*, Dec. 2010, pp. 275 –279.

[66] K. Thomas, C. Guan, C. Lau, A. Vinod, and K. Ang, "Adaptive tracking of discriminative frequency components in electroencephalograms for a robust brain–computer interface," *J. Neur. Eng.*, vol. 8, p. 036007, 2011.

[67] Y. Li and C. Guan, "Joint feature re-extraction and classification using an iterative semi-supervised support vector machine algorithm," *Mach. Lear.*, vol. 71, no. 1, pp. 33–53, 2008.

[68] R. Chavarriaga, A. Biasiucci, K. Forster, D. Roggen, G. Troster, and J. del R Millan, "Adaptation of hybrid human-computer interaction systems using eeg error-related potentials," in *2010 Annual Int. Conf. IEEE Engineering in Medicine and Biology Society.* IEEE, 2010, pp. 4226–4229.

[69] A. Buttfield, P. Ferrez, and J. Millan, "Towards a robust BCI: Error potentials and online learning," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 14, no. 2, pp. 164–168, 2006.

[70] A. Combaz, N. Chumerin, N. Manyakov, A. Robben, J. Suykens, and M. Van Hulle, "Towards the detection of Error-Related Potentials and its integration in the context of a P300 speller brain-computer interface," *Neurocomputing*, vol. 80, no. 0, pp. 73–82, 2012.

[71] Y. Zhao and Q. Ji, "A new active learning method for EEG multi-class classification," *Energy Procedia*, vol. 13, pp. 3263–3268, 2011.

[72] A. Krogh and J. Vedelsby, "Neural network ensembles, cross validation, and active learning," in *Advances in Neural Information Processing Systems*. MIT Press, 1995, pp. 231–238.

[73] M. Zhong, F. Lotte, M. Girolami, and A. Lécuyer, "Classifying EEG for brain computer interfaces using gaussian processes," *Pattern Recognition Letters*, vol. 29, no. 3, pp. 354–359, 2008.

[74] B. A. S. Hasan and J. Gan, "Unsupervised adaptive GMM for BCI," in *4th International IEEE/EMBS Conf. Neural Engineering, NER'09.* IEEE, 2009, pp. 295–298.

[75] C. Vidaurre, M. Kawanabe, P. Bunau, B. Blankertz, and K. Muller, "Toward an unsupervised adaptation of LDA for brain-computer interfaces," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 3, pp. 587–597, March 2011.

[76] Y. Li, H. Kambara, Y. Koike, and M. Sugiyama, "Application of covariate shift adaptation techniques in brain–computer interfaces," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 6, pp. 1318–1324, 2010.

[77] K. Chen and S. Wang, "Semi-supervised learning via regularized boosting working on multiple semi-supervised assumptions," *IEEE Tran. Patt. Anal. and Mach. Intel.*, vol. 33, no. 1, pp. 129–143, 2011.

[78] S. Lu, C. Guan, and H. Zhang, "Unsupervised brain computer interface based on intersubject information and online adaptation," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 17, no. 2, pp. 135–145, 2009.

[79] J. Borisoff, S. Mason, and G. Birch, "Brain interface research for asynchronous control applications," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 14, no. 2, pp. 160–164, June 2006.

[80] A. Bashashati, S. Mason, J. Borisoff, R. Ward, and G. Birch, "A comparative study on generating training-data for self-paced brain interfaces," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 15, no. 1, pp. 59–66, March 2007.

[81] G. Townsend, B. Graimann, and G. Pfurtscheller, "Continuous EEG classification during motor imagery-simulation of an asynchronous BCI," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 12, no. 2, pp. 258–265, June 2004.

[82] B. Blankertz, K. Muller, D. Krusienski, G. Schalk, J. Wolpaw, A. Schlogl, G. Pfurtscheller, J. Millan, M. Schroder, and N. Birbaumer, "The BCI competition III: Validating alternative approaches to actual BCI problems," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 14, no. 2, pp. 153–159, 2006.

[83] J. Borisoff, S. Mason, A. Bashashati, and G. Birch, "Brain-computer interface design for asynchronous control applications: improvements to the LF-ASD asynchronous brain switch," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 985–992, 2004.

[84] G. Birch and S. Mason, "Brain-computer interface research at the Neil Squire Foundation," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 8, no. 2, pp. 193–195, 2000.

[85] S. Mason and G. Birch, "A brain-controlled switch for asynchronous control applications," *IEEE Trans. Biomed. Eng.*, vol. 47, no. 10, pp. 1297–1307, 2000.

