VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI - 590018

Project Report

on

"SSVEP CLASSIFICATION"

Submitted in partial fulfilment of the requirements for the VIII Semester

Bachelor of Engineering

in

ELECTRONICS AND COMMUNICATION ENGINEERING

For the Academic Year

2016-2017

BY

Prashanth HC	1PE14EC099
Pavan Kumar D	1PE14EC094
Lakshmiraj CR	1PE14EC412
Sawan Singh Mahara	1PE14EC128

UNDER THE GUIDANCE OF

Prof. Vidya TV

Assistant Professor, Dept. of ECE, PESITBSC

PESIT SOUTH CAMPUS

Department of Electronics and Communication Engineering

PESIT - BANGALORE SOUTH CAMPUS

Hosur Road, Bengaluru - 560100

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI - 590018

Project Report

on

"SSVEP CLASSIFICATION"

Submitted in partial fulfilment of the requirements for the VIII Semester

Bachelor of Engineering

in

ELECTRONICS AND COMMUNICATION ENGINEERING

For the Academic Year

20117-2018

BY

Prashanth HC	1PE14EC099
Pavan Kumar D	1PE14EC094
Lakshmiraj CR	1PE14EC412
Sawan Singh Mahara	1PE14EC128

UNDER THE GUIDANCE OF

Prof. Vidya TV

Assistant Professor, Dept. of ECE, PESITBSC

Department of Electronics and Communication Engineering

PESIT - BANGALORE SOUTH CAMPUS

Hosur Road, Bengaluru - 560100

PESIT - BANGALORE SOUTH CAMPUS

HOSUR ROAD, BENGALURU - 560100

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the project work entitled "SSVEP Classification" carried out by ***Prashanth HC, Pavan Kumar D, Lakshmiraj CR, Sawan Singh Mahara***, bearing USN ***1PE13EC099, 1PE13EC094, 1PE12EC412,1PE14EC128*** respectively in partial fulfillment for the award of Degree of Bachelors (Bachelors of Engineering) in Electronics and Communication Engineering of Visvesvaraya Technological University, Belagavi during the year 2016-2017. It is certified that all corrections/ suggestions indicated for internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said Degree.

Signature of the Guide	Signature of the HOD	Signature of the Principal
Prof. Vidya TV Ramachandrula	**Dr. Subhash Kulkarni**	**Dr. J Surya Prasad**
Assistant Professor,ECE	**HOD, ECE**	**Principal, PESIT-BSC**

External Viva

Name of the Examiners	**Signature with Date**
1.	
2.	

Acknowledgements

It is paramount for us to express our sincerest gratitude to all of the present and past lecturers and students that guided us wholeheartedly in pursuing this venture without which, the fulfilment of this project would have been just a figment of a dream.

The college management and the very conducive Director/Principal of PESIT- Bangalore South Campus, Dr. J. Surya Prasad, also get our deepest regards for providing sterling laboratory and other campus facilities.

We would like to thank Dr. Subhash Kulkarni, Head of Department Electronics and Communication, PESIT Bangalore South Campus for giving us the support and encouragement that was necessary for the completion of this report.

We convey our gratitude also to our project guide Prof Vidya TV, Assistant Professor for providing insights and guidance as required by us.

ABSTRACT

This is a project on Brain Computer interfacing, which a system that creates a communication pathway that is a direct link between an external device and the brain. Here, we are looking at *SSVEP*, out of the many possible *BCI* paradigms. This Steady State Visually Evoked Potential is a phenomenon within which is the response of the brain to a flickering source of light. The response has been observed to have a waveform embedded in it, which is frequency and phase locked to this flickering stimulus.

The EEG waveform would be taken non-invasively from the occipital lobe of the brain by electrodes placed on the scalp of the person, and then used for classification. This project implements an SSVEP classifier using a statistical analysis of the EEG waveform. We are using a recently emerged EEG processing technique that relies on *covariance matrices* and their inherent geometry in a Riemannian space to cluster and classify EEG signals.

We have implemented the needed signal processing techniques on MATLAB and have also built an astable multivibrator circuit to control an array of flickering LEDs that act as the flickering stimulus. The dataset comes from the internet as well as recordings from the OpenBCI Hardware. The entire project material and references can be found online on github, here or on the following web link or the address in the appendix.

Keywords: BCI, SSVEP, Riemann Clustering, Riemann Manifold, covariance matrices

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter provides a high level overview of the human nervous system to readers interested in designing and using BrainComputer Interfaces (BCI). Essential neuroanatomy and physiology as well as the terminology used in BCIs are introduced.

1.1 The Central and Peripheral Nervous System

The human nervous system can be divided into two, namely the *central nervous system* (CNS) and the *peripheral nervous system* (PNS).
The CNS is made up of the brain and the *spinal cord*. is the main pathway that conveys motor-control signals from the brain to muscles all over the body and sensory feedback information from the muscles and skin back to the brain. The neurons in the spinal cord are also involved in reflex actions and rapid responses to dangerous stimuli such as fire. The reflex arc illustrating the response to an external stimulus is shown in Fig 1.1.
The PNS consists of the somatic nervous system (neurons connected to skeletal muscles, skin, and sense organs) and the autonomic nervous system (neurons that control visceral functions such as the pumping of the heart, breathing, etc.). Coming to the efferent neurons, they have the job of transmiting information from the CNS to the PNS. The afferent neurons, on the other hand, transmit information to CNS from PNS.

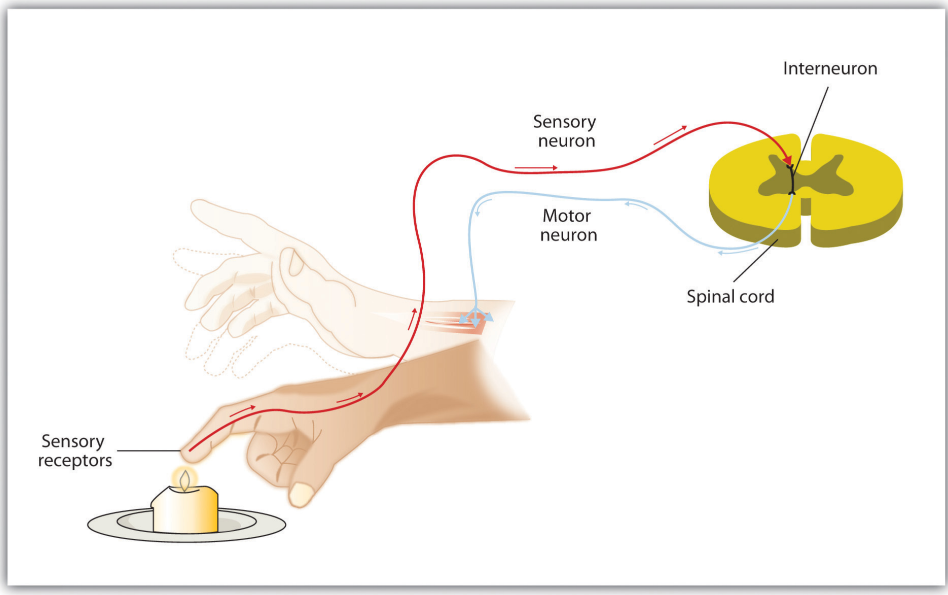

Figure 1.1: Response to an external Stimulus [2].

1.2 General description of the human brain

The brain is the central organ of the nervous system controlling, all voluntary and/or involuntary actions of body. The brain takes in all the inuts from sense organs and regulates other sensory functions like hormonal stimulations in other organs, for example. The brain is enclosed within a calcium cranium called the skull and makes up 80% of the head.

The brain can be classified as three separate parts. These are the cerebrum, the cerebellum and the medulla oblongata. We are currently focusing on electrical activity within the cerebrum or cerebral cortex. The cerebrum is made up of two cerebral hemispheres, the left and the right hemisphere which are connected by the corpus callosum. Each hemisphere, conventionally, is divided into four lobes: the frontal, parietal, temporal lobe and the occipital lobe. The last lobe is significant in this work. These lobes are yet again divided by the sulci, a.k.a the central sulcus a.k.a the fissure of Rolando, then the parieto-occipital sulcus, the lateral sulcus or Sylvian fissure, and lastly the temporal-occipital sulcus.

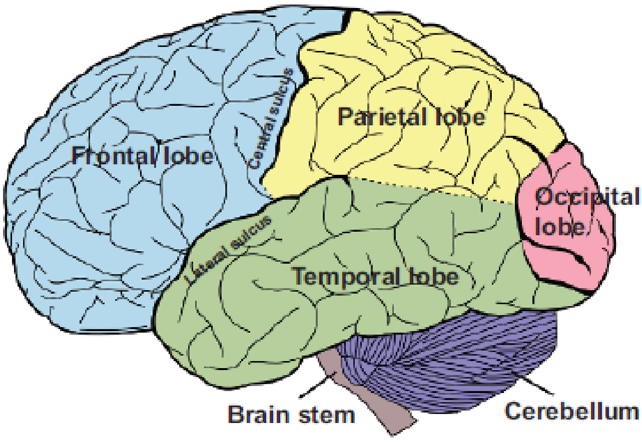

Figure 1.2: General view of brain's surfaces, main lobes and even the sulci.[1]

1.2.1 The occipital lobe

The occipital lobe and the parietal lobes are separated from each other by the occipitoparietal sulcus and from the temporal lobe by the pre-occipital notch. The primary and associative visual cortex corti are located near the occipital lobe. There also seems to be a correspondence which is systematic between all points in our visual field and the ones being represented in specific areas of V1. The relational property is termed as retinotopy.

1.2.2 The visual cortex

When the receptive field gets some visual stimuli, neurons in the corresponding cortex field fire, causing some action potentials. This region is the specific one wherein the generated potentials come from each and every neuron present here. Neuronal tuning occurs, which is the response if the neurons in question, respond to the stimuli in its field of existence. The tuning of V1 is so designated that it is able to fire to any vertical stimuli, within its receptive field. The remaining could be tuned to fire in a horizontal stimuli as well. Considering any complex tuning, it has been observed in the inferior temporal cortex. These neurons fire only under certain facial features being present. This is owing to the fact that facial recognition also takes place in this region.

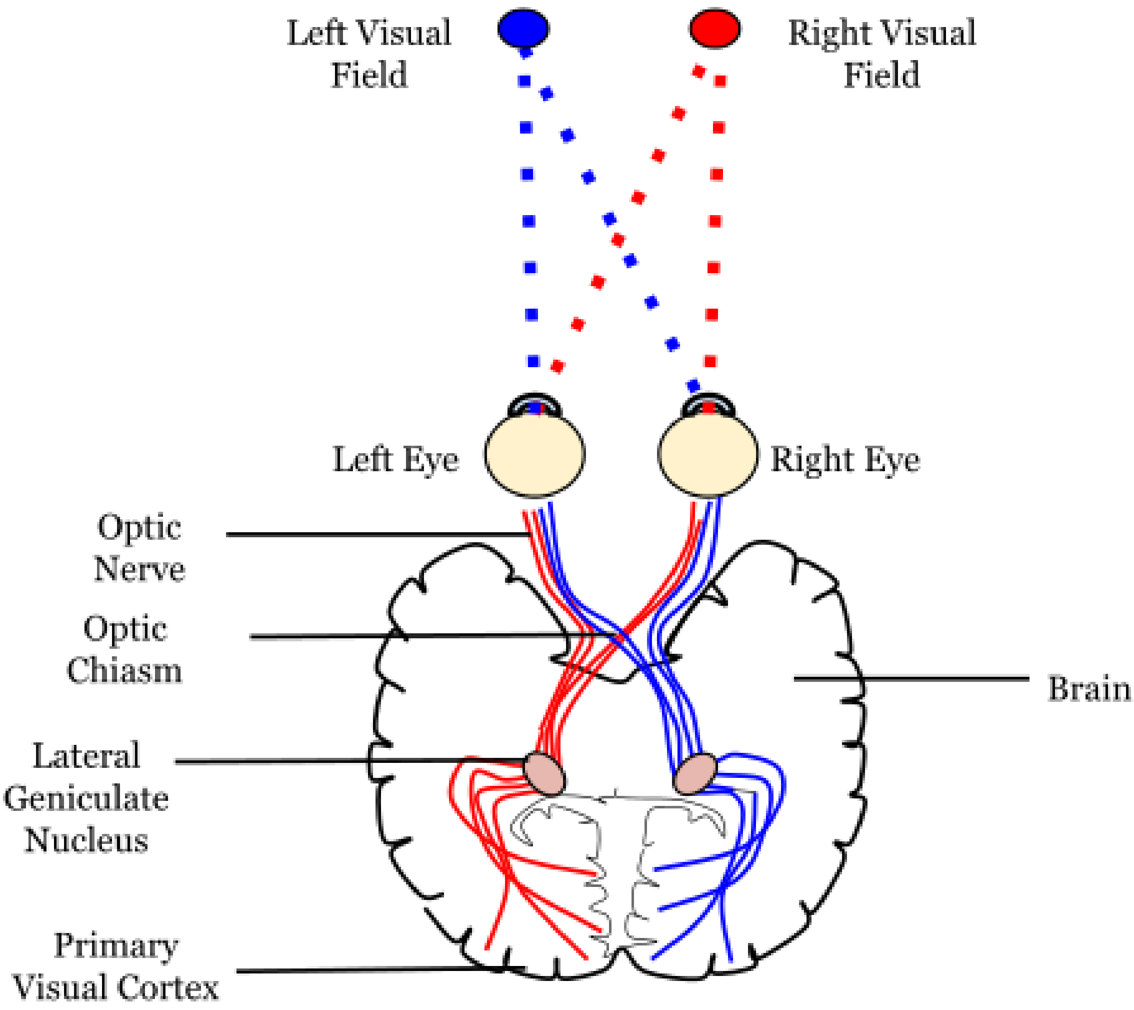

Figure 1.3: The visual pathway[3].

1.3 Introduction to Brain-Computer Interfaces

A Brain-Computer Interface or BCI is a direct communication pathway between an animal or human brain and an external device.
Designing a BCI requires knowledge of several disciplines. Brain computer interfacing architecture is summarized in figure 1.4, and it is usually consisting of 6 main stages [1]:

- **Brain activity recording** The raw signals from the user's scalp, originating from their brain are recorded. Out of the different measuring devices used, the most widely employed one is EEG (electroencephalography)
- **Preprocessing** any noise present in the recorded signal is cleaned up and re-

moved and a clean signal is passed down to the lower stages.

- **Feature extraction**The signals are described in terms of 'features' for example, the strength of a signal on an electrode or the signal frequency.
- **Classification** within a certain window a class set of features are observed from signals.This just implicates the type of brain signals which are identified according to some activity pattern. As an example, imagination of the movement of either left or right limbs can cause this. This is aptly termed as a "classifier".
- **Translation** links a command with a given pattern of brain activity which is identified in the users brain signals. In motor imagery, or example, whenever one imagines moving either right or left hand, the machine can detect this and translate it into some command, which can obviously be used to control some application like a wheelchair, or a robotic hand, etc.
- **Feedback** to inform a user about the results of his brain activity is sent. This has to be there to enable users to modulate some brain activities, thus vastly improving their control over the BCI.

Using a BCI generally demands two stages

1. An offline stage of calibration where appropriate system settings are determined and measured.
2. An online stage of operation where some user's brain activities are detected by analysing some patterns which are predetermined.

Once these stages are done, the reuslts are then employed to command some application. Research is still going on to avoid offline calibration and this work, although doens't fully avoid it, is much less computationally expensive.

