

SILESIAN UNIVERSITY OF TECHNOLOGY

DOCTORAL THESIS

**Analysis of the brain activity
spatio-temporal patterns for development
of brain-computer interfaces.**

Author:

Michał PIEŁA

Supervisors:

Marian KOTAS, PhD, DSc

Sonia Helena CONTRERAS ORTIZ, PhD

*A thesis submitted in fulfillment of the requirements
for the PhD degree*

in the

Department of Cybernetics, Nanotechnology and Data Processing

February 14, 2023

Declaration of Authorship

I, Michał PIELA, declare that this thesis titled, “Analysis of the brain activity spatio-temporal patterns for development of brain-computer interfaces.” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

*I would like to express my thanks to my supervisor
Prof. Marian Kotas for valuable remarks, dedicated support and guidance
at all stages of this dissertation.*

*I would like to thank also my second supervisor
Dr Sonia Helena Contreras Ortiz for helpful opinions and hints.*

Contents

Declaration of Authorship	iii
1 Introduction	1
1.1 Motivation for the research	3
1.2 Difficulties related to evoked potentials classification	4
1.3 Methods of EEG analysis in BCI	7
1.4 Aim of the thesis	9
1.5 Thesis	10
1.6 Overview of the thesis content	10
2 BCI experiments and databases	11
2.1 Explanation of the BCI experiment	11
2.2 Description of the BCI "Item selection" experiments	12
2.3 "Item selection" database structure	13
2.4 Description of the BCI "speller" experiments	13
2.5 "Speller" database structure	13
3 Development of a filter for spatio-temporal patterns enhancement	15
3.1 Matched filtering	15
3.2 Generalized matched filtering	17
3.3 Spatio-temporal filtering	18
3.3.1 STF filter construction	19
3.4 Generalized spatio-temporal matched filtering	22
3.5 Modified spatio-temporal matched filtering	24
3.5.1 Nontarget responses subtraction	24
3.6 The filter output interpretation	26
3.6.1 Classical decision rules	26
3.6.2 Modified decision rules	26
3.6.3 MSTMF responses as feature vectors for SVM classifier	27
3.7 Nomenclature of the methods	27
4 EEG signals preprocessing for artifacts removal	29
4.1 Types of noise artifacts disturbing the EEG signals	30
4.2 Application of DCTBS to noise removal	32
4.2.1 Detection of outliers	33
4.2.2 Artifacts blanking and interpolation	34

4.3	Application of ICA and EMD to noise removal	35
4.3.1	Artifacts detection	36
4.3.2	Algorithms for low number of channels	36
4.3.3	Algorithms for high number of channels	42
5	Numerical experiments	49
5.1	Experiments with the "Item selection" database	49
5.1.1	Signal preprocessing and artifacts removal	49
5.1.2	Parameters selection and filters construction	52
5.1.3	Tests of the methods performance	55
5.1.3.1	Search for favorable values of parameters δ and Δ	55
5.1.3.2	Verification tests	55
5.1.4	Investigation of the new ICA based approach to artifacts removal	59
5.1.5	Illustration of the influence of parameters J and τ on the accu- racy and computational time	60
5.1.6	On the execution times of the methods	61
5.1.7	Final results and comparison to the participants of the 2019 IFMBE Scientific Challenge	63
5.2	Experiments with the "speller" database	63
5.2.1	Signal preprocessing and artifacts removal	64
5.2.2	Adjustment of the developed method's parameters	68
5.2.3	Verification of the selected methods	68
5.2.4	Results and comparison to the winners of BCI challenge	70
6	Summary	73
A	Discrete cosine transform based smoothing (DCTBS)	77
B	Independent Component Analysis (ICA) for noise reduction	81
C	Empirical Mode Decomposition for artifacts rejection	83
	Bibliography	87

List of Figures

1.1	Illustrative block diagram of the brain-computer interface.	2
1.2	Illustration of the evoked potential component vs the normal activity of the brain presented on an exemplary channel recording. Left graph: result of averaging 480 evoked potentials (bold plot) and normal activity of the brain (thin plot); right graph: zoomed in part of the left graph presenting the averaged EP.	4
1.3	Construction of the feature vectors matrix: columns represent the consecutive features, in rows are gathered successive observations (feature vectors) associated with brain responses to individual flashes. Orange boxes denote the matrix entries, i.e. the EEG signal values measured during the specified time moments ($t_{i,j}$, corresponding to the i^{th} flash and the j^{th} time sample stored, following this flash).	5
1.4	Presentation of the large super-Gaussian noise components in the exemplary 64 channel EEG signal; zoomed in region illustrates part of the routine activity of the brain	6
2.1	Illustration of BCI experiment; a moment when visual stimulus (target) is captured.	12
2.2	Illustration of BCI "speller" experiment; a moment when visual stimulus (target) is captured.	14
3.1	Exemplary desired component $s(k)$ and matched filter impulse response $h(k)$, for $K=6$	16
3.2	Result of the filtering of $s(k)$ with MF filter with impulse response $h(k)$	16
3.3	Visual explanation of parameters δ and Δ on the picture showing 1.4s segment with a target response from all 8 channels. 2δ , marked in green, defines the width of the window containing time indices of the set of vectors ($\mathbf{x}^{(k)}, k \in \Psi_t(\delta)$) to be enhanced. Blue area on both sides shows the indices of vectors $\mathbf{x}^{(k)}$ to be suppressed contained in set $\Psi_s(\Delta)$. The gap between green and blue areas allows to form the proper shape of the STF filter responses to target flashes (see Fig. 3.4). The red arrow indicates a range of indices of the vector $\mathbf{x}^{(n)}$ to be enhanced.	20

3.4	Illustration of the impact of δ on the averaged filter response (over all runs within an exemplary block). Black, blue, and red plots present the averaged filter responses for optimal $\delta = 20ms$ applied in the experiments, δ two times lower, and two times greater, respectively.	21
3.5	Comparison of the filter templates constructed for OSTF (upper plot (a) in dark blue) and GSTMF (lower plots (b) in light blue). Red vertical lines separate the segments related to a given channel each containing $2J + 1 = 43$ values.	22
3.6	Comparison of the filter templates constructed for OSTF (upper plot (a) in dark blue) and GSTMF (lower plot (b) in light blue). Red vertical lines separate the segments related to a given channel each containing $2J + 1 = 43$ values.	23
3.7	Averaged target ($\bar{\mathbf{a}}$) and nontarget $\bar{\mathbf{a}}'$ patterns for the first subject are shown in the overlay plot (a), in blue $\bar{\mathbf{a}}$, and in orange $\bar{\mathbf{a}}'$. The difference of these patterns $\bar{\mathbf{a}} - \bar{\mathbf{a}}'$ is presented in plot (b). Red vertical lines separate the segments related to a given channel each containing $2J + 1 = 43$ values. Corresponding channel names are given in the upper plot (a).	24
3.8	Averaged target ($\bar{\mathbf{a}}$) and nontarget $\bar{\mathbf{a}}'$ patterns for the second subject are shown in plot (a), in blue $\bar{\mathbf{a}}$, and in orange $\bar{\mathbf{a}}'$. The difference of these patterns $\bar{\mathbf{a}} - \bar{\mathbf{a}}'$ is presented in plot (b). Red vertical lines separate the segments related to a given channel each containing $2J + 1 = 43$ values.	25
3.9	Visual explanation of the modified decision rules. The averaged filter output is plotted in black. Red, dotted line represents the position used by the classical decision rules ($k = 0$); the maximum of the response in such a case is marked with red X. Blue dotted lines indicate the larger range of search for the maximum of the averaged target response. The position found using the modified decision rules (k_0) is marked with a dotted green line, and the corresponding maximum with a green dot.	27
4.1	Exemplary occurrence of the ocular artifacts in EEG recording. The ocular artifact which spread to multiple channels is marked in a brown frame.	31
4.2	Examples of cardiac artifacts which are marked by the brown frames. Red plot is a reference ECG recording, the QRS complexes which have the greatest impact on the EEG traces are marked also by the brown frames.	32
4.3	Exemplary artifact detected. Black dashed line indicates the threshold, boundaries of the artifact are marked with red + signs and the red dotted lines.	33

4.4	Plot A: segment of the 8-channel signal where the artifact has been detected (the samples belonging to the artifact, indicated by matrix \mathbf{Z} , are plotted in red); dotted, red lines mark the artifact's borders, the channel containing the artifact is plotted in blue beyond the artifact borders; plot B: The noisy (blue) channel with the artifact's samples blanked; plot C: the outcome of DCTBS interpolation of the blanked samples, the reconstructed part is plotted in green.	34
4.5	Exemplary artifact borders estimation. Blue dots represent the positions where 1 st condition is met (first cross of the zero value), green dots: 2 nd condition (change of the derivative polarity), and red dots: 3 rd condition (second cross of the zero value). Red dots along with the dotted red vertical lines mark also the borders of the artifact.	36
4.6	Part of the measured 8-channel signal containing the artifact. The signal is scaled in time to better illustrate details of the artifact	37
4.7	Outcome of ICA performed on the signal visible in Fig. 4.6. 8 subplots present 8 independent source signals, the one plotted in red was classified as noisy in procedure described in step 2.	38
4.8	Illustration of the artifact borders (plotted by the vertical, blue, dotted lines) detected on the noisy IC (IC1).	39
4.9	Noisy source signal with the artifact zeroed	39
4.10	Noisy source signal with the artifact suppressed using EMD2 (which rejects the highly kurtotic IMFs).	40
4.11	The effect of the original measured signal (presented in Fig. 4.6) denoising. Plot A: The signal reconstructed using L-ICA1 (ICA), plot B: The signal reconstructed by L-ICA2 (ICA+EMD2).	41
4.12	Upper plot: Measured signals containing the artifacts; In red: \hat{M} signals of the highest kurtosis determined in step 3, in black: remaining signals. Blue, dotted lines represent borders of the lower plot which magnifies view on one of the largest artifacts.	42
4.13	Source signals (independent components) estimated in step 4, in red: source signals detected as noisy (above the threshold thr) according to step 7. In black: remaining, unnoisy source signals.	43
4.14	Plot A: Part of the measured signals containing the artifacts, plot B: artifacts approximated using H-ICA1 algorithm, plot C: the result of subtraction of the approximated artifacts from the corresponding measured signals.	44
4.15	Plots I, II, and III present the comparison between the noisy source signals shown in Fig. 4.13 (red plots) with the modified source signals (blue plots) in the algorithm H-ICA2 (which adds the EMD1 procedure).	45