[86] A. Bashashati, S. Mason, R. Ward, and G. Birch, "An improved asynchronous brain interface: making use of the temporal history of the LF-ASD feature vectors," *J. Neur. Eng.*, vol. 3, p. 87, 2006.

[87] P. Diez, V. Mut, E. Perona, and E. Leber, "Asynchronous BCI control using high-frequency SSVEP," *Journal of NeuroEngineering and Rehabilitation*, vol. 8, no. 1, p. 39, 2011.

[88] L. George, L. Bonnet, and A. Lécuyer, "Freeze the BCI until the user is ready: a pilot study of a BCI inhibitor," in *5th International Brain-Computer Interface Workshop*, Graz, Austria, 2011, p. accepted.

[89] R. Scherer, G. Muller, C. Neuper, B. Graimann, and G. Pfurtscheller, "An asynchronously controlled EEG-based virtual keyboard: improvement of the spelling rate," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 979–984, 2004.

[90] F. Aloise, F. Schettini, P. Aricò, F. Leotta, S. Salinari, D. Mattia, F. Babiloni, and F. Cincotti, "P300-based brain–computer interface for environmental control: an asynchronous approach," *J. Neur. Eng.*, vol. 8, p. 025025, 2011.

[91] Advanced Neuro Technology Website. [Online]. Available: http://www.ant-neuro.com/

[92] J. Jin, B. Allison, X. Wang, and C. Neuper, "A combined brain-computer interface based on P300 potentials and motion-onset visual evoked potentials," *J. Neurosci. Meth.*, vol. 205, no. 2, pp. 265 – 276, 2012.

[93] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *COLT' 98: Proc. Eleventh Annual Conf. Computational Learning Theory*. NY, USA: ACM, 1998, pp. 92–100.

[94] S. Goldman and Y. Zhou, "Enhancing supervised learning with unlabeled data," in *Proc. Seventeenth Int. Conf. MachineLearning (ICML-2000): June 29-July 2, 2000, Stanford University*. Morgan Kaufmann, 2000, pp. 327–334.

[95] W. Wang and Z.-H. Zhou, "Analyzing co-training style algorithms," in *ECML '07: Proc. 18th European Conf. Machine Learning*. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 454–465.

[96] Z.-H. Zhou and M. Li, "Semisupervised regression with cotraining-style algorithms," *IEEE Trans. on Knowl. and Data Eng.*, vol. 19, no. 11, pp. 1479–1493, 2007.

[97] D. J. Krusienski, E. W. Sellers, F. Cabestaing, S. Bayoudh, McFarland, M. Vaughan, and J. R. Wolpaw, "A comparison of classification techniques for the P300 speller," *J. Neural Eng.*, vol. 3, pp. 299–305, 2006.

[98] X. Lei, P. Yang, and D. Yao, "An empirical Bayesian framework for Brain-Computer Interfaces," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 17, no. 6, pp. 521–529, Dec. 2009.

[99] S. T. Ahi, H. Kambara, and Y. Koike, "A comparison of dimensionality reduction techniques for the P300 response," in *i-CREATe '09: Proc. 3rd Int. Convention on Rehabilitation Engineering & Assistive Technology*. New York, NY, USA: ACM, 2009, pp. 1–4.

[100] D. MacKay, "Bayesian interpolation," *Neural Computation*, vol. 4, no. 3, pp. 415–417, 1992.

[101] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[102] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, 4th ed. Chapman & Hall/CRC, 2007.

[103] O. Chapelle, B. Schölkopf, and A. Zien, Eds., *Semi-Supervised Learning*. Cambridge, MA: MIT Press, 2006.

[104] I. Cohen, F. G. Cozman, N. Sebe, M. C. Cirelo, and T. S. Huang, "Semisupervised learning of classifiers: Theory, algorithms, and their application to Human-Computer Interactions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 12, pp. 1553–1567, 2004.

[105] X. Zhu, "Semi-supervised learning literature survey," University of Wisconsin - Madison, Department of Computer Sciences, Tech. Rep., 2005.

[106] I. Shevelev, E. Mikhailova, V. Chicherov, V. Konishev, and D. Karlovskiy, "Spatial gradient of P300 in the Brain-Computer Interface paradigm," *Intl J. Psychophysiology*, vol. 69, no. 3, pp. 181 – 181, 2008.