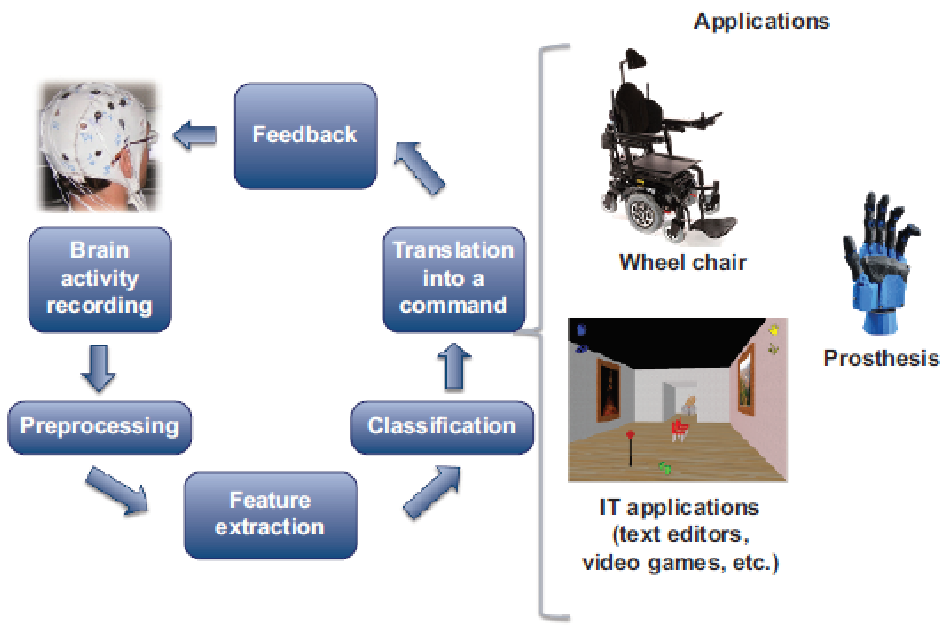

Figure 1.4: Architecture of a BCI working in real time, with some examples of application[1].

1.4 Types of BCIs

The different properties of BCIs enable us to classify them into different categories. Classification can happen as wither active or passive or reactive, dependant or independent, synchronous, asynchronous and even invasive or non-invasive and maybe hybrid.

Active/reactive/passive: If a user is involved actively when carrying out some voluntary tasks, the BCI is classified as active. If the BCI uses some imagined movement of the hand, like in the case of motor imagery, it is active BCI. If a users brain is reacting to some stimuli, it is unsurprisingly called reactive BCIs. Our proposed BCI system based on the SSVEP is reactive BCI. The BCI that doesn't analyse brain signals to perform commands and relies on the state of the user's brain fall under passive BCI.Anything dependant on user concentration or attention level is also categorised the same way.

Synchronous/asynchronous:If the actions of the user are controlled and modulated by the system such that the user can only take those actions at a certain time, it is a synchronous BCI. Allowing user interaction at all times, the existing interface would then be asynchronous. Our project is of this kind.

Dependent/independent: A BCI that is dependent on motor control is called a Dependent BCI system. Otherwise, it is independent. Since the users have to move their eyes in order to even observe stimuli, it is termed as a reactive BCI. Here, the stimulus comes from the LED circuit. If the user were allowed to control the BCI wthout moving a single muscle, it would then have been termed independent.

Invasive/non-invasive: Any invasive BCI, used data that is collected via sensors that are surgically implanted into the body. In BCI cases, it is usually into the skulls. Non-invasive paradigm do not require any surgery whatsoever. Data is collected from the skin's surface, albiet with a lot of noise.

Hybrid: Signls of various nature being combined in the same BCI is hybrid. Anything that uses hand movement (motor imagery) and also reactive BCI (SSVEP), for example is called hybrid BCI. Anything that uses muscular activity along with the other mentioned types of BCI would also be a hybrid type. Summarising, it is just a culmination of EEG brain signals along with other signals originating from the body.

Chapter 2

EEG Data acquisition and Processing

2.1 Principles of EEG acquisition

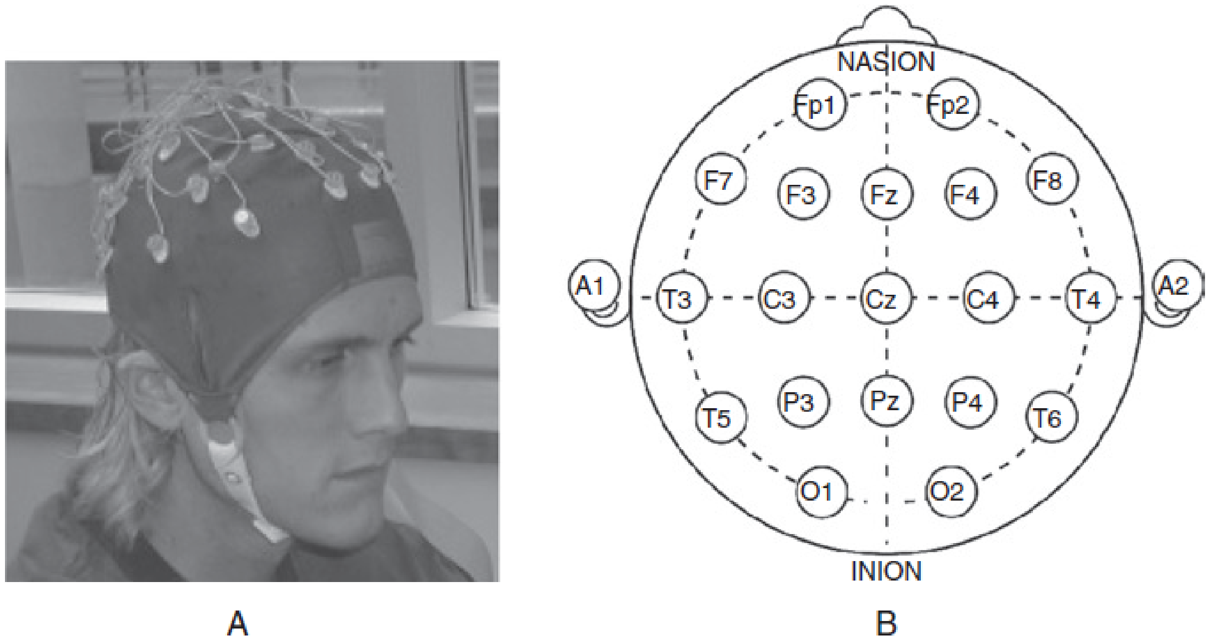

Figure 2.1: (A) Subject wearing a 32-electrode EEG cap. (B) International 1020 system for standardized EEG electrode locations on the head. C = Central, P = Parietal, T = temporal, F = frontal, Fp = frontal polar, O = occipital, A = mastoids (image A courtesy K.Miller; image B from Wikimedia Commons).

Electroencephalography or EEG provides electrical potential measurements in the form of a time signal for each electrode. Though they dont contain all the information of events transpiring in the brain, they contain significant data nonetheless about events that can be measured externally. This information is widely used in fields like neuroscience, psychology and cognitive science to study the electrophysiological nature of brain phenomenon. The information provided by EEG is not uniform, it is varied: it can show amplitude variations of the electric potential even at some certain frequencies. Also, depending on where on the scalp the signal is acquired, spatially also thre could be erratic fluctuations. Thus, if we want to observe any EEG phenomena, there are a number of different methods used to analyse signals in frequency, space or time domains.

Due to its very low amplitude (10-100 μV), EEG data only be measured only through amplifiers. Originally these EEG signal amplifiers were analog and resembled a seismograph, which used a pen to trace these signals on a roll of paper. Nowadays thanks to the the advent of moore's law and miniaturisation,the amplifiers can be digitally implemented, performing appropriate analog to digital conversion and then supplying the data in the form of sampled and quantized signals.

EEG signals indicate the average potential of many thousands of neurons firing radially from the area of measurement on the scalp. Tangential currents are not detected by EEG. Since signal attenuation is inversely proportional to the square of the distance, any potentials developed deep withing the brain will have attenuated by the time the EEG electrode picks it up. Hence EEG signals are mostly surface-level signals. EEG has poor spatial resolution, but great temporal resolution

The poor spatial resolution of EEG is caused primarily by the different layers of tissue that are interposed in between the signal and the sensor that is placed on the scalp. Each layer attenuates the signal more and more. The impulse response of these layers can be modelled as a low pass filter which acts to smear up the original signals. measured signals are in the range of a few microvolts justifying the use of powerful amplifiers and signal processing instruments for their processing.

Muscular artefacts and power-line noise (e.g., 50 Hz power-line interference) are common causes why the EEG signal gets corrupted due to its very low amplitude and vulnerability to noise. Muscle artifacts such as eye movements, talking, chewing, and head movements can all cause large artifacts in the EEG signal. Subjects are hence typically instructed to avoid all movement as much as possible, and powerful artifact removal algorithms are used toremove these artifacts. Various psychological states such as boredom and distraction can also lead to artifact generation.

2.2 EEG Montage

EEG data is acquired from electrodes laced on the scalp. Data is in the time domain and each electrode acts as a channel that is on the scalp. As only the potentials are measured, the signal does not directly relate to an electrode's electrical potential differences. Those need to be measured directly. Montages are used for measuring these potential differences..

Monopolar - The potential difference between an active site and common reference electrode is measured. The common electrode must be placed such that it would not be affected by any activity from the cerebrum. Unipolar montages have the advantage that the reference allows for comparisions that are valid in different pairings of electrodes. The drawbacks though are, obviously, that there is no ideal place to refer to. Earlobes are commonly used, however. EMG and ECG artefacts do occur here and so isn't robust.

Bipolar The potential difference between two active scalp sites is measured. Any activity in common with these sites is subtracted so that only difference in activity is recorded. The advantage in this configuration is the common noise between the two electrodes is eliminated and the signal-to-noise ratio is improved.

The choice of montage was paramount when analog amplifiers were used with their pen and paper recordings.The digital amplifiers now enable one to poseriorily process data and thus we can transform between one montage to another via codes.

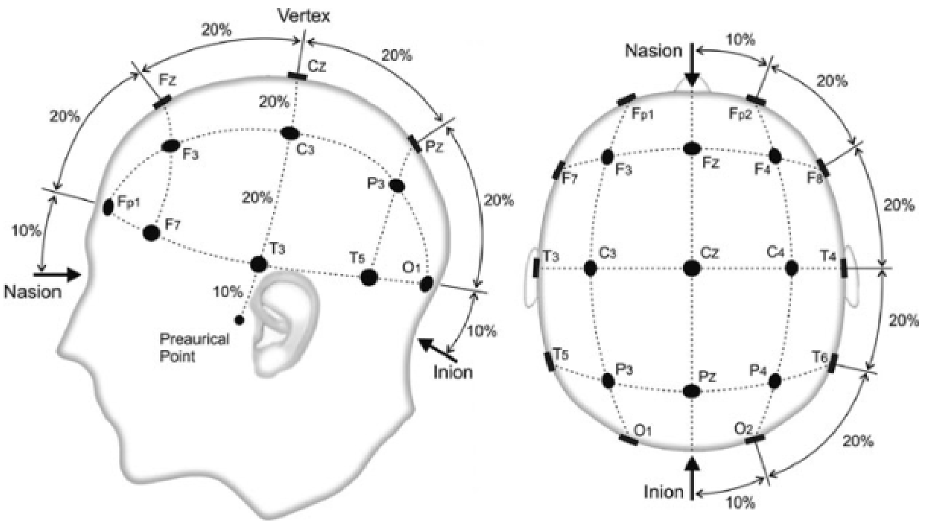

Figure 2.2: 2 International 10–20 Electrode System [5]

EEG recording and 10–20 System - It involves the subject wearing a cap or a net into which the recording electrodes are placed. In some cases, scalp locations may be prepared for recording by light abrasion to reduce impedance caused by dead skin cells. A conductive gel or paste is injected into the holes of the cap before placing the electrodes.

The international 10-20 system is a convention used to specify standardized electrode locations on the scalp. This is illustrated in Fig. 2.1. The mastoids located behind each ear (A1 and A2) can be used as reference electrodes. Other possible reference electrode locations are nasion, the top of nose, levelling with the eyes and inion, at the base of the skull namely at the midline between the back of the skull. From these points, various parameters from the skull are measured in the median and transverse or coronal planes. The locations of these elctrodes are found by dividing perimeters into 10% and 20% intervals.

The international 1020 system will ensure the consistency of the naming of the electrodes to enable reproducibility across laboratories. The number of electrodes used can vary from application to application and can be from a few for some targeted BCI applications to even 256 in some very high density arrays. Either of these methods can be used to measure EEG.

2.3 Sampling

To convert the recorded analog signal data into digital form, sampling is used. The data is measured at some instants separated by the sampling interval. This interval is the same throughout the recording in simple systems. The inverse of this interval is the sampling rate or sampling frequency. As an example, any signal sampled at 100 Hz or 100 times a second will have its sampling interval as 10 ms.

The sampling frequency determines the temporal resolution as well as the analysable frequency spectrum. According to Shannons sampling theory, the sampling frequency must be greater than two times the maximum frequency in the signal. The signal amplitude is encoded using a finite number of bits. An example being that 16 bits would provide 2^{16} values to encode the whole range of the EEG voltage value. With these 16 bits, any values between ± 600 μV have the amplitude precision of 0.0183 μV.

2.4 Segmentation

To aid processing and analysis, each event is marked with a time stamp in the signal. Each event is associated with a date and time stamp along with a label that specifies its nature. Events are marked by specific softwares, during data acquisition. It is more common to mark events during acquisition itself.

Epoching is the process where a particular segment of data is marked. The marked segment is then used for data analysis through a time window. These chunks of data are christened as epochs. In any EEG data set epochs are trials of similar events. Having these trials it is possible to classify new unkown data a priori using machine learning algorithms and other statistical methods. These trials are crucial for analysing EEG as it is possible to obtain enough data that can eventually reduce the effect of noise. Ensemble averages would then yeild ”evoked potentials”.

2.4.1 Frequency Filtering

Noise reduction: Undesirable behaviour can cause drastic signal deviations which can be easily detected and removed (e.g. because of loose electrode contact with the scalp).

Component separation: The components of the signals of interest can be directly identified in the time domain of these signals, especially if the majority is of low-frequency components. This also happens if there are large amplitude changes. This would then allow for these components to be isolated. Spatially filtering these components would help better identify the components since there are a lot of contributors to a singl component.

2.5 Feature Extraction

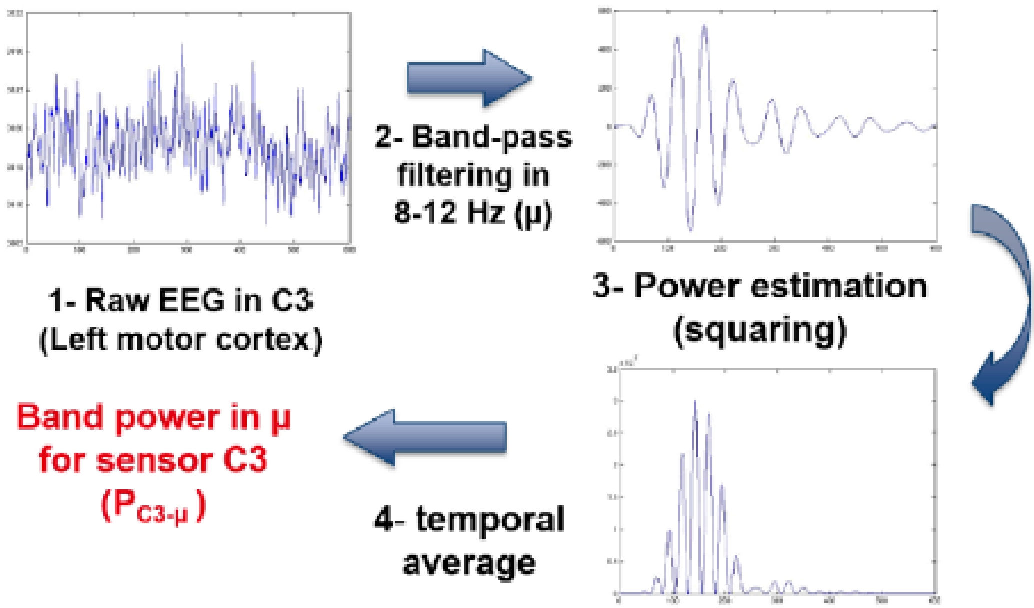

Figure 2.3: An example of band power calculation for an electrode placed at C3 in the motor cortex.