4.16	Plot A: Part of the measured signals containing the artifacts, plot B: artifacts approximated using H-ICA2 algorithm, plot C: the result of subtraction of the approximated artifacts from the corresponding measured signals.	46
4.17	Plot A: Part of the 64-channel signal, shown in Fig. 4.12, containing the artifact, plot B: modeled artifact elements using H-ICA3 algorithm, plot C: the result of subtraction the artifact elements from the noisy signal channels.	47
4.18	Visual comparison of the algorithms dedicated for databases with a high number of channels. Plot A: Noisy 64-channel signal, Plots B, C, and D: Signals denoised using H-ICA1, H-ICA2, and H-ICA3, respectively.	48
5.1	Exemplary performance of DCT based denosing applied to "Item selection" database. Plot A: the signal containing a single channel artifact (plotted in blue), plot B: the signal with artifacts rejected using DCT (the modified channel is plotted in blue).	49
5.2	Exemplary performance of DCT based denosing applied to the "Item selection" database. Plot A: the signal containing a multichannel artifact, plot B: the signal with artifacts rejected using DCT based procedure.	50
5.3	Exemplary performance of L-ICA based denosing applied to "Item selection" database. Plot A: the signal containing a single channel artifact (affected channel in light blue), plot B: estimated source signals (in red the noisy source signal), plot C: the signal with artifacts rejected using ICA; the modified channel is plotted in blue.	51
5.4	Exemplary performance of L-ICA based denosing applied to "Item selection" database. Plot A: the signal containing multichannel artifact, plot B: estimated source signals (in blue the noisy independent component (IC)), plot C: the signal with artifacts rejected using L-ICA.	52
5.5	Explanation of parameters δ and Δ on the picture showing 1.4s segments from all 8 channels. The sets given in Fig. 3.3 were readjusted to the purpose of experiments on this database. 2δ , marked in green, defines the width of the window containing time indices of the set of vectors $(\mathbf{x}^{(k)}, k \in \Psi_t(\delta))$ to be enhanced. Blue area on both sides shows the indices of vectors $(\mathbf{x}^{(k)}, k \in \Psi_s(\Delta))$ to be suppressed (these areas were additionally reduced on both sides by $J\tau$ so that the spatio-temporal vectors stored in \mathbf{B} could be properly constructed).	53

- 5.6 Visual comparison of the MSTMF and LDA performance. The plots on the left illustrate the differences $\bar{\mathbf{a}} - \bar{\mathbf{a}}'$ in MSTMF (plot A for $\delta > 0$ and plot C for $\delta = 0$). Plot E shows the analogical difference of classes means used in LDA. The plots in blue on the right present the filter templates of MSTMF with $\delta > 0$ (plot B), MSTMF with $\delta = 0$ (plot D), and LDA (plot F). These filter templates are calculated using formula (3.21) where the difference $\bar{\mathbf{a}} - \bar{\mathbf{a}}'$ is multiplied by the inverse of the noise covariance matrix: \mathbf{C}_b^{-1} . The vertical dotted lines in red separate the values belonging to the different channels (whose names are given in the upper plots); each channel contains $2J + 1 = 43$ values. 54
- 5.7 Visualization of the decision rules (modified versus classical ones): A) MSTMF responses to target flashes within one block ($I_r = 9$ consecutive runs of flashes, k on the horizontal axis is introduced in (3.23)), B) MSTMF responses to I_r flashes of a single nontarget object within the same block as in plot A, C) average target MSTMF responses (the one in blue is the mean of the plots in A; the gray ones correspond to different blocks of flashes), D) the average nontarget MSTMF response (the mean of the plots in B). The double arrow shows the extended region of the search for the maximal filter output. The red crosses mark the values of the scores obtained for $k = 0$ (expected position according to CDR), and the green points, the ones obtained using MDR (the maxima in the specified range of the search). 56
- 5.8 Left plots: histograms of the blocks averaged qualities coefficients \bar{Q}_1 for all 15 subjects for the methods: *OSTF* (plot I), *GSTMF* (plot II), and *MSTMF + MDR* (plot III). With the grayed frame marked is the subject 4 which is presented in more details (Q_1 coefficients for 350 single blocks) in the plots on the right side of the figure. Red horizontal lines indicate the detection threshold, the blocks whose Q_1 is above this threshold (equal to 1) have properly recognized target responses. 57
- 5.9 Illustration of the dependency of MSTMF+MDR accuracy on the τ parameter value. 61
- 5.10 Illustration of the dependency of the computational time on the τ parameter value. Plot A applies to the learning subset, plot B to the testing subset. 62
- 5.11 Results of outlying artifacts suppression using DCTBS: original 64-channel EEG recording; in bold black: noisy channels, in grey: acceptable channels 65
- 5.12 Results of outlying artifacts suppression using DCTBS: the modified 64-channel EEG recording; in bold black: modified channels, in grey: unchanged channels. The amplitude scale has been changed with respect to Fig. 5.11 65

5.13	Results of outlying artifacts suppression using the ICA based algorithms. Plot A: original 64-channel EEG recording, plots B, C, and D are the same 64-channel recording with the artifacts suppressed using algorithms H-ICA1, H-ICA2, and H-ICA3, respectively.	66
5.14	Visualization of H-ICA3 performance. Plot A: original 64-channel EEG recording containing both multichannel artifacts and single artifact (in one channel only, marked in blue ellipse), plot B presents the outcome of H-ICA3 denosing, and plot C is a results of H-ICA3 applied twice.	67
C.1	Plot A: The exemplary noisy source signal, plot B: the outcome of its decomposition using EMD into Intrinsic mode functions (the noisy IMFs are plotted in red), plot C: the reconstructed noisy source signal based on the noisy IMFs only, plot D: the reconstructed source signal which is a sum of the unnoisy IMFs only.	84
C.2	Details of the EMD outcome. IMFs are plotted on separate subplots, the noisy IMFs are plotted in red.	86

Chapter 1

Introduction

One of the basic humankind's way of expression is to communicate with its surrounding environment. Most natural means of such interaction involve ability to use speech, gestures, touch or facial expressions. Along with development of computer sciences and adequate numerical methods, all these mentioned types of self expression could become used not only between people themselves but also between humans and computers. Apart from a manual control of a computer mouse and a keyboard, progress of speech and gestures recognition gave a foundation for touchless ways to control the computers. Quality of the contemporary algorithms in this field is sufficiently high to enable numerous daily life applications (control by voice or gesture commands, typing with one's voice etc.). However, direct communication between the human's brain and computer seems to be another step further. The idea behind it assumes the ability to send the messages to a computer without using limb muscles, voice, or body expressions. Here, the electrical potentials generated by the brain constitute a medium of an information flow which can be interpreted by the computer system. Proper employment of these potentials in such a brain-computer interface (BCI) required developments in an electroencephalography (EEG), which aim is to acquire the brain potentials by means of electrodes placed on the skull, and material science playing a leading role in providing crucial enhancements of the EEG electrodes materials [37]. Furthermore, of a great importance was also a progress in understanding the brain physiology, particularly its reactions to a various external stimulation, typically visible as deflections from the brain waves associated with its routine activity. Thus, proper detection of such a deflection in EEG trace is a fundamental requirement made of the BCI system.

The origins of BCI systems date back to 1960s with development of the first implantable chip in 1969 which allowed to receive electrical signal from the brain using radio signals [21]. In 1973 first non-invasive system, founded on the measurements of the brain potentials from the scalp, by means of EEG, was constructed by Vidal [53]. Since then plenty of various external stimulation like visual flashes, auditory tones or tactile stimuli, whose aim was to trigger the possibly highest brain response, have been tested as the BCI control signals. Besides the enumerated external triggers, the brain responses could be induced also by the internal stimulus e.g. imagination of the limb movement. Apart from the type of the employed trigger the

working principle of BCI, presented in Fig. 1.1, remains the same. The user interface provides the commands or presents a stimulus (e.g. a visual flash as in Fig. 1.1) in a given moment of time, which is registered by the interface controller. Simultaneously, EEG signal from the user's brain is collected, preprocessed and the features are extracted. The offline part of the signal processing, which is considered as the system calibration when the proper classifier is trained, usually requires much more time and is performed prior to normal online operation of the BCI system.

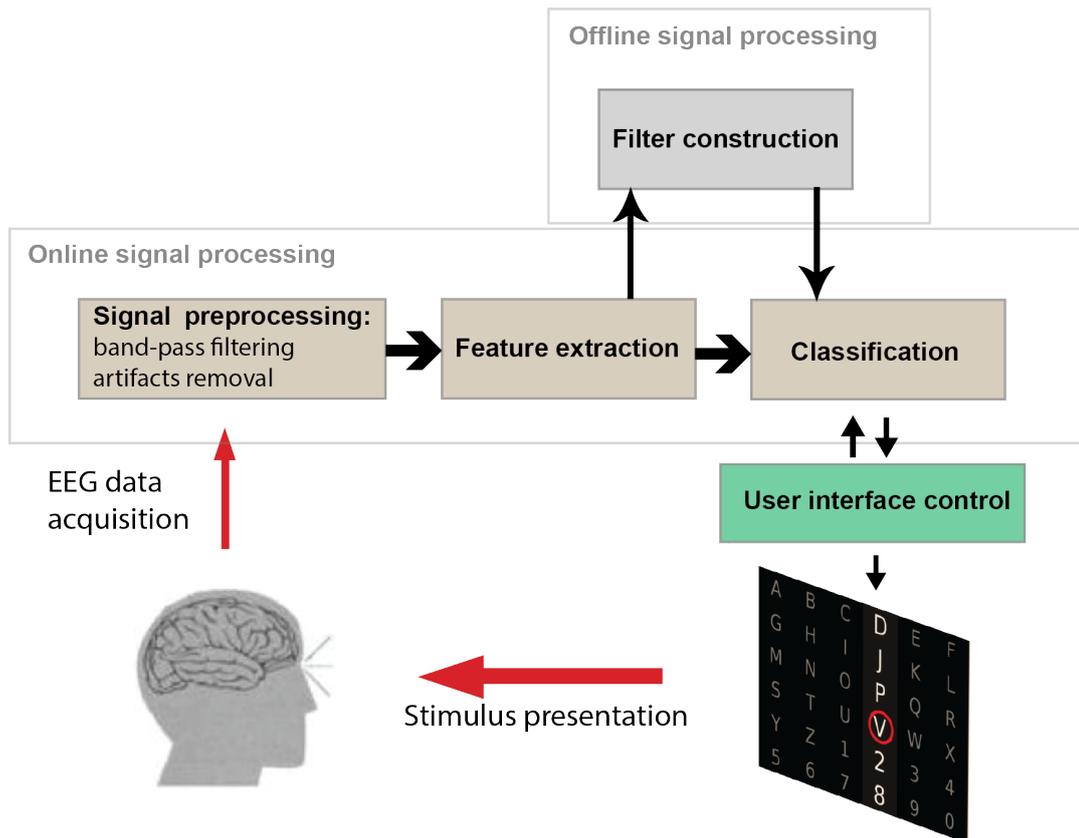


FIGURE 1.1: Illustrative block diagram of the brain-computer interface.

