

LAPPEENRANTA UNIVERSITY OF TECHNOLOGY
LUT School of Energy Systems
Electrical Engineering

Filipp Semin

**CONTROL OF A MOBILE ROBOT BY A NONINVASIVE BRAIN-COMPUTER
INTERFACE**

Supervisors: Professor Huapeng Wu, LUT
Associate professor Tuomo Lindh, LUT

Lappeenranta, 2017

ABSTRACT

Lappeenranta University of Technology
LUT School of Energy Systems
Electrical Engineering

Filipp Semin

CONTROL OF A MOBILE ROBOT BY A NONINVASIVE BRAIN-COMPUTER INTERFACE

Master's thesis
2016

55 pages, 48 figures, 1 table and 0 appendices

Examiners: Professor Huapeng Wu, LUT
Associate professor Tuomo Lindh, LUT

Keywords: Brain-computer interface, mobile robot control, electroencephalography, common spatial pattern, matched filter, genetic algorithm

In this work, an EEG-based brain-computer interface (BCI) and a mobile robot software were developed for a BCI mobile robot control application. A common spatial pattern (CSP) was applied to the recorded EEG signal filtered with a 5th order Butterworth filter and then the following features were extracted from the surrogate channels: CSP variance, Katz fractal dimension measure, Kolmogorov entropy, and Matched filter output. The Butterworth filter band frequencies, as well as the Matched filter template were optimized using a genetic algorithm. The features were classified between two classes "Go" and "No go" using a feedforward neural network.

ACKNOWLEDGEMENTS

The author wishes to express his gratitude to his supervisor, prof. Huapeng Wu, for his guidance and support during the research, and for the environment he provided to make this research possible.

The author must also express his gratitude to Amin Hekmatmanesh for sharing his profound knowledge in electroencephalography, for his help and assistance.

Special recognition goes to PhD student Aleksei Romanenko for his support in the development of mobile robot.

Lappeenranta, February 2017

TABLE OF CONTENTS

LIST OF SYMBOLS AND ABBREVIATIONS	5
1 INTRODUCTION	7
1.1 Background	7
1.2 Literature review	8
1.3 Research methodology	11
2 EEG PROCESSING	13
2.1 EEG Preprocessing	13
2.2 Detection of Lateralized Readiness Potential	18
2.2.1 Double subtraction	19
2.2.2 Matched filter	22
2.3 Common Spatial Pattern	24
2.4 Linear Discriminant Analysis	26
2.5 Data classification	28
2.5.1 Multilayer perceptron	28
2.5.2 Radial basis function network	30
2.5.3 Support vector machine	31
3 CONTROL OF A MOBILE ROBOT	32
3.1 Mobile robot	32
3.2 Neuroelectronics Enobio	35
3.3 Robot-to-Matlab interface	36
4 RESULTS	39
4.2 BCI algorithm	39
4.1 Comparison	46
5 CONCLUSIONS.....	48
6 SUMMARY	53
REFERENCES	54

LIST OF SYMBOLS AND ABBREVIATIONS

BCI	Brain-computer interface
BP	Bereitschaftspotential
CNV	Contingent negative variation
CSP	Common Spatial Pattern
EEG	Electroencephalography
EMG	Electromyography
EOG	Electrooculography
ERD	Event-related desynchronization
ERP	Event-related potential
ERS	Event-related synchronization
F	Feature matrix
f	Feature vector
LDA	Linear Discriminant Analysis
MMP	Movement-monitoring potential
MRCP	Movement-related cortical potential
P	Whitening transformation matrix
Q	Feature significance
R	Covariance matrix
RP	Readiness potential
S	Autocorrelation matrix
SMA	Supplementary motor area
V	CAR filter output
W	Transformation matrix
X	Raw data matrix
Z	CSP output

Subscripts

a	Class “a”
b	Class “b”

i	Feature/channel index
j	Epoch/sample index

Superscripts

CSP	Common Spatial Pattern filter
LAP	Laplacian filter
CAR	CAR filter
p	Feature vector item index

1 INTRODUCTION

1.1 Background

Recent advance in the field of EEG opened broad possibilities for research and development for applications using brain waves in various areas. Traditionally, EEG data was gathered by large clinical devices with either invasive sensors that were put directly under the individual's skull or noninvasive wet electrodes lubricated with a special gel. Currently the market provides devices that use noninvasive dry electrodes for EEG measurements. These devices are cheap and small compared to clinical ones and more convenient in everyday use. NeuroSky, MindFlex, Emotive, Enobio are the examples of such devices; Figure 1.1 demonstrates one of them.



Figure 1.1 – A modern EEG headset

Due to the growing availability of cheap and compact EEG headsets, it is important to investigate the possibilities of their usage in BCI systems for various scientific and practical applications. In this work, we study the use of a commercial EEG headset for control of a mobile robot.

BCI creates a communication media between human brain and computer. This allows an individual to operate an external system (e.g. a mechanical arm or a wheel-

chair). A computer system is used to analyze a given bioelectrical recording and produce control commands in response to certain events in the recording. EEG-based BCI systems do not require individual's muscular activity for the system's operation. That feature is extremely valuable for severely disabled people. An EEG-based BCI can read the individual's intention to perform a certain form of action and omit the signal associated with the action execution. Studies have shown that those signals features reflecting the movement intention are still present in EEG recordings of those people who are not able (e.g. due to a neurodegenerative disease) to execute the movement.

1.2 Literature review

History of BCIs starts with publication of prediction of the motor tasks idea by Heimholtz in 1867. After that, Sperry and Von Holst demonstrated that there is a trace of the expected voluntary movement in the electrical activity of the person's body. The nature of the original movement can be traced back by the proper analysis of this activity [1, 3, 18]. Starting from the second half of twentieth century, there have been a significant increase of a number of papers published on the analysis of the body's electrical activity for the prediction of the motor tasks.

The main goal of the BCI technology lies in allowing severely disabled individuals to communicate and classification of their decisions based on their body's electrical activity. EEG can allow registering the movement imagination in real time, therefore opening a possibility for a motor task prediction to be implemented. BCI processing EEG in real time forms a communication with external world without use of body muscles [16]. Studies have revealed a significant potential for the investigation of ERPs and brain rhythms. Thus, processing of these can be applied within a BCI system for a prediction of movement imagination. This system can be used in clinical environment for rehabilitation purposes, for example, in a way as presented in Figure 1.2.

Research has demonstrated that an online EEG recording provides sufficient amount of data for implementation of basic communication functions as controlling a cursor on screen, controlling environment (e.g. light, heat, door control), or even controlling an electromechanical prosthesis [16]. For individuals with severe neurodegenerative diseases a personal set of BCI functions can be created [7].



Figure 1.2 – A disabled person rehabilitation with use of a BCI

The term “premovement” refers to the time at which the individual’s muscles are not involved in performing any movement, but when the individual is highly aware of the movement he will perform in the near future. This time is about 0.5-2s before the movement onset, which is required for the brain to adapt/prepare for the execution of the movement.

Among the papers published on the EEG analysis for the prediction of the upcoming events there is a large fraction investigating the movement-related cortical potentials (MRCPs) [15, 19]. The MRCP is a low-frequency negative slope in the EEG recording appearing about 500ms prior to the movement onset. The MRCP reflects the brain preparation to the movement execution. Beside BCI, the MRCP also have been used for other research objectives, such as motor skill learning [19].

For a cue-based movement, the MRCP is known as contingent negative variation (CNV), while for the self-paced ones the MRCP is known as Bereitschaftspotential (BP) [15]. Both imaginary and actual movements produce the MRCP. There are three events in the MRCP, these are the BP or readiness potential (RP), motor potential (MP), and movement-monitoring potential (MMP) [15]. Studies have compared the occurrence of MRCP both for healthy individuals and for individuals with severe neurodegenerative diseases. The MRCP occurs in EEG recording not only in actual move-

ment preparation, but also in movement imagination when a person is not able to perform the movement, which makes this EEG feature useful for rehabilitation of disabled individuals.

The negative cortical potential growing from about 1.5 to 1 s before the voluntary movement execution is known as BP [15]. The BP is divided into two segments: “early BP” and “late BP”. Early BP starts about 1.5 s before the action execution in the central-medial scalp. The more rapid negative slope, the late BP, appears at around 400 ms before the action execution in the contralateral primary motor cortex and lateral premotor cortex. These segments are affected by different factors, for example, by the complexity of the intended movement which highly influences the late BP [16].

In the case of a cue-paced movement the CNV appears between the “Warning” and “Go” markers and represents a slow negative variation. Like the BP, the CNV signals the brain’s anticipation for the upcoming signal and its preparations for the execution of the action. The earlier CNV segment appears in the frontal cortex as a response to the “Warning” marker, and later (terminal) CNV starts about 1.5 s before the “Go” marker as a preparation for the motor response [1]. The terminal CNV has its maximum amplitude across the motor cortex.

Signals extracted from the EEG do not originate from the cortex. The origin of the recorded voltages can lie in eye movement, muscular and heart actions, or in the recording device itself [7]. Khorshidtalab (2011) discusses techniques for artifact removal from EEG recordings. One technique involves using artificial neural networks [7]. A neural network can successfully classify eye movement artifacts in EEG when is given the coefficients of the eye movement artifact properties. Unfortunately, the necessary use of massive training set poses limitations on the implementation of this technique in real-time systems.

To enhance the quality of the EEG recording a simultaneous EOG recording can be implemented. Since the recorded EEG signal includes both EEG and EOG, the subtraction of the separately recorded EOG from EEG can result in the better quality of the produced signal. Unfortunately, it is not applicable to remove the heartbeat artifact from EEG with the same technique; attempt will cause a classification error.

An eye movement artifact can also be removed with a threshold technique. If the signal energy value exceeds the given artifact thresholds then the sample in which it occurred is excluded from the EEG [7]. This method provides a good quality while is easy to implement.

1.3 Research methodology

The development of an algorithm for control of a mobile robot with an EEG-based BCI included the following steps:

- Development of a BCI algorithm able to precisely and with limited computational costs perform a short EEG signal window processing in order to detect a movement intention-related event;
- Development of a mobile robot with an ALU able to receive command from a computer via a wireless interface and translate them into a set of reference values for electrical drive system;
- Development of a software interface for mobile robot able to communicate with MATLAB environment and translate MATLAB requests into peripheral interface commands.

The development of algorithm of movement intention detection from an EEG recording included the implementation of a signal processing system according to the structure presented in Figure 1.3.

The practical implementation considerations included the following:

- Band-pass filter;
- Spatial filter or a signal linear transform;
- Feature extraction;
- Possible feature dimension reduction
- Classifier selection;

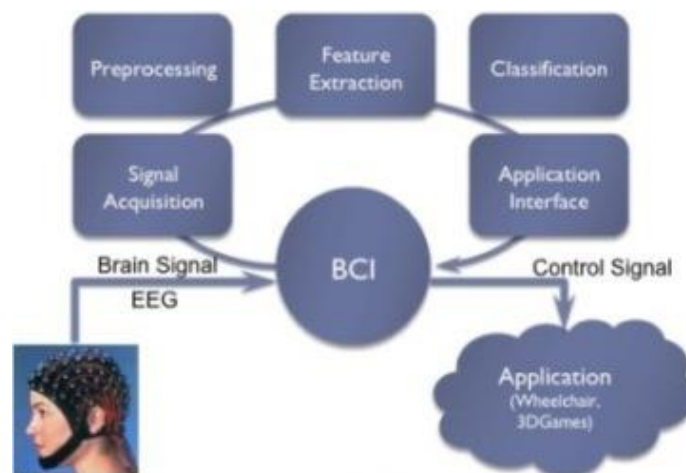


Figure 1.3 – BCI algorithm block diagram [17]

On this stage, the following tasks required a decision:

- Selection of the optimal band frequencies for band-pass filter resulting in the noise rejection and at the same time preservation of the informative components;
- Selection of a spatial filter or a derivation of the signal linear transform to enhance the signal spatial resolution. This required either a selection of the suitable spatial filter, or the derivation of the optimal linear transform coefficients;
- Selection of the most discriminative features for extraction. This required the derivation of a metric for how discriminative a feature is and selection of those features that provide the maximum value of this metric;
- In case if the amount of features used for classification results in unsatisfactory feature processing time, the reduction of feature vector dimension must be performed.;
- Selection of a classifier type suitable for the classification of the feature vector.

The evaluation of the BCI algorithm performance required an appropriate EEG signal recording performed under special conditions. The data had to be either recorded in an experimental setup or selected from published recordings of an appropriate motor task.

The development of a mobile robot aimed to physically model a wheelchair in for a wheelchair control with EEG-based BCI experiment. The EEG signal processing is assumed to be performed in a MATLAB environment, thus an interface is required for a direct control of the robot from MATLAB.

The interface was split into a software interface for MATLAB environment and a serial command interface of the robot. This required either consideration of MATLAB means for serial communication or development of the external interface.

2 EEG PROCESSING

The robot control task requires the following steps of the EEG signal processing: preprocessing, feature extraction, feature classification. Since trends in the EEG that define individual's mental state are localized in certain frequency bands, spectral filtering is necessary for better classification accuracy. Spatial filtering is a relatively novel technique that improves the spatial resolution of the EEG map. With use of spatial filtering algorithms, such as LLSF or CAR, it is possible to localize current electrical activity to its sources.

Most commonly real-time application are based on data segment analysis. For short segments it is possible, for example, to find autoregressive model coefficients and derive its evolution across time for classification. Response time is inversely proportional to accuracy in such applications.

2.1 EEG Preprocessing

Before features of the EEG signal can be extracted, the signal requires filtering. First step of the filtering process is the band pass filtering. Even though there are techniques to find optimal band frequencies for individual cases, the selection of the band frequencies is still an open question.

Most often, wide frequency range is chosen from 0 Hz to about 30 Hz. When individual frequencies are to be analyzed (e.g. alpha and gamma), a parallel filtering in those ranges can be implemented.

The slope of Bereitschafts potential, which indicates brain's preparation for motor task execution, is rather slow and stores its energy in low frequency harmonics. The task of Bereitschafts potential detection usually requires a high-pass filter to remove the DC offset of the EEG signal and a low-pass filter to remove the undesirable noise.

In general, the optimal frequency band used for the movement-related potential detection is individual for every person, type of EEG equipment used in the experimental setup, and the amount of noise. Therefore, the analysis of the impact of variations in both higher and lower bands on the output signal will be performed. Figures 2.1 – 2.4 depict the result of the EEG signal band filtering in different band.