In BCI designing, the 3 main information sources that can be used for extracting features of signals are:

1. Spatial information: This depends on where the electrodes are placed and in which cortical region of the brain. This method is used to extract brain signals from one focused brain region and ignoring the others. In the context of this project the Occipital lobe are has been widely used and the electrodes are placed in and around that region.

2. Spectral (or frequency) information: Here, the signal power band is used as a feature. This gives information about which frequency signal has the highest power. This is widely used in this project.

3. Temporal information: Describes the variation of the amplitude of the signal over time. Time windows are used to window the signal and each window is used for further processing.

Ususally, it is needed that different sources of information are required for different BCIs. Our BCI working on evoked potentials, mainly uses temporal and spatial information.

Chapter 3

Literature Survey

3.1 EEG Feature Extraction and Signal Enhancement Methods

- To extract any features of EEG there is need to consider internal and external noise factors. So, to consider that we utilize some basic pre-processing techniques.
- EEG signal is found below 30Hz which contains required information for further analysis. Hence, Temporal filtering is used to remove internal and external noise. Then, higher frequencies are removed using simple low pass filters.
- Since, we are using low EEG signals then we have to perform enhancement methods to analyze signals. Enhancement methods include a lot of factors like such the neuromechanisms, or the methodology of recording, the cardinality of the electrodes used, etc.
  So, therefore some of the Spatial filtering techniques used here are:
  1. Principal Component Analysis: It is a linear map, transforming the number of variables that could be possibly correlated, into a smller subset of uncorrelated variables. These new variables are called principle components [9]. [6]
  2. Bipolar and Surface Laplacian: Bipolar signals can be extracted to highlight the potential differences among two interested electrodes.
     Laplacian filter extracts local activity around a particular electrode by subtracting the average of the activities present in four orthogonal neighbouring electrodes [6].
  3. Common Spatial Pattern (CSP): It is also one of the spatial filtering technique that is used to find Projection matrix. By using, Projection matrix different classes of single trial EEG data are decomposed and spatial struc-

tures of event related synchronization/desynchronization in EEG context is found. H. Ramoser et al. designed optimal spatial filters by method of CSP for filtering single-trial EEG during imagined hand movements [8].

So, by using this technique we can maximize difference between certain set of classes with the help of Projection matrix. [10]. Fabien Lotte et al. has improved the BCI designs using regularized common spatial pattern technique [11] [13].

3.2 Types of BCI

3.2.1 Selective Attention Based BCIs

Selective attention based BCI requires external stimuli which is given by some BCI system. The stimuli could be auditory or even somatosensory in nature. Here, each stimuli based on particular command and users should focus on corresponding stimulus. In typical BCI settings, each of the stimuli are associated with a commands, and the users have to focus their attention to some corresponding stimuli [13]. BCIs applicable to this kind of brain patterns are as follows:

- Steady State Visually Evoked Potentials (SSVEP): Here, focusing on one particular stimuli produces SSVEP witihin the visual cortex and it would be having the same frequency as targeted flicker. In this flicker, it happens at different frequencies in the range of 6-30Hz. When we pay pay attention to any one of these flickering stimuli, it would be eliciting an SSVEP in the visual cortex having the same frequency as the target flickers [13] Cheng et al. studied results from SSVEP BCI on virtual telephone keyboard allowing 13 keyboards [20]. Notable study on SSVEP BCI has been reported in [21].

- P300 Based BCIs:P300 is a elicited brain pattern that is formed due to any selective attentions to some flashed symbol or letter. Attention to any specific flashing symbol would elicit a brain pattern called P300, which develops in centro-parietal brain areas (close to location Pz) about 300ms after the presentation of the stimulus [13]. Thulasidas et al. have implemented a text input application (speller) based on the P300 event related potential and have obtained high accuracies by using an SVM classifier [18] It is developed in centro-parietal areas of brain (close to Pz) after nearly 300ms occurrence of stimulus. Well known work on this field presented in [17].

3.2.2 Motor Imagery Based BCIs

Motor imagery means a neural activity that is produced when a subject itself imagining particular action or movement. It is a spatiotemporally neural activity which is similar to the actual movement but smaller in magnitude. Various statistical and machine learning methods can be applied to differentiate between two or more imagined movements and allowing every imagined movement to be mapped to particular control signal. A variety of machine learning algorithms can be applied to discriminate between two or more types of imagined movements, allowing each imagined activity to be mapped to a particular control signal [13].

Notable reports on this field are : The Berlin BCI system is used with minimal subject training and also covers important ML concepts for BCI [14]. Saha et al. have performed two fold classification of MI using SVM as a classifier [12]. Some motor imagery was performed by subjects using LDA as a classifier [15]. Some similar studies using CSP have been done by Lotte et al. [11] The Berlin BCI system using minimal subject training covering important ML concepts for BCI [14] and Saha et al. have performed two class classification of motor imagery using SVM as a classifier [12].

3.2.3 Steady State Visually Evoked Potentials (SSVEP) based BCI

is a phenomenon in which there is a periodic response that is evoked within the brain whenever a human looks at some stimulus that is flickering at some frequency above 6Hz. Normally it woul be detected in the occipital region on scalp. SSVEP signal arises when person is hisattention on some visual that is flickering and less demanding than other mental issues. Any increase in the power spectrum of the SSVEP would be a favourable type of input to the classifier. This is because it is based on the detection of any increment anyways. Relative information between triggered response and stimulus can be found using some simple frequency domain algorithms. For SSVEP kind of BCI flickering adjustments are required to provide visual stimulus to the subject. As long as, artefacts which do not have overlapping with stimulus SSVEP is less sensitive to artefacts. So, most application using this is for subjects which control eye movement.

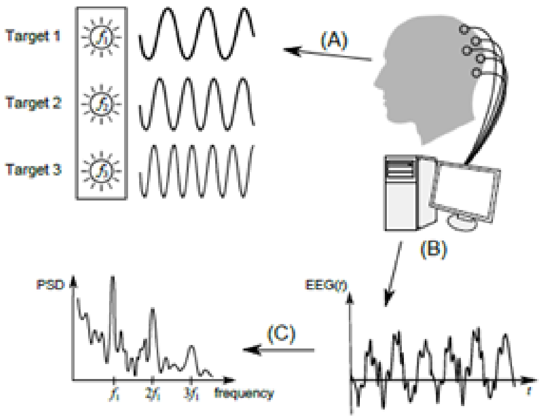

Figure 3.1: Basic idea of SSVEP decoding (A) Subject looks at a target 1 that has a flickering light at frequency $f1$. (B) EEG is recorded and then pre-procesed. (C) Some peaks at $f1, 2f1$ and $3f1$ in the obtained frequency spectrum will give a suggestion that should be the Target 1 which was the subjects choice.

3.3 Types of SSVEP Algorithms

3.3.1 Support Vector Machines (SVMs)

A Support Vector Machine (SVM) is a discriminative classifier which is based on a separating hyperplane. In other words, an optimal hyperplane which classifies new test inputs if found using given labeled training data (supervised learning). In two dimentional space, the hyperplane is a line dividing a plane into two parts with each class on either side. Given a set of some training examples, they would each be marked off as belonging to one of the two categories. The SVM training algorithm tries to build a model that will assign new examples to one of the categories. Thus it can be looked at as a non-probabilistic linear, binary classifier. Any new examples would then be mapped to the same space. It would then get predicted as belonging to one category depending on the side of the gap they lie in. [20]

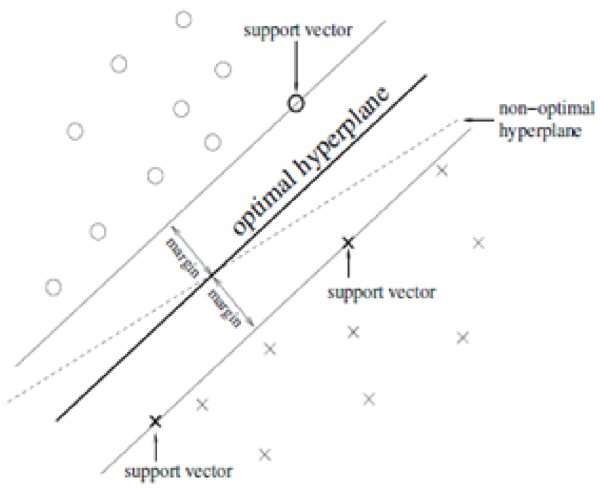

Figure 3.2: SVM finds the one optimal hyperplane generalizing.

Since SVM uses a discriminating hyperplane that identifies classes, the hyperplane selected would be the one that maximizes any margins between the classes or, put simply, the distance between nearest training points to the vector acting as a support (see Figure 3.2). Maximizing these margins has been found increasing generalization capabilities. The regularisation parameter C is used by SVMs that enables accomodation for outliers while leaving room for errors on training sets. These type of SVMs classify witj linear decision boundaries, thus earning the name linear SVM. Synchronous BCI problems succesfully have used these type of SVMs.

3.3.2 Linear Discriminant Analysis (LDA)

LDA is a method that finds linear combinations of some features that will separate two or even sometimes, more, classes events or classes. LDA (also known as Fishers LDA) uses hyperplanes that separate data which will belong to some particular classes, that are different. Two class problems will have their feature vectors depend on which side of the hyperplane they lie on. The conditional probability density functions in LDA are assumed to be normally distributed by convention.
This technique not computationally intensive and thus can be suitably used in online BCI systems The drawback however is that LDA is very linear and so complex, non-linear data from EEG will perform poorly here.

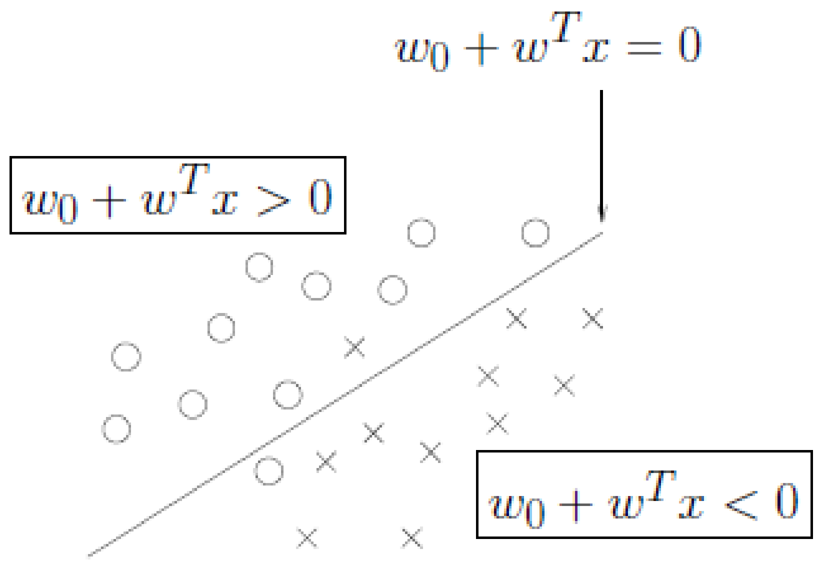

Figure 3.3: A hyperplane seperating two classes namely circles and crosses.

LDA assumes zero mean, unit variance guassian distribution of the data. It also assums that all classes have equal covariance matrice. When the projection that maximizes distances between classes is seeked, the hyperplane thus obtained is the solution to the LDA optimisation. The means and minimization of any interclass variances are also taken into account here.
Solving any N-class problem where $(N > 2)$ will definitely use several hyperplanes. The strategy employed is the famus "One versus all" stratergy. Here, the target class is the first class and all other classes are grouped as the "other", collectively. Computationally, this method is much more frugal and can be easily suited in any BCI application. LDA has been employed in a large number of BCI applications, suc-

cessfully. These include fields like motor imagery based BCI, P300 spelers, multi-class BCI or asynchronous BCI, etc. The drawback of LDA lies in its linearity that provides poor results on any complex non-linear EEG dataset [23].

3.3.3 Multivariate Linear Regression (MLR)

MLR is a statistical technique to predict the outcome of the responses of variables using several explanatory variables. MLR models the relationship between the explanatory and response variables. The model for MLR, given n observations, is:

$$y_i = B_0 + B_1 x_{i1} + B_2 x_{i2} + \ldots + +B_1 x_{i1} + E \tag{3.1}$$

$\forall i \in 1, 2..., n$ and y_i being the dependent variable , $x_i p$ being the independent variable

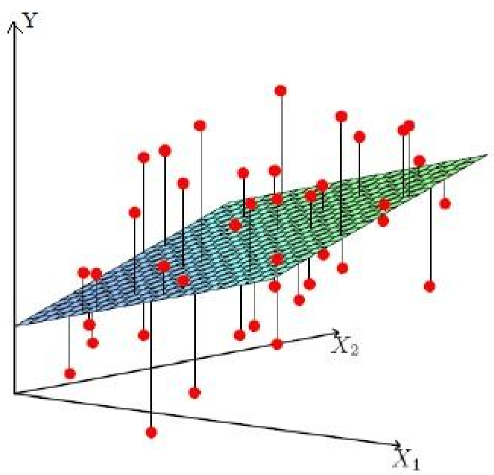

Figure 3.4: Regression performed with two independent variables from a single variate, $X : X1$ and $X2$ with one dependent variable Y [20].

A mathematical relationship among different random variables is found using MLR. Basically, MLR aims figure out how multiple independent variables will be related to the single one dependent variable. Once each of these independent factors are determined, determined to predict the single dependent variable, the obtained information on the multiple variables can now be used to create prediction that is accurate. This is on the level to which they effect the outcome variable. Creating a linear relationship to model all individual data pints is done by MLR

The multiple regression model relies on the following assumptions:

1. Dependent variables and independent variables are related, linearly.
2. The independent variables are not strongly correlated with each other.
3. Residuals should be have a 0 mean and a variance σ of its Gaussian distribution.

The multiple regression model predicts the outcomes that are based on the information given through multiple variables that are explanatory. This model, however, still doesn't achieve perfect accuracy as all the datapoints can still be offset from the outcome that the model predicts. [21].

3.3.4 Canonical Correlation Analysis (CCA)

Canonical correlation analysis (CCA) is an algorithmic method of of getting information from cross-covariance matrices. Consider two vectors $X = [X_1, ..., X_n]$nd $Y = [Y_1, ..., Y_m]$ of random variables, they can be also thought of as discrete random processes. Let us say that they have a correlation among their random variables. The canonical correlation analysis then finds the linear combination of the random processes X_i and Y_j which will give the maximum correlation with each other.

Typically, CCA can be used in experimental contexts to find the commonality among two random processes. Note that this correlation technique focuses on only two sets of variables.

Since it tries to find to find pairs of linear transformations for the random processes such that when they are transformed, the new sets of cariables will have the most correlation. The detection methods using CCA rely on the fact that any periodic pattern with a frequency same as the stimulus frequency or even one of its harmonics, can be always traced back within the brain signals. [25] CCA works on

two random processes. The variables in a set are actually signal amplitudes which can be recorded via multiple channels from local regions. The second random process can be that of the stimulus signals. As all periodic signal can be decomposed into a power series called the Fourier series. For example, any square-wave periodic signal, of a certain frequency is decomposable into the its harmonics given as fourier series coefficients.

$$y(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \\ y_3(t) \\ y_4(t) \\ y_5(t) \\ y_6(t) \end{pmatrix} = \begin{pmatrix} \sin(2\pi ft) \\ \cos(2\pi ft) \\ \sin(4pift) \\ \cos(4\pi ft) \\ \sin(6\pi ft) \\ \cos(6\pi ft) \end{pmatrix}, t = \frac{1}{S}, \frac{2}{S}, \ldots, \frac{T}{S}$$

Where f is the base frequency, S is the sampling rate and T is the number of sampling points, and Fig. 3.5 is illustrating the use of CCA in EEG signal analysis.