[107] M. Treder and B. Blankertz, "(C)overt attention and visual speller design in an ERP-based brain-computer interface," *Behavioral and Brain Functions*, vol. 6, no. 1, p. 28, 2010.

[108] P. Melville and R. J. Mooney, "Constructing diverse classifier ensembles using artificial training examples," in *IJCAI'03: Proc. 18th Int. Joint Conf. Artificial intelligence*, 2003, pp. 505–510.

[109] G. Pfurtscheller, T. Solis-Escalante, R. Ortner, P. Linortner, and G. Muller-Putz, "Self-paced operation of an SSVEP-based orthosis with and without an imagery-based brain switch: A feasibility study towards a hybrid BCI," *IEEE Trans. Neural Sys. Rehab. Eng.*, vol. 18, no. 4, pp. 409–414, 2010.

[110] G. Pfurtscheller, B. Allison, C. Brunner, G. Bauernfeind, T. Solis-Escalante, R. Scherer, T. Zander, G. Mueller-Putz, C. Neuper, and N. Birbaumer, "The hybrid BCI," *Frontiers in Neuroscience*, vol. 4, 2010.

[111] 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.*, vol. 117, no. 3, pp. 531–537, 2006.

[112] B. Allison, D. McFarland, G. Schalk, S. Zheng, M. Jackson, and J. Wolpaw, "Towards an independent brain-computer interface using steady state visual evoked potentials," *Clin Neurophysiol*, vol. 119, no. 2, pp. 399–408, Feb 2008.

[113] J. Wolpaw, N. Birbaumer, D. McFarland, G. Pfurtscheller, and T. Vaughan, "Brain-computer interfaces for communication and control," *Clin. Neurophysiol.*, vol. 113, no. 6, pp. 767–791, 2002.

[114] G. H. Klem, H. O. Lüders, H. H. Jasper, and C. Elger, "The ten-twenty electrode system of the international federation," *Electroenceph. Clin. Neurophysiol.*, vol. 52, pp. 3–6, 1999.

[115] D. Zhu, J. Bieger, G. G. Molina, and R. M. Aarts, "A survey of stimulation methods used in SSVEP-based BCIs." *Comput. Intell. Neurosci.*, p. 702357, 2010.

[116] I. Volosyak, H. Cecotti, and A. Graser, "Optimal visual stimuli on LCD screens for SSVEP based brain-computer interfaces," in *4th Intl. IEEE/EMBS Conf. Neur. Eng., 2009. NER'09.* IEEE, 2009, pp. 447–450.

[117] N. Xu, X. Gao, B. Hong, X. Miao, S. Gao, and F. Yang, "BCI competition 2003-data set iib: enhancing p300 wave detection using ICA-based subspace projections for BCI applications," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1067–1072, June 2004.

[118] 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 Trans. Biomed. Eng.*, vol. 51, no. 4, pp. 1073–1076, 2004.

[119] B. Z. Allison, "Toward ubiquitous BCIs," in *Brain-Computer Interfaces*, ser. The Frontiers Collection, B. Graimann, G. Pfurtscheller, and B. Allison, Eds. Springer Berlin Heidelberg, 2010, pp. 357–387.

[120] U. Hoffmann, E. J. Fimbel, and T. Keller, "Brain-computer interface based on high frequency steady-state visual evoked potentials: A feasibility study," in *Proc. 4th Int. IEEE/EMBS Conf. Neural Engineering NER '09*, 2009, pp. 466–469.

[121] G. Molina and V. Mihajlovic, "Spatial filters to detect steady-state visual evoked potentials elicited by high frequency stimulation: BCI application," *Biomed. Eng.*, vol. 55, no. 3, pp. 173–182, 2010.

Appendix A

Publications

1. R C Panicker, S Puthusserypady and Y Sun, "An Asynchronous P300 BCI with SSVEP Based Control State Detection", *IEEE Tran. Biomed. Eng.*, vol. 58, no. 6, pp. 1781-88, 2011.

2. R C Panicker, S Puthusserypady and Y Sun, "Adaptation in P300 Brain-Computer Interfaces: A Two-Classifier Co-Training Approach", *IEEE Tran. Biomed. Eng.*, vol. 57, no. 12, pp. 2927-35, 2010.

3. R C Panicker, S Puthusserypady, A P Pryana and Y Sun, "Asynchronous P300 BCI: SSVEP-Based Control State Detection", *Proc. European. Sig. Pro. Conf. (EUSIPCO-2010)*, Aalborg, Denmark, Aug 2010.