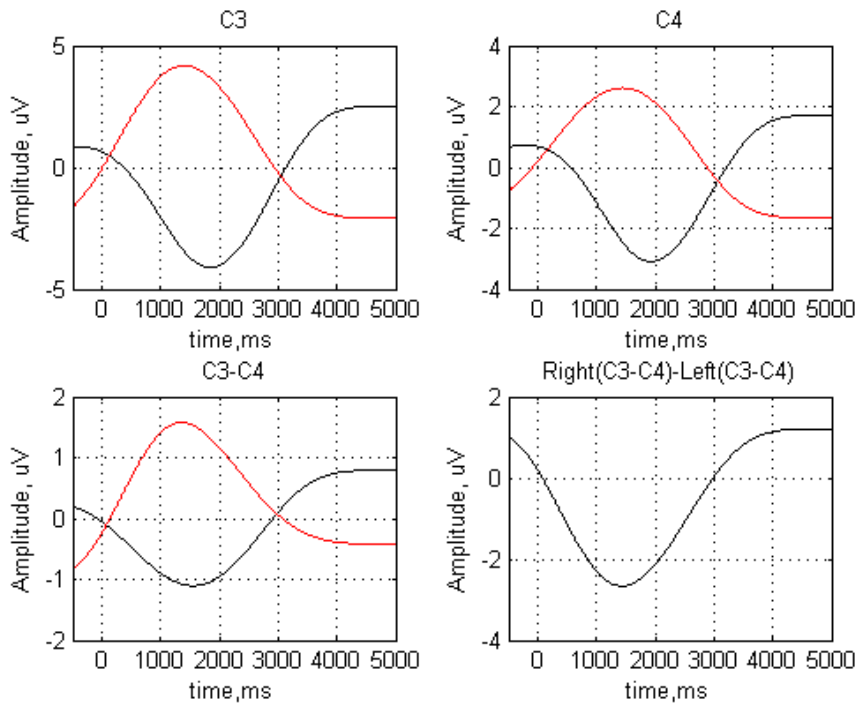


Figure 2.1 – Signal filtered in range 0.05 – 0.40 Hz. Black and right curves correspond to left and right hand movements, respectively

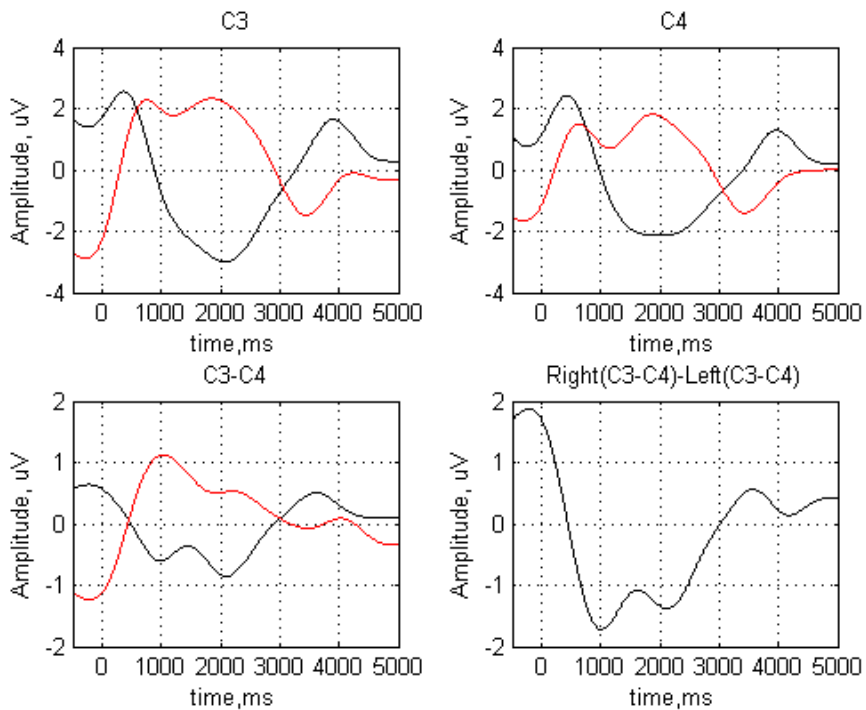


Figure 2.2 – Signal filtered in range 0.05 – 1.00 Hz. Black and right curves correspond to left and right hand movements, respectively

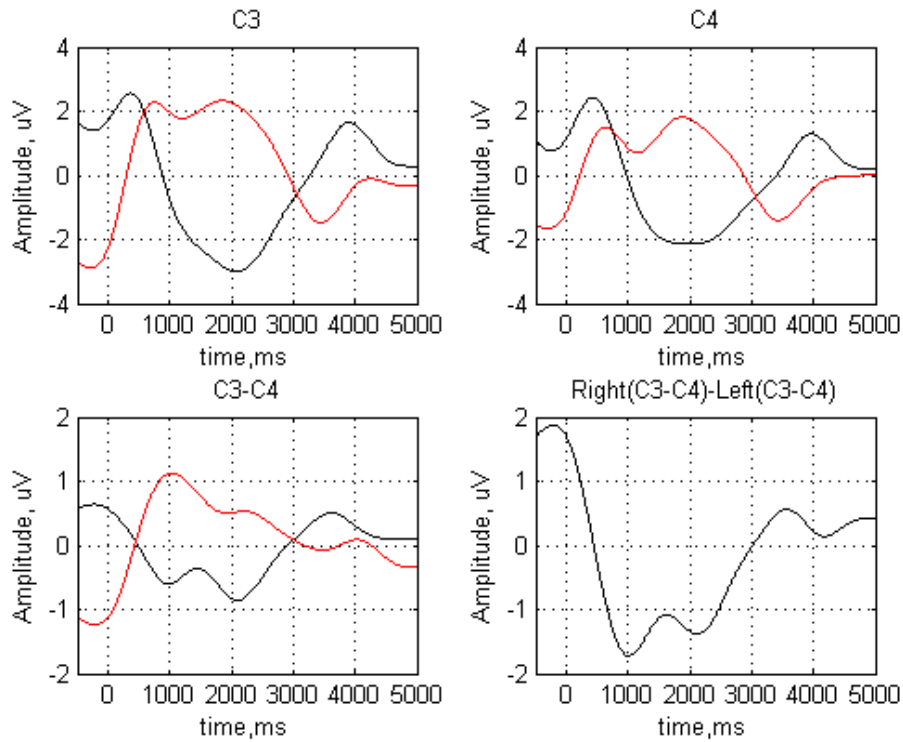


Figure 2.3 – Signal filtered in range 0.10 – 1.00 Hz. Black and right curves correspond to left and right hand movements, respectively

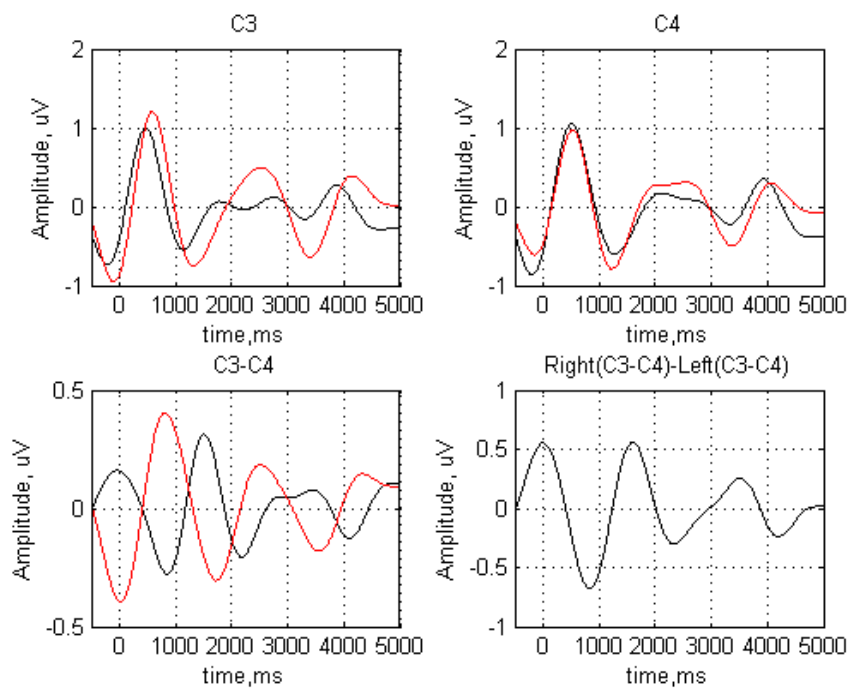


Figure 2.4 – Signal filtered in range 0.40 – 1.00 Hz. Black and right curves correspond to left and right hand movements, respectively

Figures 2.1 – 2.4 demonstrate an average of multiple epoch recordings filtered in different frequency bands. The epochs are averaged across hand movement classes and thus two curves are presented respectively for each class.

Decent spatial resolution of the EEG data is important because the target signal, for example the μ -rhythm, is weak and mixed with other strong signals of the same frequency. Raw EEG data provides poor spatial resolution due to their sparse spatial distribution.

Spatial filters are used to enhance the spatial resolution of the EEG signal. The output of a spatial filter applied to the channel under investigation is a surrogate channel with a better discriminative quality than the original one.

Selection of a certain type of spatial filter is based on the assumptions regarding the dynamics of the processes under investigation, such as the extent and location of the sources of the investigated potentials. The EEG data is highly affected by noise. Therefore, the choice of the spatial filter must be based on the understanding which type of spatial filter provides the highest signal-to-noise ratio and is able to result in better quality of the event classification. There are four main types of spatial filters: an ear reference, common average reference (CAR), small laplacian spatial filter (SLSF), and large laplacian spatial filter (LLSF) [11]. Those filters are presented in Figure 2.5.

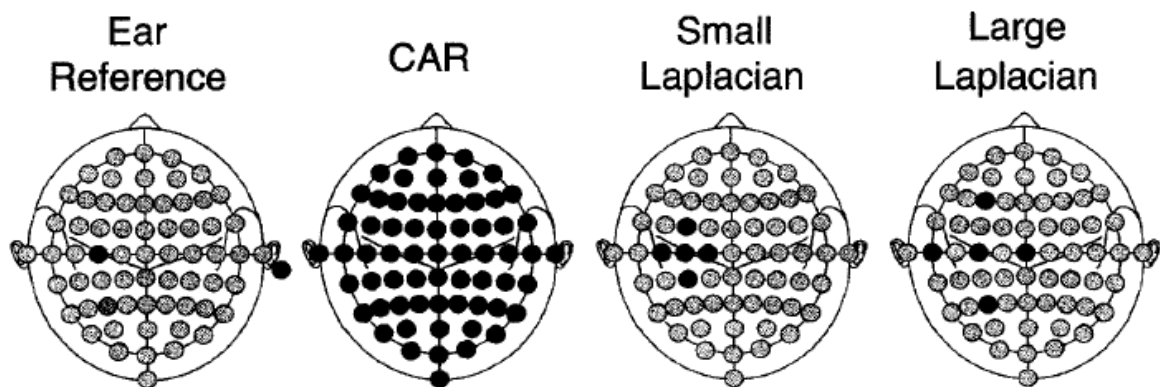


Figure 2.5 – Common types of spatial filters

The CAR filter subtracts the average of all electrodes in the montage from the investigated channel. If electrodes cover the entire head uniformly and the head potential is produced by point sources, the CAR filter produces a voltage spatial function with a zero mean. Since this assumption is not always met, the CAR filter usually results in nearly zero mean voltage distribution. The CAR filter accesses those components of

EEG signal that are present in a large group of electrodes and works as a high-pass spatial filter [11]. On the other hand, potentials that appear in most of the electrodes but absent in the electrode under investigation but undesirably appear in this electrode.

In the Laplacian filtering, the second derivative of voltage spatial function is calculated based on the data from adjoining electrodes. SLSF and LLSF access components that are present in most of the electrodes near the investigated one. The effect of LLSF on the raw EEG signal is presented in figures 2.6 and 2.7.

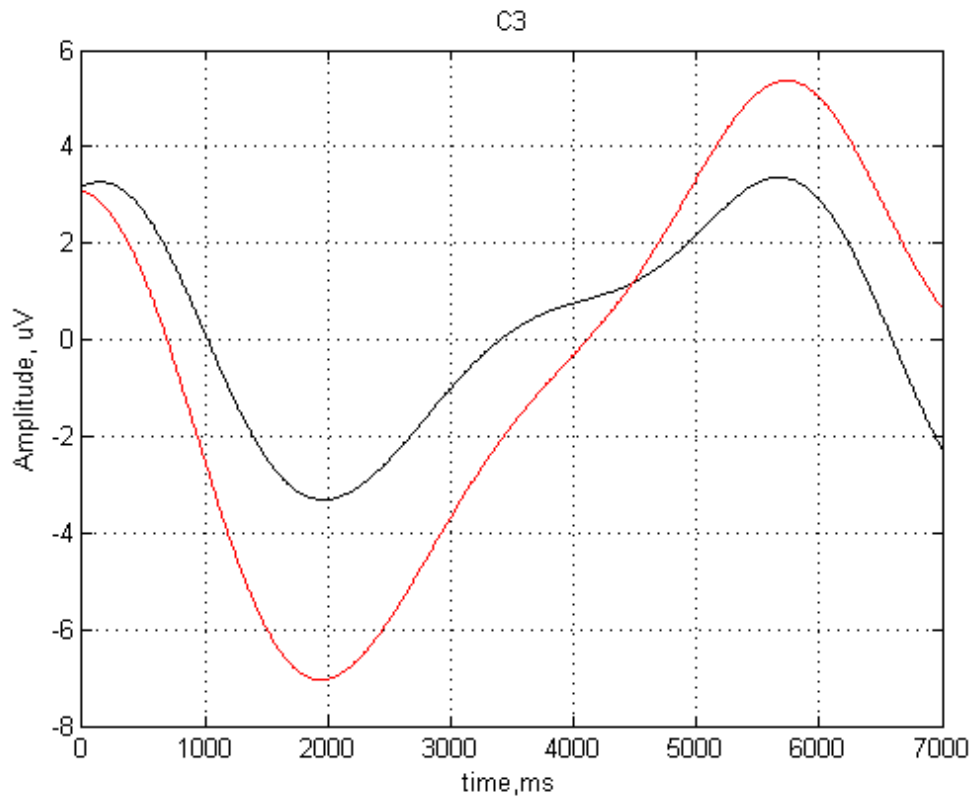


Figure 2.6 – Signal curve prior to LLSF applied to the channel

The laplacian and CAR filters are considered superior since they enhance the contribution from local potential sources to the surrogate channel signal. These results from that CAR, LLSF and SLSF are high-pass filters that reduce components present in most population of electrodes. The quality of the extracted potential depends on the relative position of the surrogate channel to the location of the potential source and the locations of noise sources.

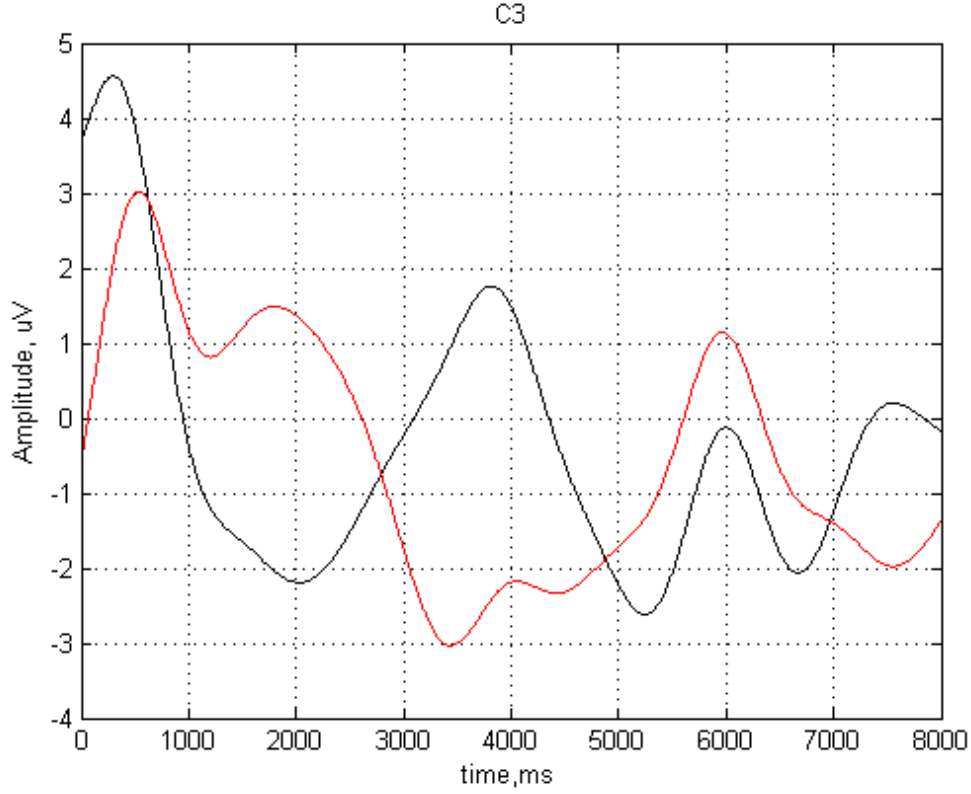


Figure 2.7 – Signal curve after LLSF applied to the channel

The Laplacian and CAR filter can be implemented in a real-time system as following:

$$V_i^{CAR} = V_i^{ER} - \frac{1}{n} \sum_{j=1}^n V_j^{ER}, \quad (2.1)$$

$$V_i^{LAP} = V_i^{ER} - \frac{1}{n(S_i)} \sum_{j \in S_i} V_j^{ER}, \quad (2.2)$$

where n denotes the number of electrodes in the montage, S_i denotes the set of nearest electrodes around the i -th electrode, and $n(S_i)$ denotes the number of items in S_i .

2.2 Detection of Lateralized Readiness Potential

Lateralized readiness potential is a feature of the EEG signal spatial distribution that can indicate that the brain starts preparing for the motor task execution. This feature can be extracted for various sources (e.g. left hand, right hand) depending on over

which part of the motor cortex the surrogate channel is located. Since the LRP signalizes the preparation for motor task execution, it is present even in case the intention to perform the action is cancelled, or if the person is physically incapable to perform the action. Therefore, the LRP detection algorithm is an eligible tool to be implemented in a BCI system.