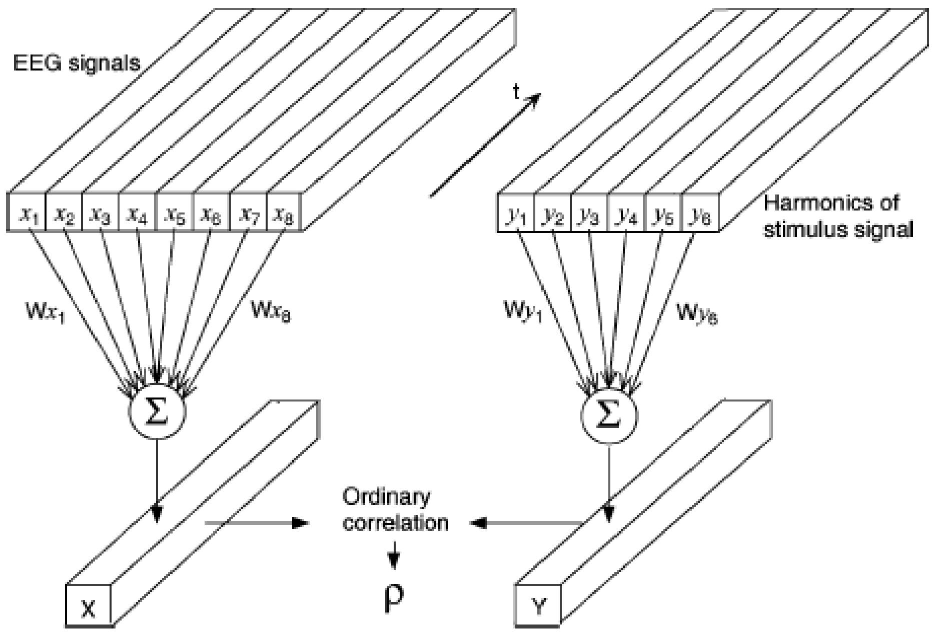

Figure 3.5: x_1 to x_8 are 8 EEG channel EEG data and y_1 to $y_6$are Fourier series representation of a period signal. The CCA finds the linear combination coefficients w_{x_1} to w_{x_8} and w_{y_1} to w_{y_6}, that give the maximum correlation between X and Y [25].

3.3.5 Power Spectral Density Analysis (PSDA)

Power Spectral Density Analysis exploits the fact that any periodic pattern with the same frequency as the stimulus or one of its harmonics, can be traced back in brain signals. Whenever SSVEP is present in the EEG, the magnitude of its magnitude spectrum or periodic pattern will only cover a narrow range of frequencies this can easily be seen in the frequency domain.

The PSDA method approximates the frequency of an SSVEP signal according to the peak of spectral amplitude. The PSD function will show the energy variation, vs the frequencies.

Power spectral density function (PSD) will show the strength of the variation in magnitude of the energy as a function of frequency. In other words, it shows the frequencies where the variations are strong and weak, graphically. As the unit of PSD is $\frac{W}{Hz}$ or energy per cycle/frequency, it is possible to obtain the total energy by just integrating the PSD over the bandwidth of interest. This PSD can be computed using the fast fourier transform or even using the autocorrelation function and then transforming it. [23].

3.3.6 Riemannian Manifold Clustering

This method is what we will be looking at and it uses the fact that any covariance matrix can be considered a point on a manifold or a surface. This surface is a subset of all possible Riemannian manifolds. The definition of this manifold is that it one which is endowed with an inner product defined on the tangent space, which varies smoothly from point to point. [3]

Basically if we could visualise this surface in 2 dimensions, we will see that EEG epochs set in one particular SSVEP frequency would have its covariance matrices clustered up. This is depicted in the figure shown below.

The cluster A would belong to one particular class and the cluster B would belong to another class. The centroid of these clusters could be found out and then that centroid would be the final trained classifier.

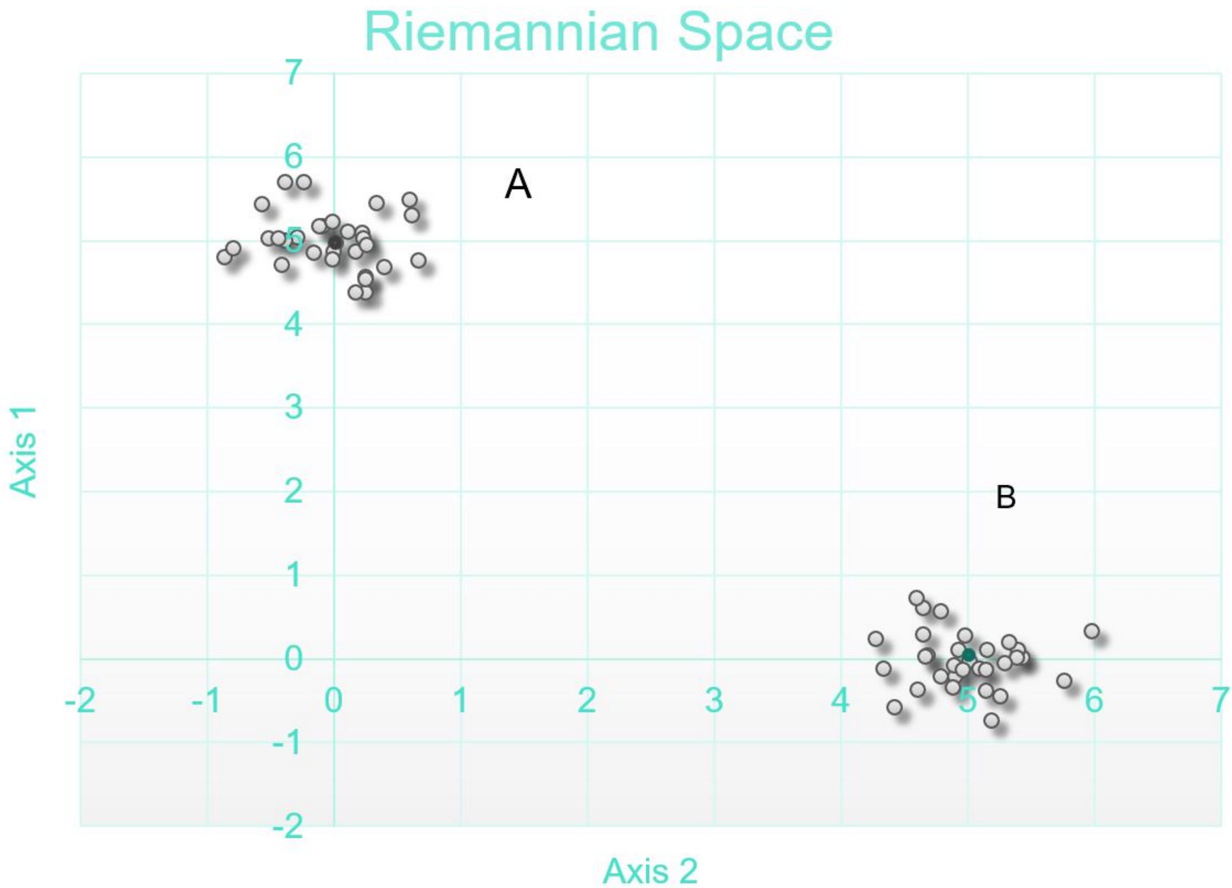

Figure 3.6: A 2D depiction of clustering in the Riemannian space. Note that it is just an illustration

This centroid, once computed will then help us classify new epochs. This would be done by utilising the notion of distance in this new space which is further dealt with in the system design and the implementation part of this report. The figure below depicts the trained classifier with the centroids for clusters A and B from figure 3.6.

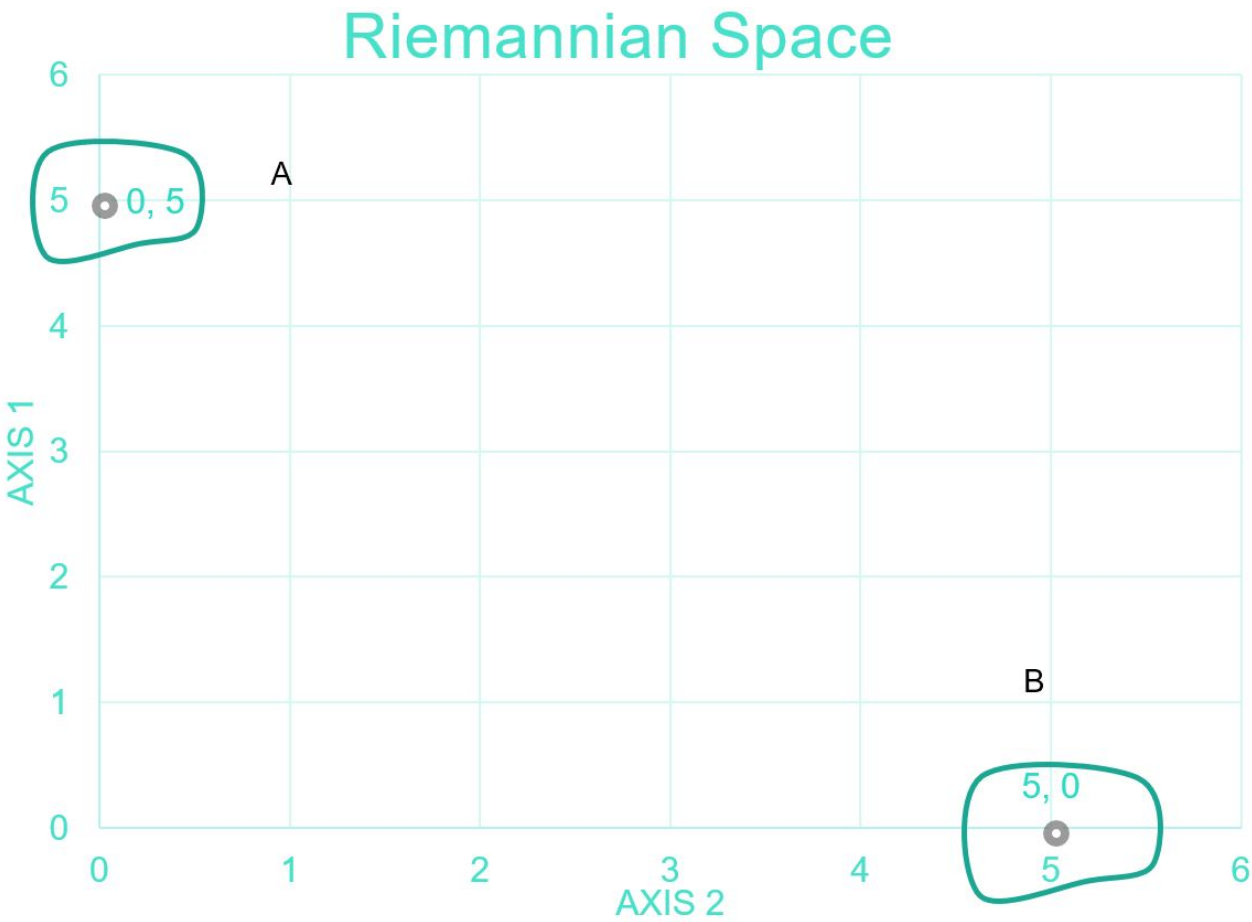

Figure 3.7: The Trained Classifier

Chapter 4

Hardware and Software Requirements Specification

4.1 Component Survey

BCI instruments can be either the invasive or the non-invasive type. The non-invasive methods will be resulting in EEG, fNIRs, MEG signals. The invasive methods outputting electrocorticography signals (ECog), local field potentials and even some single-init activity type of devices. The devices however, have a few components in common, such as the sensor, biosignal amplifiers, analog to digital converters, etc. All these devices work in tandem to relay information back to the user onto a computing device.

4.1.1 Hardware Survey

The hardwares surveyed for recording EEG signals are listed below. We have had access to only one of the components and have used datasets recorded on such a board. Since these instruments are expensive, we have not been able to purchase our own.

Device	Price	No. of channels	ADC	Brand Name
MindWave	$99.95	1	12-bit	NeuroSky
Emotiv	$399	14	16-bit	Emotiv Systems
OpenBCI	$499	8+8	24-bit	OpenBCI

Table 4.1: Some BCI hardware specifications

4.1.2 NeuroSky

In 2007 NeuroSky released the first affordable consumer based EEG along with the game NeuroBoy. This was also the first large scale EEG device to use dry sensor technology.

4.1.3 Emotiv

In 2009 Emotiv Systems had released the EPOC which is a 14 channel EEG device. This device can read 13 conscious states, 4 mental states, head movements and facial expressions. Using a dry sensor, it is the first commercial BCI to do such a thing. It can also be dampned using saline solutions in case a better connection is desired.

4.1.4 OpenBCI

This is the board that was used to record data with. The board comes from OpenBCI which is an open source brain-computer interface platform. The 32bit Board is an Arduino-compatible, 8-channel (can be extended to 16-channel) neural interface with a 32-bit processor. EEG, EMG, and EKG can all be looked at with this device. The core of this device is a PIC32MX250F128B microcontroller having the chipKIT bootloader that gives it lots of local memory and fast processing speeds. The board communicates wirelessly to a computer via the OpenBCI programmable USB dongle, which is based on the RFDuino radio module. The OpenBCI boards have a growing list of data output formats(csv, gdf, edf) , making them compatible with an expanding collection of existing biofeedback applications and tools. We have relied on the gdf format to record EEG data.

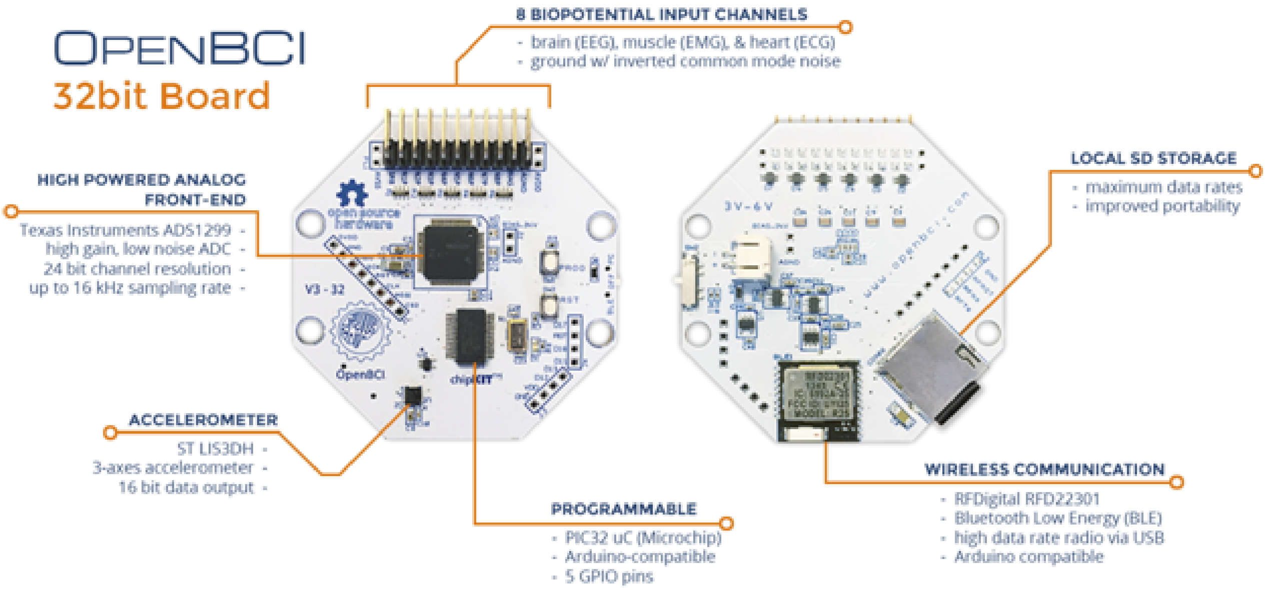

Figure 4.1: Overview of OpenBCI 32-bit Board Specifications

Technical Specifications

OpenBCI 32bit Board:

- 8 differential, high gain, low noise input channels
- Compatible with active and passive electrodes
- Texas Instruments ADS1299 ADC Analog Front End
- PIC32MX250F128B microcontroller w/chipKIT^{TM} bootloader (50MHz)
- RFduino^{TM} Low Power Bluetooth^{TM} radio
- 24-bit channel data resolution
- Programmable gain: 1, 2, 4, 6, 8, 12, 24
- 3.3V digital operating voltage
- $\pm$ 2.5V analog operating voltage
- 3.3-6V input voltage
- LIS3DH accelerometer
- Micro SD card slot
- 5 GPIO pins, 3 of which can be Analog

4.1.5 555 Timer

The well known integrated circuit chip used here is a device that can be used in a variety of oscillatory and timer applications. . The 555 can be used to provide time delays, as an oscillator, and as a flip-flop element. Derivatives provide two (556) or four (558) timing circuits in one package.[28]

The Astable circuit

The 555 timer can be used as an astable multivibrator, easily. The circuit we have implemented uses two variable resistors R_1 and R_2 that help manually adjust the dutycycle and frequency of the device to a very high accuracy and precision. The schematic was designed in Eagle software along with the circuit.