The potentials which are produced in the brain as responses to the triggering stimuli, introduced above, are called as evoked potentials (EP). This thesis focuses on detection of visual evoked potentials with dominant P300 which was employed for the first time as a control signal in BCI by Farewell and Donchin in 1988 [25]. Since then, it has become known as a good measure of evoked activity of the brain, mainly because of its relatively high amplitude, in comparison to the whole range of different potentials produced in the brain either during visual, auditory, or mental tasks. For the purpose of the experiments, being part of this dissertation, it was assumed that the efficient EP detection can be accomplished by analysis of 1s segment which starts at the onset of the visual flash presentation. It is typical time segment, commonly described in BCI related literature, which should assure the occurrence of P300 and accompanying it smaller potentials. Therefore, BCI system's main goal

is to detect the presence or absence of EP component in this search region. A BCI user interface typically contains a certain number of objects (items, letters etc.) on the screen which are blinking in a random order, one after another. During this time, a user is focused on one selected object, so called target. A target flash should, due to the mentioned user's intentional focus on it, trigger a higher brain response than the flashes of the other unobserved objects, called as nontargets. However, the amplitude of a brain response to a single flash (regardless of a type of the trigger which induced this response: target or nontarget) is very low and usually hidden in the waves associated with a routine activity of the brain. In effect, the BCI experiments are designed so that the flash of each object is repeated multiple times and by thus more brain responses are stored. By averaging these repeated responses, separately for each object, one can more accurately determine which object generated the highest averaged response. Such an object can by thus be detected as a target object (the one observed by the user).

1.1 Motivation for the research

Major motivation to perform research in the field of the BCI systems was the author's interest in the touchless communication techniques as well as in the human brain and its functionality. Because the contemporary BCIs are still too slow and insufficiently reliable to be applied and comfortably used in daily life situations, every attempt to contribute to the overall development of such interfaces seems to be desirable and inspirational.

This type of communication with machines can greatly improve life quality of people with various disabilities, particularly those associated with neuromuscular disorders (like Amyotrophic lateral sclerosis (ALS), Muscular dystrophy (MD)) who cannot use typical means of a control like their limb muscles or voice. Using the BCI system they have a possibility to communicate with their surroundings and to control the computer or other interfaced devices [41], [101]. Similar system can be used to control a prosthesis arm, especially when the peripheral nerves cannot be utilized due to a brain or a spinal injury. In such a case the user's intention to control the arm needs to be recorded directly from the brain (e.g. using EEG in noninvasive systems) [75], [78]. Dedicated BCI systems can also support post-stroke rehabilitation, which aim is to trigger neurological recovery, and by thus reduce a motor impairment of a patient. During such a therapy patients learn how to regulate their sensorimotor rhythms and thereby facilitate neuroplasticity process and recovery of the damaged neural structures of their bodies [15]. Furthermore, the system which provides the interface between the patient's brain and computer can support and enhance health diagnostics: f.e. audiovisual BCI to assess the condition of patients with disorders of consciousness (DOC) [24], auditory BCI to help in diagnosing patients with schizophrenia [87], or system supporting detection of epileptic seizure episodes [98]. Besides the medical applications, related with previously mentioned

afflictions, the interfaces allow to evaluate the condition of healthy subjects during performing crucial tasks (e.g. for safety reasons), it can attempt to examine their focus on the task [22], level of stress [91], or tiredness [29]. Another important example of BCI employment concerns neurofeedback based training of attention [3] or self control with regard to emotions and thoughts what can be particularly helpful for people highly exposed to stress due to their professions. On the other hand, BCIs can be applied in modern, touchless communication and control systems including smart home applications, and also in entertainment industry (virtual reality applications [57], video games [68], [100]).

1.2 Difficulties related to evoked potentials classification

Since years, a major difficulty inherent to the task of evoked potentials detection relates to a very low energy of such a potential in reference to the spontaneous activity of the brain. Usually, EP is no greater in amplitude than one tenth of a typical EEG trace. The core of the difficulty is illustrated in Fig. 1.2 where nearly 500 of such potentials were averaged in order to visualize the approximate shape and size of the desired component of EP. As remarked earlier, due to such immensely low signal to

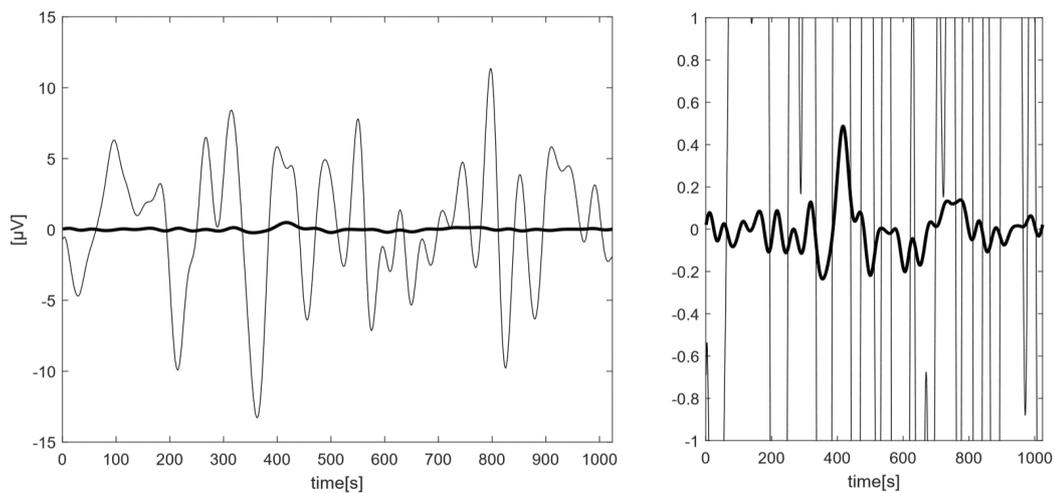


FIGURE 1.2: Illustration of the evoked potential component vs the normal activity of the brain presented on an exemplary channel recording. Left graph: result of averaging 480 evoked potentials (bold plot) and normal activity of the brain (thin plot); right graph: zoomed in part of the left graph presenting the averaged EP.

noise ratio, BCI experiments are designed so that the same stimulus, which triggers the brain reaction, is repeated multiple times. Analysis of only one response to a single trigger is insufficient to reliably detect the EP. Practical solutions involve several repetitions of the same stimuli and averaging the corresponding brain responses. Consequently, the experiment time is prolonged what is inconvenient, particularly for patients whose medical condition causes their concentration to lessen significantly with time. Such a disadvantage can be attempted to be overcome or reduced

twofold: by improving the user interface and the experiment design so that a period between the consecutive repetitions can be shortened or by developing new, more efficient classifiers which allow to reduce the number of required repetitions.

Second demanding subject concerns a massive number of features which need to be extracted to properly represent an EEG event (single brain response to a stimulus), e.g. previously mentioned 1s segment, which is analysed in terms of the presence of the EP component or its absence. Features, which are the time samples of

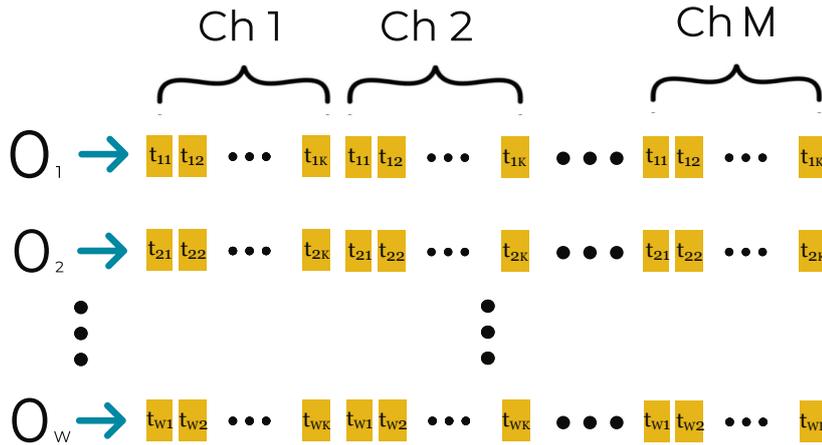


FIGURE 1.3: Construction of the feature vectors matrix: columns represent the consecutive features, in rows are gathered successive observations (feature vectors) associated with brain responses to individual flashes. Orange boxes denote the matrix entries, i.e. the EEG signal values measured during the specified time moments ($t_{i,j}$, corresponding to the i^{th} flash and the j^{th} time sample stored, following this flash).

every EEG channel, are gathered in a feature vector. Each of these vectors can be considered as an observation, set of features collected from a brain response. Exemplary structure of the matrix containing feature vectors, used to learn the classifier, is illustrated in Fig. 1.3. The matrix is composed of the consecutive observations given in rows, number of columns indicates size of the feature vector. Even though the sampling frequency and by thus the number of time samples from a single channel is usually reduced, still the dimension of the formed feature vector is relatively large. High dimensionality of the feature vector (product of a number of channels M and a number of time samples within one channel K) combined with a limited number of observations (W) available in the learning data subsets, lead to difficulties in the classifier construction which is known in literature as small sample size problem [27]. It occurs when $KM > W$ according to variable notation given in Fig. 1.3.