2.2.1 Double subtraction

Hand and leg response movements always have an associated negative potential appearing in corresponding cortex region. This lateralized readiness potential (LRP) is exceptionally important since it starts even before the movement onset and is independent from whether the movement task was performed. The onset of LRP indicates the moment when brain starts the preparation for the execution of the action [10]. The LRP represents a lateralized activation of neurons, which is valuable for the task of imaginary movement classification.

The negative wave preceding execution of a motor task is BP which can be obtained through measurement with ear-lobe reference. Shortly prior to the action onset, the wave is biased to the hemisphere associated with the hand executing the action. According to Kutas and Donchin (1980), this wave indicates preparation for the execution of motor task. The technique used for separation of localized movement-related potentials from other lateralizations is called a double subtraction technique [10].

The double subtraction separates LRP from other lateralized and distributed components, as presented in Figure 2.8. The data recorded is EEG signals from electrodes at locations close to cortex areas associated with motor tasks used in the experiment. For right and left hand finger movements, the signals are recorded from standard locations C3 and C4 for both hand responses.

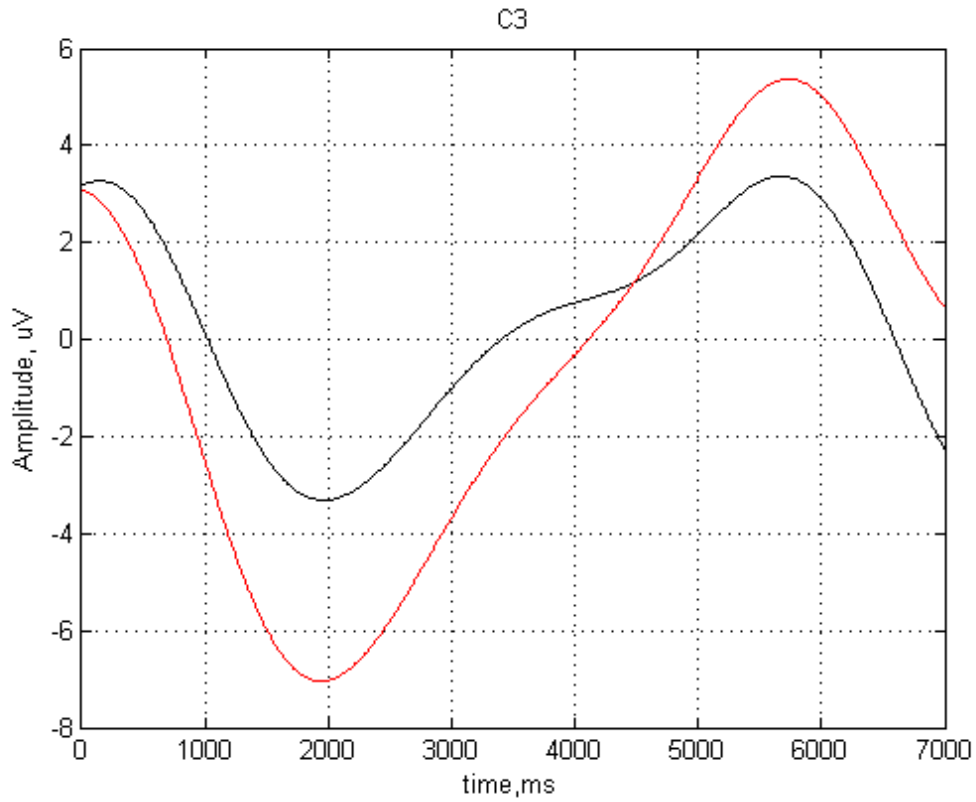


Figure 2.8 – Original signal from C3

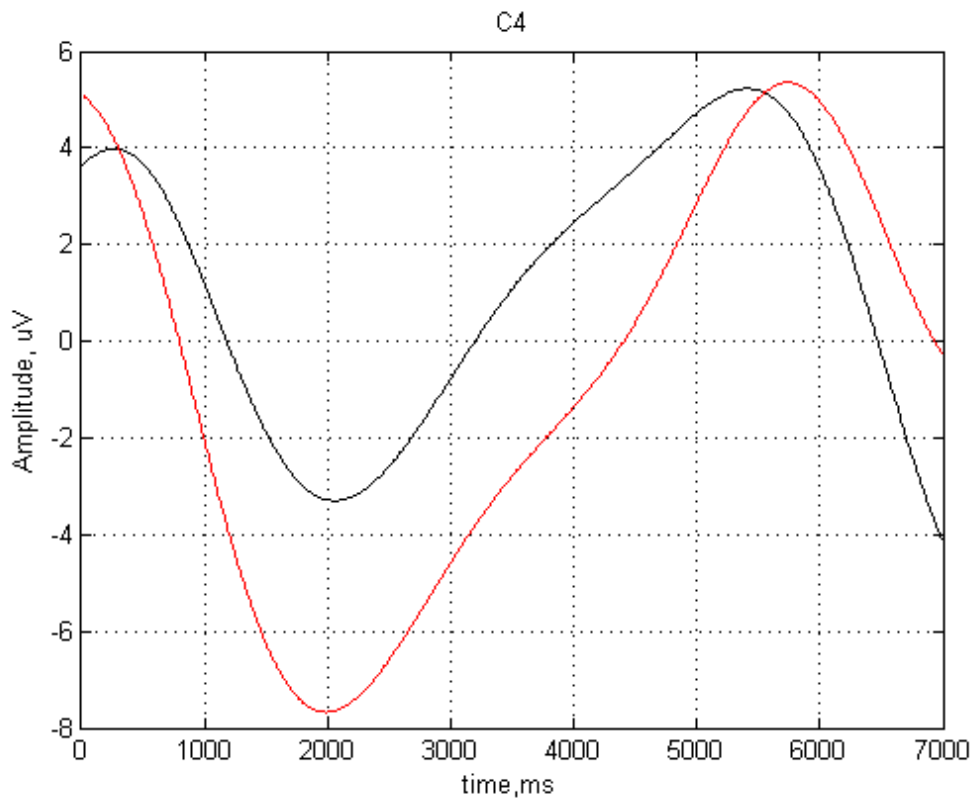


Figure 2.9 – Original signal from C4

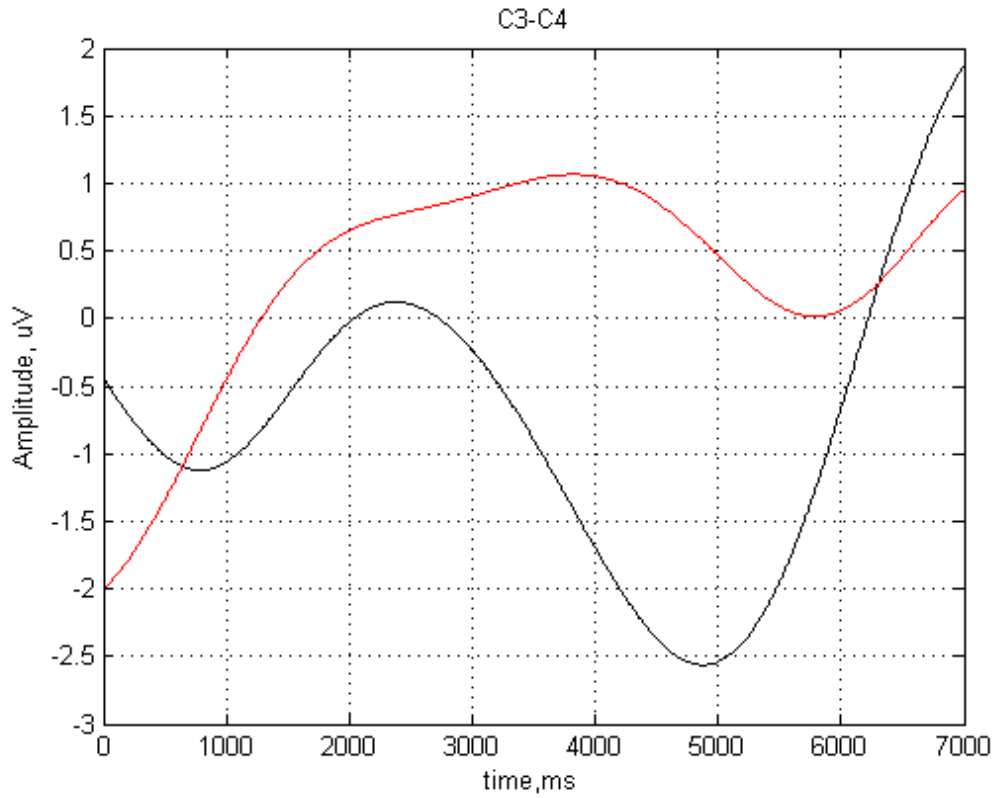


Figure 2.10 – First subtraction

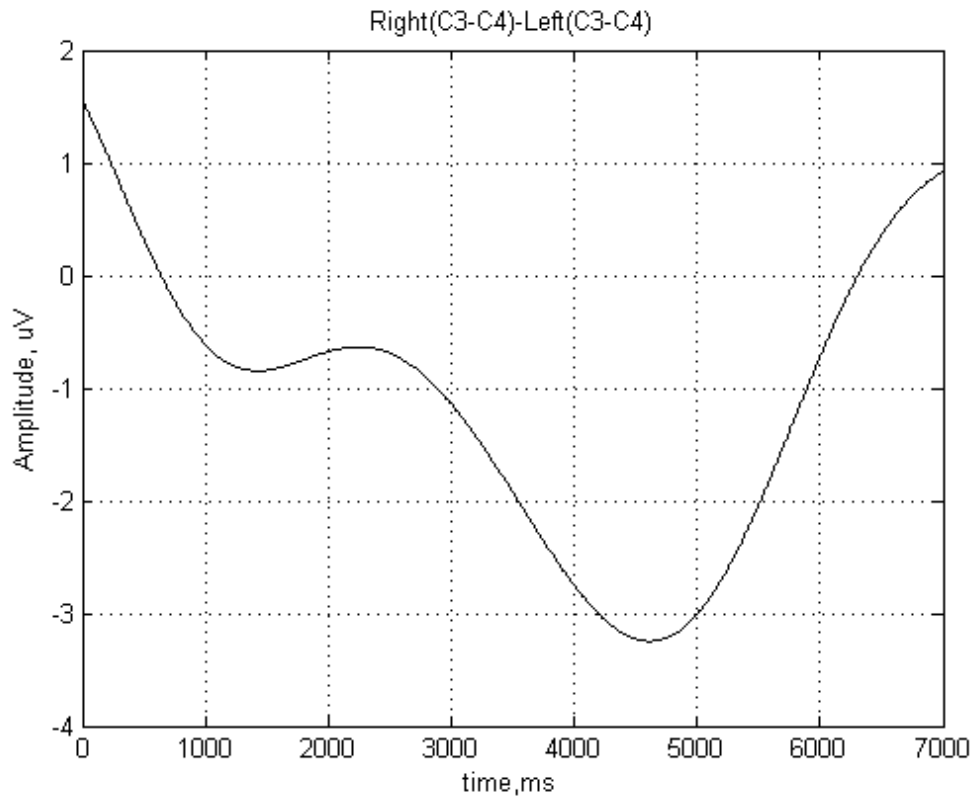


Figure 2.11 – Second subtraction

Further processing of the data requires it to be averaged across trials; typical averaged signal resulting from it is presented in figure 2.8. For the averaged data, the following discriminative expressions are derived:

$$\begin{aligned} & (C3 - C4), \text{ for right hand} \\ & -(C3 - C4), \text{ for left hand} \end{aligned} \quad (2.3)$$

or

$$\begin{aligned} & (C4 - C3)/2, \text{ for left hand} \\ & (C3 - C4)/2, \text{ for right hand} \end{aligned} \quad (2.4)$$

where C3 and C4 denote row matrices of samples of corresponding potentials.

After a right hand response the contralateral hemisphere electrode (C3) receives a preponderance in negative wave, however, the ipsilateral hemisphere electrode (C4) does not. Therefore, the value of the first expression in (2.3) will decrease. Similarly, output of second expression in (2.4) will decrease for left hand response. In both expressions from (2.4) ipsilateral hemisphere potential is subtracted from more negative contralateral hemisphere potential, thus copying the behavior of (2.3). All non-lateralized potentials that appear both in C3 and in C4 are cancelled out because they appear in both matrices with similar sign. Therefore, double subtraction technique takes into account only lateralized components; any preparation being reflected in both electrodes potentials is cancelled out.

2.2.2 Matched filter

Detection of LRP in surrogate channel signal relies on correlation of subsequences in that signal with given template extracted from the signal and is known to indicate the presence of LPR. The template is extracted from the initial negative phase associated with the voluntary movement execution. The implementation of matched filter is equivalent to the operation of convolution of the signal with time-reversed pattern of LPR. The filtering is performed as following:

$$y[n] = \sum_{k=-\infty}^{\infty} h[n - k]x[k]. \quad (2.5)$$

The output of matched filter represents the correlation of the signal with the template.

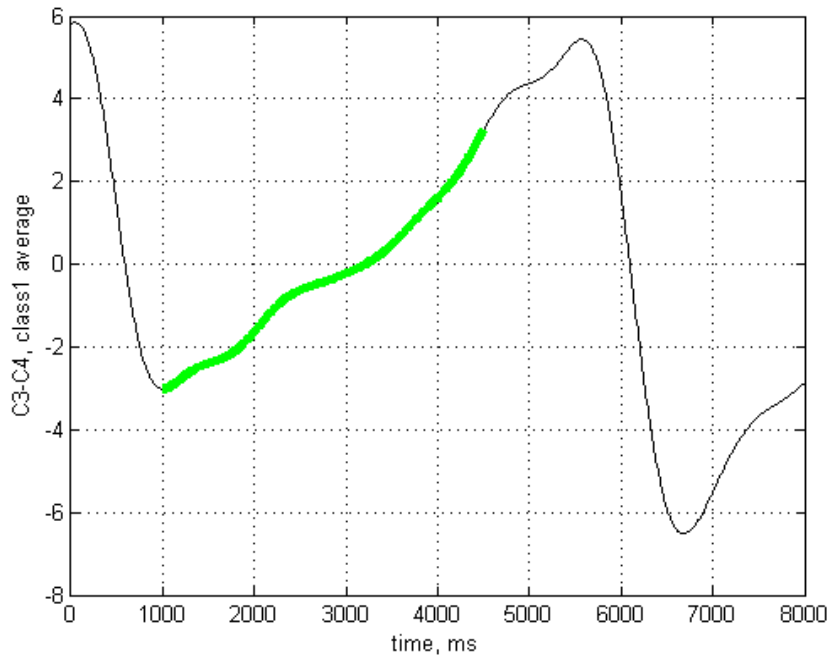


Figure 2.12 – LRP template extracted from averaged data

After the matched filter is applied to the original signal, a threshold can be used to indicate the occurrence of the searched event in the signal. Typical output of the filter and the implementation of threshold are presented in figure 2.13.

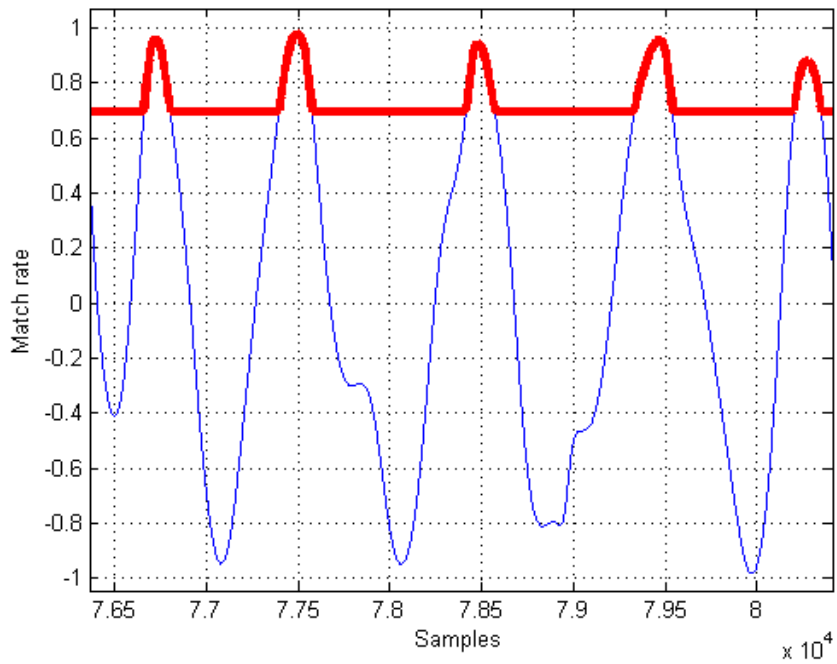


Figure 2.13 – Matched filter output with a threshold (red)

On order to limit the output of matched filter between -1 and 1 the pattern and the sub-signal are normalized according to:

$$X = X/\sqrt{XX^T}. \quad (2.6)$$

The use of matched filter provides the resemblance of the recorded signal to the template that represents the LRP. Moreover, matched filter minimizes the effect of noise in the filtered signal.