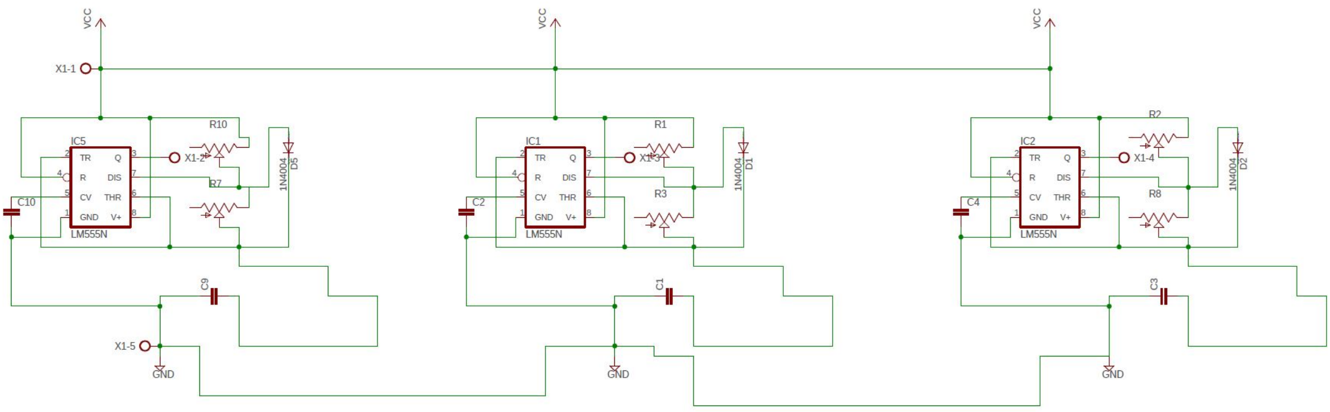

Figure 4.2: The circuit schematic of the astable multivibrator designed by us.

We have used three astable multivibrator circuits to achieve compactness and also to allow for control of 6 LEDs. Two PCBs were designed as 6 circuits could not be made onto a small enough board. The PCB schematic is shown below.

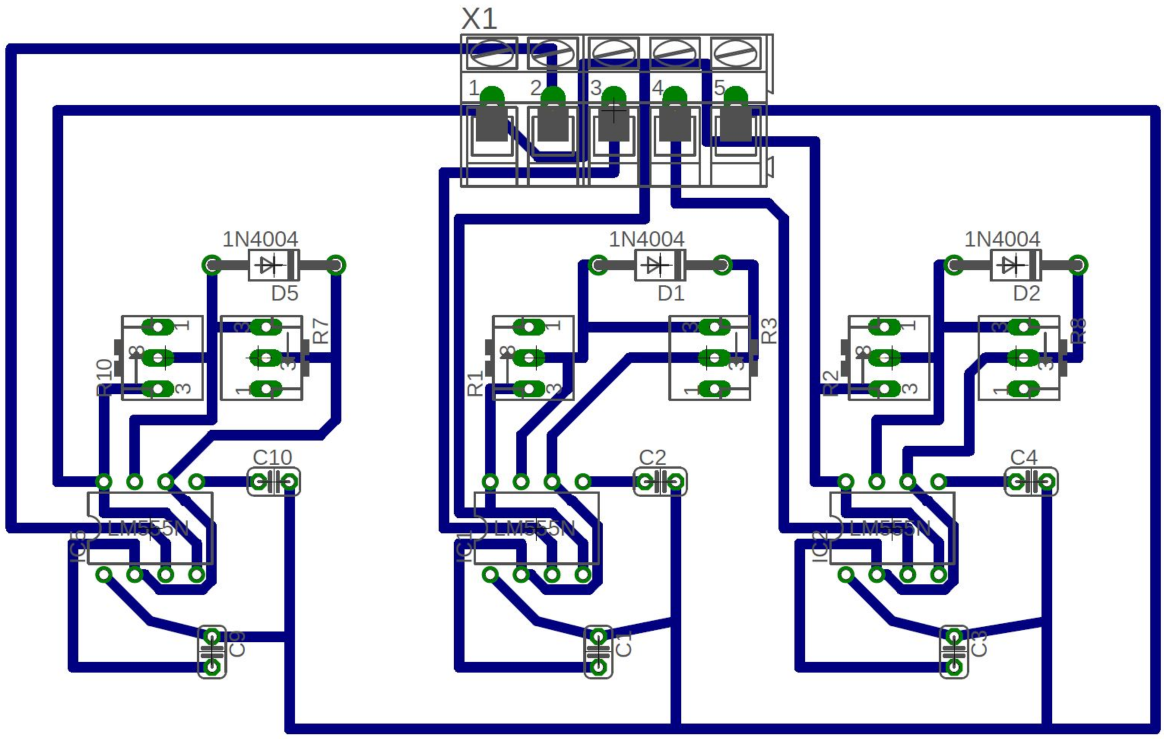

Figure 4.3: The PCB layout schematic of the astable multivibrator.

The routing was manually done on EAGLE and a 1mm drill was used to make the connecting rails. The board is a single layered design to accommodate for simplicity.

4.2 Software Survey

4.2.1 MATLAB

Proprietary scientific computing software from Mathworks. The capability of MATLAB can be extended by using external toolboxes such as:

- EEGLAB/BCILAB
  (a) Developed since 2010 at Swartz Center for Computational Neuroscience, UCSD (precursors dating back to 2006)
  (b) BCILAB is a MATLAB-based, cross-platform, offline and online analysis.
  (c) Largest collection of BCI algorithms.
  (d) Complex Internal Framework requires expertise to extend and is relatively little native support for acquisition systems but can tie into real-time experimentation frameworks (BCI2000, LSL)

- BioSig
  (a) BIOSIG toolbox contains many useful functions for biomedical signal processing.
  (b) Developed at TU Graz since at least 2002
  (c) One of the oldest open-source BCI toolboxes, for MATLAB/Octave (cross-platform)
  (d) Large amount of functionality : Various Biomedical Data Formats supported, Quality Control and Artifact Processing, Signal Processing and Feature extraction,Statistical analysis, False Discovery Rate (FDR), Statistical Toolbox for data with missing Samples(for handling NaN)
  (e) Offline analysis only. No real-time hardware or computation support
  (f) GNU license, General Public License.

- Barachant Covariance toolbox
  (a) This toolbox contain a set of matlab functions dedicated to covariance matrices estimation and manipulation.
  (b) The key functions mainly focus on Riemanian geometry of Symmetric Positive Definite matrices.

(c) Functionality include: Various Geodesic distance metrics, Function for finding Mean of covariance matrices, tangent space of covariance matrices, multiclass classification.

(d) GNU license, General Public License.

- Filter Designer

(a) This toolbox contains appropriate matlab functions to implement different digital filters.

(b) IIR and FIR filters can be designed just by inputting the passband and stop band frequencies and attenuation.

(c) The tool also enables one to implement a code that accepts a digital stream of data as input and puts out a stream of filtered signals as its output.

(d) The designer can be brought up by typing 'filterDesigner' on the matlab command line

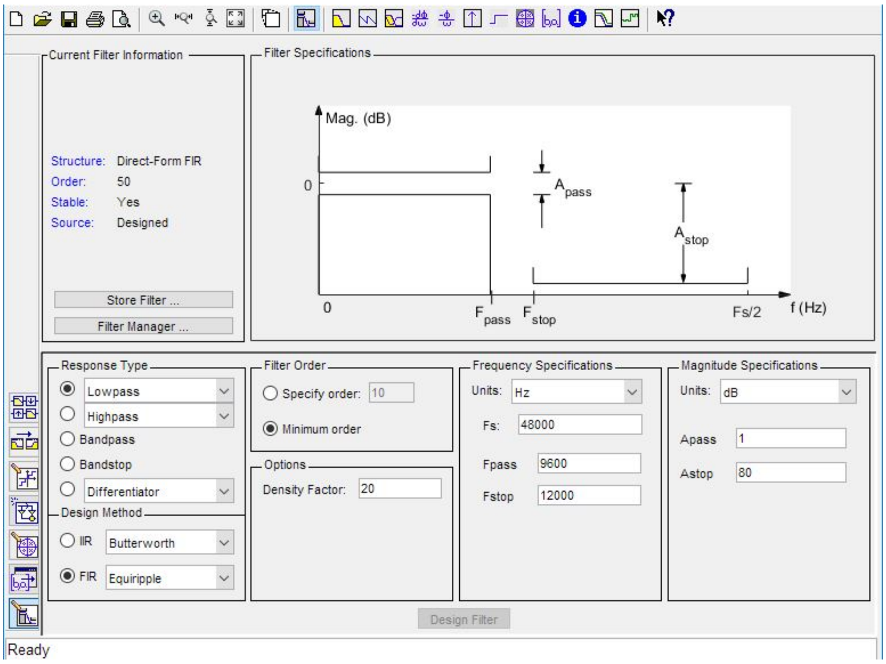

Figure 4.4: The filter designing tool in MATLAB.

4.2.2 Eagle

Overview

EAGLE is a EDA (electronic design automation) software tool which is scriptable with PCB (printed circuit board) and CAM (computer aided manufacturing) features. It was developed by American Software Company AUTODESK. EAGLE is an acronym described by Easily Applicable Graphical Layout Editor. It is one of user friendly and powerful layout tool which consist of useful libraries for electronic circuit designers.

- EAGLE software application consist of following features that are important for any circuit design,
- a. Schematic Editor: EAGLE tool consist of Editor for making design of circuits which is the first task to be done in preparing PCB. The Schematics from this tool are stored under. SCH extension and electronic parts are stored under .LBR extension.
- b. Layout Editor: After schematic of circuit is made layout is prepared in Layout Editor. Here, the files stored are called board files and stored under .BRD extension. Based on the connections made in schematics this feature makes use of back-annotation(netlist data) and auto-routing for correct connections of circuit.
- Most of the design firms uses Standard file formats namely Sieb Meyer and Excellon formats which will be supported in EAGLE application tool and it supports Gerber and PostScript formats which are most common formats in EAGLE.
- EAGLE design software tool is an inexpensive and affordable tool for PCB designs and it runs on Linux, Mac and Windows.

Chapter 5

System Design

BCI Application

5.1 Overview

Here we deal with the algorithm used in the final BCI project in the offline methodology. For the training stage, the data is first pre-processed by first having the epochs extracted, then appropriate filtering done on it. Then the centres of the covariance matrices clusters are found and then the classifier is trained. To use it, new epochs are then run through the classifier with the same pre-processing. When SSVEP is detected, the overall accuracy is plotted on the screen along with the confusion matrix.

The basic idea of any brain computer interface is as shown in the figure. From the occipital lobe of the brain the SSVEP stimulus can be sensed on the scalp of a person, as this lobe is the one dealing with the visual processing. The electrodes used are gold cupped with some electro conductive gel applied onto it to facilitate an electrical connection.

An analog to digital converter is used to sample and quantise the EEG signals. This ADC is interfaced to a microcontroller that then relays these quantised bits to the PC via a bluetooth module. The signal is then processed as needed by the PC via a suitable program. An OpenBCI board was used for data acquisition. The board's specifications are given in the appendix and literature review.

For pre-processing the EEG data received onto the PC is passed through a digital IIR Chebyshev, type 1 band pass filter. Depending on the type of frequency we are trying to cluster up, the centre frequency changes for this filter. Once the frequency is chosen, an appropriate band pass filter is used with a bandwidth of 2Hz. This concludes the temporal filtering phase of the EEG epochs. Another temporal filter can be added to remove the effects of power-line noise or any other electrical activity, by using a notch filter around 50Hz. After this, the epochs are then subject to the algorithms discussed further.

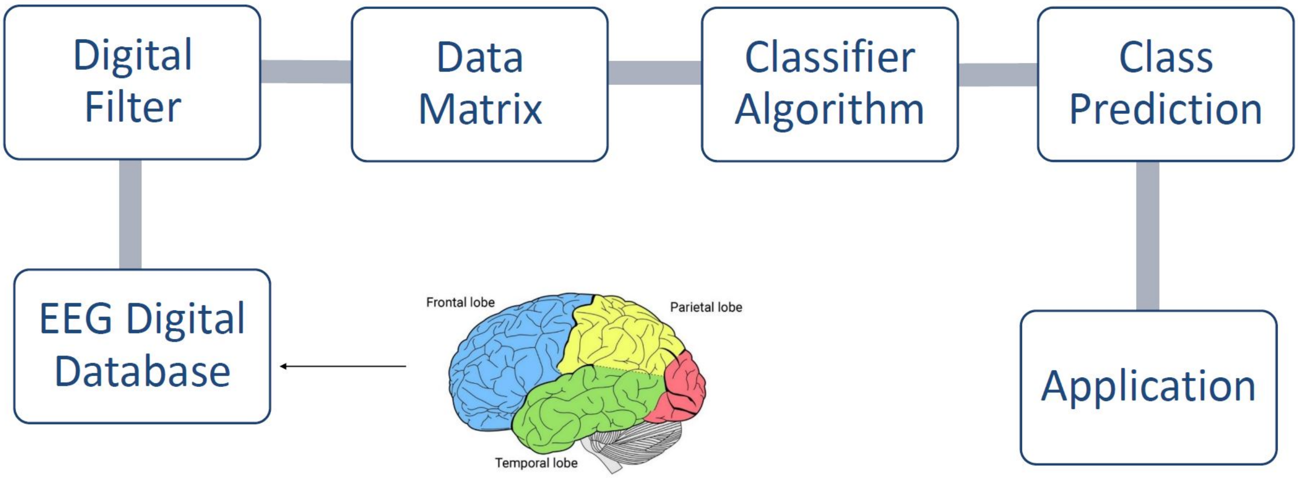

Figure 5.1: General setup of a BCI.

5.2 Offline Algorithm

5.2.1 Algorithm 1

Prerequisites:
The data stored in the form of a matrix, X where each of the trials out of I for a particular frequency, are stored in the rows. Each column contains the different frequencies out of F chosen.

$$X = \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1I} \\ \vdots & & \ddots & \\ f_{F1} & f_{F2} & \cdots & f_{FI} \end{bmatrix} \tag{5.1}$$

Each of the f sub matrices are the EEG epochs arranged as a $C \times N$ matrix of numbers, where C is the number of channels and N is the number of samples

$$f = \begin{bmatrix} sample_{11} & sample_{12} & \dots & sample_{1N} \\ \vdots & & \ddots & \\ sample_{C1} & sample_{C2} & \dots & sample_{CN} \end{bmatrix} \tag{5.2}$$

After this is done, the following algorithms are run.

Overview:

Inputs: The data matrix X. The labelling is inherently present in this structure.
Outputs: The predicted centres of the clusters.

