<u>İSTANBUL TECHNICAL UNIVERSITY</u>
<u>ELECTRICAL - ELECTRONICS ENGINEERING FACULTY</u>

RAW EEG DATA CLASSFICATION AND APPLICATIONS USING SVM

BSc Thesis by

Onur Varol

040060307

Department: Electronics and Communication Engineering

Programme: Electronics Engineering

Supervisor: Assoc. Prof. Müştak Erhan Yalçın

MAY 2010

FOREWORD

I would like to thank my advisors, Assoc. Prof. Müştak Erhan Yalçın for their inspiring guidance and support and the opportunities they provided me in this thesis. It was a great pleasure for me to carry on this thesis under his supervision.

In addition, I would like to thank Tuba Ayhan for her help in my thesis, which would not be carried out without her. She helps for managing the problems that were encountered in this study, and she was always patient and understanding for me.

I also appreciated for the student branch OTOKON. It is always like a family, and I learned the most important lessons of my life; friendship, cooperation and most of my technical knowledge.

Finally, I especially want to thank my family, my father Cüneyt VAROL, my mother Berrin VAROL, and my sister Beril VAROL, for their continuous support, sacrifice and understanding during my educational life.

May 2010 Onur VAROL

INDEX

ABBREVIATIONS

BCI	: Brain Computer Interface
EEG	: Electroencephalography
SVM	: Support Vector Machines
EMG	: Electromyography
EKG	: Electrocardiography
ADC	: Analog-Digital Converter
ICA	: Independent Component Analysis
UI	: User Interface
FFT	: Fast Fourier Transform

LIST of TABLES

LIST OF FIGURES

SUMMARY

BCI systems are become more significant in recent years. As a result of developments in the field of biomedical science, signals are acquired and processed more successfully. Studies in biomedical signal processing have gained importance in recent years. The use of BCI systems and computer interfaces, such as robot control and other human-machine system are achieved easily.

In this thesis, various applications are developed for testing and performing demo applications by using EEG signal processing. Performances of these applications are controlled with various tests and effects on the performance criteria were examined.

In the first part of the thesis work focused on EEG signals and biological origins. Some basic concepts are described in this part. BCI systems are mentioned in the next section, and essential factors of the BCI systems are told. In the third part of the thesis, information about Support Vector Machines is given which is used for classification of the EEG patterns.

After creating the theoretical background of the thesis projects, applications written for this project are demonstrated. EEG test tools and other applications such as robot control and human-machine interfaces are explained.

Finally performances of the EEG classification under different conditions are examined. Classification performance under different environmental and physiological conditions is examined while users come across with different tasks. Environment change of the test and training on the classification performance are measured.

ÖZET

BCI sistemler günümüzde önemli bir noktaya gelmektedir. Biyomedikal alanında gerçekleşen gelişmelerin sonucunda işaretlerin daha sağlıklı alınması ve işlenmesi mümkün olmaktadır. Bu alanda yapılan çalışmalarda son yıllarda önem kazanmaktadır. BCI sistemlerin kullanılması ile robot kontrolü ve bilgisayar arayüzleri gibi uygulamaların geliştirilmesi mümkün olmaktadır.

Bu tez içerisinde EEG işaretlerinin alınması ve işlenmesi ile çeşitli uygulamalar geliştirilmiştir. Bu uygulamaların başarımı çeşitli testler ile kontrol edilip performansı etkileyen kriterler incelenmiştir.

Tez çalışmasının birinci bölümünde EEG işaretleri ve biyolojik kökenleri üzerinde durulmuştur. Burada temel bir takım kavramlar açıklanmıştır. Sonraki bölümde BCI sistemlerden bahsedilmiştir ve bir BCI sistemin sahip olması gereken temel öğelere yer verilmiştir. Tezin üçüncü bölümüne gelindiğinde ise sınıflandırma araçlarından bu proje içerisinde kullanılan Destek Vektör Makineleri hakkında bilgi verilmiştir. Kullanılan öznitelikler ve sınıflandırmada izlenen yol hakkında bilgi verilmiştir.

Teorik altyapı oluşturulduktan sonra tezin dördüncü bölümüne gelindiğinde projede yazılmış olan uygulamardan bahsedilmektedir. EEG işleme konusunda ihtiyaç duyulabilecek temel uygulamalardan robot kontrolüne kadar olan geniş bir çerçevede yapılan uygulamalar açıklanmıştır.

Son bölüm içerisinde ise EEG sınıflandırma üzerinde elde edilen sonuçların değerlendirilmesi ve kullanıcılar üzerinde yapılan deneylerin açıklanmasına yer verilmiştir. Burada kullanıcılar farklı ortam şartları altında robot kontrolü için çalışmakta ancak ortam değişikliklerine olan tepkileri gözlenmekte ve bir BCI sistemin başarımının nelerden etkilendiği anlatılmaya çalışılmıştır.

1. INTRODUCTION

This thesis describes the principles of the Brain Computer Interface (BCI). In this graduate project several programs are written to control different interfaces using Electroencephalography (EEG) data. The brain activity monitoring is a main part of the programs to perform different tasks such as controlling a mobile robot or changing mouse position or triggering events on these tools.

EEG data is being recorded while users are concentrating on given tasks. Activity on different regions on brain is measured. Using these measurements EEG data is classified for different cognitive actions.

Raw EEG data is acquired from a product called Emotiv EPOC. It is developed for game programmers to interact with player on games more realistically. It has 14 sensors and also 2 axis gyros for 2 dimension control of the head pose.

Applications written on different environments aim to gain maximum benefits in case of productivity and creativity. Programs are implemented in C# and MATLAB environments for different purposes. MATLAB is a useful tool for testing algorithms and working with high dimension data. Furthermore, C# is beneficial to accessing Raw EEG data from the headset via Emotiv SDK. Online applications are needed to acquire live data, which can be gathered with SDK, so applications are mostly coded in C# language.

In the first part basics about EEG and Signal Processing will be explained. In Section 2, BCI system will be presented and Emotiv EPOC will be presented. After that feature selection and classifications will be explained and Support Vector Machines (SVM) will be presented in Section 3. In Section 4 and 5, experiments and applications will be given.

2. BRAIN STRUCTURES AND CENTERS

Brain is the most complex organ of all creatures. All physical and mental tasks are done in the brain. This section gives an overview of the structure and the functions of the brain.

2.1 Structure of the Brain

Brain is the main part of the central nervous system, which consist of large brain, brainstem and spinal cord as shown in Figure 2.1. Brainstem is the part that connects large brain and spinal cord. Anatomically, brain can be divided into parts such as hind brain, mid brain and fore brain. Hind brain consists of the myelencephalon which means the spinal cord, above the myelencephalon Cerebellum and fourth ventricle locates. The second part, mid brain consists of the mesencephalon which consists of the tectum and tegmentum and cerebral aqueduct. Last part of the brain is fore brain, which consists of two parts called diencephalon and telencephalon [1].

After a short brief, functional perspective of these parts which is more important for this work will be given. It can be divided in three parts in terms of functionality. First part is called large brain, which is also known as forebrain or cerebrum. This part controls higher mental activity such as analytical thinking and language and contains diencephalon and telencephalon. Second part is called the brainstem which is responsible to visual and auditional functions. The brain stem is located in the mesencephalon. Third part is the cerebellum, and it handles the motor control and movement of the limbs and body.

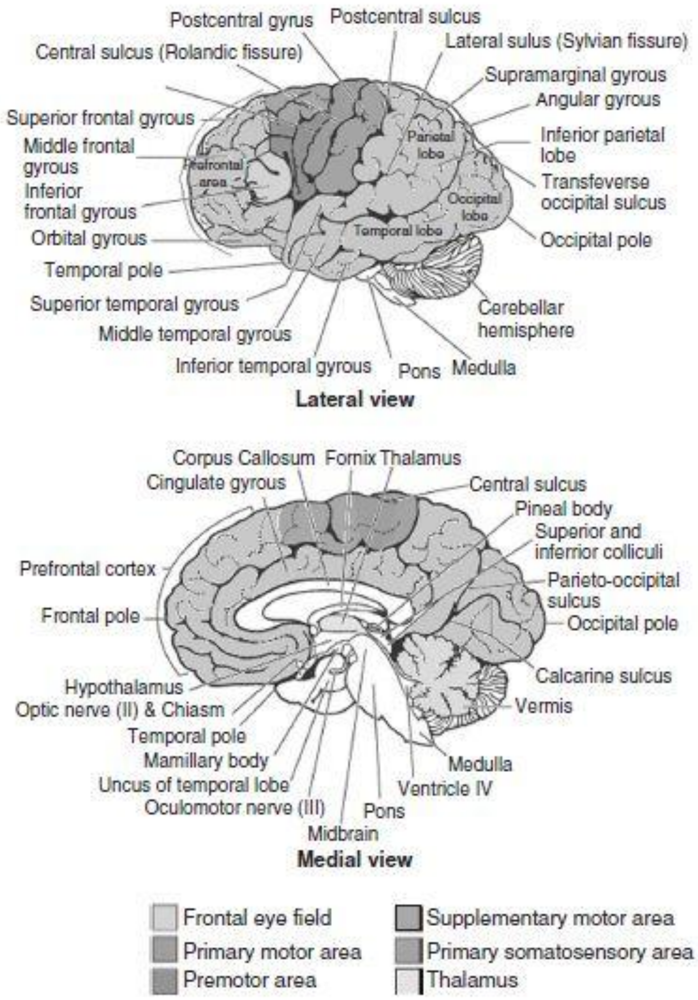

Figure 2.1: Major parts of the brain [2]

2.1.1 The Cortex

It is the dominant part of the cerebrum and it consists of 1010 to 1012 neurons arranged on different layers. The cortex is 2-3mm thin but its total area is large compare to the area of the skull because different shapes of the cortex surface such as fissures, which are folding that, is divided into two hemispheres and two frontal and temporal lobes.

The left and right hemispheres of the brain are the major parts of the higher cognitive actions. The left part of the brain is related to language and verbal materials and also positive emotions, whereas the right hemisphere is related to visio-spatial functions and negative emotions. These two hemispheres are connected and they communicate via corpus collosum, which contains millions of nerve fibers run across two different hemispheres.

Soma motor cortex is the center of the bodily functions located on the brain. The key rule in localization of the centers of the organs on the motor cortex is accuracy on the area is related. Area on the brain such as lips and tongue is covering larger area than legs.

Somatosensory cortex is also laid similar to the somatomotor cortex. Representations of these areas are shown as a homunculus in Figure 2.2.

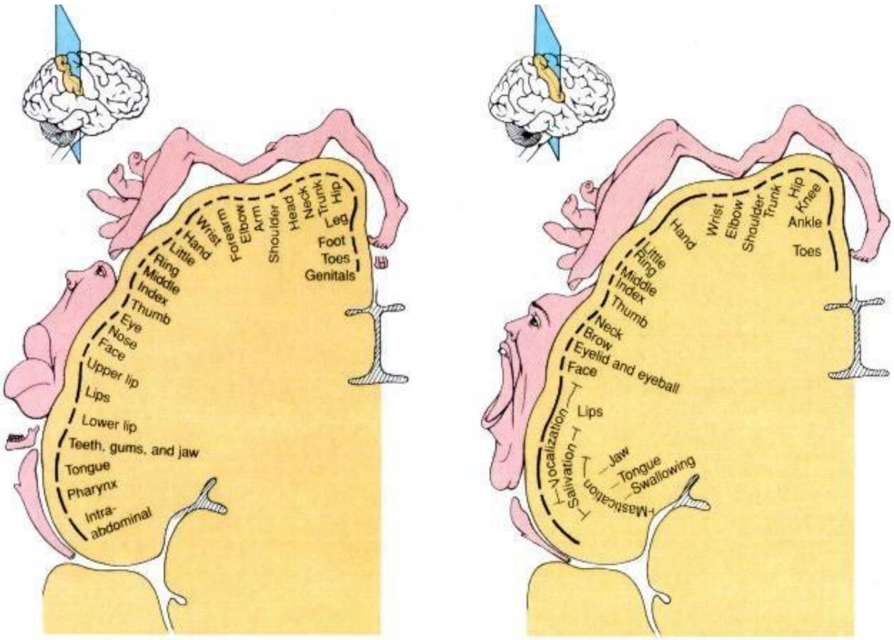

Figure 2.2: Somatomotor and Somatosensory cortex [3]

2.1.2 The Lobes of the Cortex

Human brain can be explained in four different lobes. These lobes have different functionalities. These lobes are called frontal, temporal, parietal and occipital.

The frontal lobes control complex cognitive action, language programs and execution of the motor patterns. Damage of this area may cause Parkinson, Alzheimer or Schizophrenia diseases.

Temporal lobe is used for visual memory, processing of the language and other audio related functions. Hippocampus which is related to memory is also located in the temporal lobe.

Brain takes care of senses and other outputs in the parietal area. Parietal area is connected all sensing and processing the center of brain. It is an ancient area for all mammals.

The occipital lobe is the center of all visual related tasks. This is the place where the visual information from eyes transforms directly. It is also playing a role on recognition of the object and processing of them.

2.1.3 Basal Ganglia

Basal ganglia are the place for controlling body movements by integrating sensory and motor information from other areas of the brain. Diseases such as Parkinson are also originated from that area.

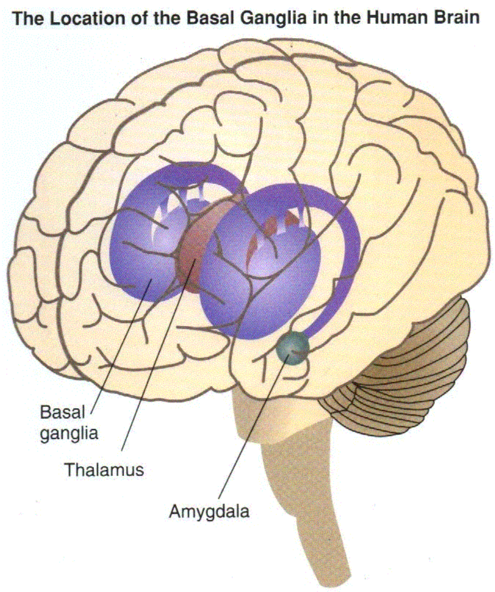

Figure 2.3: Basal Ganglia [3]

2.1.4 The Hypothalamus and Thalamus

The hypothalamus is responsible for balancing body in terms of temperature, thirst, hunger, etc. by controlling hormones. The balance of the body is called homeostasis.