There are two considerations for the implementation of the matched filter in the classification system. First is to generate the best match for every window processes, thus considering the value as a temporal feature of the signal. Second is to plot the match against time and analyze not only the value itself, but also the fluctuation of the value in time. The second case is expected to provide better results, however it requires more computational resources and can be challenging to implement within a real-time BCI system.

2.3 Common Spatial Pattern

Classification features can be extracted from EEG using the Common spatial pattern (CSP) method. Studies have shown that motor imagery results in the rise of local neural oscillation called Event-related desynchronization (ERD) or in the reduction of it called Event-related synchronization (ERS) [17]. ERD and ERS are presented in Figure 2.14.

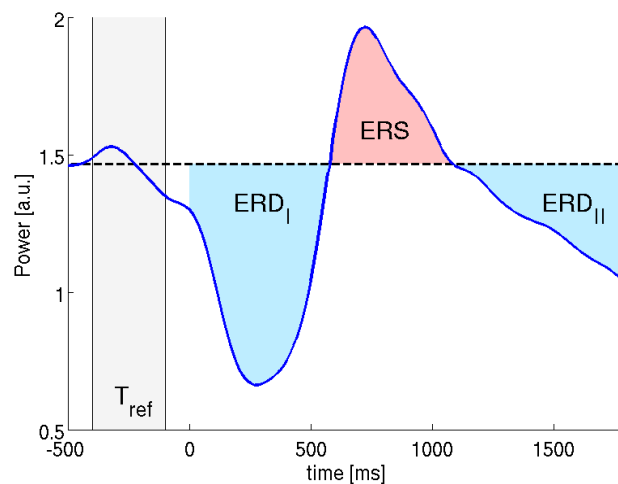


Figure 2.14 – ERD and ERS as deviations from average power calculated prior to the event

Event-related desynchronization (ERD) event can be detected successfully with use of CSP [9]. As a result, CSP produces new time series that are optimal for the classification between two EEG events. This method comprises a parallel diagonalization of two covariance matrices.

Let $R_a(i)$ and $R_b(i)$ denote the i -th trial spatial covariance matrices from classes a and b, respectively.

$$\begin{aligned} R_a(i) &= X(i)X^T(i); \quad i = 1:n_1, \\ R_b(i) &= X(i)X^T(i); \quad i = 1:n_2, \end{aligned} \quad (2.7)$$

where n_1 and n_2 respectively correspond to the number of trials in classes a and b and $X(i)$ is the $N \times T$ matrix of EEG recording for i -th trial. N and T correspond to the number of channels and the number of samples, respectively.

The following equations are used to calculate the averaged normalized covariance matrices:

$$\begin{aligned} R_a &= \frac{1}{n_1} \sum_{i=1}^{n_1} R_a(i), \\ R_b &= \frac{1}{n_2} \sum_{i=1}^{n_2} R_b(i). \end{aligned} \quad (2.8)$$

The composite covariance matrix is:

$$R = R_a + R_b. \quad (2.9)$$

Because R is symmetric, it can be decomposed as follows:

$$R = U_0 \Lambda_c U_0^T, \quad (2.10)$$

where U_0 comprises the characteristic vectors of R and Λ_c is a diagonal matrix that comprises characteristic values of R . T in U_0^T denotes the transpose operation.

The whitening transformation of R is performed with the following equation:

$$P = \Lambda_c^{-\frac{1}{2}} U_0^T. \quad (2.11)$$

Individual covariance matrices R_a and R_b can be transformed as follows:

$$\begin{aligned} S_a &= P R_a P^T, \\ S_b &= P R_b P^T. \end{aligned} \quad (2.12)$$

S_a and S_b have equal characteristic vectors, therefore:

$$\begin{aligned} S_a &= U \Psi_a U^T, \\ S_b &= U \Psi_b U^T. \end{aligned} \quad (2.13)$$

If $S_a + S_b = I$, then $\Psi_a + \Psi_b = I$. The characteristic vector that includes the largest characteristic value for S_a includes the smallest characteristic value for S_b and vice versa.

Matrix $W_{CSP} = U^T P$ produces the projection from $X(i)$ to $Z(i)$ with the following equation:

$$Z(i) = W_{CSP}X(i). \quad (2.14)$$

In the end, the equation for the variances of the signal processed with CSP for the first and last r rows in Z :

$$f^p(i) = var(Z^p(i)); \quad p = 1, \dots, r, n - r + 1, n - r + 2, \dots, n. \quad (2.15)$$

where n corresponds to the number of rows in Z and $Z^p(i)$ is the p -th row in $Z(i)$.

In this work, r is assigned with the typical value of 1 [13].

Figure 2.15 show a scatter plot of SCP signal for two most significant channels. The blue stars and red circles depict averaged epoch samples for class 1 and class 2, respectively. The CSP linear transform resulted in maximized variance in the averaged data between those two classes.

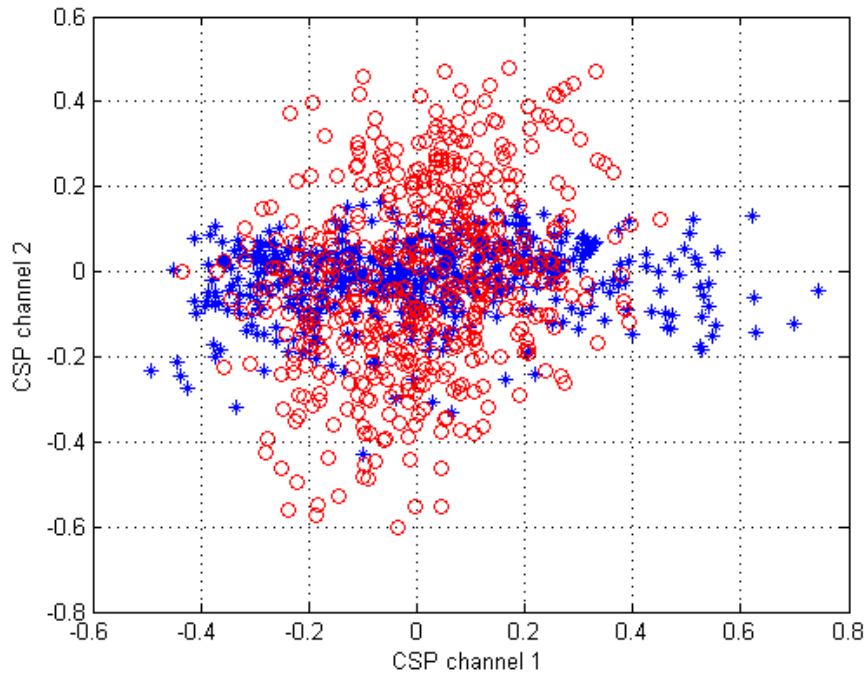


Figure 2.15 – Single-trial CSP signal scatter plot for two most discriminative channels

2.4 Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is method that constructs a linear combination of features that result in the best separation between classes. LDA projects input

data on a lower-dimension space in such a way to maintain the class-discriminatory information. The use of LDA in a BCI system significantly reduces the computational costs.

The goal of LDA is to create a feature subspace projection of a d-dimensional data X onto which provides the optimal separation within the new data Z, as presented in Figure 2.16. The linear transform is defined with matrix ω representing the subspace:

$$Z = \omega^T X. \quad (2.16)$$

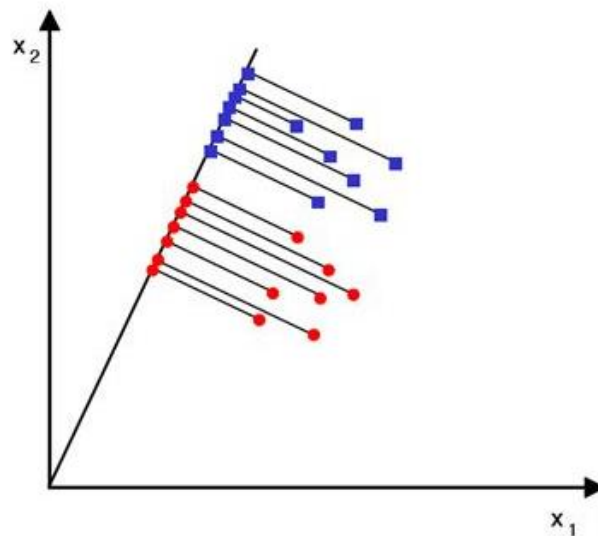


Figure 2.16 – Feature subspace providing the optimal between-class separation

One way to construct the optimal subspace is to implement the gradient descent algorithm. The cost function, therefore, is defined as following:

$$J(\omega) = \frac{(m_1 - m_2)^2}{s_1 + s_2}, \quad (2.17)$$

where m_1 and m_2 denote the averages within classes 1 and 2, respectively, and s_1 and s_2 denote the variances within classes 1 and 2, respectively.

The value of the cost function (2.17) has to be maximized over the subspace space ω in order to find the solution $\hat{\omega}$ representing the optimal subspace.

The expression (2.17) can be transformed as follows:

$$\begin{aligned} (m_1 - m_2)^2 &= (\omega^T \mu_1 - \omega^T \mu_2)(\omega^T \mu_1 - \omega^T \mu_2)^T = \\ &= \omega^T (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T \omega = \omega^T S_B \omega, \\ s_1 + s_2 &= \omega^T \Sigma_1 \omega + \omega^T \Sigma_2 \omega = \omega^T (\Sigma_1 + \Sigma_2) \omega = \omega^T S_w \omega, \end{aligned} \quad (2.18)$$

The cost function (2.17) then looks like:

$$J(\omega) = \frac{\omega^T S_B \omega}{\omega^T S_W \omega}. \quad (2.19)$$

After calculation of the optimal projection $\hat{\omega}$ the optimal threshold z_0 for classification may be chosen. The classification algorithm is defined as following:

$$\begin{aligned} x \in \{class1\}, \omega^T x &\geq z_0 \\ x \in \{class1\}, \omega^T x &< z_0 \end{aligned} \quad (2.20)$$

For example, a common solution is $z_0 = (m_1 + m_2) / 2$.

2.5 Data classification

Features extracted from the investigated temporal window are quantities to represent a particular control state. Fit values extracted by matched filter as well as temporal features of the surrogate channel signal are considered as points in a multivariate space. The task of EEG event detection and classification is to divide this space into regions representing feature values combinations corresponding to the occurrence of certain events (e.g. dormant state, left hand movement, right hand movement). The subject of this section is to investigate the types of universal classifiers applicable for solving this task and to select the preferable one.

The types of classifiers considered are multilayer perceptron, radial basis function network, and support vector machine. The neural network approach is most common for the task of statistical data classification, while support vector machine is rather new tool and also proved to result in decent precision [6].

2.5.1 Multilayer perceptron

An artificial neural network consisting of layers of neurons that result in one-sided propagation of signal is known as a multilayer perceptron (MLP). Every neuron in network, except for input neurons, is a processing unit that applies a nonlinear activation function to the weighted sum of its inputs. In a scope of EEG signal feature processing and classification, MLP most importantly provides the non-linear separation of the input data. The structure of MLP is presented in Figure 2.17.

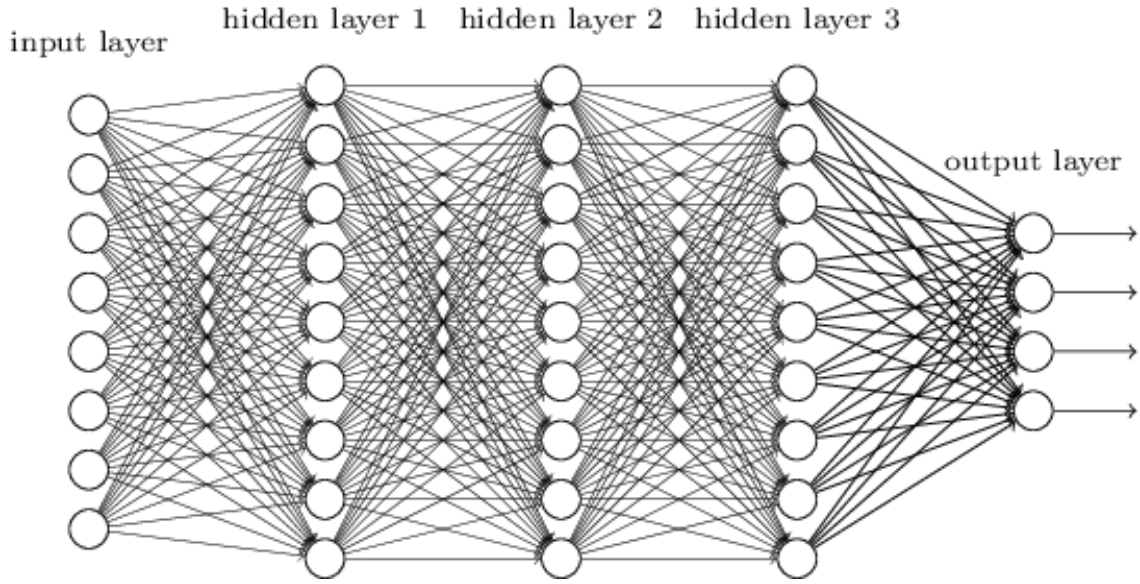


Figure 2.17 – Structure of multilayer perceptron

Each neuron consists of input nodes x_i , input weights w_i , bias b , and an output node y . Relation between them can be written as follows:

$$y = \varphi(\sum_{i=1}^n w_i x_i + b) = \varphi(\mathbf{w}^T \mathbf{x} + b), \quad (2.21)$$

where φ denotes the nonlinear activation function of the neuron.

The ability of the perceptron to match input vectors to the corresponding output vectors are obtained by adjusting weights of each neuron in the network. The most common method of MLP learning is the backward propagation of errors, or backpropagation.

Backpropagation calculates the partial derivatives of a loss function with respect to the weights of the neurons in the network and iteratively optimizes the weights in order to minimize the loss function. Backpropagation uses known output for each input vector in the training set to perform this iterative optimization, therefore it is considered to be a supervised learning algorithm. By moving from output to input across the network backpropagation iteratively calculates derivatives for each layer, which is a generalization of the delta-rule. This algorithm assumes the neuron activation function has finite values of its first derivative in all points.

The loss function used in minimization task represents the cumulative distance of all output components from the desired values:

$$E(n) = \frac{1}{2} \sum_j e_j^2(n), \quad (2.22)$$

where $e_j(n) = d_j(n) - y_j(n)$ is the error between desired \mathbf{d} and actual \mathbf{y} values produced by perceptron.

According to the gradient descent method, the weight adjustment is calculated as:

$$\Delta w_{ji}(n) = -\eta \frac{\partial E(n)}{\partial w_{ij}}, \quad (2.23)$$

Where η is a learning ratio and w_{ij} is a weight value between i-th neuron from layer k-1 and j-th neuron from layer k.

2.5.2 Radial basis function network

Radial basis function network (RBFN) is a particular type of artificial neural networks that implements a radial basis function as activation function. Output of RBFN is interpreted as a linear combination of activation functions applied to the inputs of RBFN. The Gaussian function, the most common activation function of RBFN, is presented in Figure 2.18.