Third difficulty, intrinsic to the acquisition of bioelectric signals, is a presence of artifacts, particularly those of very high amplitude. In terms of BCI experiments, the waves related to normal activity of the brain in EEG recording are considered as noise. The efficient filter should attenuate these waves and enhance the desired

EP components. However, the high amplitude spikes occurring in the signal (as ones presented in Fig. 1.4), which can be regarded as super-Gaussian artifacts [93], can have a detrimental impact on this major filter's task and result in its decreased performance. Methods based on the matched filtering applied alone, like general-

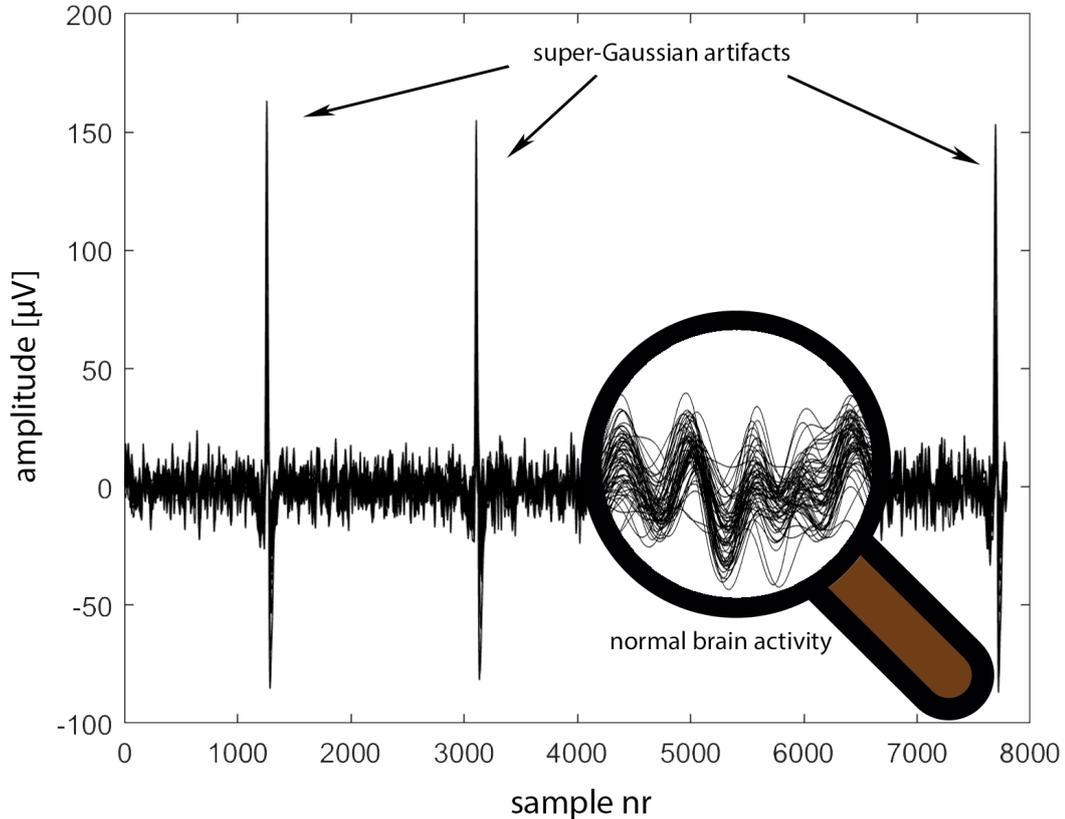


FIGURE 1.4: Presentation of the large super-Gaussian noise components in the exemplary 64 channel EEG signal; zoomed in region illustrates part of the routine activity of the brain

ized matched filtering (GMF) being efficient for detection of a known pattern embedded in colored Gaussian noise [54], are prone to fail in case of presence of such non-Gaussian artifacts. Similarly, the applied classifiers can produce false decisions. Therefore, prior removal of these artifacts should be essential.

Finally, the exact moments of brain responses, measured with latency (time from the onset of the stimulus to the occurrence of the brain response potential), can be slightly shifted from the expected positions. It can be caused by the inter-subject variability of latency, intra-subject latency jitter [106], [51], or result from the measuring equipment, experiment design etc. Regardless of the source of these shifts, their proper recognition and correction seem to be meaningful for BCI classifier to properly discriminate the brain potentials. When the database is affected by latency jitter, the brain responses stored as the feature vectors in the matrix illustrated by Fig. 1.3 are shifted with respect to each other, and the time moments $t_{w,k}$, $w = 1, 2, \dots, W$,

which should indicate exactly the same phase within the respective W brain responses, fail to do so. It can hinder the process of the classifier learning and by thus deteriorate the proper classification. Such a problem applies to the majority of the classifiers used in the field of BCI.

1.3 Methods of EEG analysis in BCI

Aim of the classifier in BCI is to distinguish whether the brain response was induced by a flash of a target (the object the user was focused on) or a flash of a nontarget (any other object neglected by the user). Among the classifiers used to accomplish this task are those which analyze temporal, spatial or spatio-temporal (containing spatial information from various channels of the EEG signal and temporal, taking into account the appropriate time segments of the signal) patterns of the brain responses. The most popular ones, which assure good performance, are based on support vector machines [95] (and its extensions like ensemble of support vector machines [80], [64], [66]), random forest [1], common spatial patterns (CSP), CSP combined with fuzzy classifier [76], linear discriminant analysis (LDA) [34] and its Fisher's and Bayesian's versions (FLDA, BLDA) [40] or xDAWN algorithm [84]. Despite the linear methods are considered as the most suitable to classify EPs [62], recently, the nonlinear algorithms grow in popularity. These methods are based on convolutional neural networks [84], [99], [65], self-organizing fuzzy neural networks [18], extreme learning machines [58], [50], or deep learning techniques which utilize neural networks or CapsNet model [71].

Because the methods applied within this thesis can be positioned in a group of classifiers which share some foundations of linear discriminant analysis and its extensions, it is eligible to describe the core of LDA in more detail.

LDA ability to discriminate between two classes is based on statistical information about these classes: their means, and the within-class and between-class distances [107], [94]. The method searches for the optimal transformation hyperplane (given by \mathbf{w}), basing on the training data, which can be used later to classify new, test data according to

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b, \quad (1.1)$$

where \mathbf{x} denotes the test sample to be projected and classified. When LDA is used as a classifier in BCI, the function values defined by (1.1) are calculated for the responses to all flashes. The flash for which the highest score has been obtained is regarded as the target one. It is obvious that for such purpose the value of parameter b is meaningless, and only the weights vector \mathbf{w} must be calculated.

It was assumed that learning samples of all classes are gathered in $\mathbf{X}_{W \times KM} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_W]^T$, where \mathbf{x}_o is the o^{th} observation, and the labels of classes

in \mathbf{y} , where y_o indicates the label (1 or 2) of the o^{th} observation in \mathbf{X} . In turn, \mathbf{Z}_i gathers observation data of the i^{th} class ($i = 1, 2$). The number of samples (observations) in each class is given as $n_i = |\mathbf{Z}_i|$.

In order to find the optimal \mathbf{w} the method performs simultaneous minimization of the within-class scatter matrix \mathbf{S}_w and maximization of the between-class scatter matrix \mathbf{S}_b using formula

$$\mathbf{w} = \underset{\mathbf{w}}{\operatorname{argmax}} \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}, \quad (1.2)$$

where the scatter matrices are defined as

$$\begin{aligned} \mathbf{S}_b &= (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T \\ \mathbf{S}_w &= \sum_{i=1,2} \sum_{\mathbf{x} \in \mathbf{Z}_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T \end{aligned}$$

and i^{th} class mean is given as follows:

$$\mathbf{m}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in \mathbf{Z}_i} \mathbf{x}, \quad i = 1, 2$$

If \mathbf{S}_w is not singular, \mathbf{w} can be solved by applying the eigendecomposition to $\mathbf{S}_w^{-1} \mathbf{S}_b$. As a result \mathbf{w} can be calculated using

$$\mathbf{w} = \mathbf{S}_w^{-1} (\mathbf{m}_2 - \mathbf{m}_1) \quad (1.3)$$

If \mathbf{S}_w is singular, additional techniques of regularization are required.

Although the currently used classifiers (e.g. based on LDA) are part of the state of the art solutions, they still require large amount of learning data to work efficiently with the feature vectors of a high dimensionality used in BCI (as mentioned in Section 1.2). Significant limitations of these classifiers stem from the fact that their performance is affected by the insufficient number of available brain responses (observations) in the learning datasets. The low number of observations with regard to the large number of features leads to ill-conditioned calculations and by thus to improper solution. As an example, for LDA, in such conditions a within-class scatter matrix \mathbf{S}_w tends to be singular and therefore its inversion, required to determine the LDA classifier, is unrealizable without additional techniques of regularization of this matrix.

There have been numerous attempts suggested to circumvent the negative consequences of the mentioned problem. They usually concern a modification or an alternative estimation of the construct equivalent to LDA's \mathbf{S}_w which have a tendency to become singular on a certain point of the mentioned algorithms. The basic approach involves addition of a small, constant number $\mu > 0$ to all diagonal elements of the within-class-scatter matrix [26]. The resulting, nonsingular matrix is given as $\mathbf{S}_w + \mu \mathbf{I}_m$, where \mathbf{I}_m is an identity matrix. However, this method requires a search for

an optimal μ value (too high value can cause loss of information, on the contrary too low value will not be effective). In [5] the authors proposed to reduce the dimension by means of principal component analysis (PCA) prior to application of LDA, so that the within-class scatter matrix used in LDA is no more singular. The solution, however, is fraught with high risk of losing some discriminant information during PCA processing. Other techniques involve Penalized LDA [36], Pseudo-inverse LDA [27], [63], and LDA/GSVD (generalized singular value decomposition) [77]. All the referred methods try to improve the classical LDA by using different forms of regularization or dimension reduction. However, limiting their operation to feature vectors matrices described in Fig. 1.3, containing the brain responses triggered by rather small number of flashes (produced during learning sessions which cannot last too long because of the users fatigue), they leave the fundamental problem of a low number of learning data unresolved. Moreover, using the vectors located at exactly the same position with respect to all the individual flashes almost precludes finding an effective solution to the varying latencies problem. Similarly, removal of the nonstationary super-Gaussian spiky noise seems to be rather difficult using this approach.

Therefore, a different approach to improve the BCI systems accuracy is the objective of this study. Instead of using more and more advanced classifiers, application of signal processing methods, and particularly of the spatio-temporal filtering (STF) [61] is proposed. The STF filter bears some similarity to the LDA classifier. It operates on similar input data, covering the whole responses of the assumed length, and it is constructed by maximization of the similar objective function (like (1.2)). However, the fundamental difference consists in different construction of the covariance matrix to be inverted (analogical to \mathbf{S}_w). Whereas for LDA it is estimated on the basis of rather small number of observations (W , corresponding to the number of flashes during the learning sessions), for STF the whole recorded EEG signal can be exploited for this purpose. Thus the fundamental difference between the approaches referred and the proposed one relies on the use of the whole EEG signals recorded as a source of information instead of the brain responses to a limited number of flashes, only.

1.4 Aim of the thesis

The aim of the thesis is to apply spatio-temporal filtering to EEG data processing for brain responses classification, and by the proper adjustment of the method to achieve a substantial improvement of the classification accuracy, allowing for more successful operation of the BCI systems involved.

The STF method was first proposed for detection of low amplitude fetal QRS complexes. In this application, construction of the STF filter template was based on maximization of the filter response to the desired signal (segments containing the fetal QRS complexes) and minimization of its response to noise (the other signal

segments). Followingly, the filter weights were calculated by a generalized eigendecomposition of the corresponding two covariance matrices.

However, this approach, very effective in the application considered, does not necessarily lead to a successful processing and classification of the EEG signals. It results from much lower ratio of the desired evoked potentials to noise, and the eigendecomposition method sensitivity to the latter. To overcome this problem, the optimized objective function should be modified, so as to replace the eigendecomposition based calculations with the ones that are more immune to noise. Moreover, a proper method to reject the spiky super-Gaussian artifacts before these calculations should be developed. Concluding, the STF filter learning phase should be modified to be more effective for the tackled type of data. Apart that, the filter application can also naturally be exploited for overcoming the problem of the brain responses varying latencies. To this end, the applied rules of STF output interpretation should be modified. Instead of the filter application to brain responses gathered in the matrix illustrated by Fig. 1.3, only, it should be calculated for all necessary signal segments. This would allow for a search of the filter highest response, possibly shifted because of the changed latency. Accomplishment of these aims should lead to a significant improvement of the BCI systems accuracy in the brain responses classification.