Thalamus plays a role as a relay station of all sensory data. Thalamus integrates and passes on somatosensory and somatomotor information [1].

2.2 SIGNAL TYPES (BRAIN RHYTHMS)

EEG signal is a complex signals, which are described in terms of rhythmic and transient which shown in Figure 2.4. The rhythmic activity is divided into bands by frequency. EEG signal of a healthy adult may vary in amplitude and frequency when it is recorded in different states such as in sleep or awake. Moreover, the characteristic of wave changes with age.

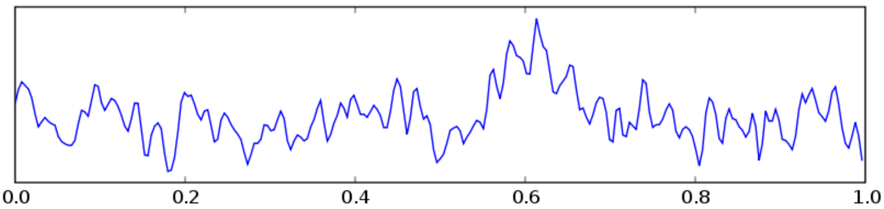

Figure 2.4: Mixed Channel EEG Data [4]

There are five major brain waves forms that can be distinguished by their frequency ranges. These fre uency bands from low to high fre uencies are called alpha , theta , beta , delta and gamma .

2.2.1 Delta Waves

Delta waves lie within the range of 0.5-4 Hz. This wave type is associated with deep sleep and may be present in awake state. These signals are very likely to be confused with the large muscle artifact signals. Delta rhythm is decreased with age, and it is not normal to detect delta waves in healthy people while they are awake.

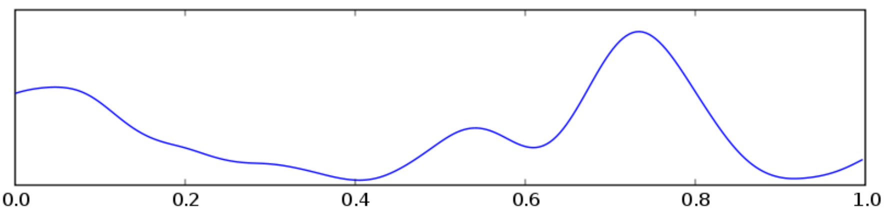

Figure 2.5: Delta waves [4]

2.2.2 Theta Waves

Theta waves lie within the range of 4-7.5 Hz. Theta waves associated with access to unconscious materials, creative thinking and deep meditation. Origin of this wave name comes from Thalamus. Furthermore, there is a link between emotions such as disappointment and frustration.

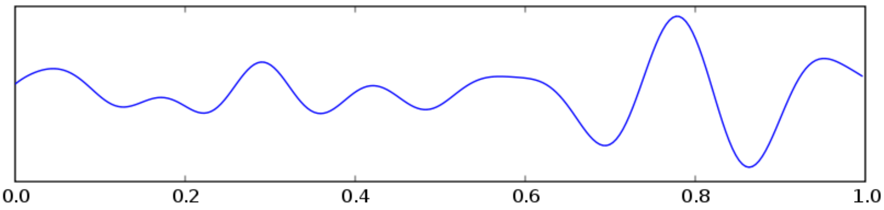

Figure 2.6: Theta waves [4]

2.2.3 Alpha Waves

Alpha waves appear on a posterior part of the head and usually found over an occipital area of the brain. Alpha wave frequency lies within the range of 8-13 Hz. These waves have comparatively higher amplitude to other wave types. It is best seen when eyes are closed and patient is in mentally relaxed. In some cases, alpha waves interfere with -rhythm. This wave type is useful to trace mental effort because of its higher amplitude.

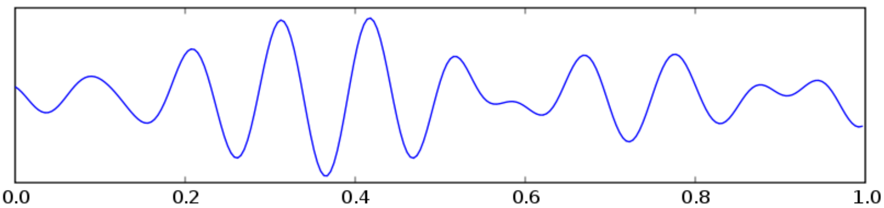

Figure 2.7: Alpha waves [4]

2.2.4 Beta Waves

Beta waves have a large range of the frequency spectrum. It is between 13-50 Hz practically. Beta rhythm can be measured from frontal and central regions of brain. The central beta rhythm is related to Rolandic -rhythm and can be blocked by motor activity and operation of planning to move.

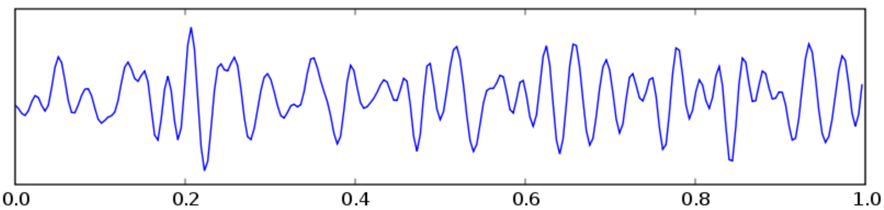

Figure 2.8: Beta waves [4]

2.2.5 Gama Waves

Gama waves are sometimes called fast beta rhythm and lie within the range of 30Hz and higher frequencies. These waves have very low amplitude so it is very rare to observe them. It is used to detect high cognitive activities and gives some clues about mental diseases.

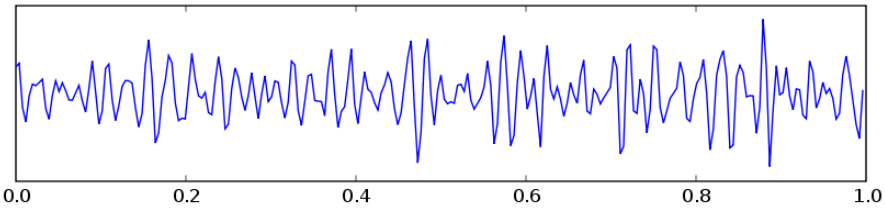

Figure 2.9: Gama waves [4]

2.2.6 Mu Rhythm

Mu Rhythm which is also called Rolantic -rhythm is related to posterior alpha rhythm in frequency and amplitude, but its importance is different from alpha waves. stands for motor and rhythm is strongly connected with motor activities. This rhythm is very anti-symmetric and easy to detect. Most of the times face muscle activities and eye movements can be seen as an artifact of EEG signal. This rhythm is mostly seen 8-11 Hz frequency and is easily detected using C3 and C4 electrodes of the standard 10-20 system.

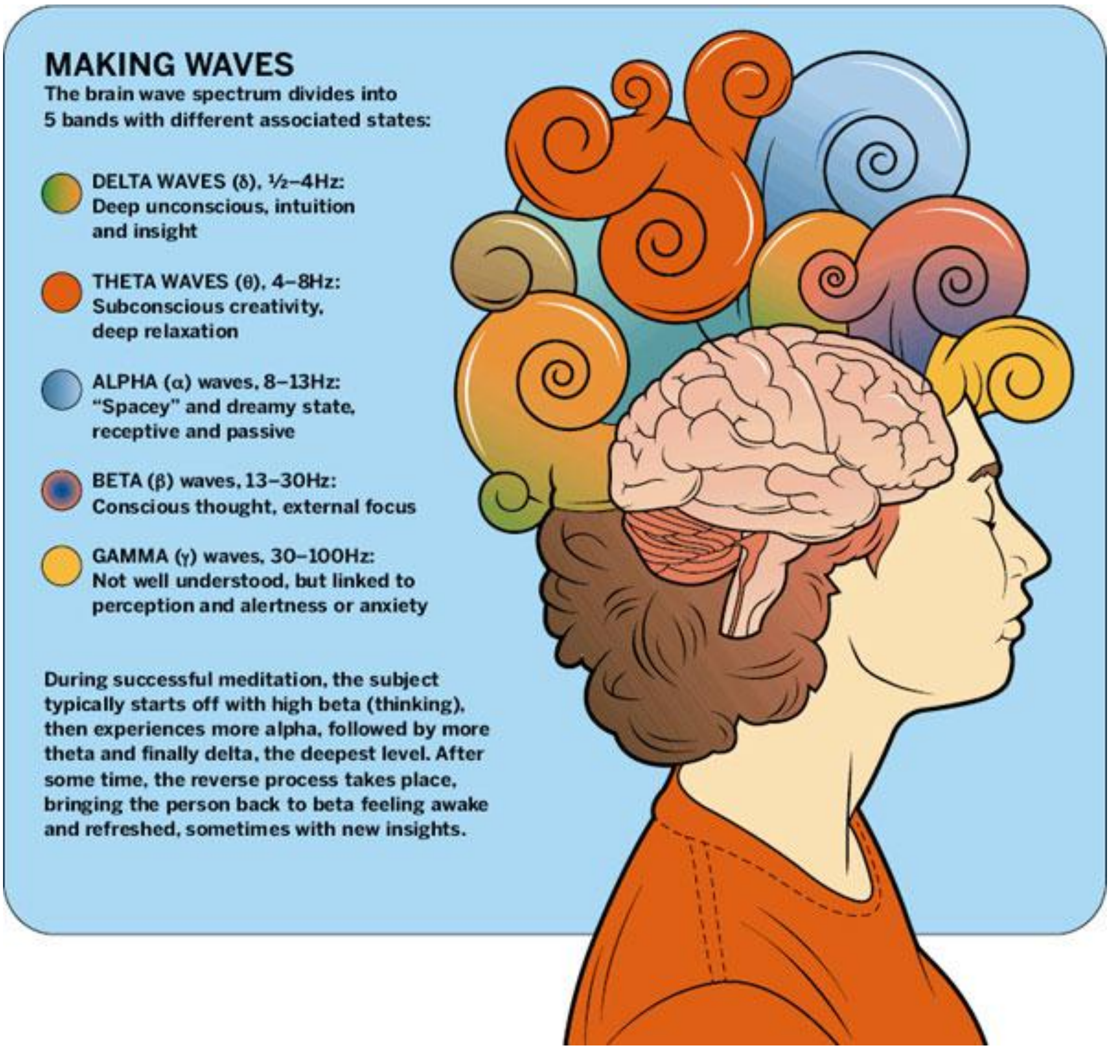

Figure 2.10: EEG Rhythms [5]

2.3 ACQUISITION OF EEG SIGNALS

In biomedical sciences several type of electrical signals are measured. EEG is one of the biological signal, which is called Electroencephalography in means that signal measuring from sculp. The principle of EEG measurement is to calculate the potential difference of two electrodes. In this system, reference electrodes are used two determine background electric field of the skull and they are placed on the ear lobes or mastoids. The placement of the reference electrodes is important because if they are too close to the brain they are affected from the brain activity and also if it is on another part of the body, it is likely to be affected from muscle s electrical activity, especially from the hearth. [2]

Recent EEG devices consist of the number of accurate electrodes, amplifiers for each electrode followed by filters. Analog EEG signals are needed to be converted to digital data and sampling frequency is limited to catch up with the speed of the conversation on ADC.

Filtering system is needed to remove the artifact. There are low pass and high pass filters to remove unwanted frequencies, removing EMG (Electromyography), EKG (Electrocardiography) signals and 60 Hz coming from the ground loop [6].

Moreover, resolution of the data is important in terms of transferring and processing. So sampling frequency, sampling rate and sensor numbers are important parameters.

The EEG recording electrodes also differ from each other in terms of method that they use to acquire data. Some types of electrodes that are often used in EEG recording systems are:

- Disposable (gel-less and pre-gelled types),
- Reusable disk electrode,
- Headbands and electrode caps,
- Saline-based electrodes,
- Needle electrodes.

2.3.1 Artifacts

Artifacts are not related to brain activity but affects the signal measured making it unusable and difficult to interpret. There are several categories of artifacts.

Most important artifact is caused by the impedance of the system. Second major artifact is 50 and 60 Hz artifact, which caused by ground loop.

Biological signals such as EMG and EKG are also affecting the EEG signals. These biological artifacts can be useful for example; EMG artifact comes from eye movements, so it may give an insight on status of mental status such as drowsiness, awareness and asleep.

2.3.2 Sensors

EEG sensors are not placed to the head randomly the standard that is used to determine places of the sensors is called 10-20 System. In this system position are determined as follows: One of the reference points, the nasion, which is delve at the top of the nose, lie on the same level with the eyes. Inion is the bony lump at the base of the skull on the midline at the back of the head as shown in Figure 2.11. From these points, the skull perimeters are measured in the transverse and median planes. Electrode locations are determined by dividing these perimeters into 10%, 20% intervals.