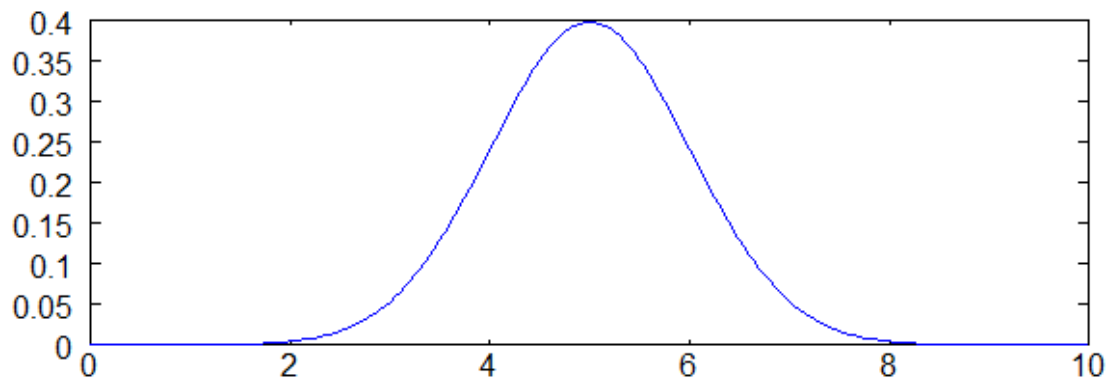


Figure 2.18 – Gaussian activation function

The Gaussian function produces a value that depends on the distance between input vector and the center of the neuron function. This function can be calculated as follows:

$$\rho(\|x - c_i\|) = \exp[-\beta\|x - c_i\|^2]. \quad (2.24)$$

Most commonly, there are three layers in an RBFN. First layer performs routing task – it distributes input vector components between neurons of the second layer. Second layer comprises neurons with radial basis activation function. Finally, the third layer includes neurons with linear activation function; the only purpose of this layer is to generate a weighted combination of the values produced by the radial basis function neurons.

2.5.3 Support vector machine

Support vector machine (SVM) is a data processing model with its own supervised learning algorithms that are used for classification purposes. The SVM learning algorithms build a model based on a given training set to separate the input instances into two classes. After that, the SVM can predict the class to which the new data belongs. SVM considers instances as points in multidimensional space and classifies them based on a hyperplane constructed by learning algorithms (Figure 2.19).

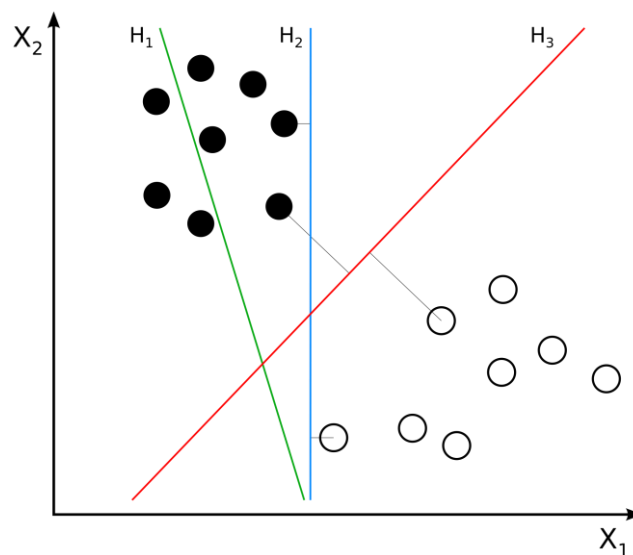


Figure 2.19 – Data separation with maximum margin

3 CONTROL OF A MOBILE ROBOT

To physically model a wheelchair used to provide a disabled subject mobility a mobile robot is used. The following section discusses the means by which the robotic system is implemented and interface to control the system is established. EEG headset that is used within the system is also described below.

3.1 Mobile robot

The goal of the EEG-based BCI is to allow a disabled person to control a wheelchair. In order to evaluate the BCI's performance a mobile robot was used to physically model a wheelchair system. The robot is presented in Figure 3.1.



Figure 3.1 – Mobile robot used in the work

Microcontroller used for robot actuation is presented in Figure 3.2. The microcontroller is ATxmega128B1; it provides PWM, USART, and is equipped with an LCD display for visual feedback.



Figure 3.2 – Microcontroller used for robot control

The robot includes two servomotors for the control of front and rear wheels angular position control. The servomotors receive PWM signals from microcontroller with the duty cycle in range from 30% to 100%. Middle point 60% corresponds to the neutral position of wheels. One of the servomotors is presented in Figure 3.3.

The movement is produced with a DC motor that is also controlled with PWM with same duty cycle ranges. Middle point for DC motor corresponds to zero velocity. The motor is presented in Figure 3.4.

The microcontroller implements uC-OS II operating system [8] with a program executing reading incoming USART messages, editing of the DC motor and servo drive references, and applying the PWM signals to corresponding ports of the microcontroller.

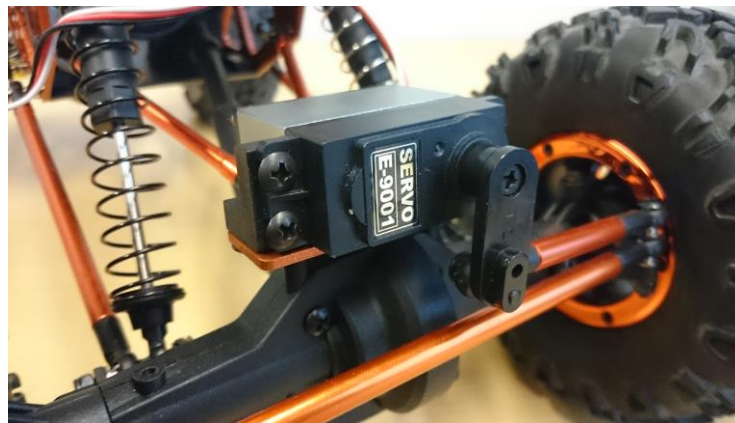


Figure 3.3 – Steering servomotor



Figure 3.4 – Driving DC motor

Reference values for PWM channels are transmitted to the microcontroller from PC via serial connection in a form of messages. According to the device protocol, the command to rewrite all reference values consists of:

1. Request byte (0x7F),
2. Set all references byte (0x06),
3. First channel reference low byte,
4. First channel reference high byte,
5. Second channel reference low byte,
6. Second channel reference high byte,
7. Third channel reference low byte,
8. Third channel reference high byte,
9. Fourth channel reference low byte,
10. Fourth channel reference high byte,
11. Cyclic redundancy check.

The data is transmitted to the controller via XBee modules that implement ZigBee communication protocol. An XBee module is presented in Figure 3.5. ZigBee modules consume less power than Bluetooth or Wi-Fi ones and are cheaper.

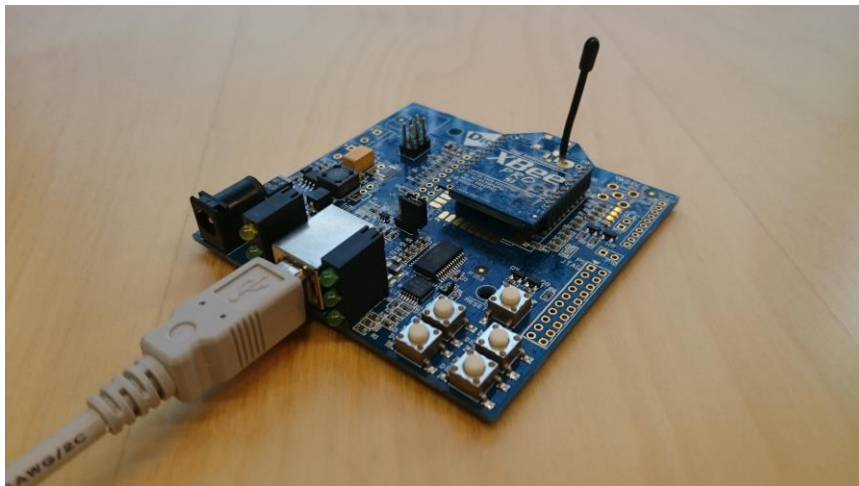


Figure 3.5 – XBee wireless module

Further, an interface of direct control of a mobile robot directly from MATLAB environment is established.

3.2 Neuroelectrics Enobio

Neuroelectrics Enobio headset is the equipment by which the EEG signal is recorded and transferred into MATLAB environment. The headset is presented in Figure 3.6. The headset provides a 32-channel EEG recording with decent signal-to-noise ratio and high recording frequency.



Figure 3.6 – Neuroelectrics Enobio EEG headset

The Neuroelectrics Instrument Controller (NIC) software provides a Matlab plugin with which it is possible to operate the recording and marker placement during training and simulation sessions. The data is stored in a form of .easy file that can be accessed and recognized as a Matlab variable with EEGLAB toolbox.

The main window of NIC is presented in Figure 3.7.

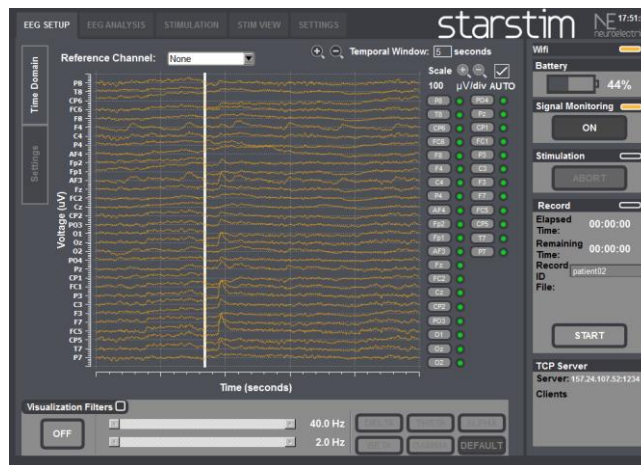


Figure 3.7 – Neuroelectrics Instrument Controller main window

MatNIC plugin for Matlab (MatNIC) provided by Neuroelectrics allows communication between Matlab environment from where the system is controlled and Neuroelectrics Enobio. This is possible because COREGUI includes a TCP/IP server for clients to control the application remotely. This plugin provides a set of Matlab functions to connect to this TCP/IP server and remotely operate the NIC software.

Among those functions are:

- MatNICConnect – Connect to COREGUI TCP/IP server,
- MatNICStartEEG – Start EEG streaming,
- MatNICMarkerConnectLSL – Make a connection to COREGUI with a certain stream. This allows to send markers to COREGUI from Matlab,
- MatNICMarkerSendLSL – Send a marker to COREGUI.

3.3 Robot-to-Matlab interface

Write about Qt program that was developed, as well as about TCP/IP protocol

The interface to control the robot directly from the MATLAB environment was developed with use of Qt framework. The communication diagram using this interface is presented in figure 3.8.

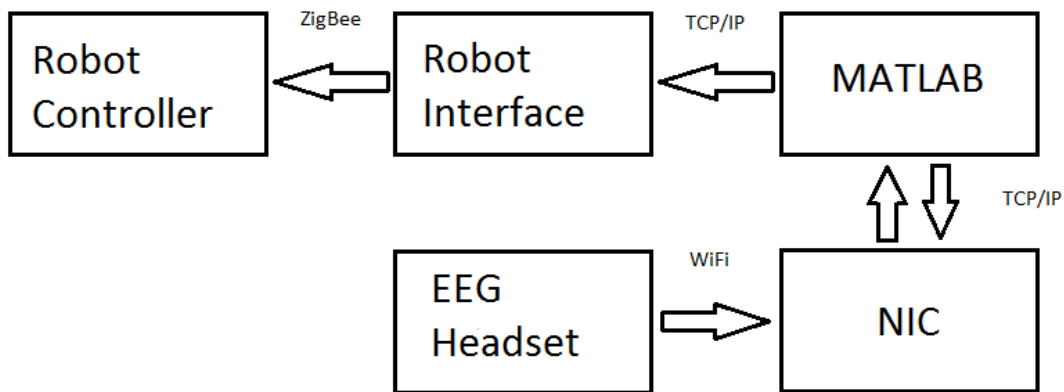


Figure 3.8 – Communication diagram

The interface runs a local TCP/IP server providing the MATLAB environment with a function to connect to and transfer reference values for robot drives. The block diagram for the MATLAB function *updateReference.m* is presented in figure 3.9.

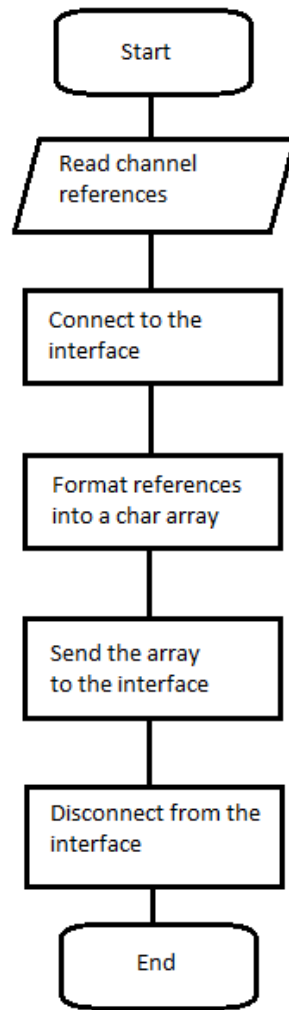


Figure 3.9 – Block diagram for *updateReference.m*

The program GUI presented in figure 3.10 includes means to connect to a serial port and buttons to control the robot manually.

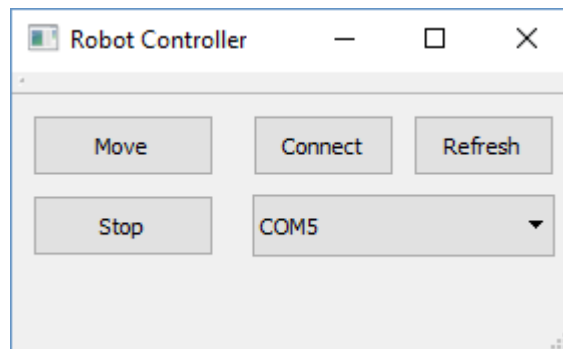


Figure 3.10 – Mobile robot software interface GUI window

The reference values for motor drives are stored as unsigned 16-bit variables and are updated after the message reception from MATLAB. The message must contain the values for channels in text format separated by space characters.

4 RESULTS

The performance of the BCI algorithm will be evaluated based on a 2-fold cross validation based on IVa datasets from BCI competition III. The results will be compared with ones provided by existing approaches. The robot-computer interface will be established to provide possibility to practically implement the BCI.

4.2 BCI algorithm

This work uses the IVa datasets provided by “BCI Competition III” to estimate the BCI algorithm performance. The BCI is constructed with implementation of the following methods:

- Butterworth band-pass filter;
- CSP;
- Matched filter;
- Katz fractal dimension;
- Kolmogorov entropy.

Butterworth 5-th order band-pass filter was used for signal preprocessing. For each subject the corresponding frequency band was determined using genetic algorithm. Matched filter template was extracted from the CSP most discriminative channel. Template indices were determined using genetic algorithm as well with respect to the cross-validation accuracy.

Parameters for genetic algorithm index estimation were chosen as follows:

- Population size – 100;
- Number of generations – 10;
- Creation function – Feasible population;
- Fitness scaling function – Rank;
- Selection function – Stochastic uniform;
- Mutation function – Adaptive feasible.

Since genetic algorithm is an approach to find the global minimum of the function, the cross-validation precision with the opposite sign was used as the cost function to make target values correspond to the cost function global maximum.

The indices selected using the genetic algorithm are:

- Start index: 205;
- End index: 344.

The indices correspond to 408ms and 686ms offsets, respectively.

Based on those techniques a feature vector was constructed for the future classification. An instance of the feature vector consists of 20 items and is constructed as follows:

- CSP variance of channel #1;
- CSP variance of channel #2;
- CSP variance of channel #3;
- CSP variance of channel #4;
- CSP variance of channel #5;
- CSP variance of channel #6;
- Katz fractal dimension measure of channel #1;
- Katz fractal dimension measure of channel #2;
- Katz fractal dimension measure of channel #3;
- Katz fractal dimension measure of channel #4;
- Katz fractal dimension measure of channel #5;
- Katz fractal dimension measure of channel #6;
- Kolmogorov entropy of channel #1;
- Kolmogorov entropy of channel #2;
- Kolmogorov entropy of channel #3;
- Kolmogorov entropy of channel #4;
- Kolmogorov entropy of channel #5;
- Kolmogorov entropy of channel #6;
- Matched filter best match of channel #1, pattern for class #1;
- Matched filter best match of channel #1, pattern for class #2;

The channels 1 to 6 are the first 6 channels selected from the CSP output.

Figures 4.1 – 4.7 show signals extracted for subject “al” from the IVa dataset.