1.5 Thesis

Based on the performed analysis of the factors limiting the accuracy of evoked potentials classification in BCI systems, the following thesis was formulated: it is possible to apply the method of spatio-temporal filtering to brain responses enhancement and classification, and by the proper modification of its learning phase and the applied interpretation rules a significant growth of the classification accuracy can be achieved.

1.6 Overview of the thesis content

The rest of the thesis is organized as follows: Chapter 2 contains descriptions of the databases used to evaluate the proposed solutions, and related BCI experiment designs. The developed methods details are gathered in Chapter 3. Subsequently, the whole Chapter 4 is dedicated to the problem of artifacts disturbing the EEG recordings. It presents also the proposed algorithms which aim in the super-Gaussian artifacts detection and their rejection. The results are presented and discussed in Chapter 5. Finally, Chapter 6 summarizes the thesis.

Chapter 2

BCI experiments and databases

This chapter introduces rules of the BCI experiment which are explained on the basis of the "smart home" control system [40]. Furthermore, two databases ("Item selection" and "speller") selected to verify the algorithms proposed in this thesis, and related experiments used to create these databases are described. During search for the databases the emphasis was put on possibility to compare with other researchers' algorithms. Both aforementioned databases have been frequently utilized to test different methods applied to EP detection, therefore, the results one can compare to are abundant in the literature. Moreover, these datasets were part of BCI competitions: 2019 IFMBE Scientific Challenge and BCI Competition III Challenge 2004, respectively.

2.1 Explanation of the BCI experiment

All BCI experiments are designed according to the common rules which utilize a selective response of the brain to the visual stimulus which is consciously perceived by the observer. During a single, elementary segment of the experiment (so called run), the user faces a screen filled with the different objects (letters, signs or pictures etc.), and is focused on the selected one. Fig. 2.1 offers a glimpse of the "smart home" experiment conditions where the user interface contains 6 household appliances, and the user focuses on the telephone. During a run, all 6 objects are flashed one at a time in a random order [40]. In this case, each flash lasts 100 ms and is followed by a constant 400 ms period when all the objects remain dim. The flash of the item the user is focused on (telephone) is called as the target event, flashes of the other items are the nontarget ones. It is expected that the target elicits higher response of the brain than the nontargets. Once all the objects have been flashed (1 target and 5 nontargets), the elicited brain responses can be compared. It is accomplished by evaluating the outputs of the classifier applied to these 6 events. The one with the highest score indicates the picture classified as a target. However, due to a very low amplitude of brain responses to the visual stimuli, it is very rare to achieve proper discrimination at this early step of the experiment. Therefore, the experiment involves multiple repetition of such runs of flashes and averaging the outputs for every individual picture. Typically, taking into account more subsequent runs, the

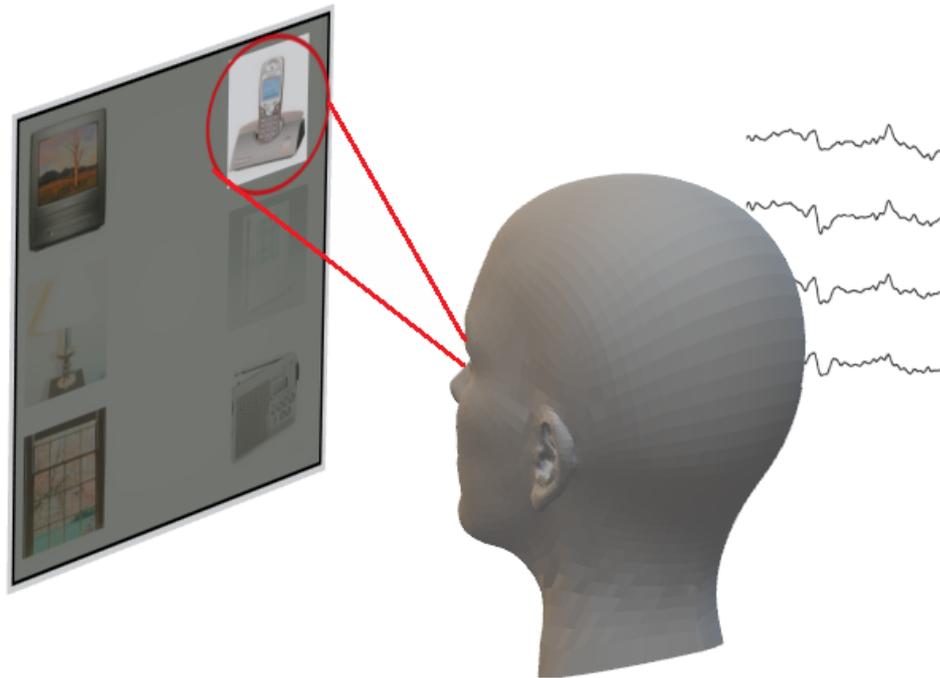


FIGURE 2.1: Illustration of BCI experiment; a moment when visual stimulus (target) is captured.

averaged response to a target stimulus is increased when compared to the averaged responses to the nontarget ones. The number of the averaged runs that is necessary to reliably differentiate a target from nontargets is highly dependent on the efficiency of the classifier, but also on the quality of the data. The series of runs, whose count is 20 in this case, is called as a block. During the experiment, this process (block) is repeated for every object (on the screen) regarded as a target. In turn, all blocks, each having assigned a unique object as a target (6 in the considered example), form a session. Usually, depending on the experiment design, available time and resources, the number of blocks within a session can differ and include more blocks of the same type (with the same object assigned as a target).

2.2 Description of the BCI "Item selection" experiments

Experiments which originally allowed to prepare the dataset utilized the virtual reality scene filled with 8 objects (books on a shelf, a radio on top of a dresser, a printer on a shelf, a laptop on a table, a ball on the ground, a corkboard on the wall, a wooden plane hanging from the ceiling, and a picture on the wall) [88]. Each of 15 participants, diagnosed with autism spectrum disorder, recorded 7 sessions which were split on the training and testing parts. Data were acquired using g.Nautilus

system with 8 electrodes positioned at C3, Cz, C4, CPz, P3, Pz, P4, and POz. The virtual environment was presented using the Oculus Rift Development Kit 2 headset. During a run all the objects were flashed, one at a time in a random order, according to the experiment scheme typical for BCI experiments described in Section 2.1. Each flash lasted 100 ms and was followed by 200 ms of interstimulus interval. Within one block containing multiple runs, the user was uninterruptedly focused on a selected item, assigned as the target in a given block. The targets were changed in consecutive blocks to involve all items visible in the scene several times each.

2.3 "Item selection" database structure

Database contains 15 packages of data, one for every participant of the experiments. Each such a package is split on the learning and testing data subsets. Learning subset for a given subject stores 7 sessions and one session contains 20 blocks. A block, which is a part of the experiment when the user is focused on the same target object, consists of 10 runs. In turn, a run gathers brain responses for single flashes of every items available in the user interface scene (7 nontargets and 1 target). The structure of testing subset is analogous. The only difference concerns the number of runs included in one block. On the contrary to the learning part, this number is not constant and varies within range 4-10.

2.4 Description of the BCI "speller" experiments

The "speller" experiment aims in typing the letters or signs using the interface shown in Fig. 2.2. Similarly to the "smart home" experiment, during a block the user is focused on the selected character (out of 36 available). However, the way the letters or signs are flashed is different. Here, the whole columns or rows are flashed one at a time. Therefore, within one run there are 12 events (6 flashes of columns and 6 flashes of rows). Because every character belongs to a given column and a given row, the flash of either this particular column or row is the target event. As a consequence, runs gather 2 target events and 10 nontarget ones. The brain responses elicited by all events can be compared after the run is completed. The classifier is applied to 12 events (2 targets and 10 nontargets). As every of the characters is represented two times (column, row), its final score is an average value of the column score and the row score. Having all 36 characters outcomes calculated, the one with the greatest score is classified as the target character.

2.5 "Speller" database structure

The "speller" database was prepared for the purpose of the BCI Competition III Challenge 2004 [9]. Originally, two subjects recorded 5 sessions, during which they were asked to spell the selected character (letter/sign). Each session contains multiple

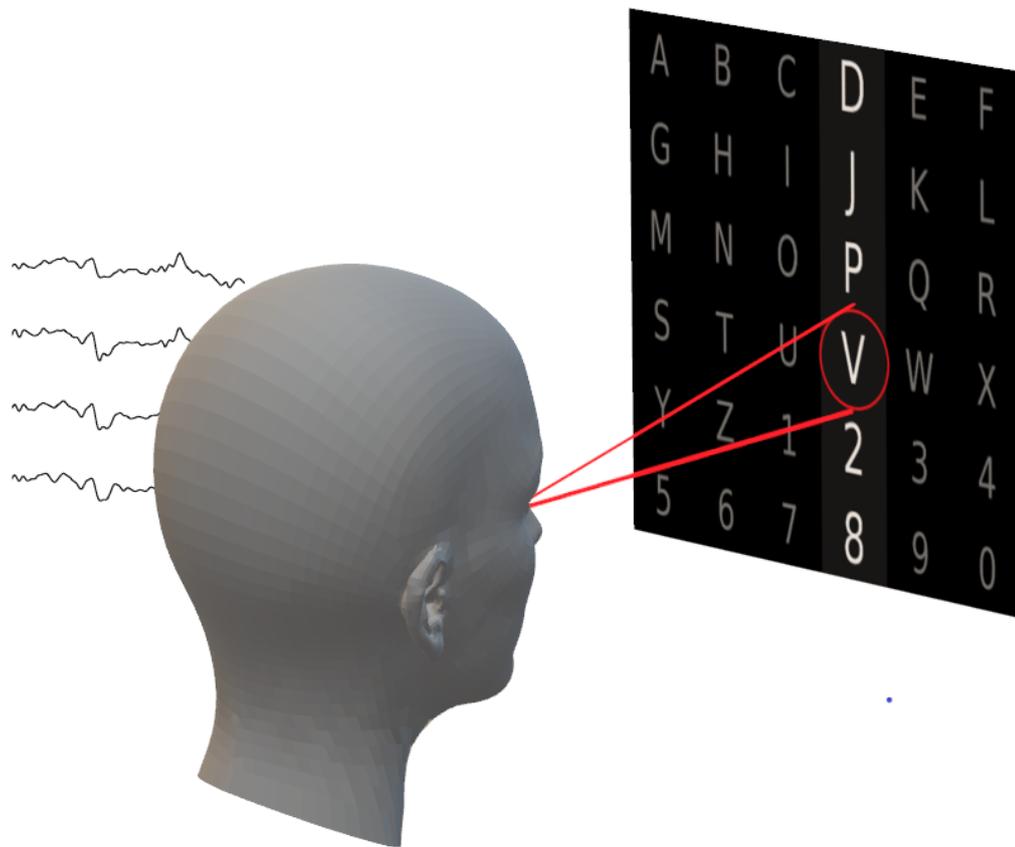


FIGURE 2.2: Illustration of BCI "speller" experiment; a moment when visual stimulus (target) is captured.

blocks, and similarly as in "smart home" experiments a single block consists of repeatable runs. Every block contains 15 runs, where one run gathers brain responses to all 12 flashes. The BCI organizers split the data of every subject to the learning and testing subsets, containing 85 and 100 blocks, respectively.