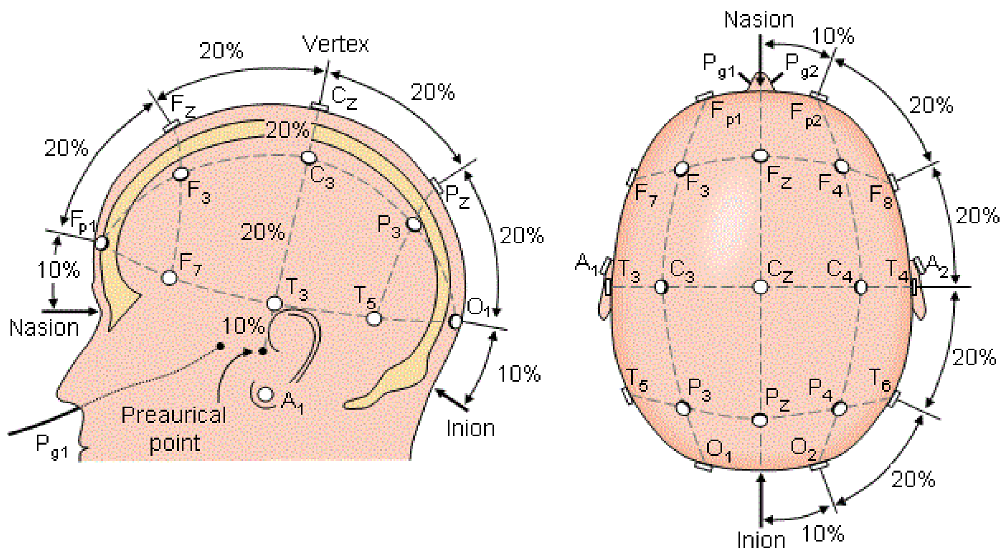

Figure 2.11: Electrode placement [7]

In this system electrode numbers can vary but the rule is simple. Enumerating each sensor is based on a simple rule. Letters correspond to different places of the brain as shown in Figure 2.12. A stands for Ear Lobe, C for Central, Pg for nasopharyngeal, P for parietal, F for frontal, Fp for frontal polar and O for occipital area.

This placement rule is standardized by the American Electroencephalographic Society [2].

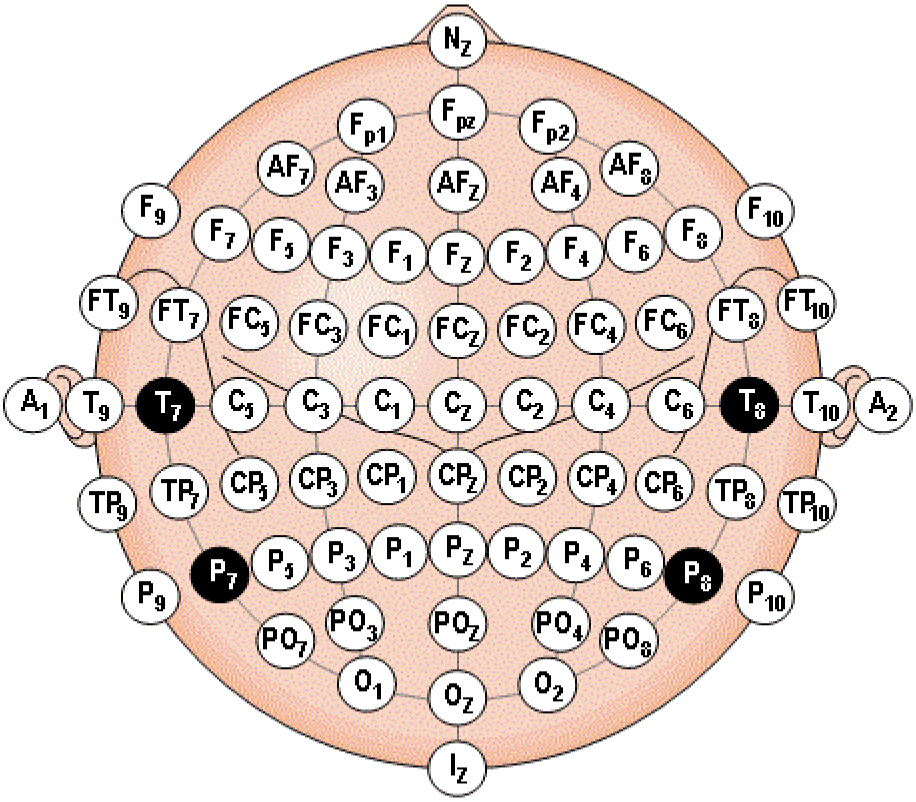

Figure 2.12: 10-20 System [7]

3. BCI

Brain Computer Interface (BCI) is a communication system that recognizes the commands from processing EEG signals and reacts according to this pattern. For this purpose a classifier is trained using data of the user.

In BCI system no peripheral nerves or muscles are required for communication, only brain activity is required. Therefore controlling computer or wheelchair for a disabled person can be possible with using brain activities.

BCI systems are built in several steps in order to operate successfully. The first step is the preparation of detection equipment. The second is building an algorithm for detecting the stimulus in the brain. The third one is classification or decision of the action. Finally actual control of the system is achieved. These are listed and explained below.

3.1.1 Preparation of the System

BCI systems consist of electrodes that acquire signals from the skull. These sensors are placed in a pattern which suits the 10-20 system. Furthermore, device should be calibrated to adjust electrode impedances and amplification of the signals.

3.1.2 Detection

Detection is mostly used in P300 type evoked potential applications. Detection of the stimulus is important to understand the function of the focused responses then these responses can be classified more accurately. Activity patterns usually stimulate the upper cortex of the brain where a good quality of the EEG signals can be measured.

3.1.3 Classification

EEG signals are complex signals because of its nature. That makes working with EEG data harder. Classifying the data of EEG signal requires some techniques to determine differences between signals. Classification can be done on different domains such as frequency or time. Methods using classification and feature extraction differ from problems characteristic. Classification is implemented on the windowed data to extract the eigen values of signals for selected feature. After determination of the eigenvalues and preparation of the data set are completed, the packet of data and eigenvalues is sent to the classifier. Details of the classification are presented in Section 3.5. Several classifiers are used for classification of EEG data, from Neural Networks to Statistical classifiers. Classifiers are run for each data packet recorded for certain window length. Training is the key factor of the classification, training data must be accurate and sufficient to teach the patterns to the system.

3.1.4 Control

After training and separating the data stream to different classes control is the next step for a BCI system. Actuators vary for different purposes such as controlling wheelchair, playing games, operating a robot, etc. Matching the EEG patterns and actuator task is not complex after making successful classification.

3.2 EMOTIV EPOC

Emotiv EPOC is an EEG Headset which supplies 14 channels EEG data and 2 gyros for 2 dimensional controls. Emotiv s sensors are saline based so it is easy to use for many applications. Emotiv is developed for game programmers to interact with the user with their avatar in the game. Its features are adequate for a useful BCI in case of resolution and bandwidth.

It is also advantageous that being wireless and having long life batteries. Emotiv EPOC sends EEG data encrypted via Wi-Fi. Academic version of the Emotiv can access the raw data which is decrypted using Emotiv Control Panel. Connecting the TCP port of the Control Panel, decrypted data is received and used in the applications.

	SDK HEADSET
Number of channels	14 (plus CMS/DRL references, P3/P4 locations)
Channel names (International 10-20 locations)	AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4
Sampling method	Sequential sampling. Single ADC
Sampling rate	128 SPS (2048 Hz internal)
Resolution	16 bits (14 bits effective) 1 LSB = 1.95μV
Bandwidth	0.2 - 45Hz, digital notch filters at 50Hz and 60Hz
Filtering	Built in digital 5th order Sinc filter
Dynamic range (input referred)	256mVpp
Coupling mode	AC coupled
Connectivity	Proprietary wireless, 2.4GHz band
Power	LiPoly
Battery life (typical)	12 hours
Impedance Measurement	Contact quality using patented system

Figure 3.1: Emotiv EPOC Specifications [8]

Emotiv EPOC s sensor layout is planned carefully to gain optimum benefits for human machine interaction. Sensors are mostly located in the frontal cortex, so it is useful to detect upper face gestures and also determine Alpha waves while concentrating on the task. Emotiv s sensor layout is shown in Figure 3.1.

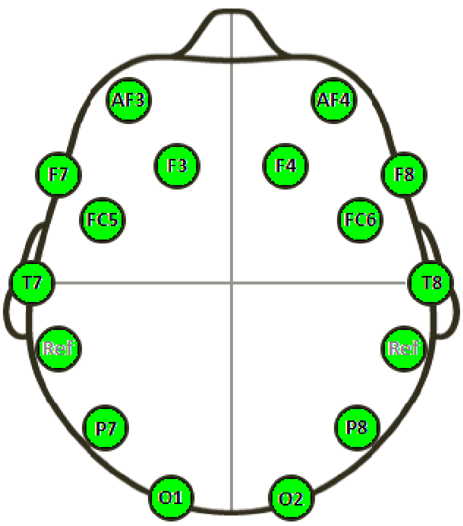

Figure 3.2: Emotiv's Sensor Layout

3.3 DATA ACQUISITION

In BCI application, working with raw data is essential. Proper data is needed to apply some filtering process and converting raw data into the digital form. Digitalized data is used in applications after it is classified.

Emotiv System is applying some main filters on hardware. The measurements are filtered with high pass and low pass filters on the device. C-R based high pass filter at 0.16 Hz cutoff frequency and low pass filter at 83 Hz are applied before amplifying the data. Raw ADC (Analog-Digital Converter) collection rate is 2048 sec/channel. This data is filtered with a 5th order sinc filter to notch 50 Hz and 60Hz. This processed data is down sampled to 128 sec/channel to eliminate the main harmonics. Overall effective bandwidth is 0.16-43 Hz [8].

Applications in this thesis are used both offline and online data acquired from Emotiv Headset. Offline data is recorded in edf file format and used in MATLAB to work with. Online data is used with the application that is written for real time control.

Offline data is recorded easily with Emotiv s Academic version product in edf file format. This format is commonly used in many other EEG processing applications such as EEGLAB and BCI2000. Edf files are imported by EEGLAB toolbox of MATLAB and converted to mat files which MATLAB can process for reusability.

3.3.1 EEGLAB EDF file converter

EEGLAB is a rich toolbox for MATLAB to process EEG data. It can apply ICA (Independent Component Analysis), import channel locations, and do artifact rejection and time-frequency analysis [9].

While recording offline data Emotiv s product is used to monitor EEG data at the same time as shown Figure 3.2.

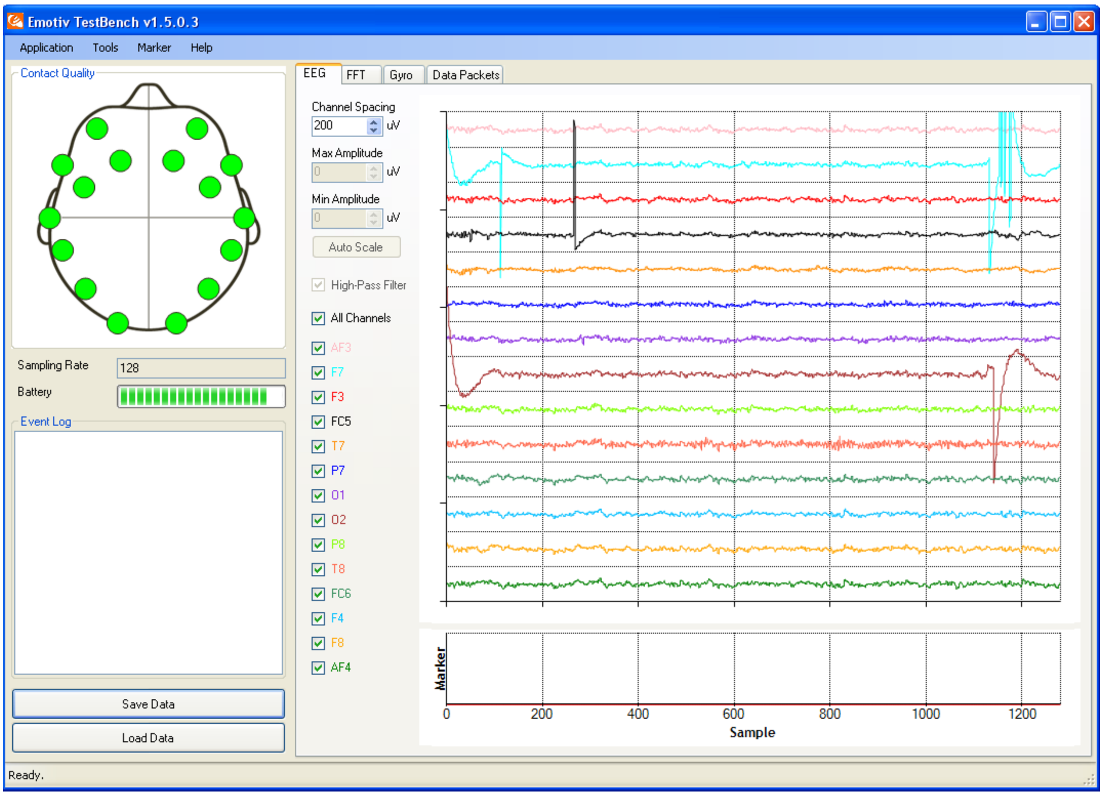

Figure 3.3: Emotiv Testbench

Recorded data is saved in edf format, so it can be converted other formats such as csv or text files. Recorded data is processed in MATLAB as shown in Figure 3.3.

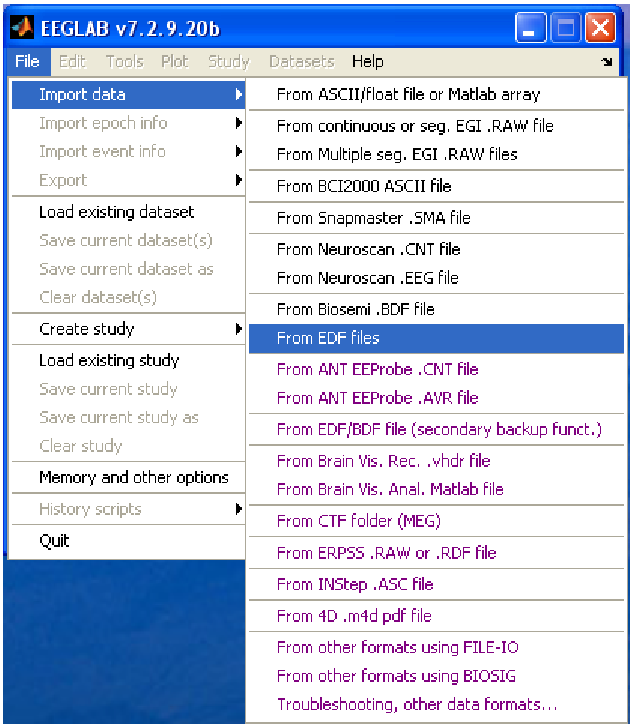

Figure 3.4: EEGLAB EDF File Converter

Using EEGLAB, raw data file is imported to workspace of MATLAB, so data can be saved or processed in MATLAB environment.