1. Each of the sub-matrices f in a row is passed through a Chebyshev type 1 band-pass filter.

2. Next, each of these filtered sub-matrices then have their individual covariance matrices computed. The result will be a $C \times C$ matrix, which then replaces the original f matrix.

3. We are left with a structure named X_{cov} where

$$X_{\text{cov}} = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1I} \\ \vdots & & \ddots & \\ c_{F1} & c_{F2} & \dots & c_{FI} \end{bmatrix} \tag{5.3}$$

4. Each of the c covariance matrices are now points on a Riemann manifold where the distance between each point can be computed with

$$\delta(\Sigma_1, \Sigma_2) = \|Log(\Sigma_1^{-1}\Sigma_2)\|_F = \sum_{c=1}^{C} log^2(\lambda_c) \tag{5.4}$$

   Here $\|\cdot\|_F$ indicates the frobinius norm and here, is defined for the matrix the λ_c of the matrix $\Sigma_1^{-1}\Sigma_2$ as the sum of the squares of the logarithm of its eigen values

5. Knowing this distance, the centroid of the entire cluster of trials obtained can be found as Σ which is another covariance matrix that satisfies the following optimisation problem

$$\mu(\Sigma_1, ..., \Sigma_I) = \underset{\Sigma \in M}{\arg\min} \sum_{i=1}^{I} \delta^2(\Sigma_i, \Sigma) \tag{5.5}$$

6. This is repeated for all the F classes, each having I trials

7. The result is the matrix χ that contains the covariance matrices' centres as arranged below.

$$\chi = \begin{bmatrix} \Sigma_1 \\ \vdots \\ \Sigma_F \end{bmatrix} \tag{5.6}$$

5.2.2 Algorithm 2

Overview:

Inputs: The covariance centres χ, the EEG epoch ε to be classified
Outputs: The predicted class of the epoch ε.

1. The epoch ε is taken as it is, without any filtering and converted to its covariance matrix, which should be of the same size as the ones used in training i.e $C \times C$

2. Calling this covariance matrix as κ, the distance between this and all the $\Sigma \in \chi$ is computed using equation 5.4.

3. The result is the $F \times 1$ matrix Δ containing distances between this κ and all other covariance centres χ

$$\Delta = \begin{bmatrix} \delta(\Sigma_1, \kappa) \\ \vdots \\ \delta(\Sigma_F, \kappa) \end{bmatrix} \tag{5.7}$$

4. The row j with the smallest number is found and from this, the covariance trial κ is classified as belonging with the cluster centred around Σ_j

5. The output prediction is the class labelled j

5.2.3 SSVEP Visual Stimulus

The visual stimulus to generate SSVEP signals is taken from a the led array that flickers a red led.

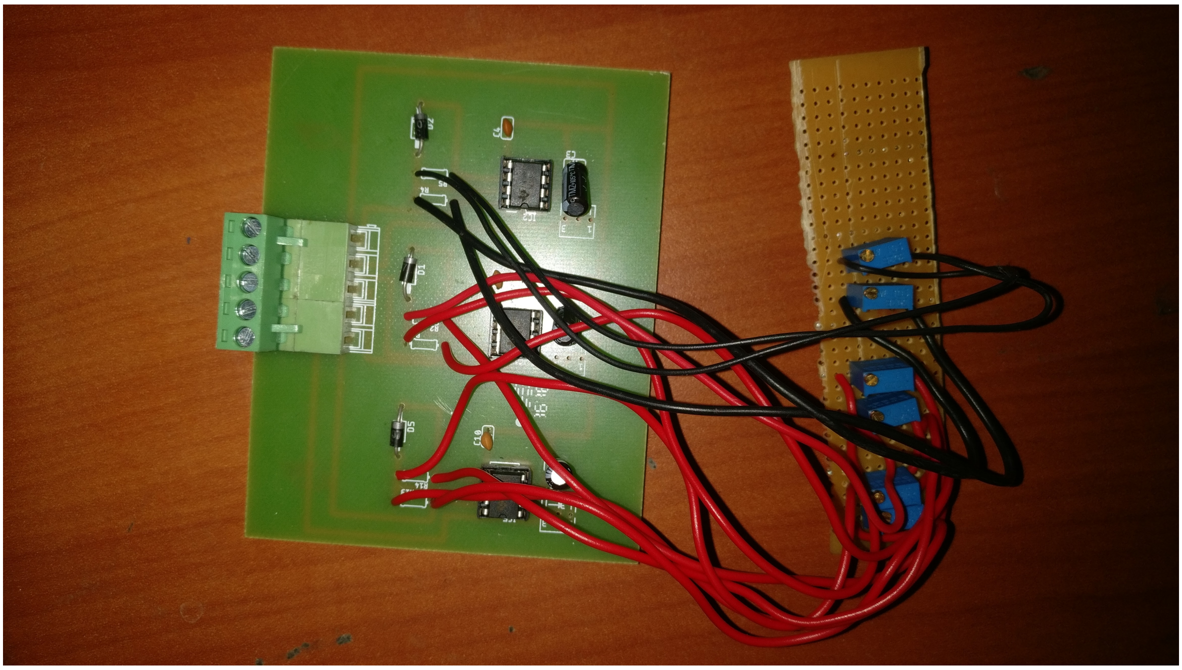

Figure 5.2: The LED controller circuit

5.2.4 SSVEP Experimental Protocol

1. To record data, the subject was made to look at the led array for a few seconds, when prompted.
2. After 5 seconds, the user is prompted to look away or rest for another 5 seconds or so
3. The user is again prompted to look at an led which is flickering at the same fixed frequency.
4. This is repeated for 8 times and then the user is made to repeat the same with another set of frequencies.

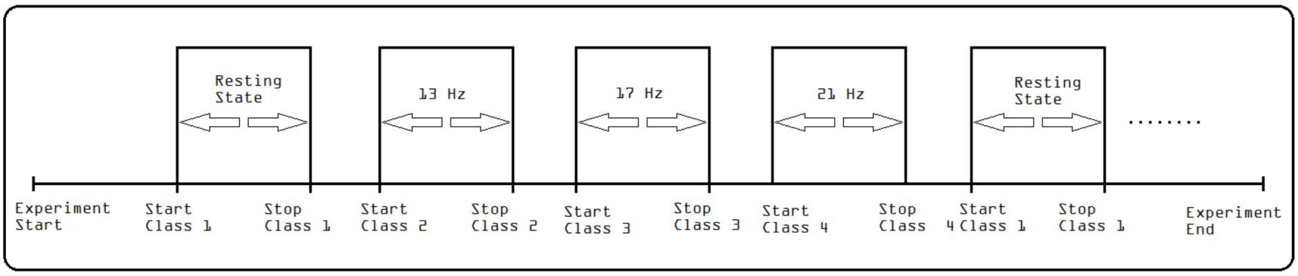

Figure 5.3: The SSVEP Protocol

Chapter 6

Implementation

6.1 Classification in MATLAB

6.1.1 Covariance

The implementation followed here closely replicates the algorithm and procedures depicted in the paper written by Emmanuel Kalunga. As mentioned earlier, the pre-processing is done using a Chebyshev type 1 filter. Then the covariance matrices are formed using an ignorance model of pdf and then, the clustering algorithm is performed this, as detailed in chapter 5.
The general formula for covariance that we use is the sample covariance matrix where:

$$\Sigma_{scm} = \frac{1}{N-1}\sum_{n=1}^{N}(f_n-\mu_f)(f_n-\mu_f)^T$$

Where μ_x is a vector having the means of each feature/channel in the EEG epoch f

In general, if there are tree random processes A, B, and C then the covariance matrix for them would be

$$C = \begin{pmatrix} cov(A,A) & cov(A,B) & cov(A,C) \\ cov(B,A) & cov(B,B) & cov(B,C) \\ cov(C,A) & cov(C,B) & cov(C,C) \end{pmatrix}$$

Which is a symmetric matrix that has positive eigen values.

6.1.2 Distance

For any function $d(x,y)$ to be called a distance, it must satisfy the following properties

1. $d(x,y) \geq 0$
2. $d(x,y) = 0 \implies x = y$
3. $d(x,y) = d(y,x) \geq 0$
4. $d(x,z) \leq d(x,y) + d(y,z)$

All this is satisfied by the distance metric

$$\delta(\Sigma_1, \Sigma_2) = \|Log(\Sigma_1^{-1}\Sigma_2)\|_F = \sum_{c=1}^{C} log^2(\lambda_c) \tag{6.1}$$

Here $\|\cdot\|_F$ indicates the frobinius norm and here, is defined for the matrix the λ_c of the matrix $\Sigma_1^{-1}\Sigma_2$ as the sum of the squares of the logarithm of its eigen values
The MATLAB code used is included in Appendix C.

6.1.3 Filter

A Chebyshev type 1 filter is an IIR filter with ripples in the passband and a smooth stopband. The one used by us, for a 17 Hz centre frequency has been computed to have an order of 13. The magnitude and phase plot obtained is as shown for a 17 Hz band pass filter.

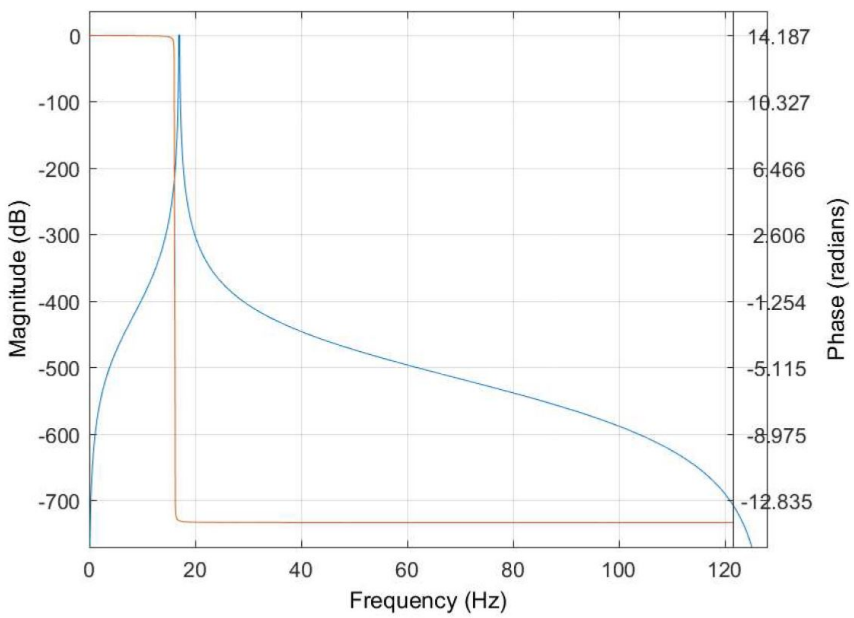

Figure 6.1: Magnitude and phase response of the bandpass filter used.

6.1.4 GUI

A graphical user interface was also created to help predict classes. The code was designed in a way so as to give the user the choice of choosing which classes they would like to run the algorithm against. The result is that an overall accuracy bar graph is plotted, along with a confusion matrix.
The GUI created is as shown in the picture below.

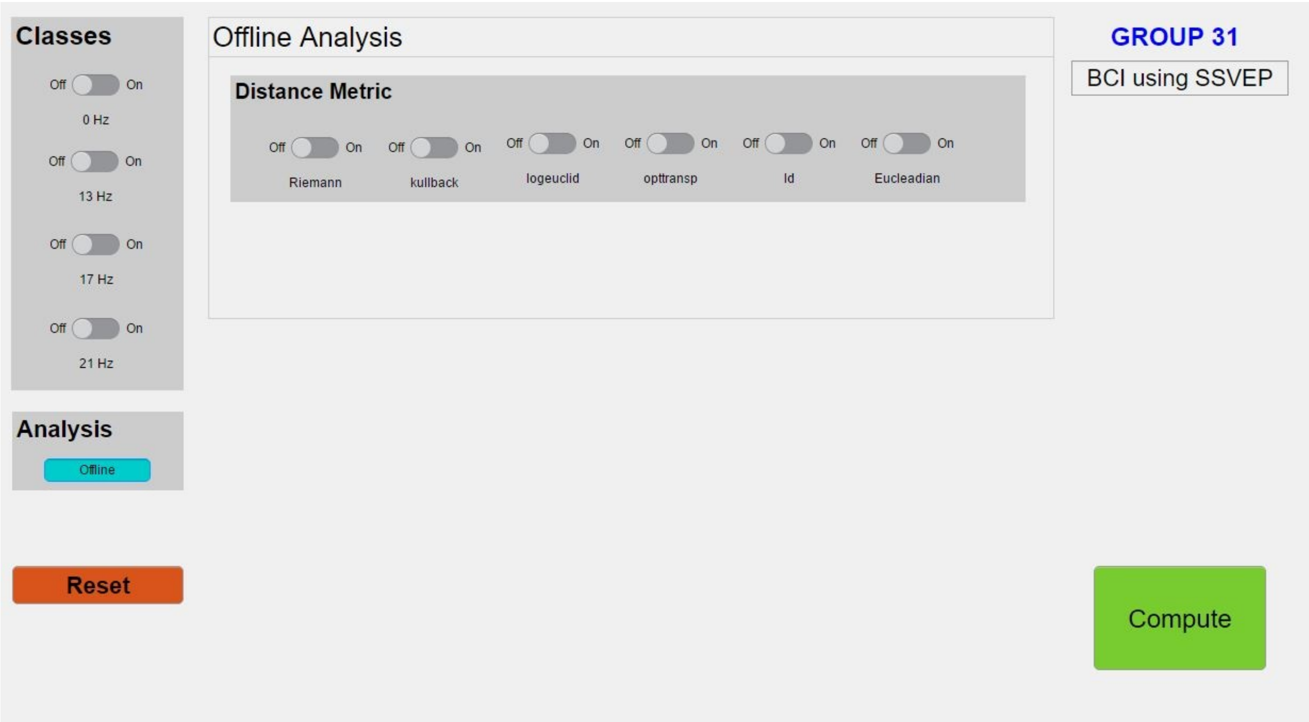

Figure 6.2: GUI Snapshot

Chapter 7

Result Analysis

The final outcome of our project is highlighted here. The offline analysis has achieved a GUI implementation on MATLAB that can allow for one to choose the number of classes to evaluate and different distance metrics being applied to them. The results are shown in the following sections

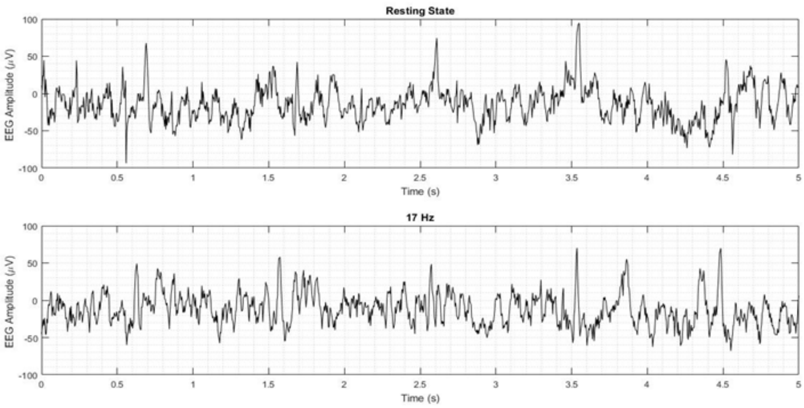

Figure 7.1: Time domain depiction of SSVEP. Normal brain activity is also depicted to contrast with.

For the offline analysis, the outcome was an overall accuracy score that looked at the amout of times a particular class was predicted correctly and more general confusion matrix, which gives a clearer picture of the behaviour of the classifier.

7.1 General Accuracy

As explained earlier, this was obtained by attempting to count the nnumber of times a particular class was correctly predicted, by feeding a labelled test set. The accuracy percentage was defined as:

$$Acc\% = \frac{\text{number of correct predictions for a class}}{\text{number of times that class was tested}} \tag{7.1}$$

Results are shown only for the Riemannian distance metric as that was observed to give the most consistently good performance overall, (empirically tested). The results for two class, three class and four class are summarised below. This was trained with 48 different EEG epochs and tested on 16.