Chapter 3

Development of a filter for spatio-temporal patterns enhancement

Since the desired component of EP, to be detected in BCI experiments, is of a known shape (or can be determined by averaging of a large number of repeated brain responses, like shown in Fig. 1.2), the methods of matched filtering (MF) can pose a good alternative [85]. MF has been used in numerous research studies [31], [92], [105], primarily, to detect a known waveform in a signal submerged in white Gaussian noise. However, detection of this waveform's presence in case of correlated noise is a more demanding task which requires additional step - prewhitening of the noise. This step, along with the simultaneous enhancement of the searched waveform component and suppression of correlated noise is provided by use of generalized matched filtering (GMF) [54]. However, the analysis of the temporal or spatial information on EP separately entails a difficulty to properly capture the properties of evoked potential due to its three dimensional nature. Further increase of capabilities of GMF is provided by spatio-temporal filtering (STF) which was originally developed for enhancement of the fetal ECG [61], and applied recently also to the atrial arrhythmia waveforms detection in the ECG [32]. The advantage of the method stems from the possibility of simultaneous analysis of temporal and spatial patterns existing in the multichannel signal what was not applicable neither in MF, nor GMF.

This section presents the matched filtering (MF) based methods, which allow to utilize temporal information only as well as its extensions, beginning with the spatio-temporal filtering (STF), which enable a simultaneous processing of temporal and spatial information.

3.1 Matched filtering

Matched filtering (MF) aims at detection of a component of a known shape $s(k)$, submerged in stationary, zero mean, white Gaussian noise $w(k)$. For detection purpose two hypotheses which relate to the content of the analyzed signal $x(k)$ are considered.

$$\begin{aligned} H_0 : x(k) &= w(k) \\ H_1 : x(k) &= s(k) + w(k) \end{aligned} \quad (3.1)$$

MF allows to discriminate between H_0 ($x(k)$ contains noise alone) and H_1 ($x(k)$ represents the desired component submerged in noise), considering the assumed detection threshold.

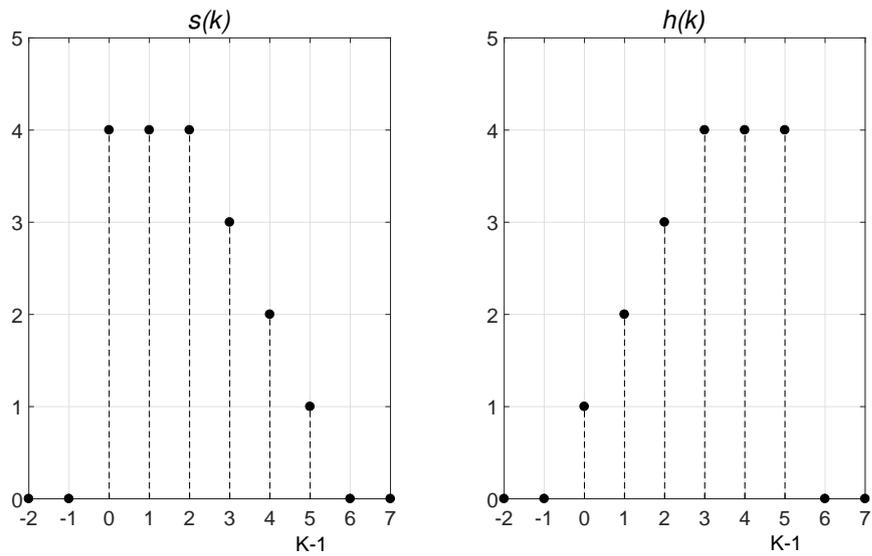


FIGURE 3.1: Exemplary desired component $s(k)$ and matched filter impulse response $h(k)$, for $K=6$.

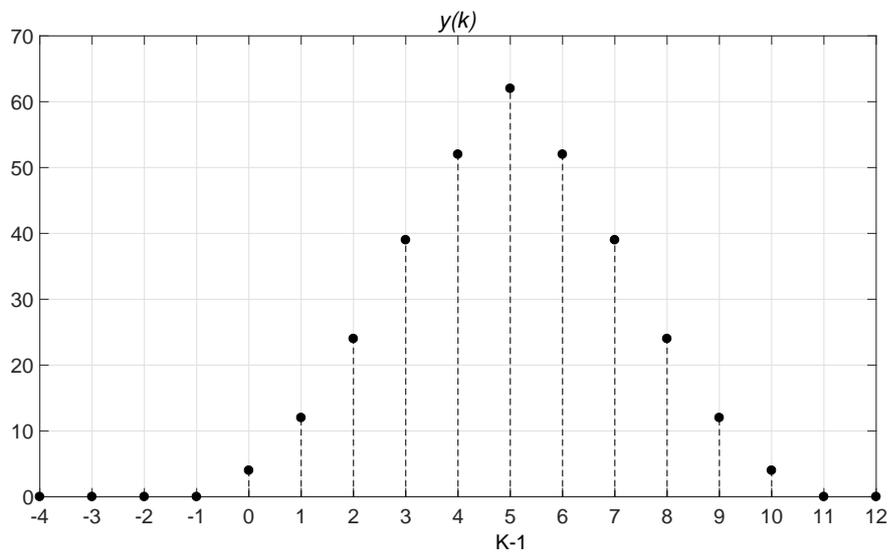


FIGURE 3.2: Result of the filtering of $s(k)$ with MF filter with impulse response $h(k)$.

The matched filter impulse response is defined as the desired signal reversed in time

$$h(k) = \begin{cases} s(K-1-k), & 0 \leq k < K; \\ 0, & \text{elsewhere,} \end{cases} \quad (3.2)$$

where K is the desired signal length.

A very important property of matched filtering is that it maximizes the SNR ratio once hypothesis H_1 is true [54].

Using vector notation: $\mathbf{h} = [h(K-1), h(K-2), \dots, h(1), h(0)]^T$, $\mathbf{s} = [s(0), s(1), \dots, s(K-1)]^T$ and $\mathbf{w} = [w(0), w(1), \dots, w(K-1)]^T$, SNR can be expressed as the energy of the filter response to the deterministic desired component divided by the expected value of the noise variance at the filter output

$$SNR = \frac{(\mathbf{h}^T \mathbf{s})^2}{E[(\mathbf{h}^T \mathbf{w})^2]} = \frac{(\mathbf{h}^T \mathbf{s})^2}{\mathbf{h}^T E[\mathbf{w} \mathbf{w}^T] \mathbf{h}}. \quad (3.3)$$

For a diagonal covariance matrix of a zero mean white Gaussian noise ($E[\mathbf{w} \mathbf{w}^T] = \sigma^2 \mathbf{I}$, where \mathbf{I} is the identity matrix), SNR takes form [54]

$$SNR = \frac{(\mathbf{h}^T \mathbf{s})^2}{\mathbf{h}^T \sigma^2 \mathbf{I} \mathbf{h}} = \frac{(\mathbf{h}^T \mathbf{s})^2}{\sigma^2 \mathbf{h}^T \mathbf{h}}. \quad (3.4)$$

For $\mathbf{h} = \mathbf{s}$ (see the definitions, above) the maximal value

$$SNR = \frac{1}{\sigma^2} \mathbf{s}^T \mathbf{s}, \quad (3.5)$$

can be achieved, equal to the desired signal energy divided by the noise variance.

Exemplary impulse response $h(k)$ of the matched filter designed to detect the given desired component $s(k)$ is presented in Fig. 3.1. In turn Fig. 3.2 shows the result of matched filtering which is the convolution of $s(k)$ and $h(k)$.

As it should have been expected, the maximal value at the filter output was obtained for $k = K - 1$.

3.2 Generalized matched filtering

When the noise is of zero mean but not white, performance of a matched filter is deteriorated. It is caused by an increase of covariance between noise and the matched filter template, and as a result, the growth of denominator in (3.3). To overcome the problem, before application of matched filtering, power spectrum of the noise can be estimated and the proper pre-whitening filter applied [79]. The method of generalized matched filtering applies a more straightforward approach: the SNR defined by (3.3) is used as the objective function, maximized to derive the formula for the

filter impulse response

$$Q(\mathbf{h}) = \frac{(\mathbf{h}^T \mathbf{s})^2}{E[(\mathbf{h}^T \mathbf{w})^2]} = \frac{\mathbf{h}^T (\mathbf{s}\mathbf{s}^T) \mathbf{h}}{\mathbf{h}^T \mathbf{C}_w \mathbf{h}} \quad (3.6)$$

with a crucial difference regarding \mathbf{w} which denotes the colored noise, instead of the white one, as in (3.3).

If covariance matrix \mathbf{C}_w is of full rank, a solution maximizing this objective function, and adequately the SNR at the filter output is given by

$$\mathbf{h} = \mathbf{C}_w^{-1} \mathbf{s}. \quad (3.7)$$

3.3 Spatio-temporal filtering

Whereas MF and GMF are useful to enhance the desired component in the single channel, they overlook the spatio-temporal dependencies which are inherent in the multichannel signal. Such interrelation between the channels stems from the fact they usually are different linear combinations of the same source signals captured by the electrodes placed in proximity to each other. Spatio-temporal filtering was designed to enable processing of the spatio-temporal vectors created on the basis of the current and adjacent time samples from all available signal channels. This approach, which takes benefit of the inter-channel relations among different signal components, is particularly effective in noise minimization.