3.3.2 C# Program for Real Time Acquisition

Raw EEG data can be accessed using Emotiv SDK [9], so applications are recorded data to save data in text file also in another mode of applications EEG data can be used to control other systems. Emotiv Dll is decrypted the raw data gathered from the headset. Headset is connecting to Control Panel, so raw data is sending to the application via using TCP socket.

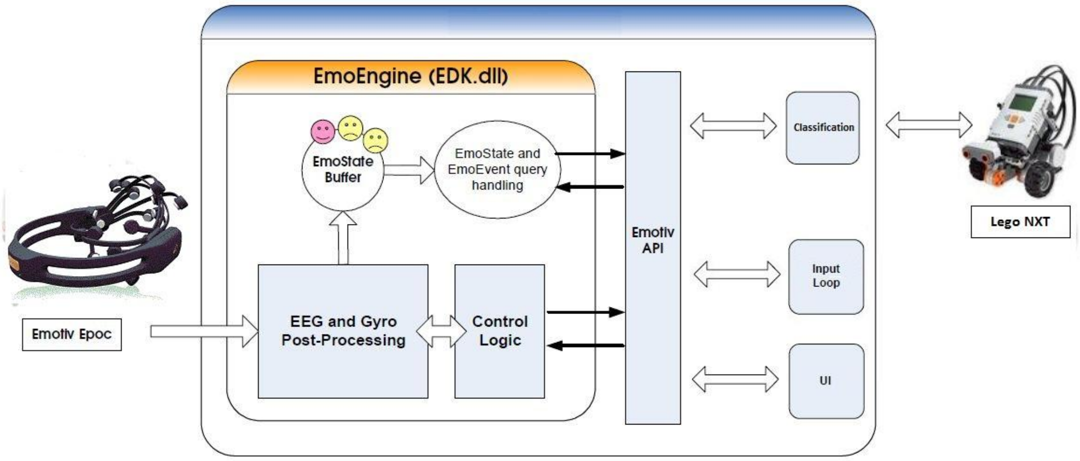

Figure 3.5: Program flow of EEG Control

Emotiv DLL behaves like in Figure 3.5. EEG data called by the application and decrypted data are received via Emotiv SDK. Decrypted data are used on different modules such as refreshing the UI (User Interface) of the program and classification module. After sending data to the classification module control unit determines the action of the robot, mouse coordinates or other systems by using the result of the classification module.

3.4 SVM

Support Vector Machine (SVM) is a supervised learning method which is used for classification and regression. SVM conceptually implements the following idea: input vectors are non-linearly mapped a very high dimension feature space. In this feature space, a linear decision surface is constructed. Special properties of the decision surface ensure the high generalization ability of the learning machine [10].

In simple word, it can be explained as given a set of training examples, each sample is marked with the category that belongs to. SVM training algorithm builds a model that predicts whether a new example falls into one category or the other. Intuitively, SVM model is a representation of the examples as a point in the feature space. Examples are used for the find out the gap between data to split them to the different region for different classes. Test data are then mapped into this space and predicted to belong to a category based on which side of the gap they fall on [11].

Technical problem of implementing the algorithm is how computationally to treat such high-dimensional spaces: to construct polynomial of degree 4 or 5 in a 200 dimensional space it may be necessary to construct hyper planes in a billion dimensional feature spaces. The conceptual part of this problem was solved in 1965 [10] for the case of optimal hyper planes for separable classes. An optimal hyper plane is here defined as the linear decision function with maximal margin between the vectors of the two classes. It was observed that to construct such optimal hyper planes one only has to take into account a small amount of the training data, the so called support vectors, which determine this margin [10].

3.4.1 Linearly Separable Classification

Theory behind the SVM explained on linearly separable data and then improved to non-linear classification. Firstly, L training point where each input x_i has D attributes and is in one of the classes.

$$\{x_i, y_i\} \qquad i = 1 \ldots L, y_i \in \{-1,1\}, x \in R^D \tag{3.1}$$

This training data set is assumed linearly separable, that mean a straight line on a graph split points in two different class.

This hyper plane can be described by$w.x + b = 0$. In this linear system w is stands for normal vector to the hyper plane and $\frac{b}{\|w\|}$ is the perpendicular distance from the hyper plane to the origin.

Support vectors are the examples closest to the separating hyper plane and aim of the SVM is to orientate this hyper plane in such a way as to be as far as possible from the closest members of the classes as shown in Figure 3.6.

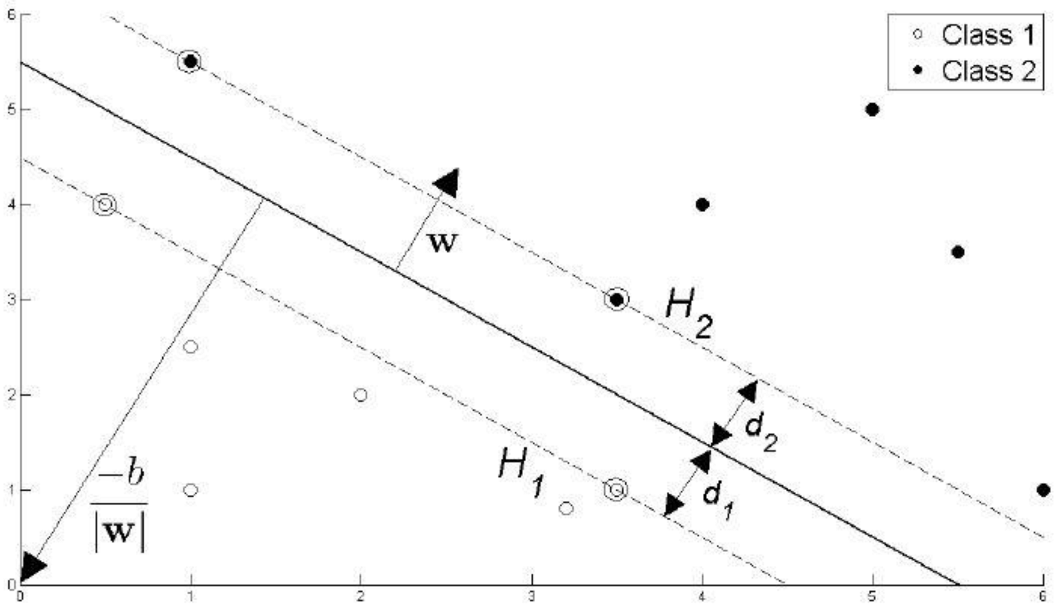

Figure 3.6: Hyperplane of two Linearly Separable Class [12]

It can be seen on the Figure 3.6 that points are separated by the line crossing between them. Each point provides one of the equations below so it can be determined which class it belongs to.

$$x_i.w + b \geq +1 \quad y_i = +1 \tag{3.2}$$

$$x_i.w + b \leq -1 \quad y_i = -1 \tag{3.3}$$

Lines passing between data limit the interval of classes. Distance between lines H1 and H2 would be bigger for better classification. This length SVM margin is tried to maximize. By using vector geometry shows that the margin is equal to $\frac{1}{\|w\|}$ and maximizing it which is also similar to minimizing$\|w\|$. For minimum value of $\|w\|$ Equation 3.4 is also true.

$$y_i(x_i.w + b) - 1 \geq 0 \; for \; \forall i \tag{3.4}$$

Minimizing $\|w\|$ is equivalent to minimizing $\frac{1}{2}\|w\|^2$ and the use of this term makes it possible to perform Lagrange multipliers for minimization.

Assume that Lagrange multipliers $_i$ 0 for $\forall_i$.

$$L_P = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{L} \alpha_i [y_i(x_i.w + b) - 1] \quad (3.5)$$

Lagrange multipliers is used to find w and b which is minimized and which maximizes Equation 3.5. It can be found by differentiating L_P with respect to w and b and this derivative equal to zero.

$$\frac{\partial L_P}{\partial w} = 0 \rightarrow w = \sum_{i=1}^{L} \alpha_i y_i x_i \quad (3.6)$$

$$\frac{\partial L_P}{\partial b} = 0 \rightarrow w = \sum_{i=1}^{L} \alpha_i y_i \quad (3.7)$$

Substituting 3.6 and 3.7 into 3.5 gives a new formulation, which being dependent on , it is needed to be maximized.

$$L_D = \sum_{i=1}^{L} \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j x_i.x_j \quad (3.8)$$

Such that $a_i \geq 0 \ \forall i \ \sum_{i=1}^{L} a_i y_i = 0$

$$H_{ij} = y_i y_j x_i x_j$$

By using the H_{ij} and Equation 3.8 L_D can be write shorter form.

$$L_D = \sum_{i=1}^{L} \alpha_i - \frac{1}{2}\alpha^T H \alpha \quad (3.9)$$

To find the minimum values, maximum value of L_D has to be found.

$$\max_\alpha \left[\sum_{i=1}^{L} \alpha_i - \frac{1}{2}\alpha^T H \alpha\right] \quad (3.10)$$

By using Equation 3.10, can be found. Using the α value in Equation 3.6 w value is calculated and b should be calculated finally.

Any data point satisfying Equation 3.7 which is a Support Vector x_s will have the form:

$$y_s(x_s.w + b) = 1 \quad (3.11)$$

Substituting this in Equation 3.6:

$$y_s(\sum_{m \in S} \alpha_m x_m y_m . x_s + b) = 1 \quad (3.12)$$

where S donates the set of the indices of the Support Vectors. S is determined by finding the indices i where $_{i}$. Multiplying through by ys and then using $y_s^2 = 1$ from Equation 3.2 and 3.3

$$y_s^2 \left(\sum_{m \in S} \alpha_m y_m x_m . x_s + b\right) = ys \tag{3.13}$$

$$b = y_s - \sum_{m \in S} \alpha_m y_m x_m . x_s \tag{3.14}$$

Instead of using an arbitrary function Support Vector x_s, it is better to take an average over all of the Support Vectors in S.

$$b = \frac{1}{N_s} \sum_{s \in S} \left(y_s - \sum_{m \in S} \alpha_m y_m x_m . x_s\right) \tag{3.15}$$

Variable w and b which is used find to need for classification.

3.4.2 Non-separable Data

Linearly separable data is easily classified but sometimes data is not separated by using a linear function. In order to extend SVM methodology to handle data that is not linearly separable, outlier data are allowed to misclassify as shown in Figure 3.6. This is done by introducing a positive slack variable $_{i}$, i , L

$$x_i . w + b \geq +1 - \xi_i \;\text{ for }\; y_i = +1 \tag{3.16}$$

$$x_i . w + b \geq -1 + \xi_i \;\text{ for }\; y_i = -1 \tag{3.17}$$

$$\xi_i \geq 0 \quad \forall_i \tag{3.18}$$

These Equations used to generate general form of the equation.

$$y_i(x_i . w + b) - 1 + \xi_i \geq 0 \quad where \quad \xi_i \geq 0, \forall_i \tag{3.19}$$

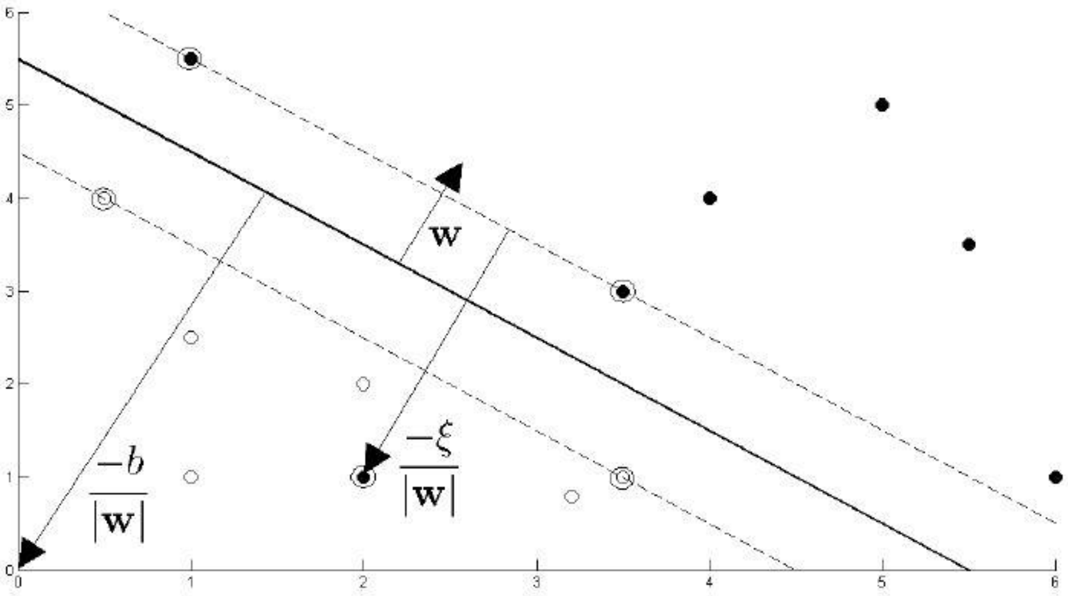

Figure 3.7: Hyper plane through non-separable classes [12]

It is intended to minimize the misclassified data and increase the classifications success rate. Penalty term is added to energy function for each misclassified point.

$$min\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{L}\xi_i \tag{3.20}$$

$$y_i(x_i.w + b) - 1 + \xi_i \geq 0\ \forall_i \tag{3.21}$$

In Equation 3.20 parameter C controls the trade-off between the slack variable penalty and the width of the margin. By using the Lagrangian, we can found minimized values of w, b and ${}_i$ with respect to .

$$L_P = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{L}\xi_i - \sum_{i=1}^{L}\alpha_i[y_i(x_i.w + b) - 1 + \xi_i] - \sum_{i=1}^{L}\mu_i\xi_i \tag{3.22}$$

Differentiating Equation 3.22 with respect to w, b and ${}_i$ and setting the derivatives equal to zero:

$$\frac{\partial L_P}{\partial w} = 0\ \rightarrow w = \sum_{i=1}^{L}\alpha_i y_i x_i \tag{3.23}$$

$$\frac{\partial L_P}{\partial b} = 0 \ \rightarrow \sum_{i=1}^{L} \alpha_i y_i = 0 \tag{3.24}$$

$$\frac{\partial L_P}{\partial \xi_i} = 0 \ \rightarrow C = \alpha_i + \mu_i \tag{3.25}$$

To find the unknowns w and b, LP needed to solve by using Equations 3.23, 3.24 and 3.25.

$$\max_{\alpha} \left[\sum_{i=1}^{L} a_i - \frac{1}{2} \alpha^T H \alpha\right] \tag{3.26}$$

From this form of the equation b is solved by using 3.21 through in this instance the set of Support Vectors used to calculate b is determined by finding the indices I where 0 $_{i}$ <C.