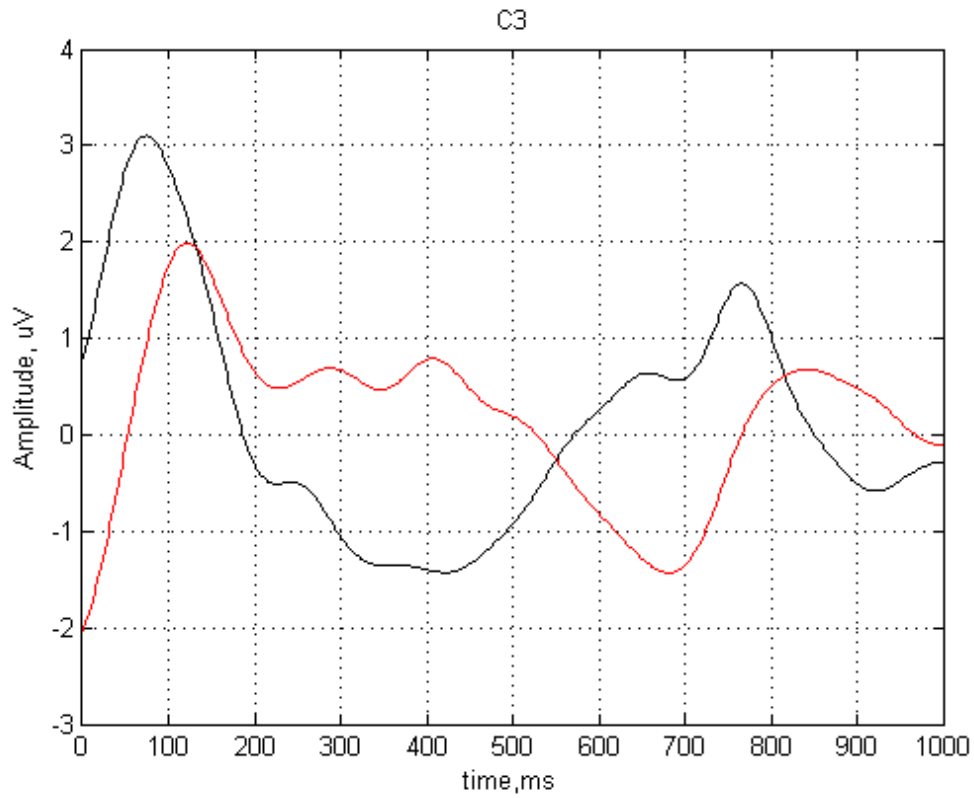


Figure 4.1 – Averaged trial epoch for channel C₃

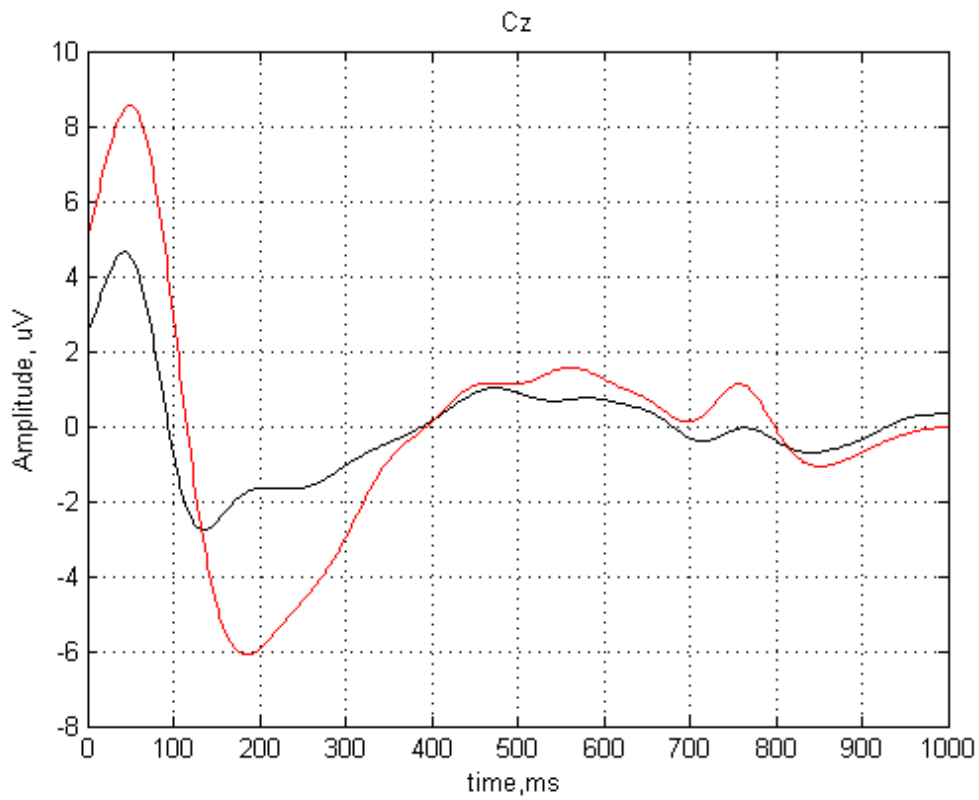


Figure 4.2 – Averaged epoch for channel C_z

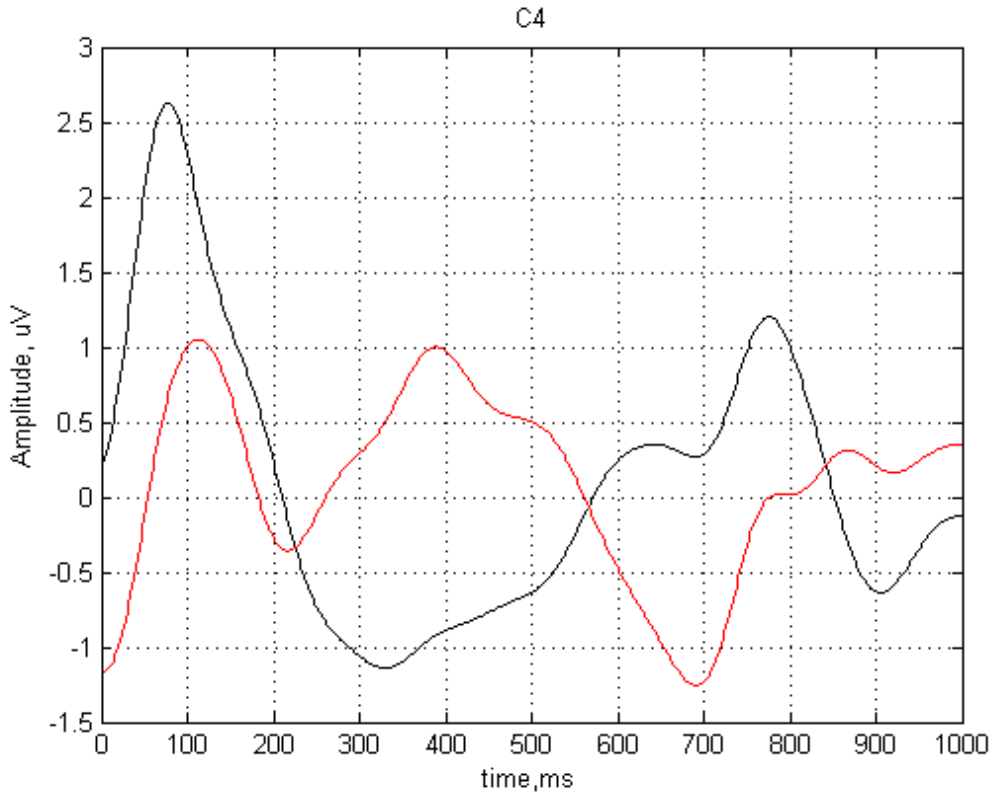


Figure 4.3 – Averaged epoch for channel C₄

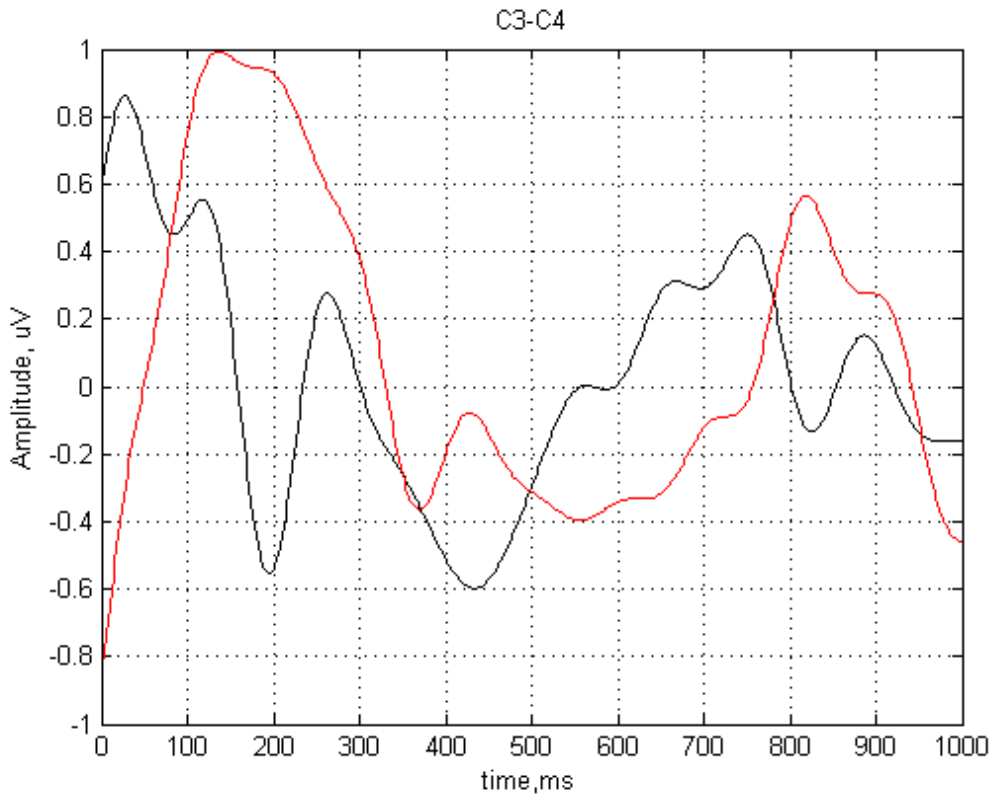


Figure 4.4 – Averaged epoch for surrogate channel C₃-C₄

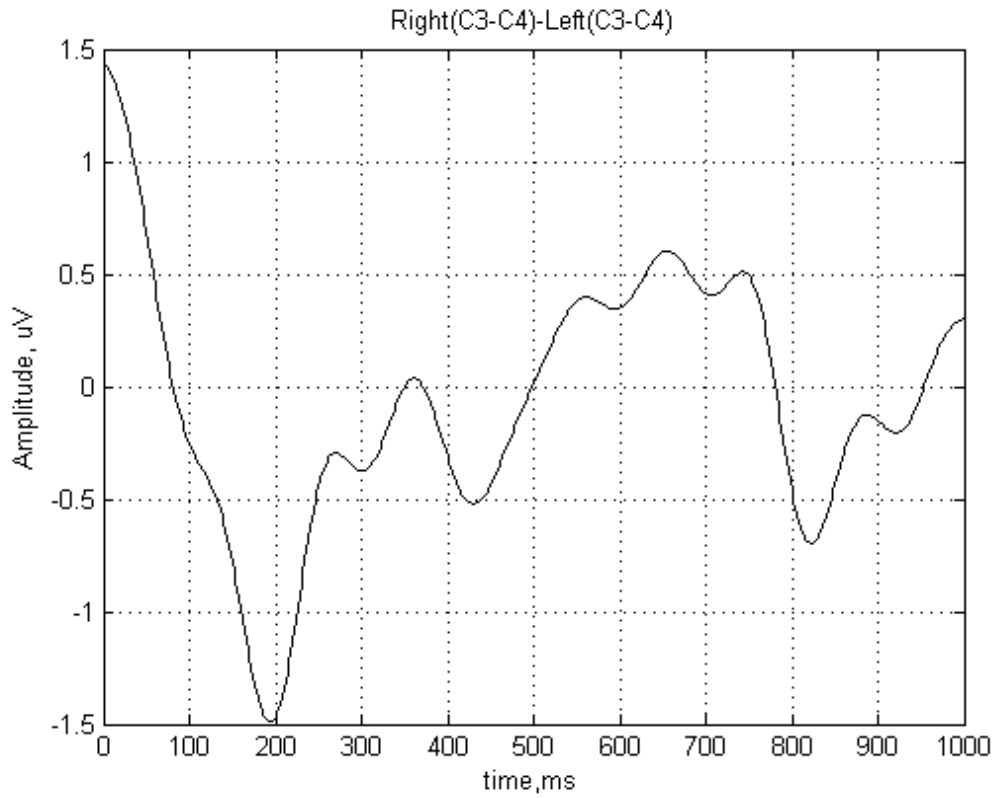


Figure 4.5 – Difference between average epochs for both classes

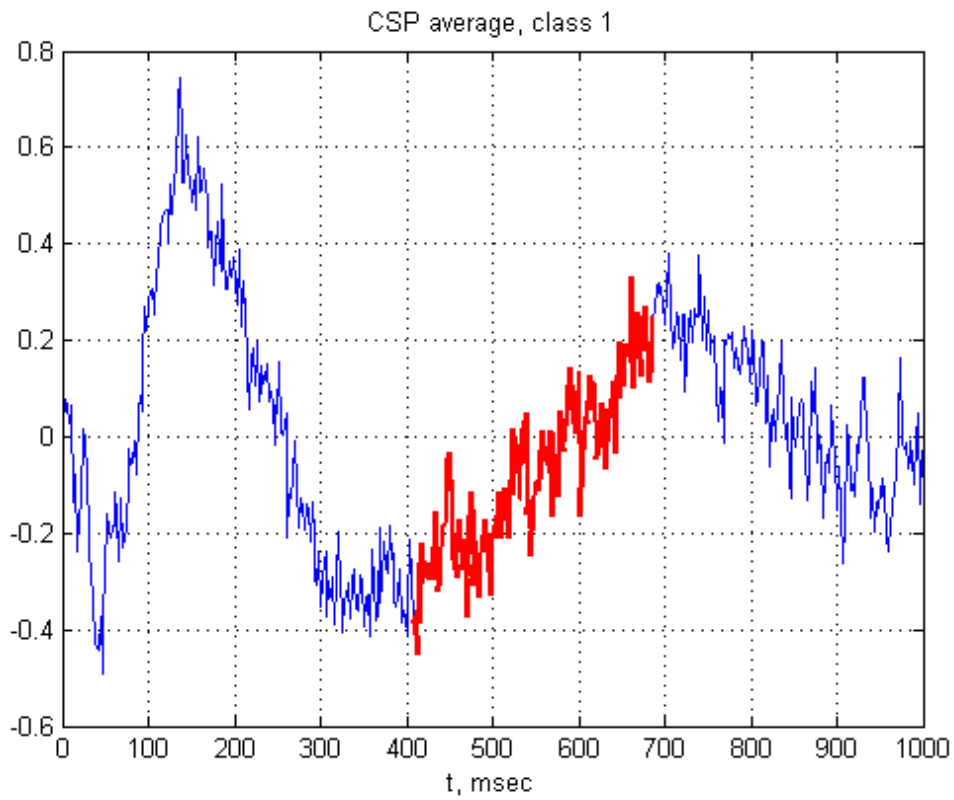


Figure 4.6 – Matched filter template extracted for class 1

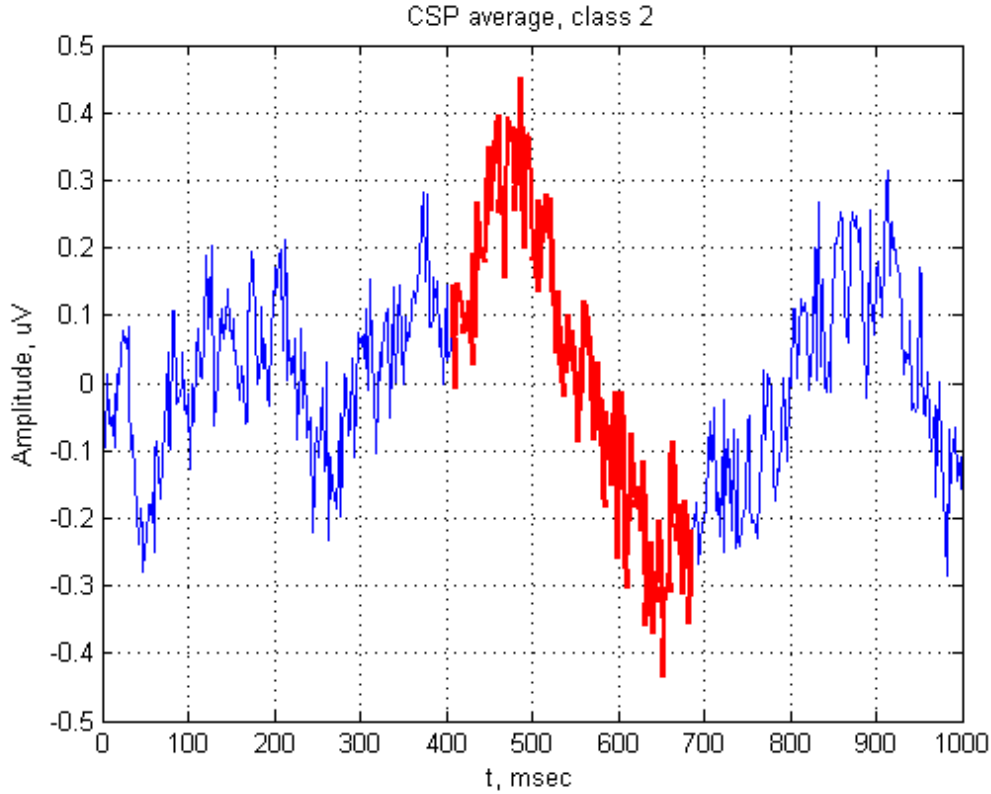


Figure 4.7 – Matched filter template extracted for class 2

The classification of features was performed with use of a feedforward neural network with the following parameters:

- Number of layers: 2;
- Number of neurons in first layer: 10;
- Number of neurons in second layer: 4;
- Activation function: hyperbolic tangent sigmoidal;
- Bias node: yes;
- Training algorithm: Levenberg-Marquardt algorithm;
- Performance measure: Mean squared error.