For the M channel signal vector given as $\mathbf{x}(k) = [x_1(k), x_2(k), \dots, x_M(k)]^T$, its spatio-temporal representation is defined as the extended vector [61]

$$\mathbf{x}^{(k)} = \begin{bmatrix} x_1(k - J\tau) \\ \dots \\ x_M(k - J\tau) \\ x_1(k - (J - 1)\tau) \\ \dots \\ x_M(k - (J - 1)\tau) \\ \dots \\ x_1(k + J\tau) \\ \dots \\ x_M(k + J\tau) \end{bmatrix} \quad (3.8)$$

whose size is $p = (2J + 1)M$ (it contains $2J + 1$ time samples from M channels available). For $J = 0$, obtained is the original vector $\mathbf{x}(k)$, i.e. M values corresponding to the respective M channels recorded. Such a vector can be regarded as a spatial pattern of the electric potentials distribution on the head of the BCI system user. For $J > 0$, many such spatial patterns, corresponding to different moments of time,

are contained in the extended vector $\mathbf{x}^{(k)}$. Thus the extended signal vectors can be regarded as the spatio-temporal patterns referred to in the title of the dissertation.

Parameters τ and J indicate which samples undergo filtering operation expressed as

$$y(k) = \mathbf{h}^T \mathbf{x}^{(k)}. \quad (3.9)$$

Parameter τ introduces a kind of decimation to reduce the size of spatio-temporal vector $\mathbf{x}^{(k)}$. The initial experiments allowed to make an assumption that the resulting frequency f_s/τ (where f_s is the original sampling frequency) should not be decreased below 40Hz. Thus, τ can be calculated using following formula: $\tau = \left\lfloor \frac{f_s [Hz]}{40} \right\rfloor$, where $\lfloor \cdot \rfloor$ denotes the largest integer not greater than the argument. When setting the value of J one should have regard to a length T_p of the signal pattern being detected. As the filter length ($2J\tau$) should be greater than T_p , J can be determined according to: $J = \left\lceil \frac{f_s \cdot T_p}{2 \cdot \tau} \right\rceil$, where $\lceil \cdot \rceil$ denotes the smallest integer not smaller than the argument.

3.3.1 STF filter construction

Procedure of STF filter construction (learning phase of STF) requires to indicate the signal patterns to be maximized by the filter and those unwanted, to be suppressed. The former, in BCI, involves the target responses. It is assumed that a target brain response starts at the target flash onset, whose location is denoted as k_i (where $i = 1, 2, \dots, I_t$; I_t is the total number of target flashes), and ends at $k_i + 2J\tau$. This time segment corresponds to the spatio-temporal vector $\mathbf{x}^{(k_i+J\cdot\tau)}$ which should be of the highest amplitude on the filter output. Because the filter response is given in form of a peak with a finite slope, more samples around the target predicted location are selected to be magnified. Therefore, the set containing the spatio-temporal vectors to be magnified can be defined as follows [60]

$$\Psi_t(\delta) = \{k \mid |k - (k_i + J \cdot \tau)| \leq \delta \cdot f_s, i = 1, 2, \dots, I_t\}, \quad (3.10)$$

where δ limits the assumed time range. The vectors to be suppressed should contain time samples distant from the $\Psi_t(\delta)$. Therefore, the auxiliary set $\Psi_t(\Delta)$, whose argument $\Delta > \delta$, can be used to define the set containing the vectors to be suppressed

$$\Psi_s(\Delta) = \Psi - \Psi_t(\Delta), \quad (3.11)$$

where Ψ denotes the set of all time indices for which vectors $\mathbf{x}^{(k)}$ have been constructed. Illustration of these sets along with the visual explanation of the parameters δ and Δ is shown in Fig. 3.3.

The influence of δ value on the shape of the averaged filter response is presented in Fig. 3.4, where the averaged filter responses were plotted. Black plot represents the response for the favorable value of δ . When this parameter is decreased two times, the response is slightly thinner and a little decreased. On the contrary, when

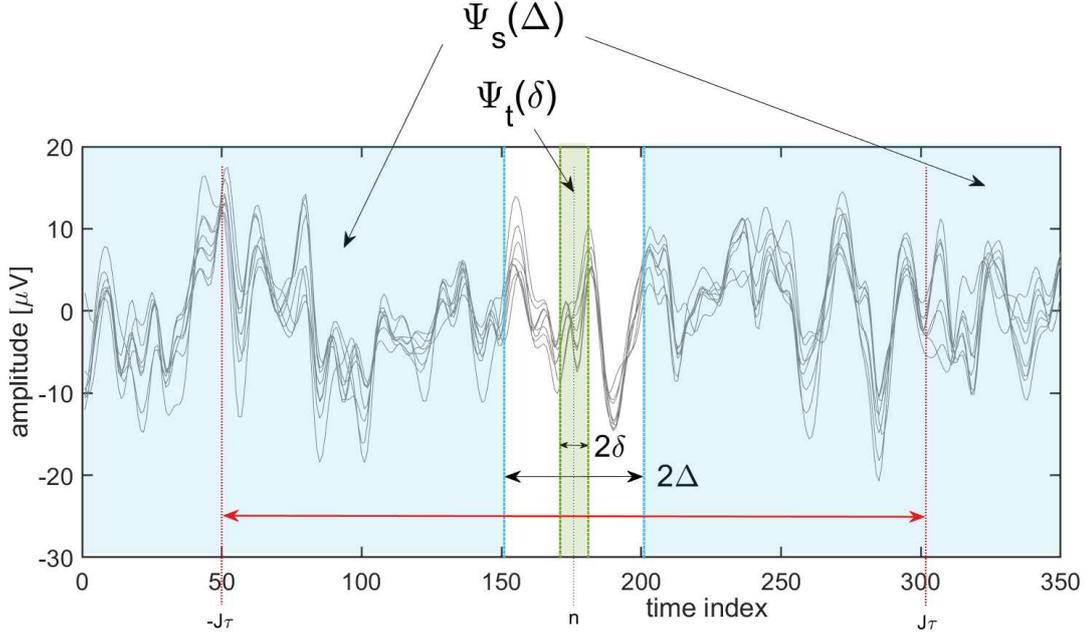


FIGURE 3.3: Visual explanation of parameters δ and Δ on the picture showing 1.4s segment with a target response from all 8 channels. 2δ , marked in green, defines the width of the window containing time indices of the set of vectors $(\mathbf{x}^{(k)}, k \in \Psi_t(\delta))$ to be enhanced. Blue area on both sides shows the indices of vectors $\mathbf{x}^{(k)}$ to be suppressed contained in set $\Psi_s(\Delta)$. The gap between green and blue areas allows to form the proper shape of the STF filter responses to target flashes (see Fig. 3.4). The red arrow indicates a range of indices of the vector $\mathbf{x}^{(n)}$ to be enhanced.

δ is doubled, the response is much wider and contains double peaks with a kind of an inconvenient valley in the middle. These differences confirm the necessity to use the proper value of δ .

As the STF filter should maximize the energy of the $\mathbf{x}^{(k)}$ patterns indicated by set $\Psi_t(\delta)$ and suppress those indicated by $\Psi_s(\Delta)$, the following objective function is maximized

$$Q(\mathbf{h}) = \frac{\frac{1}{|\Psi_t(\delta)|} \sum_{k \in \Psi_t(\delta)} (\mathbf{h}^T \mathbf{x}^{(k)})^2}{\frac{1}{|\Psi_s(\Delta)|} \sum_{k \in \Psi_s(\Delta)} (\mathbf{h}^T \mathbf{x}^{(k)})^2} \quad (3.12)$$

where $|\cdot|$ denotes the set cardinality.

The spatio-temporal vectors to be enhanced are gathered in matrix $\mathbf{A}_{p \times n_a} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{n_a}]$, in turn, the vectors to be suppressed in matrix $\mathbf{B}_{p \times n_b} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_{n_b}]$ where $n_a = |\Psi_t(\delta)|$ and $n_b = |\Psi_s(\Delta)|$. Thereby, every single sample in either of the sets is used to create a spatio-temporal vector of size p .

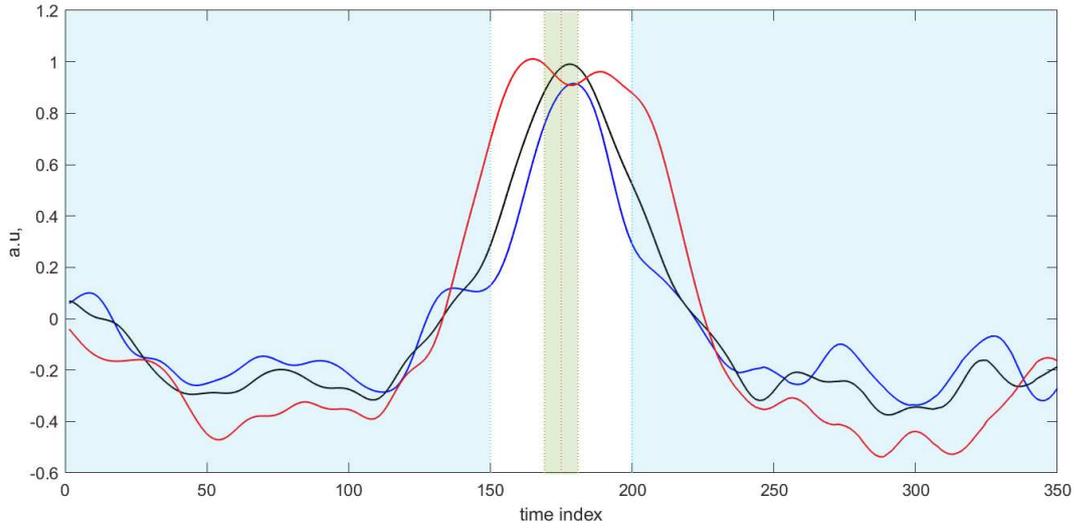


FIGURE 3.4: Illustration of the impact of δ on the averaged filter response (over all runs within an exemplary block). Black, blue, and red plots present the averaged filter responses for optimal $\delta = 20ms$ applied in the experiments, δ two times lower, and two times greater, respectively.

As a result the objective function can be changed accordingly

$$Q(\mathbf{h}) = \frac{\frac{1}{n_a} \|\mathbf{h}^T \mathbf{A}\|^2}{\frac{1}{n_b} \|\mathbf{h}^T \mathbf{B}\|^2} = \frac{\mathbf{h}^T \mathbf{C}_a \mathbf{h}}{\mathbf{h}^T \mathbf{C}_b \mathbf{h}}, \quad (3.13)$$

where $\mathbf{C}_a = \frac{1}{n_a} \mathbf{A} \mathbf{A}^T$ and $\mathbf{C}_b = \frac{1}{n_b} \mathbf{B} \mathbf{B}^T$. Construction of \mathbf{C}_b based on the larger set $\Psi_s(\Delta)$ enables to avoid the problem of matrix singularity mentioned earlier in Section 1.2.