3.4.3 Advantages of SVM using on EEG Classification

Classification problems mostly deal with high dimensional spaces, many numbers of classes and time and memory management problems [13].

SVM can work on high dimension spaces. In EEG classification problem, dimension of space is high because the dimension of the patterns increase with the number of features selected and the number of the sensor channel used for acquisition. Since SVM use over fitting protection, which does not necessarily depend on the number of the features, it has the potential to handle these large feature spaces.

SVM is used for classification problems containing many classes. Each class is represented with different labels, so the classification problem solved for different classes. The dominating approach for doing so is to reduce the single multiclass problem to multiple binary classification problems. Each of the problems yields a binary classifier, which is assumed to produce an output function that gives relatively large values for examples from the positive class and relatively small values for examples belonging to the negative class. Two common ways to produce such binary classifiers where each classifier distinguishes between two methods are given below:

- One versus all: One of the labels to the rest is done by a winner-takes-all strategy in which the classifier with the highest output function assigns the class.
- One versus one: This method is implemented between every pair of classes.

Classification is done by a maximum is winning voting strategy, in which every classifier assigns the instances to one of the two classes, then the vote for the assigned class is increased by one vote and finally the class of the input determines with the most voted class [11].

3.5 FEATURE SELECTION

Feature selection is the first step of the classification. Classification of the data set is impossible in some cases. Feature selection is important because it provides a transformation to different space where data can be classified.

EEG signals are complex signals seem like noise, so gathering information from the time domain is hard. In time domain data seems irrelevant, but it is possible to gather information from that data.

Different spaces give different eigenvectors of this space and different conditions can be shown by the linear combination of these eigenvectors. Transforming data to other spaces obtain different features. Using different spaces guarantee that two different features are not linearly dependent.

Time domain is the main space of the EEG data collected as shown in Figure 3.8. Frequency is also used as a feature space as shown in Figure 3.9. Various transforms can be possible for gathering information on different spaces.

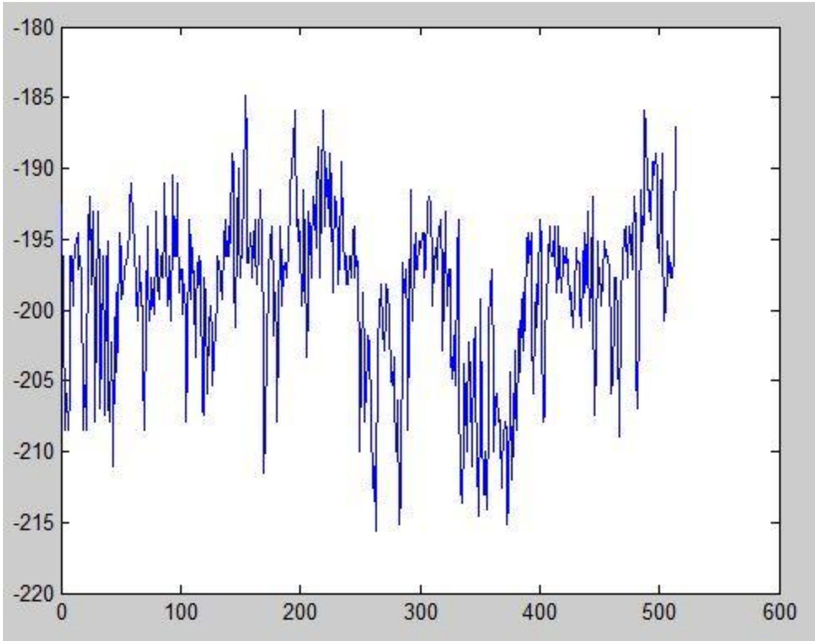

Figure 3.8: EEG Signal in Time Domain

EEG data in time domain is separated in distinct interval parts. These intervals are called Window Length which is also a parameter for classification. Window Lengths should be power of two because FFT (Fast Fourier Transform) transform is faster if the time series length is power of two.

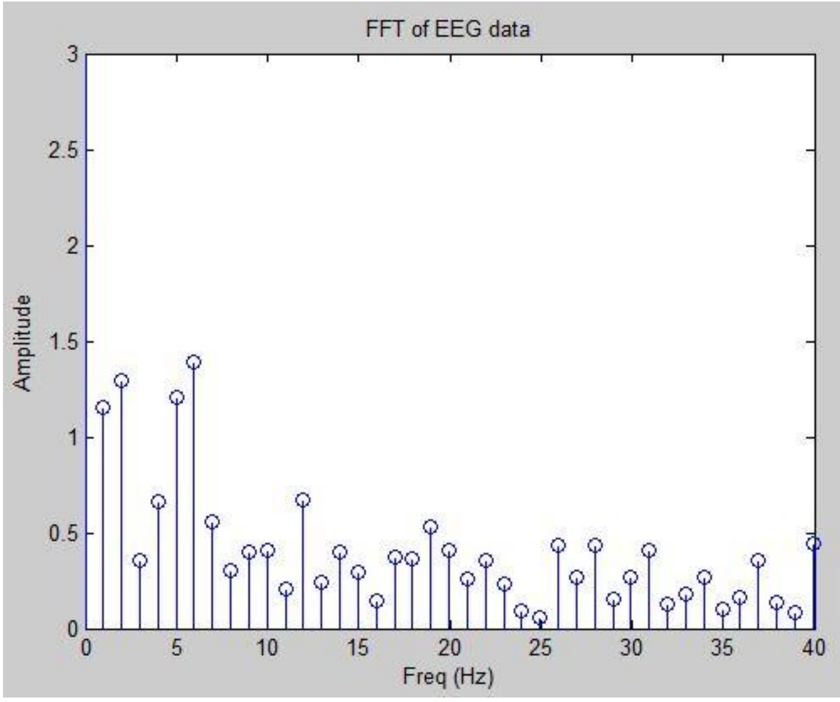

Figure 3.9: FFT of EEG Signal

3.5.1 Features

Feature selection is important for the performance of the classification. EEG signals are very complex and patterns of the EEG data are different for each user. Because of this problem, it is not easy to determine a generic classification method and features.

Different features are selected for the system and Best classification performance is obtained when the features are altered according to the problem.

3.5.1.1 Mean Value

Each sensor channel represented x contains particular number of value. Mean value of the each channel is used as a feature.

$$\mu_{channel} = \frac{1}{N}\sum_{i=0}^{N} x_i \tag{3.27}$$

In equation 3.5.1, N represents Window Length. Each channel s mean value is calculated and mean values of each channel are used as a feature in the considered time interval.

3.5.1.2 Standard Deviation

Standard deviations are used as a feature. Each channel has a N value for a single window. Mean values are calculated as a feature also the standard deviation calculated as follow and uses as a feature.

$$\sigma_m = \sqrt{\frac{1}{N}\sum_{i=0}^{N}(x_i - \mu_m)^2} \tag{3.28}$$

3.5.1.3 Maximum Amplitude

Maximum amplitude is calculated by determining the maximum value of data in each window. This feature is likely to sample each window when the value is maximum amplitude.

3.5.1.4 Dominant Brain Rhythm

EEG signal can be split by frequency ranges. These frequency ranges are called Brain Rhythms. FFT of the EEG signal gives frequency spectrum of the signal and dominant rhythm can be determined from the spectrum.

3.5.1.5 Fast Fourier Transform

Fast Fourier Transform transports EEG signal to the frequency domain. Frequency domain is more efficient for the EEG processing. Common ways for processing EEG use frequency domain features.

EEG Headset acquire 2-45 Hz frequency interval of signal shown in Figure 3.9, so it can be determined which frequency interval is used as a feature. In this work, 64 point FFT is done on time domain response of each of the sensors. Therefore, amplitudes regarding to 64 points in frequency axis are generated for every signal sensor.

3.5.2 Dimension Reduction

EEG signal processing deals with the large amount of data. Emotiv headset is also supplying 14 channels EEG at 128bit/sec transmission rate for each channel [8]. This data is converted to other feature spaces and used in classification in real time applications. Dimension reduction is used to maximize performance by losing insignificant data.

Correlation between features presented above is more important and which one is less significant. For some applications, a specific region of the brain gives response, therefore sensors that are not related to that region remain stable. These sensors can be removed from dataset to reduce the dimension.

4. APPLICATIONS

In this thesis, I have developed many applications by using Emotiv EPOC. Also tools for EEG Processing are written by me. In this section, applications are explained and details are given about them.

These applications are written in different platforms like .NET and MATLAB also some communication tools for data transfer are implemented to combine these platforms' benefits.

4.1 Robot Control

Robot control application is written in C   language and Emotiv s libraries are used to access the raw data. Navigating robot is a task after processing and classifying the EEG data. Navigating robot is achieved by 3 different actions these are moving forward, turning right and left. These actions are associated with the actions that are determined from EEG signals.

4.2 NXT Robot

Robot that is used in this project is Lego Mindstorm NXT, which is mostly used as a test bed for many robotic applications as shown in Figure 4.1. NXT uses Bluetooth connection which is used as a COM port while sending commands to NXT robot.

The main part of the robot is intelligent brick. It can receive input from up to 4 sensors and control up to 3 motors via RJ12 cable.

Figure 4.1: Lego Mindstorm NXT Robot

4.2.1 NXT Software

Application developed for controlling NXT control has two different modes. First one is testing mode for controlling the robot movements regularly. In the other mode robot is controlled by EEG while the headset is connected to the system. In the application shown in Figure 4.2, there are 4 way controls for robot, these are forward, backward, left and right. Also tachometer response from the robot is shown in the application.

Figure 4.2: NXT Control Software

Bluetooth communication is made via virtual serial communication on COM port by AForge library for NXT Robots.

4.2.1.1 Aforge.NET Library

AForge.NET framework provides set of classes allowing manipulation of Lego robotics kits, such as RCX and NXT kits. With the framework's API, it is possible to control robot's outputs (motors) as well as read sensors' values, which allows starting programming robotics applications very quickly [14].

AForge Library takes control of all sensor inputs and motor outputs. By using AForge API robot control class implemented with combining these API methods for purpose. Class diagram of AForge is shown in Figure 4.3. NXT Brick processes commands and sends responds to computer via Bluetooth.

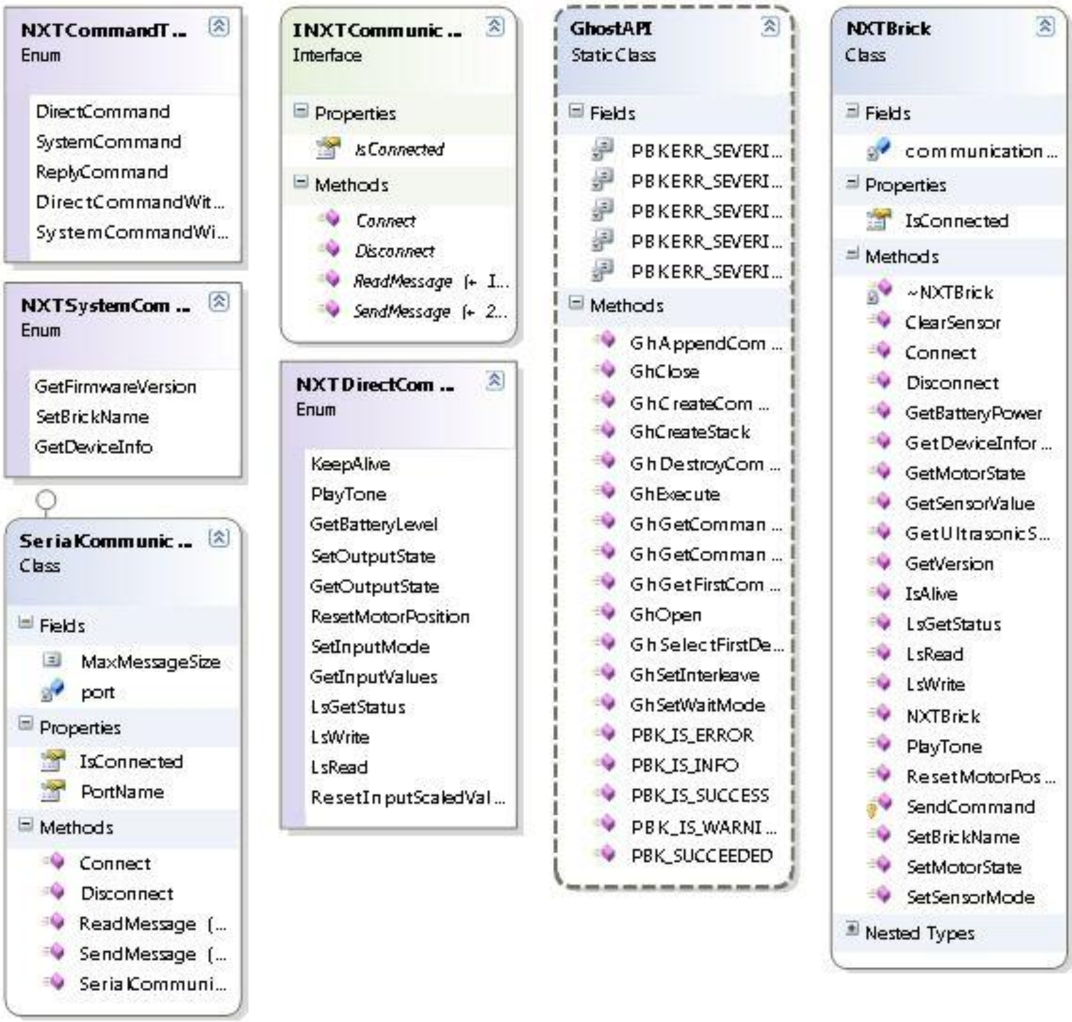

Figure 4.3: AForge API Class Diagram [1]

4.2.1.2 Mouse Emulators

Mouse Emulator is implemented for creating BCI to control mouse cursor position and click events of it as shown in Figure 4.4. This application is developed for testing Emotiv gyros. Mouse position controlled by acceleration data acquired from Emotiv Headset. Gyro gives information for one axis and two gyros is used to control the mouse cursor in 2 dimensions of the screen.