Figures 4.8 – 4.10 are the scatter plots showing the distribution of three most significant features extracted from the training and validation datasets. The significance of features was measured as a sum across all pairs between two classes of squares of normalized feature distances:

$$Q_i = \sum_{j=\{class1\},k=\{class2\}} (F_{i,j}/\max(\{|F_i|\}) - F_{i,k}/\max(\{|F_i|\}))^2. \quad (4.1)$$

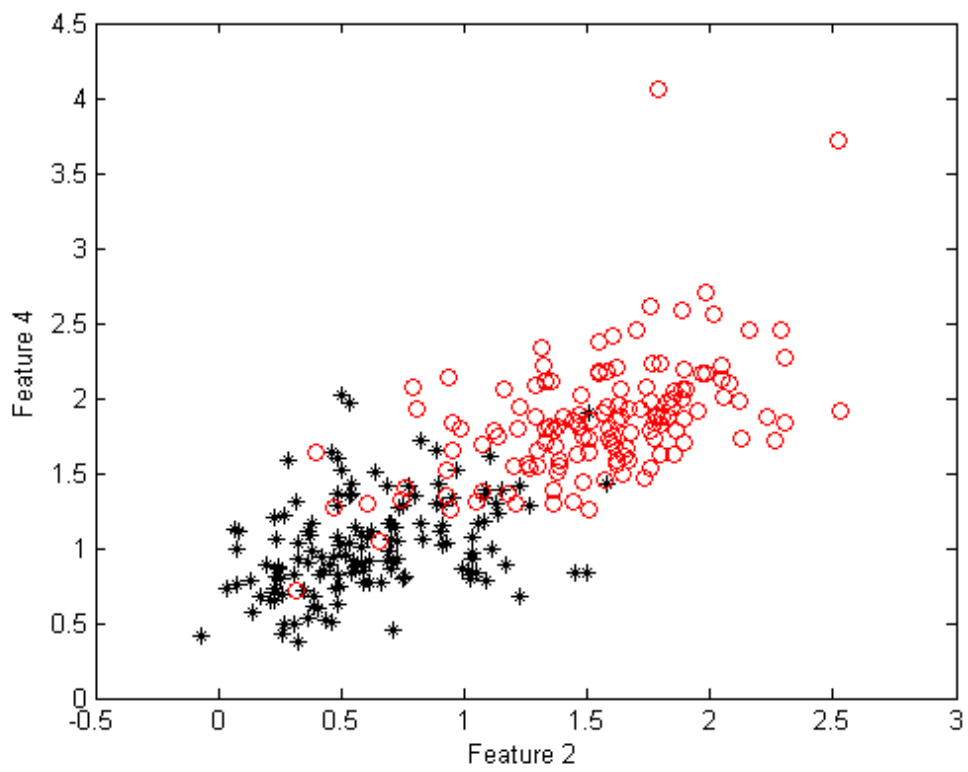


Figure 4.8 – Scatter plot for classification features 2 and 4

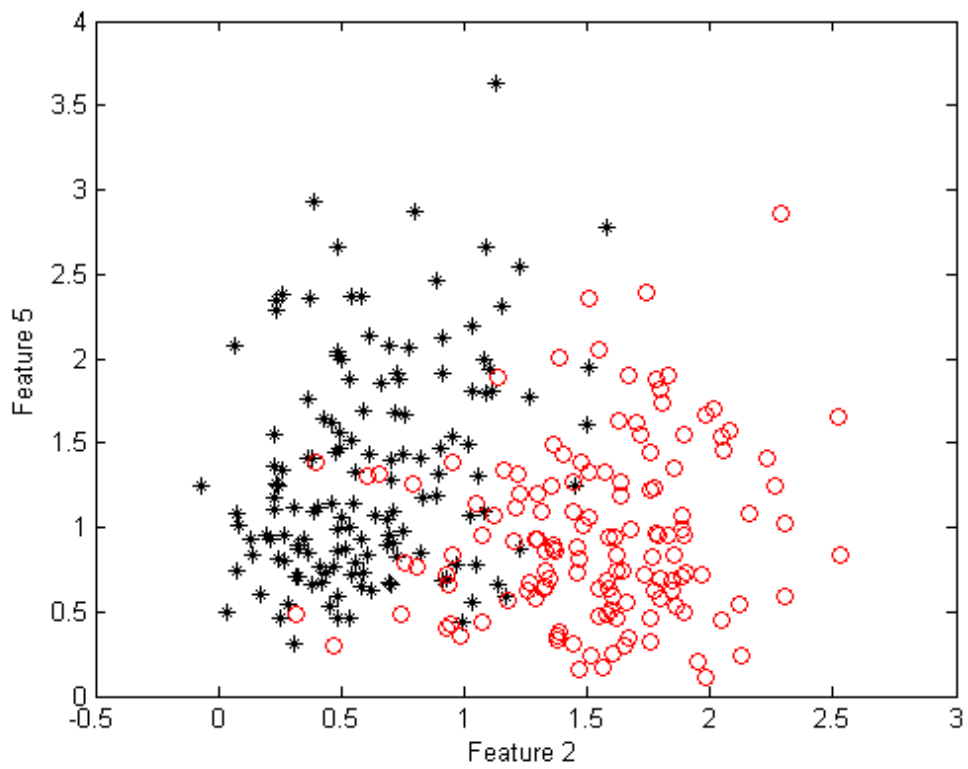


Figure 4.9 – Scatter plot for classification features 2 and 5

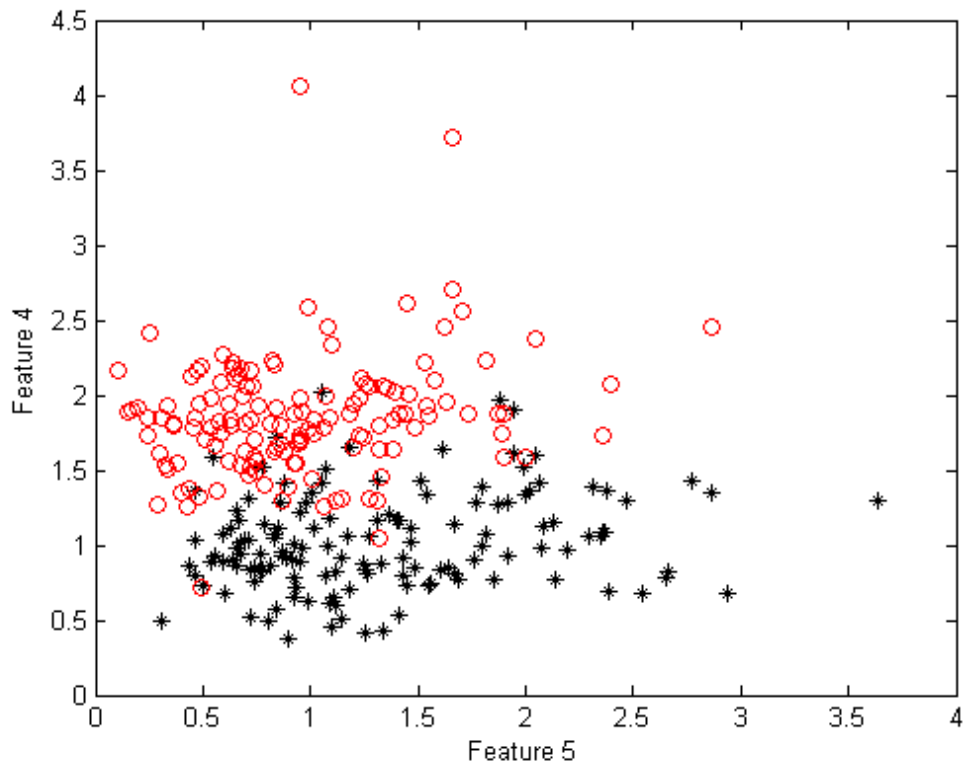


Figure 4.10 – Scatter plot for classification features 4 and 5

4.1 Comparison

Figure 4.11 demonstrates performance of the algorithm tested on 5 subjects: “aa”, “al”, “av”, “aw”, “ay” from dataset IVa. Table 4.1 demonstrates 2-fold cross-validation results.

Table 4.1 – Cross-validation results

		Subject				
		“aa”	“al”	“av”	“aw”	“ay”
Precision, %		77.10±3.18	98.63±1.56	62.07±5.18	99.40±0.95	91.67±2.79
Optimized parameters	Frequency range, Hz	60.6-92.2	58.2-102.2	58.7-145.7	52.0-146.9	46.6-141.0
	Matched filter template range, ms	353-760	343-651	172-655	455-603	195-413

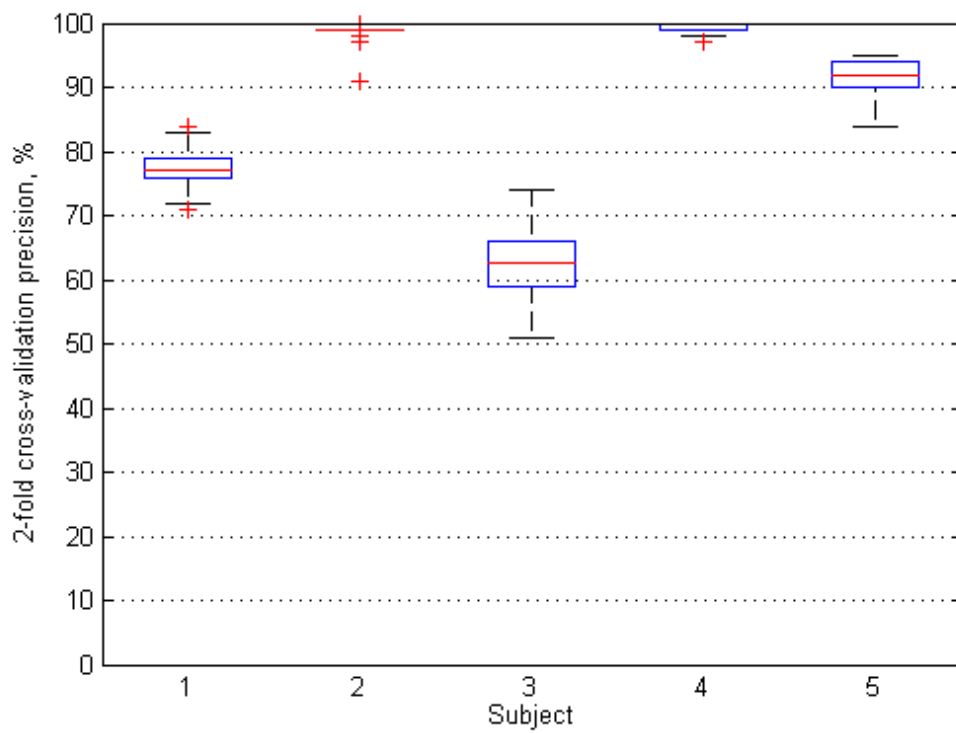


Figure 4.11 – Box plot of BCI algorithm performance across all subjects

5 CONCLUSIONS

The scatter plots obtained after feature extraction demonstrate the feature variance between two classes. The measure of how well a feature depicts the classes can be obtained using (4.1). The equation reflects the averaged square of normalized distance between values of the feature between windows for different classes. Figures 5.1 – 5.5 demonstrate values of feature significance for all of the features used in the experiment.

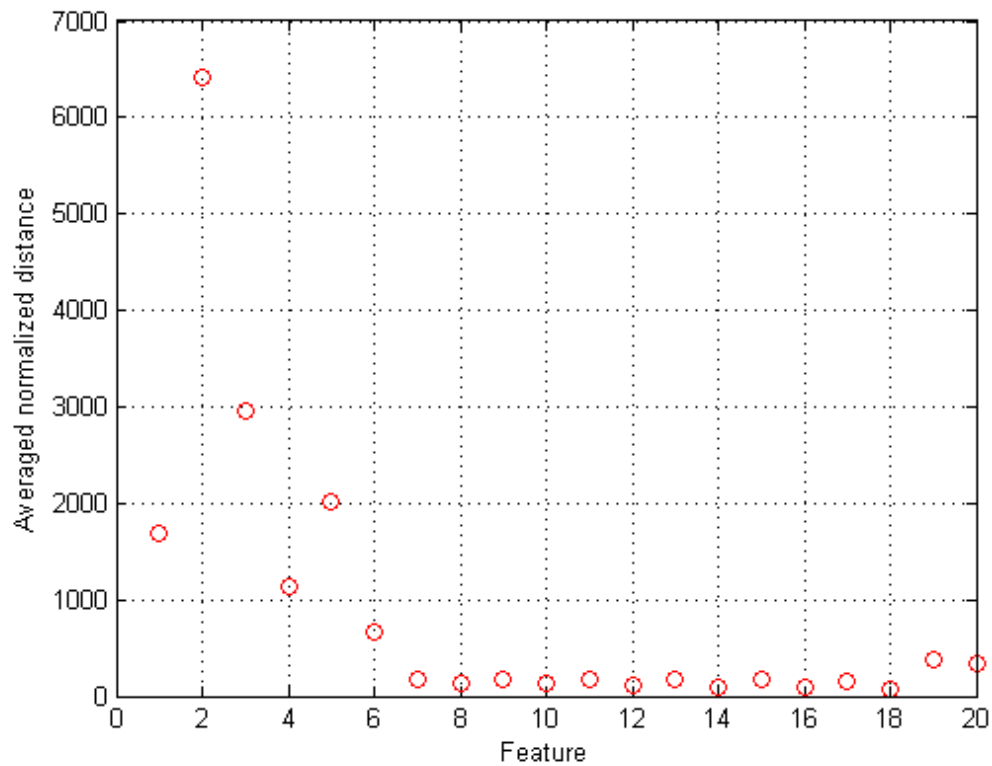


Figure 5.1 – Feature significance for subject “aa”