- For two class, we have

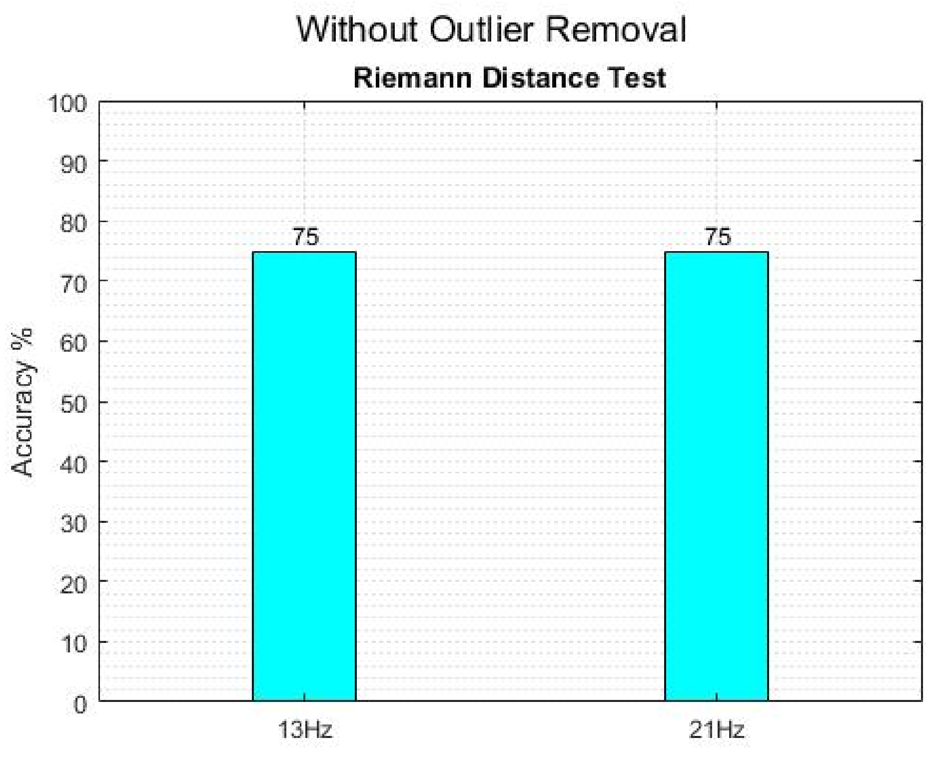

Figure 7.2: Considering only 13Hz and 21Hz frequencies

- For three class, we have

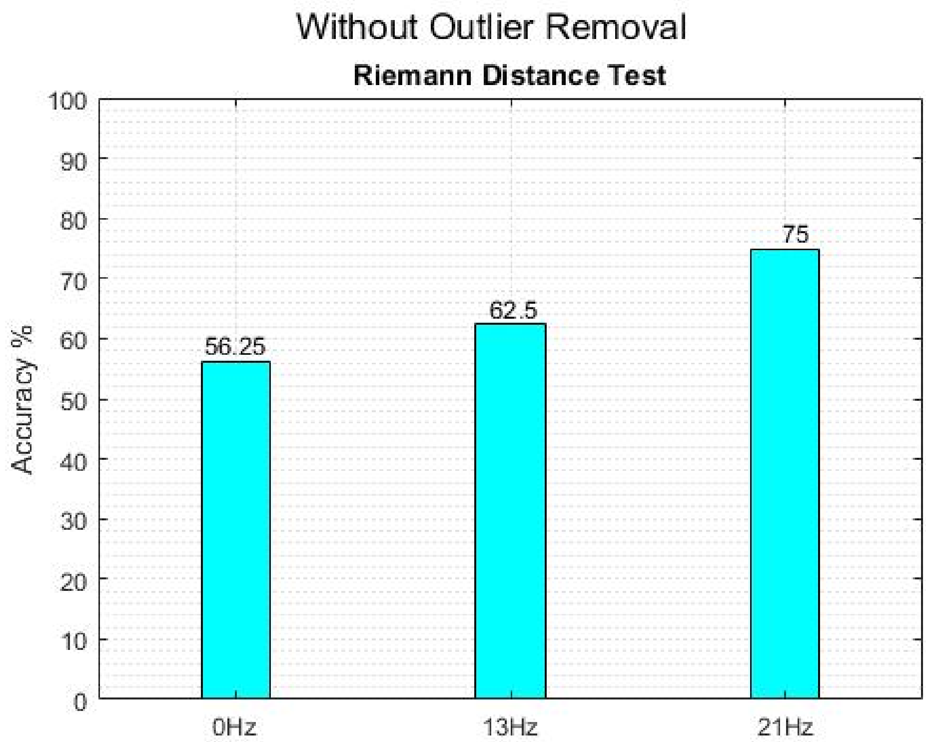

Figure 7.3: Considering no SSVEP as one class along with 13Hz and 21Hz frequencies

- For four class, we have

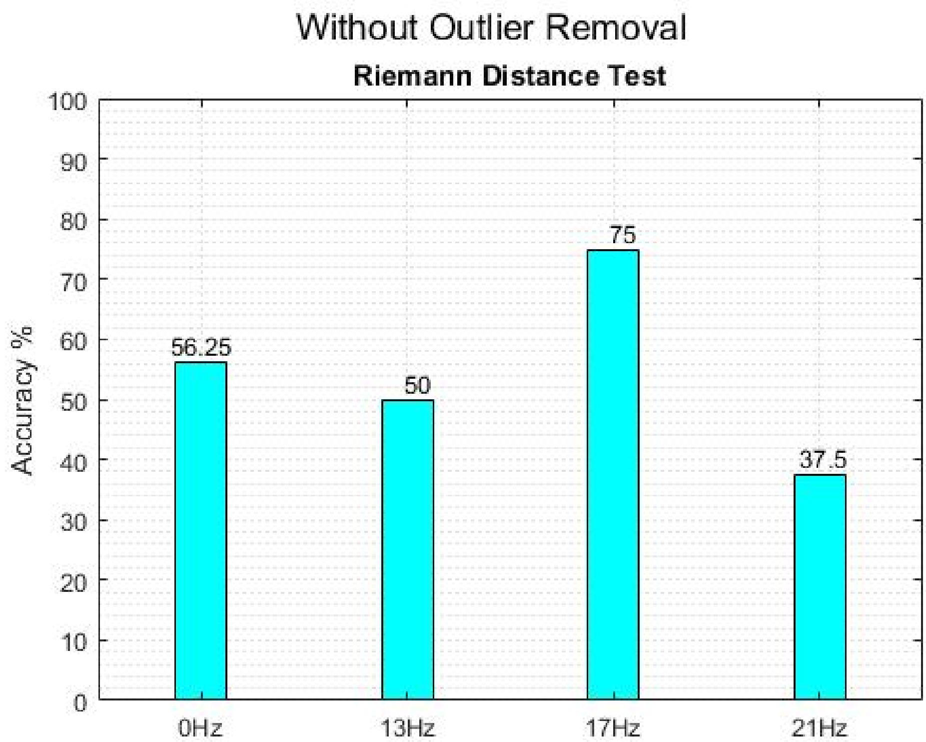

Figure 7.4: Considering four classes, namely 0Hz, 13Hz, 21Hz and 17Hz

1. As can be seen above, the main benefit of using the Riemannian clustering approach is the fact that no SSVEP conditions can also be classified. Unlike the generally used CCA method, that provides no way for the user to classify 0Hz. This means that in a CCA classifier, the machine would output one class even if the user is not looking at a flickering stimulus, thus can be prone to random fluctuating outputs.

2. Another thing to be noted here is that the accuracies fall off as the number of classes increase. This is due to the fact that on the riemannian manifold, the covariance matrices' clusters are close to one another.

7.2 Confusion Matrix

A brief look at what a confusion matrix is would help in understanding the following results if one is uncertain about it. This part can be skipped if one is already familiar with it.

7.2.1 Overview

1. A confusion matrix is just a table often used to describe the overall performance of a classifier and in turn, a classification model. This table considers all possible input and output combinations and tabulates them all.

2. A few parameters are also defined for this table to take into account the performance of the classifier by considering different aspects of the results obtained.

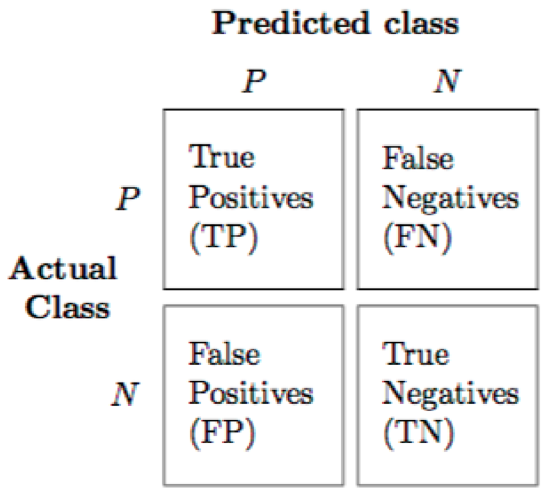

Figure 7.5: An illustration of a confusion matrix

Confusion Matrix Parameters:

1. **True Positive (TP)**: A true positive is when the prediction is correct for ”positive class”.

2. **False Positive (FP)**: A false positive occurs when the prediction falsely classifies the output as belonging to the ”positive class”.

3. **False Negative (FN)**: A false Negative occurs when the prediction falsely classifies the output as belonging to the ”negative class”.

4. **True Negative (TN)**: A true negative occurs when the prediction correctly classifies the output as belonging to the ”Negative class”.

”Positive class” and ”negative class” simple are the two broad terminologies used to help understand the classification results. A class is ”positive” if it is the desired output under consideration. A class is ”negative if it is any other class other than the positive.

Performance Evaluation Parameters

1. Accuracy:

$$Accuracy = \frac{\sum TP + \sum TN}{\sum Total\ Population}$$

   Accuracy is the measurment of the degree of closeness of any measurement of some quantity to its true value

2. True Positive Rate (TPR) or Probability of Detection:

$$TPR = \frac{\sum TP}{\sum Condition\ Positive}$$

   True Positive Rate, also called sensitivity or hit-rate will measure the proportion of positives that were correctly identified as such .

3. False Positive Rate (FPR) or Probability of False Alarm:

$$FPR = \frac{\sum FP}{\sum Condition\ Negative}$$

   False Positive Rate defines the probability of a false alarm.

4. False Negative Rate (FNR) or Miss Rate:

$$FNR = \frac{\sum FN}{\sum Condition\ Positive}$$

False Negative Rate describes the number of missed detections in the experiment .

7.2.2 Confusion Results

The following section displays the results obtained from the confusion matrix.

1. Using all the two classes, the confusion matrix is shown along with the tabulation of the error and hit rates.

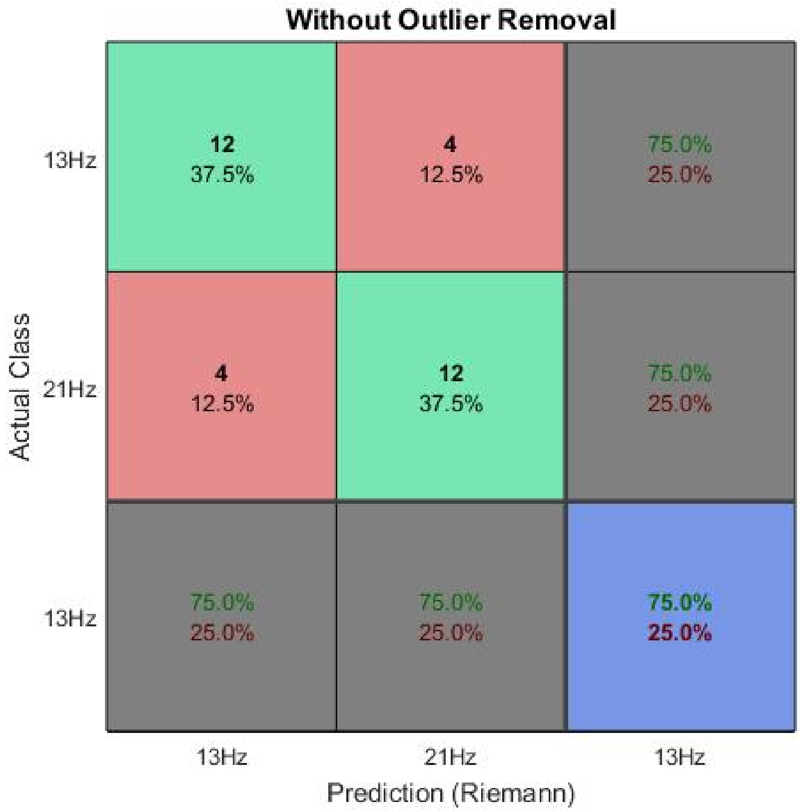

Figure 7.6: The confusion matrix for two classes

C/P	False Negative Rate	False Positive Rate	True Positive Rate	True Negative Rate
13Hz	0.2500	0.2500	0.7500	0.7500
21Hz	0.2500	0.2500	0.7500	0.7500

Table 7.1: Two Class Results

2. Using all the three classes, the confusion matrix is shown along with the tabulation of the error and hit rates.

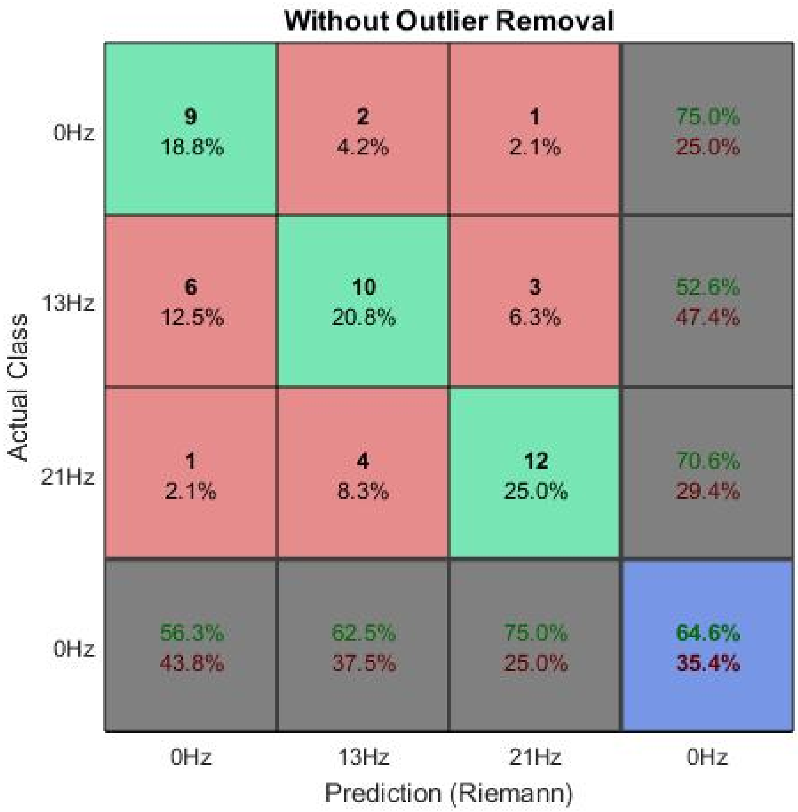

Figure 7.7: The confusion matrix for three classes

C/P	False Negative Rate	False Positive Rate	True Positive Rate	True Negative Rate
0Hz	0.1944	0.2500	0.7500	0.8056
13Hz	0.2069	0.4737	0.5263	0.7931
21Hz	0.1290	0.2941	0.7059	0.8710