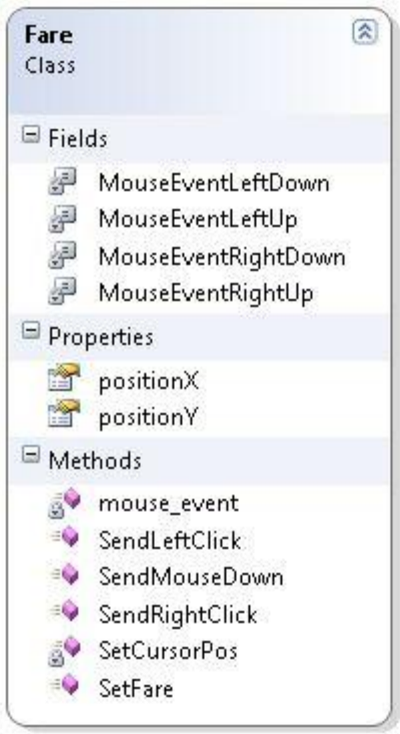

Figure 4.4: Cursor Hook Class [15]

4.2.2 Gyro Data

Data acquired from gyro is acceleration of the movement. It is the periodic sinus wave if the head position changes periodically, but if one certain position is pointed with head, amplitude of the wave changes in positive and negative sites of the wave.

Gyro data is used in 2 different ways first one is acceleration based control, which works smoother. Another method is transforming matrix method in which transformation between head orientation and mouse coordinate is calculated.

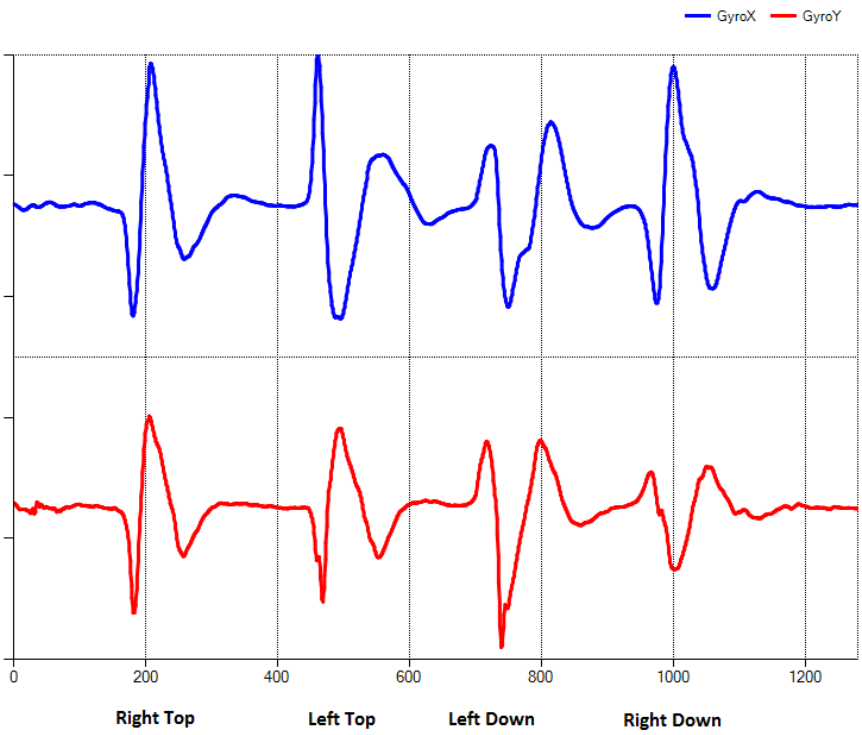

Figure 4.5: Gyro Outputs

In Figure 4.5 different gyro output signals are shown. The main similarities of these waves are integral of the wave is positive or negative, if the movement vector is positive direction. If user points top, integral of Gyro Y output is positive. Similarly, if user points right, integral of Gyro X output is positive as shown in Figure 4.5. It is calibrated and calculated by determining maximum and minimum values of the integrals. Also precision adjustment can be done by user.

4.2.2.1 Acceleration Based Control

Acceleration is the output of the headset. It can be used in time to control mouse speed and by the way, its position. Moving head to a particular position change the form of the wave. Sharp decrease or increases of data are detected to provide the value for acceleration to control the cursor.

Summing the output in time gives the change of the acceleration and these positive or negative values are used to direct mouse different directions. Refreshing the sum in a particular interval is important to avoid from the drift of the sensor.

4.2.2.2 Transform Matrix Method

Transform matrix is giving the direct transform between head position and screen coordinates. Calibration of the system is important in this task two different corner position and cursor coordinates matches and 2x2 transform matrix elements are calculated.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (4.1)$$

Application implemented for this purpose shown in Figure 4.6, contains a start button which should be pressed while looking directly to the middle of the screen. After that acceleration value is integrated two times to acquire position value. Two buttons on the each site of the program is used to match with the position values and cursor positions and transform matrix is calculated.

Main problem of this application is the error coming from the numerical integration. Integration constants are extended by time and error from this source effect the application. Also the drift of the sensor causes an error.

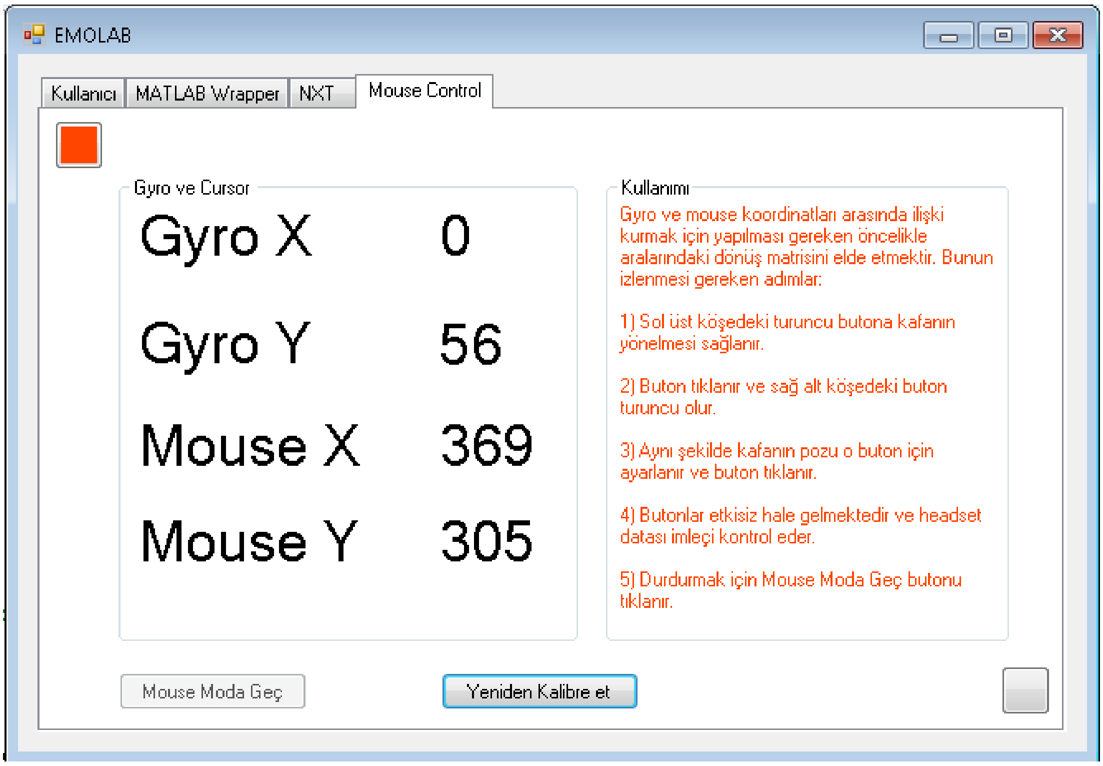

Figure 4.6: Mouse Emulator

4.3 Presentation Software

Presentation software is developed for the slide presentation or reading e book as shown in Figure 4.7. This application controls the PowerPoint or other applications. There are two different classes for changing page.

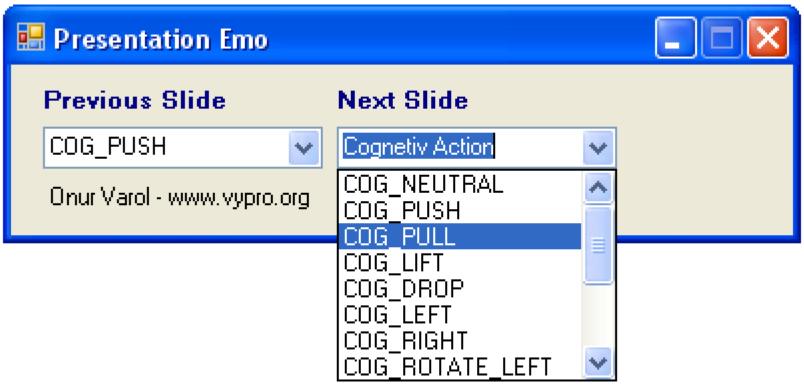

Figure 4.7: Presentation Software

Main structure of the project is same with the other applications. After detecting the EEG pattern, it is used for controlling different interfaces such as mouse cursor or robot.

4.4 MATLAB Test Software

Working with EEG needs different methods and different trials to improve the methods. Tools that are used to improve the algorithm should be easy to implement and flexible for changes in application.

MATLAB is a good platform for testing algorithms and methods. Visualization and plotting the data is also important and MATLAB is very useful in this manner. Test software is written in MATLAB because of all these advantages.

4.4.1 Recorded Data FFT Plotter

Observing the EEG Rhythms is important to determine which sensors are dominant and change on these sensors are important. EEG data which is recorded in the time domain is converted to the frequency domain and categorized for different EEG rhythms such as alpha, beta, gamma, and delta.

Two different datasets are compared with this software, and the software is used for observing the correlation between these datasets. Sixteen channel data are plotted at the same time. Recorded data can replay with this software and observe the changes in time as shown in Figure 4.8.

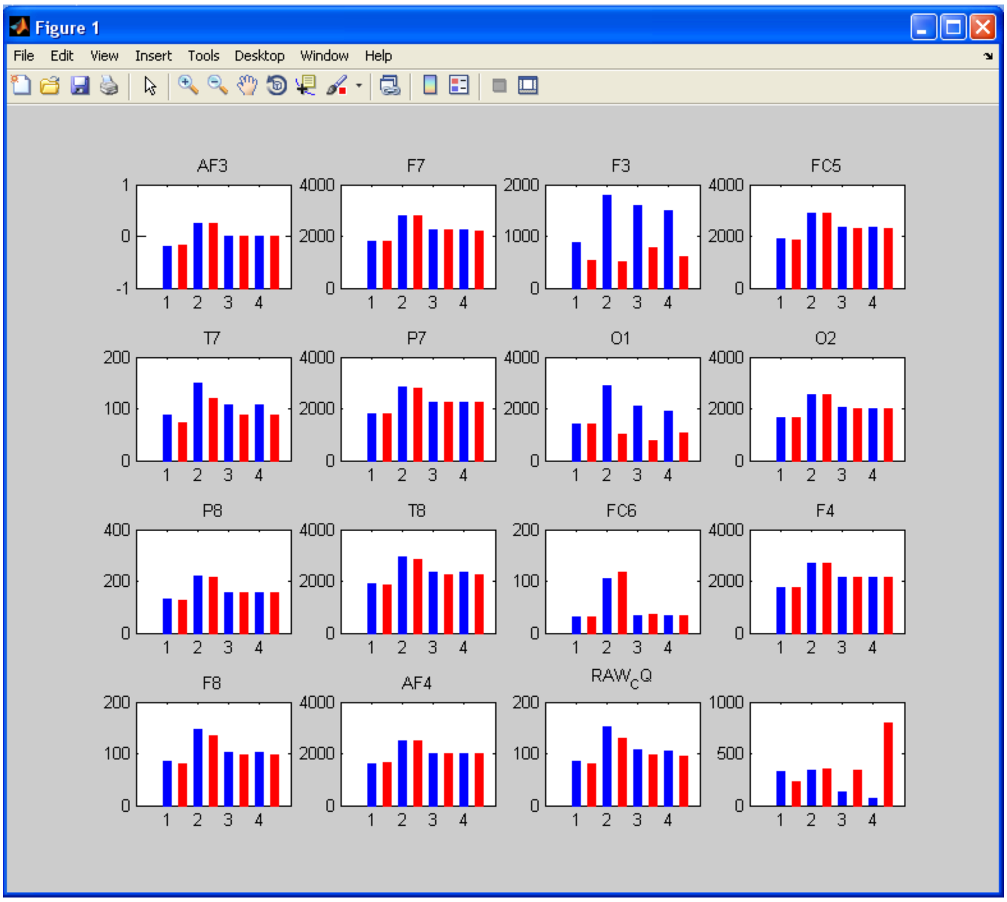

Figure 4.8: FFT Player

4.4.2 TCP-IP Communicator with .NET

The data should be transferred properly in order to run the real time applications on MATLAB correctly. There are different ways to do this such as accessing library files via MATLAB or socket communication. Socket communication implemented using TCP-IP protocol is an efficient way to work at live data on MATLAB.