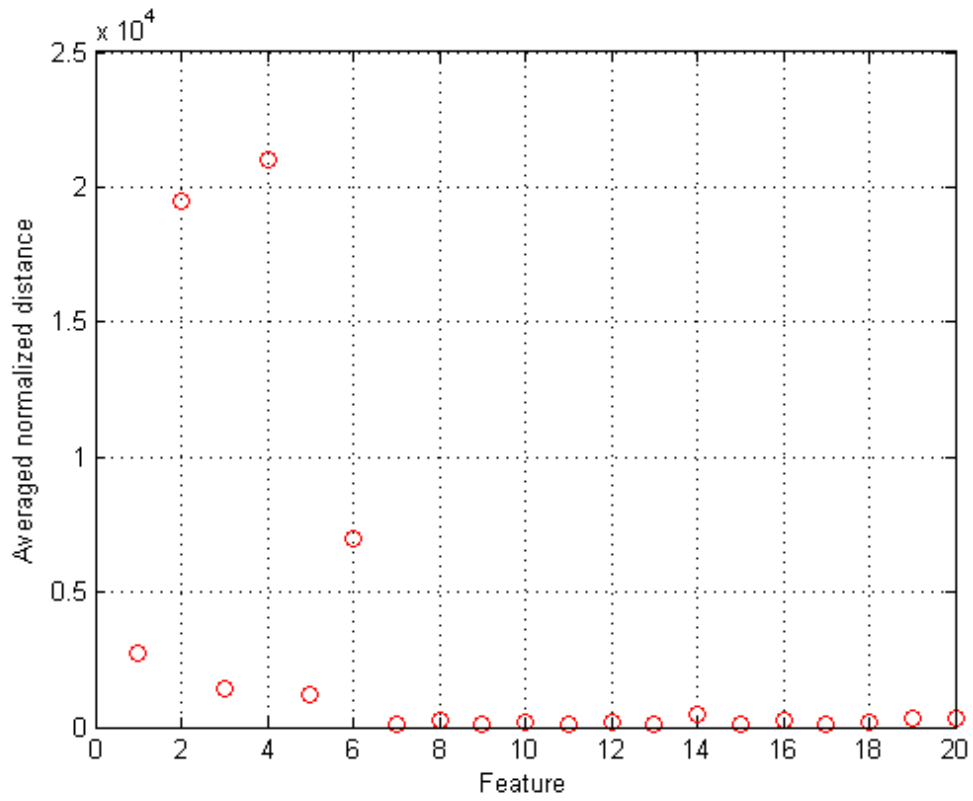


Figure 5.2 – Feature significance for subject “al”

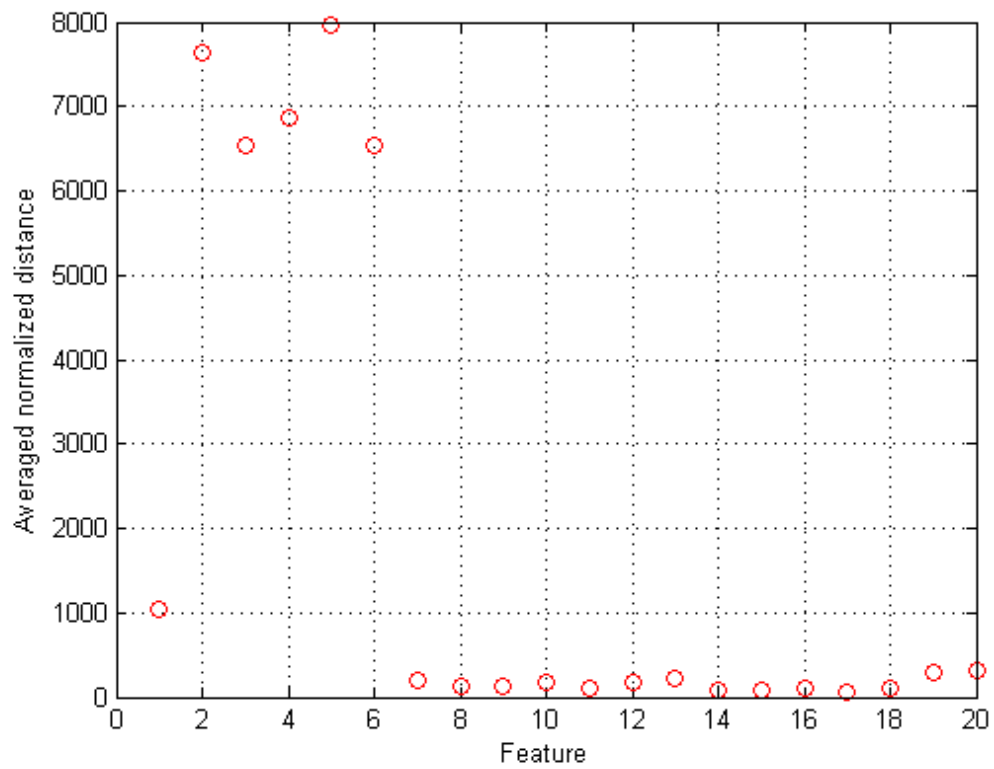


Figure 5.3 – Feature significance for subject “av”

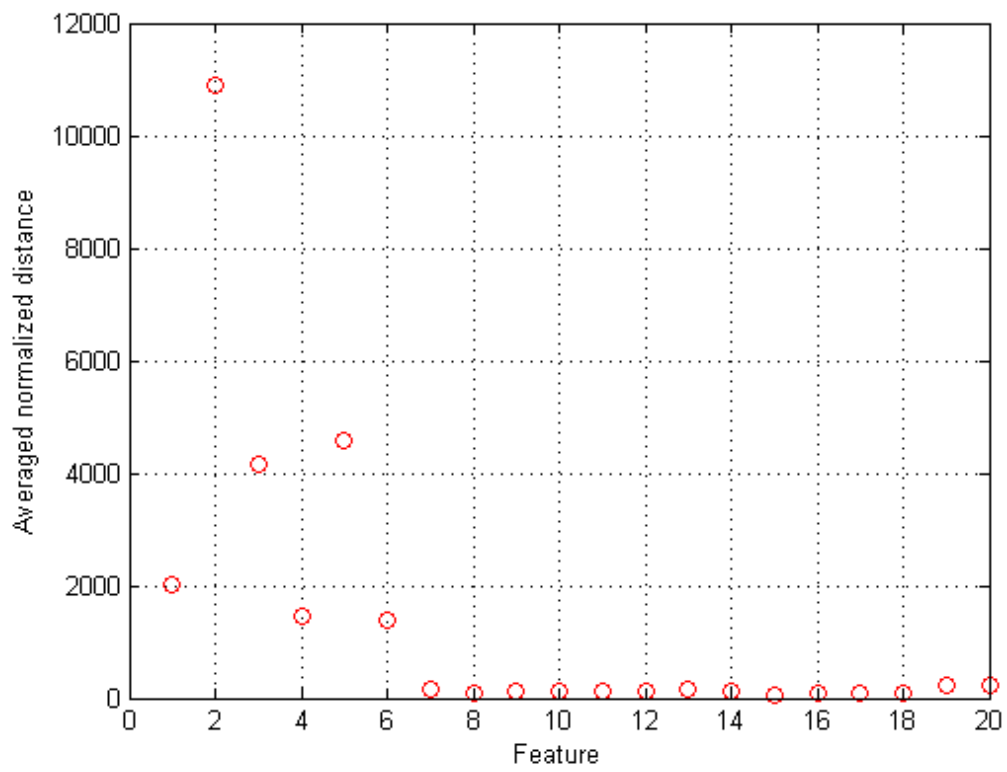


Figure 5.4 – Feature significance for subject “aw”

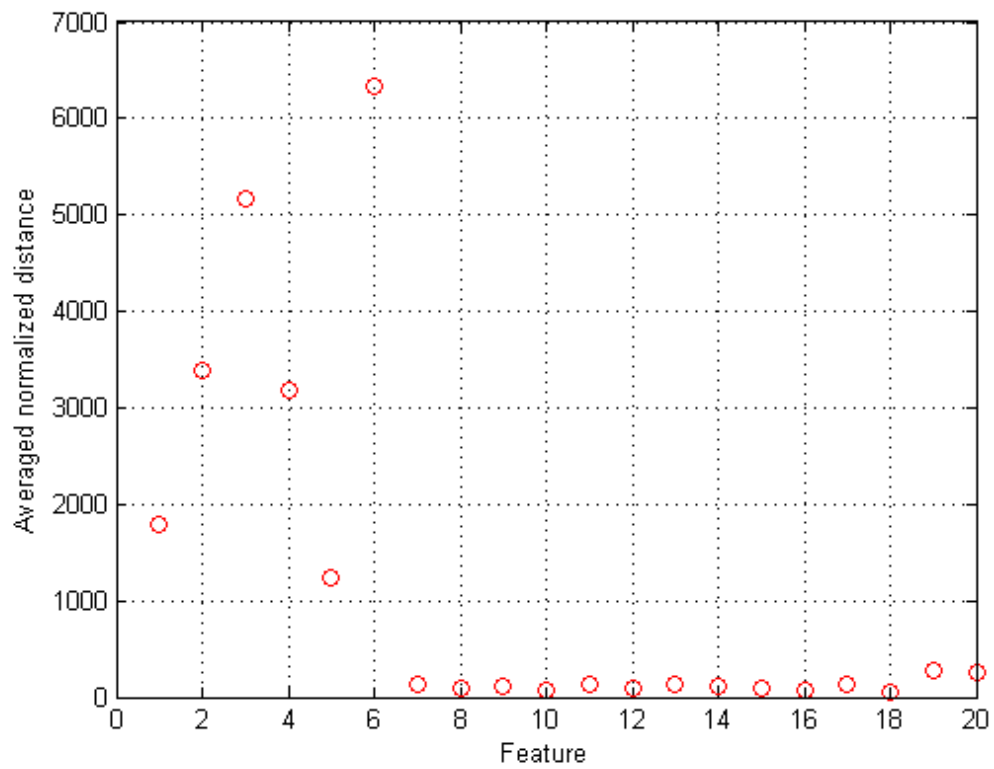


Figure 5.5 – Feature significance for subject “ay”

The Matched filter is a feature that requires considerably more time to compute than any other feature. The implementation of Matched filter is limited in a real-time system for which the algorithm is designed and, if more components are to be processed with the Matched filter, requires significant optimization. The matched filter should either not be used in a real-time system due to low efficiency, or modified, for example, by using adjustable average for template extraction.

The features other than CSP variance and Matched filter output, namely the Kolmogorov entropy and Katz fractal dimension measure, also shown poor significance for channels other than first and therefore will be removed in the future if subsequent modifications to the algorithm will require time cost optimization.

The tasks of band-pass filter band frequency and matched filter template selection were solved by using a genetic algorithm approach. This method has demonstrated decent performance. The optimization task arguments were attracted to the global minima of the cost function after 10 to 20 generations of population of 100 genotypes.

It is worth mentioning that due to the high variance within best system parameters for different subjects the empirical optimization approaches, such as the genetic algorithm used in this work are preferable. In some cases, even a closed set optimization is a viable approach.

Further research will focus on the following:

- Development of spatial filtering. The most important concept under development here is a multi-class CSP filter. The development of this is aimed to create an EEG signal linear combination that provides maximal variance between multiple windows. Study of the combination of different filter types, such as, CSP filter and LLSF, is the desired direction of the research.
- Feature vector optimization. This aims at finding the best possible set of signal feature that result in higher values of cross-validation accuracy.
- Development of the empirical optimization method. This means separating the algorithm into fixed part and optimization task, so that the individual parameters can be optimized with respect to some cost function. The goal is to find such optimization task structure and optimization approach to reduce the time needed for calculation while conserving the cost function convergence to the global minima.

- Experiments on the direct use of the BCI for control of the mobile robot. This requires further experiments of a practical use of the developed BCI. The reliable behavior of the robotic system according to the motor imagery commands are expected.
- Noise elimination. The equipment used in the research has shown strong exposure to noises. Although 5th order band-pass Butterworth filter with optimized band frequencies resulted in better system performance, the development of a filtering technique is still a topic for consideration.

6 SUMMARY

In this work, an algorithm of movement intention detection from a continuous EEG window was developed to classify signal sequences between two classes (“Go” and “No go”). The algorithm uses CSP spatial filter to separate the source activity into additive components that provide maximal variance between the classes. The preprocessing stage of the algorithm includes band-filtering of the data using a Butterworth filter. Band frequencies of the filter are adjusted using a genetic algorithm with respect to the cross-validation accuracy. A feedforward neural network is used to classify the following features extracted from the CSP signal: CSP variance, Katz fractal dimension measure, Kolmogorov entropy, Matched filter class 1 match, Matched filter class 2 match.

The algorithm performed reasonably well, providing a decent cross-validation accuracy ($77.10\pm 3.18\%$, $98.63\pm 1.56\%$, $62.07\pm 5.18\%$, $99.40\pm 0.95\%$, $91.67\pm 2.79\%$). The algorithm has shown viable results, as the comparison between this algorithm and existing approaches show. Moreover, the computation time (about 200 milliseconds in MATLAB) shows the suitability of the algorithm for a real-time robot control application.

In future research, the algorithm will be used together with a mobile robot to synthesize an optimal mobile robot control system based on the noninvasive EEG. The results of this work show the possibility of such system.

REFERENCES

- [1] Ahmadian, P., Cagnoni, S., Ascari, L., “How capable is noninvasive EEG data of predicting the next movement? A mini review,” *Frontiers in Human Neuroscience*, vol. 7, article 124, 2013.
- [2] Allen, P.J., Josephs, O., Turner, R., “A method for removing imaging artifact from continuous EEG recorded during functional MRI,” *NeuroImage* 12, pp. 230-239, 2000.
- [3] Blakemore, S.J., Goodbody, S.J., Wolpert, D.M., “Predicting the consequences of our own actions: the role of sensorimotor context estimation,” *The Journal of Neuroscience*, vol. 18, no. 18, pp.7511-7518, 1998
- [4] Blankertz, B., Muller K.-R., Krusienski, D.J., Schalk, G., Wolpaw, J.R., Schlogl, A., Pfurtscheller, G., Millan, J. del R., Schroder, M., Birbaumer, N., “The BCI competition III: Validating alternative approaches to actual BCI problems,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14 (2), pp. 153-159, 2006.
- [5] Bogacz, R., Markovska-Kaczmar, U., Kozik, A., “Blinking artefact recognition in EEG signal using artificial neural network,” *4th Conference on Neural Networks and Their Applications*, Zakopane p. 6, 1999.
- [6] Chang, C.-C., Lin, C.-J., “LIBSVM: A library for support vector machines,” 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [7] Khorshidtalab, A., Salami, M.J.E. (2011), “EEG signal classification for real-time brain-computer interface applications. A review,” *4th International Conference on Mechatronics (ICOM)*, pp. 1-7, 2011.
- [8] Labrosse, J.J., “MicroC/OS-II, The Real-Time Kernel,” *CMP Books*, 2002.
- [9] Lemm, S., Blankertz, B., Curio, G., Muller, K.-R., “Spatio-Spectral Filters for Improving the Classification of Single Trial EEG,” *IEEE Transactions on Biomedical Engineering* vol. 52, pp. 1541-1548, 2005.
- [10] Luck, S.J., Kappenman, E.S., “The Oxford Handbook of Event-Related Potential Components,” *Oxford University press*, pp. 209-210, 2011.
- [11] McFarland, D.J., McCane, L.M., David, S.V., Wolpaw, J.R., “Spatial filter selection for EEG-based communication,” *Electroencephalography and Clinical Neurophysiology*, 103 (3), pp. 386-394, 1997

- [12] Niazy, R.K., Beckmann, C.F., Iannetti, G.D., Brady, J.M., Smith, S.M., “Removal of fMRI environment artifacts from EEG data using optimal basis sets,” *NeuroImage* 28, pp. 720-737, 2005.
- [13] Novi, Q., Guan, C., Dat, T.H., Xue, P., “Sub-band Common Spatial Pattern (SBCSP) for Brain-Computer Interface,” *3rd International IEEE EMBS Conference on Neural Engineering*, pp. 204-207, 2007.
- [14] Pfurtscheller, G., Lopes da Silva, F.H., “Event-related EEG/MEG synchronization and desynchronization: basic principles,” *Clin. Neurophysiol.* 110, no. 11, pp. 1842-1857, 1999.
- [15] Shakeel, A., Navid, M.S., Anwar, M.N., Mazhar, S., Jochumsen, M., Niazi, I. K., “A review of techniques for detection of Movement Intention Using Movement-Related Cortical Potentials,” *Computational and Mathematical Methods in Medicine* 13, 2015.
- [16] Shibasaki, H., Hallett, M., “What is the Bereitschaftspotential?” *Clinical Neuropsychology*, vol. 117, no. 11, pp.2341-2356, 2006.
- [17] Sun, G., Hu, J., Wu, G., “A novel frequency band selection method for Common Spatial Pattern in Motor Imagery based Brain Computer Interface,” *Neural Networks (IJCNN), The 2010 International Joint Conference*, pp. 1-6, 2010.
- [18] Wolpert, D.M., Flanagan, J.R., “Motor prediction,” *Current Biology*, vol. 11, no. 18, pp. 729-732, 2001.
- [19] Wright, D.J., Holmes, P. S., Smith, D., “Using the Movement-Related Cortical Potential to Study Motor Skill Learning,” *Journal of Motor Behavior*, vol. 43