Table 7.2: Three Class Results

3. Using all the four classes, the confusion matrix is shown along with the tabulation of the error and hit rates.

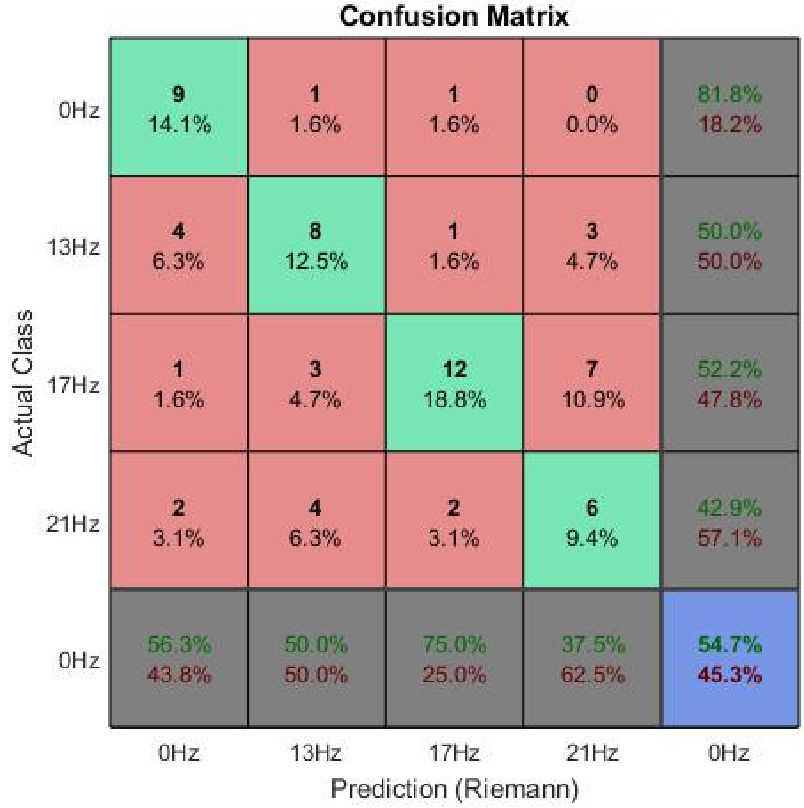

Figure 7.8: The confusion matrix for all four classes

C/P	False Negative Rate	False Positive Rate	True Positive Rate	True Negative Rate
0Hz	0.1321	0.1818	0.8182	0.8679
13Hz	0.1667	0.5000	0.5000	0.8333
17Hz	0.0976	0.4783	0.5217	0.9024
21Hz	0.2000	0.5714	0.4286	0.8000

Table 7.3: Four Class Results

As we can see here again, the accuracy goes down with the number of classes. A side-note to be considered is that the ”Without Outlier Removal” heading refers to the fact that the classification was done by also including data points that are far off in the training set and can cause errors. The outlier removal designed was not used since it tend to over-fit the data and couldn’t be generalised.

Chapter 8

Conclusion and Future Scope

8.1 Conclusion

This report is the culmination of work done in the field of brain computer interfacing for our final year project. Having explored the SSVEP paradigm, we have accomplished the task of building a statistical classifier that is more friendly and easier to implement on other larger modules. Focusing on offline analysis, we have understood the core idea of covariance of data giving us a vector metric to help cluster objects. The offline analysis was used to test the algorithm used and the results were looked at using the confusion matrix. MATLAB and its various toolboxes were used to help us process, extract and test the contrived codes, from the algorithms

The Achievements of our project were as follows

- Reviewing and understanding of different BCI algorithms used in classifying SSVEP signals.
- Understanding the Riemannian clustering and classification approach both in online and offline scenarios [3].
- Implementing an offline algorithm that has the capability to train and test as many classes as desired, using the riemannian clustering approach.
- Writing a custom MATLAB code for offline processing.
- Building a GUI for the code.

8.2 Future Scope

Since the availability of open source BCI equipment is slowly gaining mass popularity, it would only be a matter of time before it becomes mainstream. Although limited processing capabilities do exist for these boards, upgrades over time would certainly help create more engaging BCI applications. SSVEP can be seen in the following applications:

- Wheelchair Control: Some people that are paralysed from neck down would be unable to operate a joystick to move around a wheelchair. An array of LEDs flickering in front of them could be utilised as stimulus that can be assigned to different directions of movement. A person looks at one to move forward, another to go left, etc. The SSVEP classifier can then be used to control the movement of the chair.
- Writing: A robotic arm can be made to move a pen across a writing surface, controlled by the patient's SSVEP signals.
- BCI game development: Virtual reality can see the use of SSVEP to perform various tasks in games later in the future as both BCI and VR are slowly gaining traction.

References

[1] *Brain-Computer Interfaces: Foundations and Methods*; Bougrain, L., Lotte, F., Clerc, M. (2016). MIT Press

[2] *Brain-computer interfacing: an introduction*; Rao, R. P. (2013). Cambridge University Press.

[3] *Online SSVEP-based BCI using Riemannian geometry*; Emmanuel Kalunga, Sylvain Chevallier, Quentin Barthelemy, Karim Djouani, Eric Monacelli,et al.. . Neurocomputing, Elsevier, 2016, 191, pp.55-68.

[4] *Braincomputer interfaces: A gentle introduction.*;Graimann, B., Allison, B., Pfurtscheller, G. (2010). In Brain-Computer Interfaces (pp. 1-27). Springer Berlin Heidelberg

[5] *Signal Processing and Classification Approaches for Brain-computer Interface.*; Al-ani, T., Trad, D. (2010).INTECH Open Access Publisher.

[6] *Brain-computer interfacing: an introduction. Cambridge University Press.*; Rao, R. P. (2013). Cambridge University Press.

[7] *Stages for Developing Control Systems using EMG and EEG signals: A survey*; Rechy-Ramirez, E. J., & Hu, H. (2011). School of Computer Science and Electronic Engineering, University of Essex.

[8] *Optimal spatial filtering of single trial EEG during imagined hand movement.*; Ramoser, H., Muller-Gerking, J.,Pfurtscheller, G. (2000). Rehabilitation Engineering, IEEE Transactions on, 8(4), 441-446.

[9] *A survey of signal processing algorithms in braincomputer interfaces based on electrical brain signals.* ;] Bashashati, A., Fatourechi, M., Ward, R. K., Birch, G. E. (2007). . Journal of Neural engineering, 4(2), R32.

[10] *Optimizing spatial filters for robust EEG single-trial analysis.*;Blankertz, B., Tomioka, R., Lemm, S., Kawanabe, M., & Muller, K. R. (2008). Signal Processing Magazine, IEEE, 25(1), 41-56.

[11] *Regularizing common spatial patterns to improve BCI designs: unified theory and new algorithms.*; Lotte, F., & Guan, C. (2011). Biomedical Engineering, IEEE Transactions on, 58(2), 355-362.

[12] *Twofold classification of motor imagery using common spatial pattern.*;Mohanchandra, K., Saha, S., & Deshmukh, R. (2014, November). In Contemporary Computing and Informatics (IC3I), 2014 International Conference on (pp. 434-439). IEEE.

[13] *Braincomputer interfaces: A gentle introduction*;Graimann, B., Allison, B., & Pfurtscheller, G. (2010). In Brain-Computer Interfaces (pp. 1-27). Springer Berlin Heidelberg.

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

[15] *Shortlived brain state after cued motor imagery in naive subjects.*;Pfurtscheller, G., Scherer, R., MllerPutz, G. R., & Lopes da Silva, F. H. (2008). European Journal of Neuroscience, 28(7), 1419-1426.

[16] *Machine learning for real-time single-trial EEG-analysis: from braincomputer interfacing to mental state monitoring.* ;Mller, K. R., Tangermann, M., Dornhege, G., Krauledat, M., Curio, G., & Blankertz, B. (2008). Journal of neuroscience methods, 167(1), 82-90.

[17] *Review of brain-computer interfaces based on the P300 evoked potential.*; Elshout, J. A. (2009).

[18] *Robust classification of EEG signal for brain-computer interface.*;Thulasidas, M., Guan, C., & Wu, J. (2006). Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 14(1), 24-29.

[19] *An asynchronous P300 BCI with SSVEP-based control state detection.* ; Panicker, R. C., Puthusserypady, S., & Sun, Y. (2011). Biomedical Engineering, IEEE Transactions on, 58(6), 1781-1788.

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

[21] *A survey of stimulation methods used in SSVEP-based BCIs.* ; Zhu, D., Bieger, J., Molina, G. G., & Aarts, R. M. (2010). Computational intelligence and neuroscience, 2010, 1.

[22] *Discriminative Feature Extraction via Multivariate Linear Regression for SSVEP-Based BCI.*; Wang, Haiqiang, et al. IEEE Transactions on Neural Systems and Rehabilitation Engineering 24.5 (2016): 532-541.

[23] *Comparison of PSDA and CCA detection methods in a SSVEP-based BCI-system*;] Hakvoort, Gido, Boris Reuderink, and Michel Obbink. (2011).

[24] *A review of classification algorithms for EEG-based braincomputer interfaces.*; Lotte, Fabien, et al. Journal of neural engineering 4.2 (2007): R1.

[25] *A CWT-based SSVEP classification method for brain-computer interface system.*; Zhang, Zimu, Xiuquan Li, and Zhidong Deng. Intelligent Control and Information Processing (ICICIP), 2010 International Conference on. IEEE, 2010.

[26] *Frequency recognition based on canonical correlation analysis for SSVEP-based BCIs.*; Lin, Zhonglin, et al. IEEE Transactions on Biomedical Engineering 53.12 (2006): 2610-2614.

[27] *Multiple channel detection of steady-state visual evoked potentials for brain-computer interfaces.*; Friman, Ola, Ivan Volosyak, and Axel Graser. IEEE Transactions on Biomedical Engineering 54.4 (2007): 742-750.

[28] *Linear LSI Data and Applications Manual* Signetics. 1985. Archived from the original on April 5, 2016.

Appendix A

Project Planning

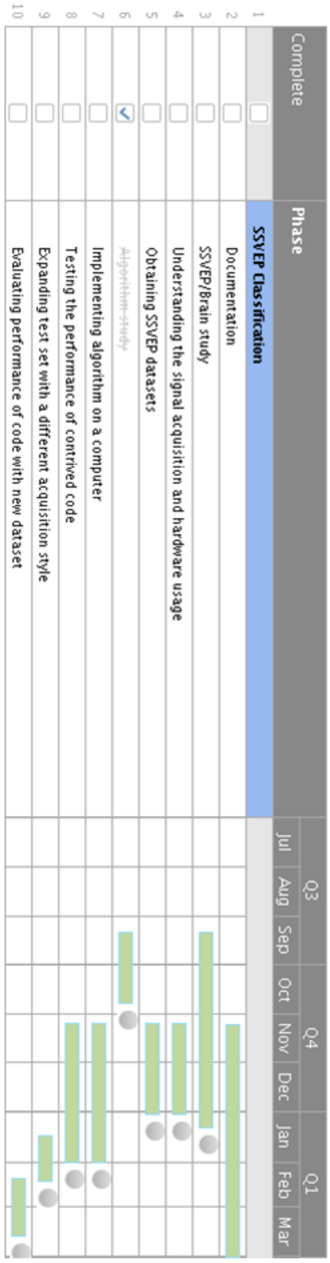

Figure A.1: The Gantt Chart

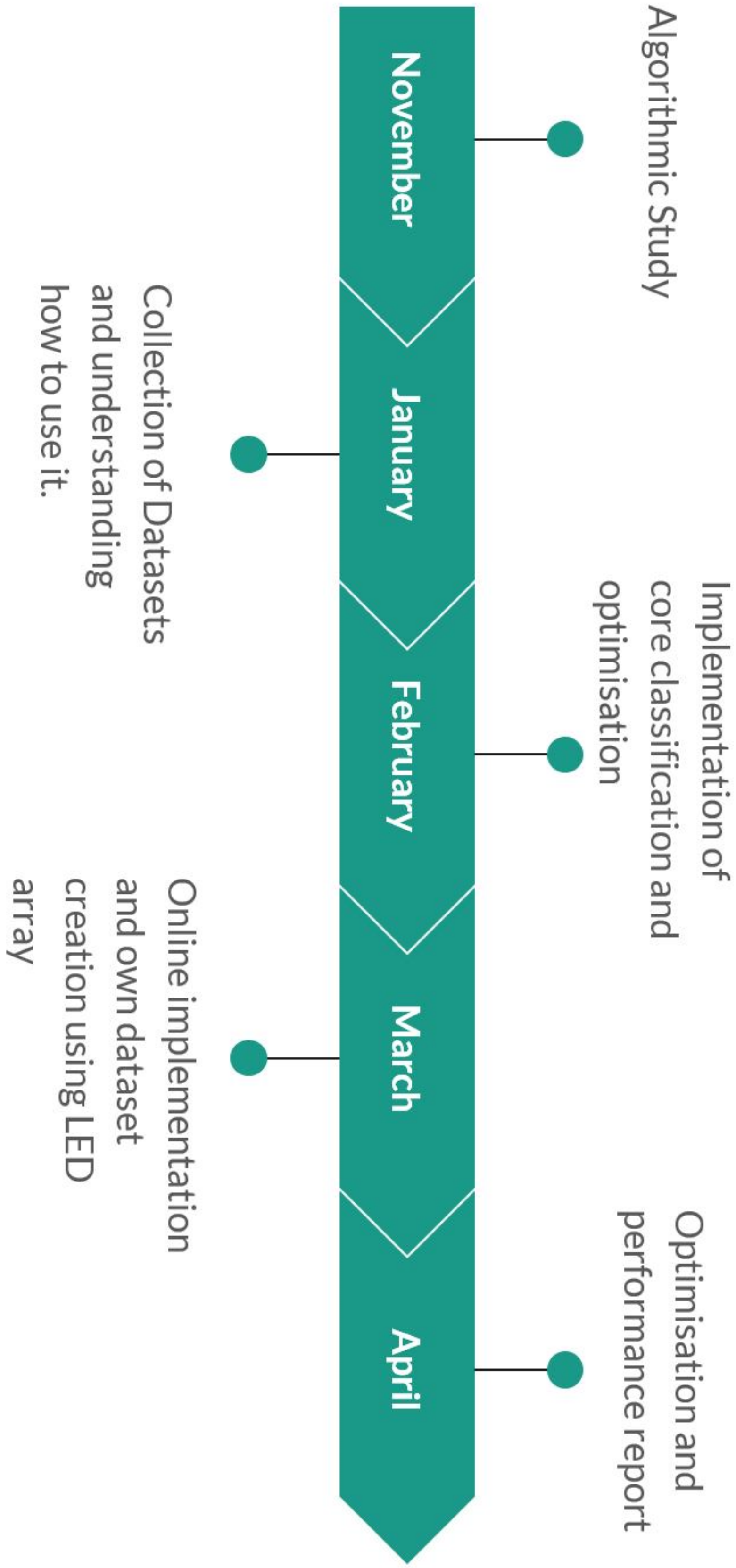

Figure A.2: Overall Plan

Appendix B

Technical Specifications

OpenBCI 32-bit Board[27]	
• 8 differential, high gain, low noise input channels	• Programmable gain: 1, 2, 4, 6, 8, 12, 24
• Compatible with active and passive electrodes	• 24-bit channel data resolution
• Texas Instruments ADS1299 ADC Analog Front End	• ±2.5V analog operating voltage
• PIC32MX250F128B microcontroller w/chipKIT^{TM}bootloader (50M Hz)	• 3.3–6V input voltage
• RFduino^{TM} Low Power Bluetooth^{TM} sradio	• LIS3DH accelerometer
• Micro SD card slot	• 5 GPIO pins, 3 of which can be Analog

OpenBCI Dongle [27]
• RFD22301 radio module from RFdigital^{TM}
• FT231X USB-to- serial converter from FTDI
• Can upload code to the OpenBCI board or the dongle
• Fully broken out and pin-compatible w/ RFduino form factor

The Github repository hosting the entire project: `https://github.com/SmellingSalt/Group31`