For the communication interface between different platforms, server and client applications are developed. Acquiring raw data and sending them to listening application is the role of the server application. C# console application is developed for this task. Communication work on the certain IP: 127.0.0.1.

Client application is written in MATLAB to listening server socket to obtaining data and lost data and data received from different order is arranged using control channel, which sends triangle shaped data for using reordering data packages.

4.4.3 Performance Analyzer

Performance analyzer is the testing software written in MATLAB to test the classification performance of algorithms and different features. Data sets are easily applied into this application. Visualization and comparing the result with other data sets are easy with this application. A public available SVM classifier, LIBSVM [16] is used for classification with linear kernel.

5. EXPERIMENTS

BCI systems highly depend on the environmental changes. Therefore, a system that is trained in a certain environment may not provide the expected performance when it is retested in another environment. Especially the physiological and physical changes of the user may decrease the performance.

For a healthy experiment and gathering qualified data, user should be fully concentrated, sitting relaxed and eyes open. These are important for the quality of the data, which are collected for the training of the system.

Experiment data are collected from two different users while user concentrating on 3 different cognitive action. These cognitive states are chosen to implement for several applications. These cognitive states are left, right and neutral states of the users. Aim is to discuss the effects of different conditions on classification results. The conditions are:

- In rest position,
- Standing,
- Thinking on mathematical questions.

The most important factors that affect performance of the systems are determined by experiments. There are also other factors, which are less important than the factors listed below.

One of them is the users. Users need to know how to train and how to think about the actions. Expert users are more successful at training the system, because expert users know what kind of mental activity (i.e. focusing the action, thinking of muscular movements) provide the best data for the system. In these experiments one experienced and one untrained users are applied to different tasks.

In BCI system feature extraction is made on EEG data. Blueprints of the data are gathered and classification is applied by using SVM. Firstly, a three class problem is considered. Two users are wanted to focus on moving an object to right or left each separately. The third class is the neutral state of the user. Then the partially collected data are windowed with intervals of 0.3 second and combined for a better visualization. The classes are labeled with numbers as given in Figure 5.1. Right pattern shown as 3, left shown as 1 and neutral state is represented as 2. In rest conditions training performance is quite high. If the test and training data are qualified, the conditions that test and training data acquired are identical, classification performance on the test set is between 93% and 98%.

Certain features are combined in order to form a fingerprint for the considered window. Then the fingerprints are labeled with the class that they belong to, so that a set of labeled fingerprints is created. The labeled fingerprints set is divided into two subsets, training set and test set. The training set, which contains randomly chosen 70% of the labeled fingerprints set, is used for building the classifier model with linear kernel. The rest of the fingerprints are used for calculating the performance of the model. The percentage of correctly guessed fingerprints to total amount of test data is assumed to be the performance of the system for the chosen features. Both time and frequency based features are used in classification.

Maximum amplitude and FFT are used to generate fingerprint of each window. As given in Section 3.5.1.5 and 4, features for each sensor are extracted from frequency domain analysis. FFT feature is 64 dimensions for each sensor. Maximum amplitude of each sensor in the evaluated window forms another feature. Overall, 64+1 features are extracted from each sensor information and a 14x65 dimensional pattern is generated for each window. Total number of patterns is 14x65x(number of windows).

In real applications, the end user may not be able to train the system on his own, so system may be preferred to be trained at set up. However, conditions of the user that are given above, cannot be always identical at training phase and using phase. In this section the effect of the test conditions on performance of the trained system will be investigated. After the system is trained with the clear data collected when the user is relaxed, test data is acquired for different conditions for 20 seconds each. Then the same features with the training data are extracted.

Table 1: Rest condition of the users

EXPERIENCED USER	UNEXPERIENCED USER

optimization finished, #iter = 2007
nu = 0.363533
obj = -193.030869, rho = -0.203574
nSV = 233, nBSV = 218
.
optimization finished, #iter = 823
nu = 0.161300
obj = -56.675088, rho = 0.140469
nSV = 93, nBSV = 78
.*
optimization finished, #iter = 991
nu = 0.190505
obj = -80.784603, rho = -0.760733
nSV = 124, nBSV = 110
Total nSV = 360
Accuracy = 97.3958% (187/192) (classification)
	optimization finished, #iter = 2379
	nu = 0.403273
	obj = -219.517459, rho = -0.033665
	nSV = 257, nBSV = 243
	.*.*
	optimization finished, #iter = 1477
	nu = 0.203164
	obj = -86.875332, rho = -1.215531
	nSV = 133, nBSV = 117
	.*
	optimization finished, #iter = 1039
	nu = 0.166512
	obj = -60.434509, rho = 0.258464
	nSV = 95, nBSV = 84
	Total nSV = 385
	Accuracy = 91.6667% (176/192) (classification)

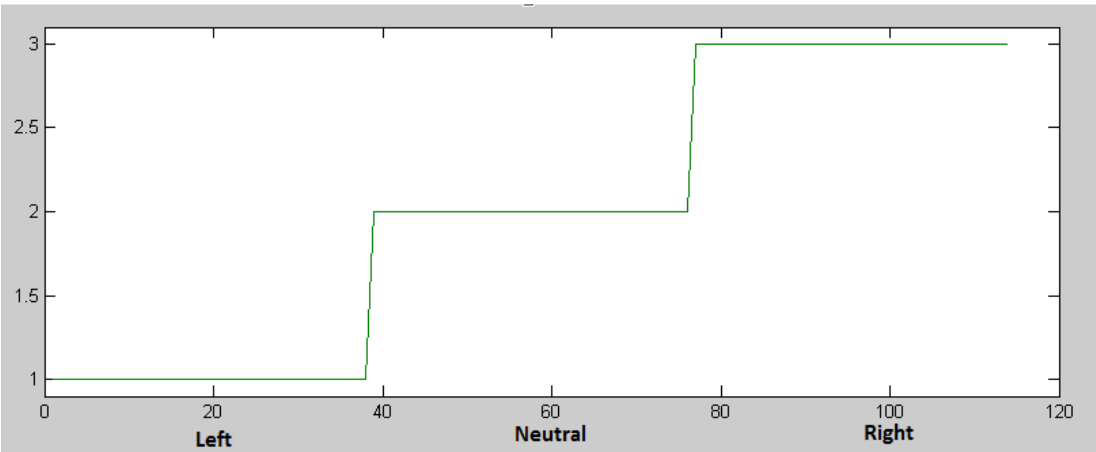

Figure 5.1: Representation of the Classes

5.1 Changing The Test Environment

Test environment is like the workplace of the applications. Performance of the test is nearly same as the real time performance. Training sessions are quite different than the test environment. Because of this difference performance in real time is worse than the test results.

User is effected many different factors. Lighting, sounds, physiological changes are affecting the system [6].

In experiment user brain activity is recorded while they are standing. Experiment result is shown below. Training data set is recorded while user is relaxed, however test data is recorded in changed test environment and classification results are compared with the test results of unchanged conditions.

Table 2: Users in standing conditions

EXPERIENCED USER	**UNEXPERIENCED USER**

optimization finished, #iter = 104
nu = 0.204764
obj = -72.345352, rho = 6.225184
nSV = 115, nBSV = 106
*
optimization finished, #iter = 285
nu = 0.447130
obj = -179.607435, rho = 0.624384
nSV = 245, nBSV = 234
.
optimization finished, #iter = 541
nu = 0.297255
obj = -101.821937, rho = -2.278184
nSV = 166, nBSV = 153
Total nSV = 409
Accuracy = 95.539% (257/269) (classification)
	optimization finished, #iter = 1985
	nu = 0.396750
	obj = -216.624726, rho = 0.103371
	nSV = 254, nBSV = 238
	.*.*
	optimization finished, #iter = 1357
	nu = 0.168156
	obj = -60.313122, rho = -0.008488
	nSV = 97, nBSV = 81
	.*..*
	optimization finished, #iter = 1941
	nu = 0.188866
	obj = -80.627606, rho = -1.283009
	nSV = 126, nBSV = 110
	Total nSV = 381
	Accuracy = 92.8367% (324/349) (classification)

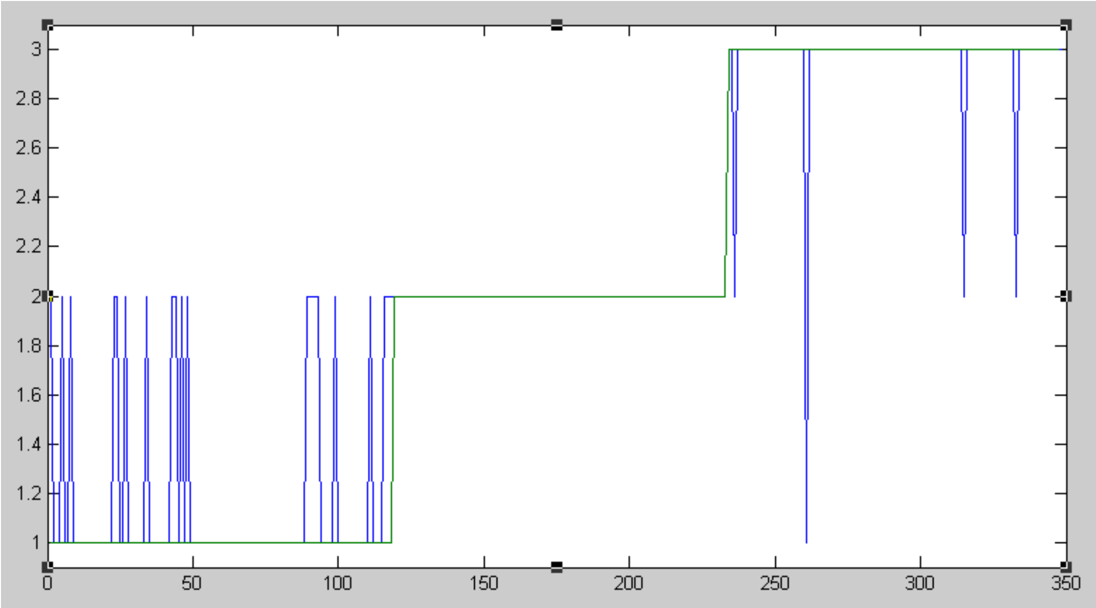

Figure 5.2: Responses while inexperienced user standing

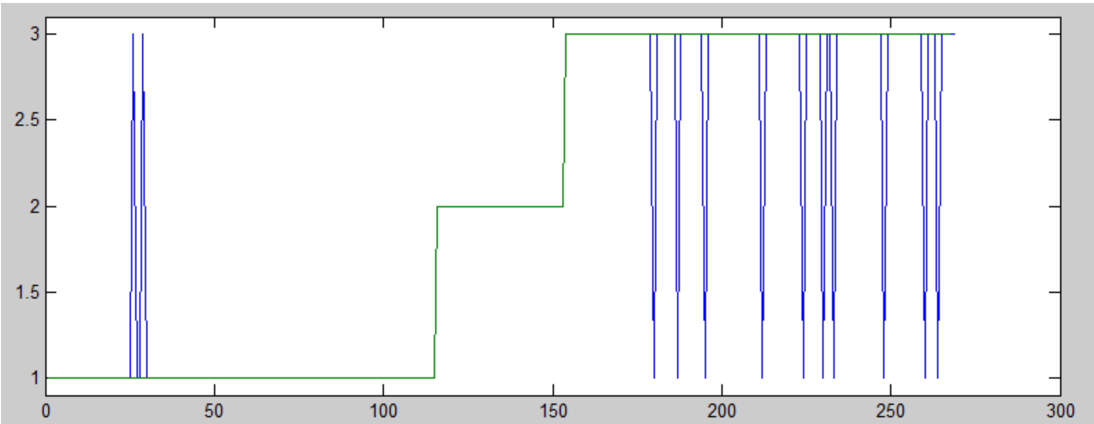

Figure 5.3: Responses while experienced user standing

As it seen from the results in Table 2, classification performance of the experienced user (95%) is worse than the case when training and test data are collected together. However, for the performance of the inexperienced user does not change with the standing condition. In Figure 5.2 and Figure 5.3, green line shows the actual class and blue lines are the guesses of the classifier for the pattern. Pattern number is given with the horizontal axis. Note that, the classifier is not likely to misclassify the neutral state; on contrary it is likely to be confused in other states and classify them as no action for the data collected from inexperienced user. It can be discussed that, an inexperienced user s thinking of left or right is not very different than his unfocussed neutral state. However, for the experienced user, classifier is confused between left and right states It is seen that changing in the environment conditions affects the performance of the classification, especially for the experienced user.

5.2 Changing the Recording Action

In BCI applications one the most important factor is users performance on training. Users are needed to be educated how to train and use the BCI system because the way the user train system and test it should be same. Expert users training and test results are more adequate because they are experienced of how to use the BCI system.

Patterns of thoughts are often different from user to user. Some users figure how the action actually done, some visualize it and some users think of moving their limbs. Performance and EEG patterns diverse from each other by these behaviors, so users have to behave the way the system is trained otherwise the performance of the system would be low.

In this part performance is tested in different conditions. Training data are collected while the user is fully concentrated and at sitting position. Training data are used as a reference for our testing results. Test data are collected in two different methods. One of them is recording the data while the user concentrates on both moving robot and solving multiplication problems on mind and the other is moving limbs while concentrating on the task.

5.2.1 Testing while Analytical Thinking

In this test effect of different thoughts are examined. Users concentrate on moving the robot but with different thoughts are on their mind. Most effective thoughts are the most ones that we need to concentrate on most, so mathematical problems are the best task for this test. The users are wanted to give answer to simple multiplication problems when they are focused on moving the robot.

Result of the test shows that performance decreases while the user tries to concentrate two different thought as we expected. As it is shown in Figure 5.4, some patterns belonging to the neutral state are misclassified. It can be discussed that analytical thinking in the neutral case may interfere with the trained actions such as thinking to move left or right for the inexperienced user. However, as in the previous condition, performance is not changed for inexperienced user; results are given in Table 3. In order to eliminate the neutral state problem, more recordings should be used for the neutral case especially when the users lose their concentration.

Table 3: Users in analytically thinking condition

EXPERIENCED USER	**UNEXPERIENCED USER**
optimization finished, #iter = 159	optimization finished, #iter = 1003
nu = 0.194596	nu = 0.378439
obj = -68.220383, rho = -5.983371	obj = -204.791925, rho = -0.089145
nSV = 109, nBSV = 99	nSV = 241, nBSV = 228
* .*	*.*
optimization finished, #iter = 520	optimization finished, #iter = 918
nu = 0.304924	nu = 0.168180
obj = -104.749120, rho = -2.210396	obj = -59.907734, rho = 0.374119
nSV = 169, nBSV = 158	nSV = 97, nBSV = 85
*	*.*
optimization finished, #iter = 231	optimization finished, #iter = 1357
nu = 0.440362	nu = 0.195815
obj = -175.210306, rho = 1.371363	obj = -84.907293, rho = -0.760847
nSV = 238, nBSV = 228	nSV = 128, nBSV = 112
Total nSV = 418	Total nSV = 370
Accuracy = 94.7955% (255/269) (classification)	Accuracy = 92.4119% (341/369) (classification)

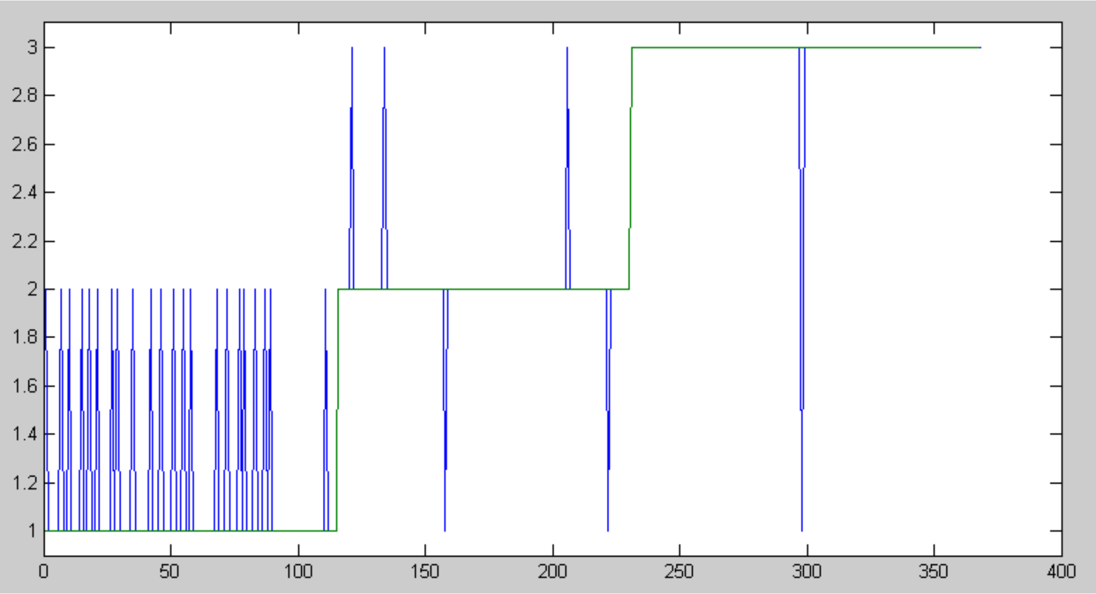

Figure 5.4: Responses while inexperienced user analytically thinking

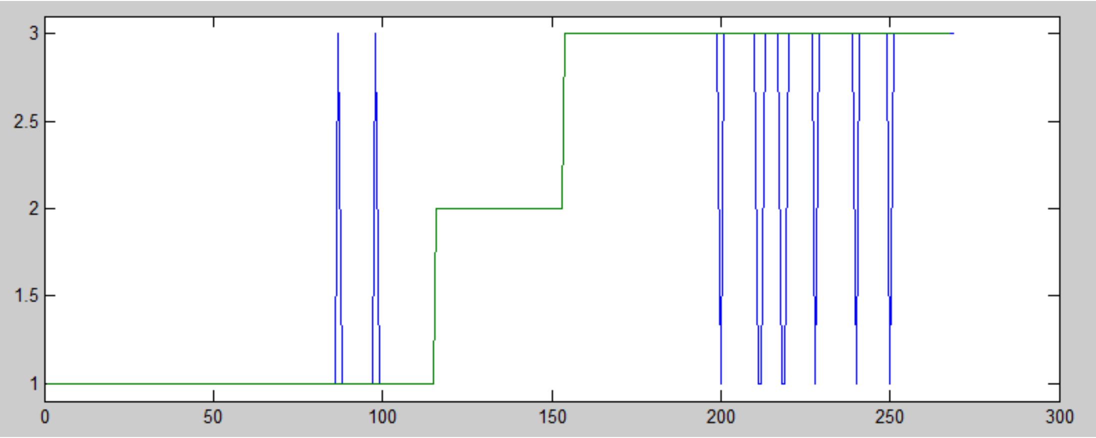

Figure 5.5: Responses while experienced user analytically thinking

5.2.2 Testing while Moving Limbs

In this test motor cortex s effect on the training is examined. Users concentrate on moving the object while they are moving their limbs. Training data is recorded while user visualizing the moving the object by thinking of moving limbs for collecting the test data. Comparison of the test data recorded while moving limbs and test data in rest conditions, results are better while moving limbs. If the effect of thinking of an action and doing the action on performance of EEG data classification, it can be stated that muscular movements can be classified more accurately.

Table 4: Users in moving limbs condition

EXPERIENCED USER	UNEXPERIENCED USER

optimization finished, #iter = 152
nu = 0.210137
obj = -72.865678, rho = 6.106700
nSV = 118, nBSV = 108
*
optimization finished, #iter = 276
nu = 0.459164
obj = -182.432701, rho = 1.765062
nSV = 249, nBSV = 239
*
optimization finished, #iter = 324
nu = 0.310649
obj = -108.399277, rho = -2.230768
nSV = 169, nBSV = 159
Total nSV = 421
Accuracy = 97.026% (261/269) (classification)
	optimization finished, #iter = 1646
	nu = 0.390619
	obj = -210.991820, rho = 0.401864
	nSV = 249, nBSV = 237
	*
	optimization finished, #iter = 496
	nu = 0.157155
	obj = -55.450721, rho = 0.154258
	nSV = 90, nBSV = 78
	*
	optimization finished, #iter = 2231
	nu = 0.196391
	obj = -84.064264, rho = -0.926501
	nSV = 130, nBSV = 115
	Total nSV = 377
	Accuracy = 92.1965% (319/346) (classification)

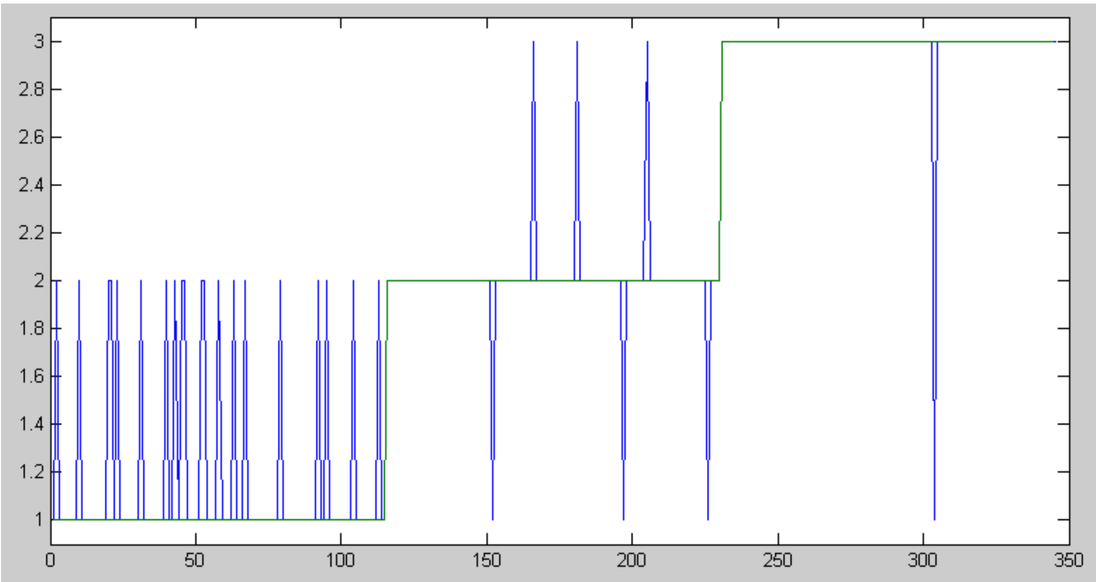

Figure 5.6: Responses while inexperienced user moving limbs

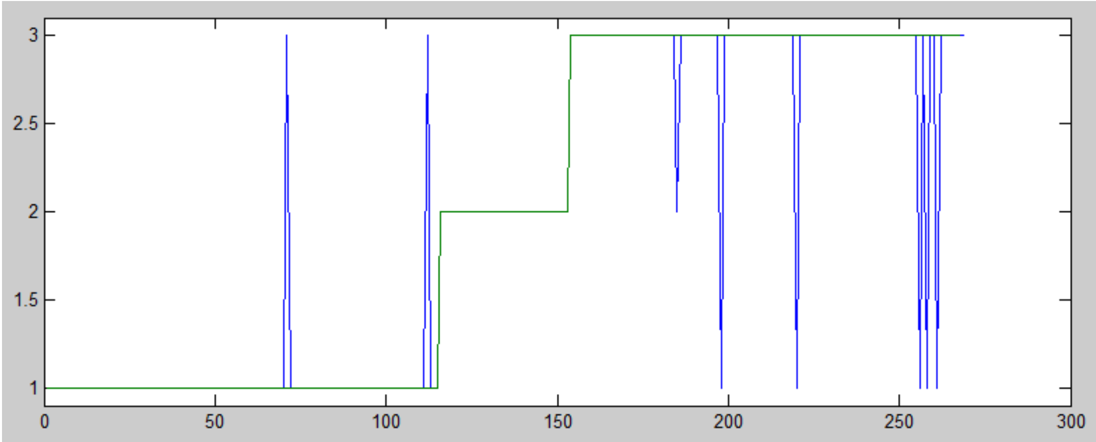

Figure 5.7: Responses while experienced user moving limbs

In different conditions effects of the environment changes are watched and decreases of the performance is measured and summarized in Table 5.

Table 5: Performance comparison

	Normal Conditions	**Standing**	**Analytical Thinking**	**Moving Limbs**
Experienced User	**97.4**	**95.54**	**94.80**	**97.03**
Inexperienced User	**91.7**	**92.84**	**92.41**	**92.20**

Experienced users performances are much better than inexperienced ones. Standing and performing mental task while concentrating on another other task decreases the success of the classification. Concentrating on other task is decreases the attention on a simple task. However, moving limbs increases the performance. Motor cortex responses are more easily detected on the EEG signal. Activation for disabled people is the same for the healthy people, so muscular activity can be used to train for wheelchair control.

Inexperienced user s results are not as uniform as experienced users. Loss of concentration is one of the problems. As seen on Table 5, inexperienced users performance is different than experienced users.

6. CONCLUSION

In this thesis EEG signals are processed and classified using SVM. Emotiv Headset is used and numerous applications are developed using Emotiv Headset. Classification problem is one of the main topics of this project and different features are used for classification, performance and environment effect on the performance is measured.

Robot control and different BCI applications are developed to show performance of the system. It is seen that after successful classification anything can be controlled using EEG data. In the Nonlinear system laboratory, EEG processing is a new topic and many tools are written for future usage. These applications will be improved and new applications will be developed by using the tool written in this project including new approaches to robot navigation problem. For example, robot control application can be expanded by using two headsets [17].

To compare the performance of the classifications, different experiments are done and results of these experiments are examined. Effects of the environment and conditions of the users are discussed. These tests give an insight about the performance of the system when it is used for real time applications.

In this thesis, classification performance of only SVM is evaluated. For future work, different classification methods, such as statistical ones, should be examined because characteristics of the data determine the classification tool that may give the best performance. Progress of the project is written on the project web page and new applications are presented [17].

Feature extraction can be improved by selecting the most beneficial features. The computational cost decreases in both feature generation and classification if number of features can be reduced while keeping the performance.

REFERENCES

[1] **Nykopp and Tommi**, Statistical Modeling Issues for the Adaptive Brain Interface, June 2010.

[2] **Saeid Sanei, J.A. Chambers**, 2007. EEG Signal Processing, Wiley.

[3] http://www.zbynekmlcoch.cz, December 2010.

[4] http://en.wikipedia.org/wiki/Eeg, December 2009.

[5] http://blog.makezine.com/archive/2008/12/the_brain_machine.html, December 2009.

[6] **Drongelen, Wim Van**, 2006. Signal Processing for Neuroscientists, Academic Press.

[7] http://www.bem.fi/book/13/13.htm, December 2010.

[8] Emotiv Research Plus Specification, February 2010.

[9] EEGLAB Wiki Documentation.

[10] **C. Cortes and V. Vapnik**, 1995. Support-vector networks, Machine Learning, 20:273-297.

[11] http://en.wikipedia.org/wiki/Support_vector_machine, March 2010.

[12] **Nello Christianini, John Shawe-Taylor**, 2001. An Introlduction to Support Vector Machines and other kernel-based learning methods, Cambridge University Press.

[13] **Joachims and Thorsten**, 1997. Text Categorization with Support Vector Machine: Learning with Many Relevant Features.

[14] http://www.aforgenet.com/framework/features/lego_robotics.html, June 2009.

[15] http://www.vypro.org, August 2009.

[16] **Chih-Chung Chang and Chih-Jen Lin**,2001. LIBSVM: a Library for Support Vector Machines

[17] http://eeg.vypro.org, May 2010.

RESUME

Onur VAROL was born in İzmir at 12 Temmuz 1988. He studied High School in Karşıyaka Anatolian High School and graduated in 2006. He attended Istanbul Technical University Electronics Engineering at 2006. He is also a Double Major student in Physics. He will continue his academic career in Computer Sciences.