Objective function $Q(\mathbf{h})$ has the form of a Rayleigh quotient; in order to maximize it, the stationary points are determined according to

$$\frac{\partial Q(\mathbf{h})}{\partial \mathbf{h}} = 0. \quad (3.14)$$

As a result of further transformations [61], a solution based on the generalized eigen-decomposition formula is obtained

$$\mathbf{C}_a \mathbf{h} = \lambda \mathbf{C}_b \mathbf{h}. \quad (3.15)$$

By solving it, the STF filter coefficients \mathbf{h} can be determined as the generalized eigenvector that corresponds to the greatest eigenvalue [61]. However, \mathbf{h} determined in such an operation is affected by the sign ambiguity of the eigenvector (\mathbf{h} equals \mathbf{v}_1 or $-\mathbf{v}_1$) [11]. This concern is solved by applying both version of filters to the spatio-temporal vectors stored in \mathbf{A} and checking for which one the averaged filtered

response is positive.

This original version of STF, which incorporates the eigendecomposition based calculations, will be denoted as OSTF.

3.4 Generalized spatio-temporal matched filtering

Matrix \mathbf{A} storing the vectors to be enhanced (defined in Section 3.3.1) contains not only the desired signal pattern but also noise. Whereas enhancement of the former is desirable, the filter determined on the basis of the objective function defined by (3.12) enhances also the latter. In order to reduce this effect the vectors stored in \mathbf{A} can be averaged as follows

$$\bar{\mathbf{a}} = \frac{1}{n_a} \sum_{i=1}^{n_a} \mathbf{a}_i. \quad (3.16)$$

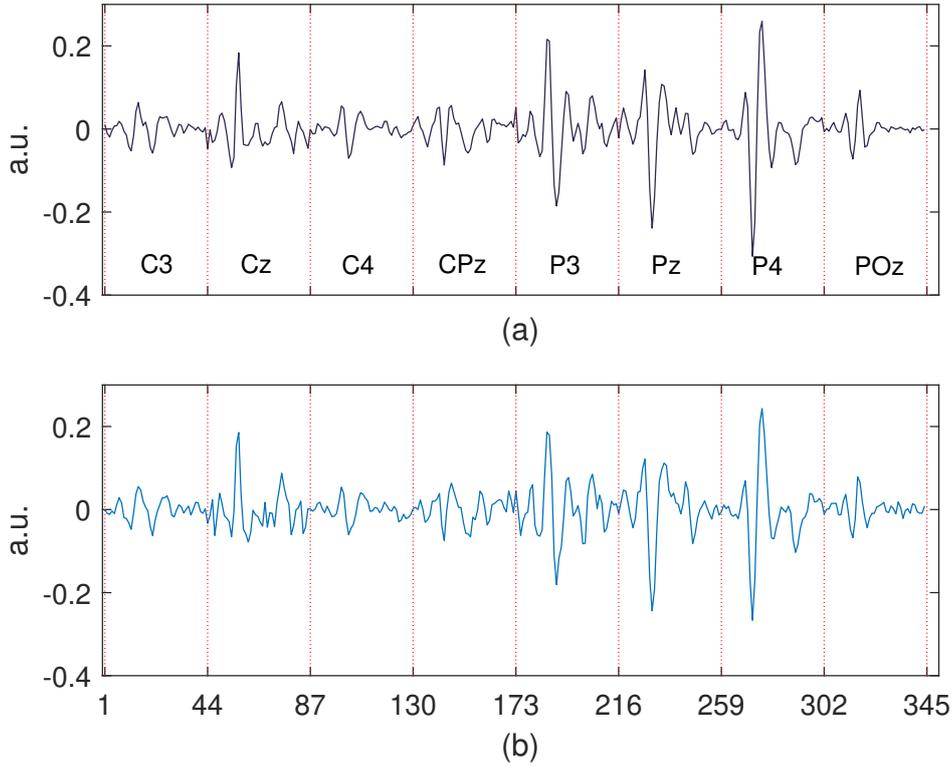


FIGURE 3.5: Comparison of the filter templates constructed for OSTF (upper plot (a) in dark blue) and GSTMF (lower plots (b) in light blue). Red vertical lines separate the segments related to a given channel each containing $2J + 1 = 43$ values.

Then, the objective function is defined accordingly

$$Q(\mathbf{h}) = \frac{\mathbf{h}^T (\bar{\mathbf{a}} \bar{\mathbf{a}}^T) \mathbf{h}}{\mathbf{h}^T \mathbf{C}_b \mathbf{h}}, \quad (3.17)$$

and its solution (assuming that \mathbf{C}_b is of full rank) as

$$\mathbf{h} = \mathbf{C}_b^{-1} \bar{\mathbf{a}}. \quad (3.18)$$

This solution allows to construct a kind of a spatio-temporal model of the target responses which is more immune to noise. Computation of the filter above is similar as for the generalized matched filters (defined by (3.7)), therefore, following [60], the method is called as generalized spatio-temporal matched filtering (GSTMF).

Figures 3.5 and 3.6 illustrate the difference between the filter templates constructed for OSTF and GSTMF and the level of filter enhancement introduced by the averaging step added in GSTMF. Whereas the former figure shows rather similar filter templates created for both methods, a significant improvement of the filter template is visible on the latter. Here, the OSTF filter is more noisy than the GSTMF one.

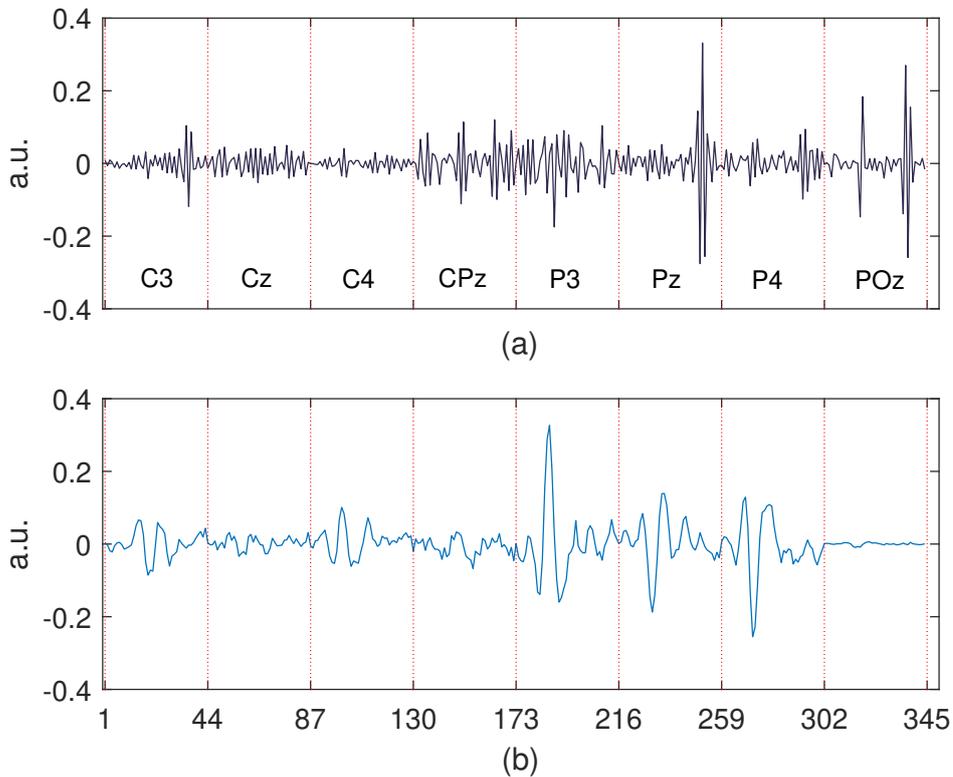


FIGURE 3.6: Comparison of the filter templates constructed for OSTF (upper plot (a) in dark blue) and GSTMF (lower plot (b) in light blue). Red vertical lines separate the segments related to a given channel each containing $2J + 1 = 43$ values.

3.5 Modified spatio-temporal matched filtering

Better classification of the EP requires additional modifications of GSTMF. In the methods described earlier both routine activity of the brain and nontarget responses were treated as noise during the learning phase (both were gathered in set $\Psi_s(\Delta)$ defined by (3.11) in Section 3.3.1). Much as the latter are neglected by the users of BCI, they also generate tiny deflections in the EEG recording and have an impact on the classification results [40]. Therefore, the nontarget responses analysis is incorporated in the modified spatio-temporal matched filtering (MSTMF).

3.5.1 Nontarget responses subtraction

The modification concerns formation of the nontarget pattern (analogical to the one given for the target responses in (3.16)) and its involvement in the filter construction. The set containing the indices of nontarget responses can be defined similarly to the

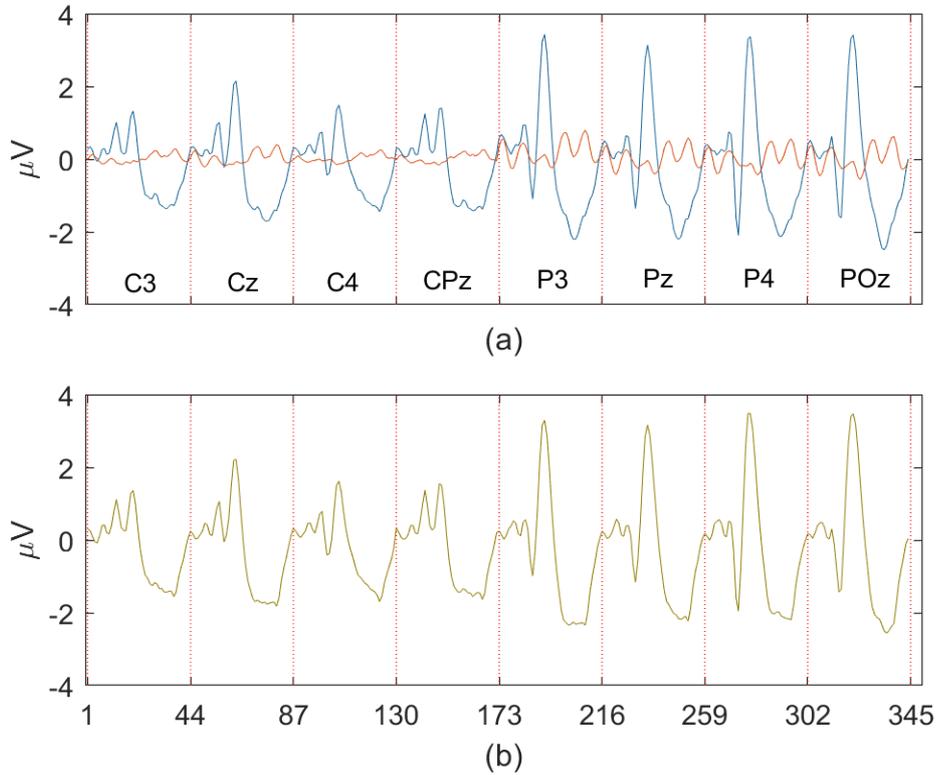


FIGURE 3.7: Averaged target ($\bar{\mathbf{a}}$) and nontarget $\bar{\mathbf{a}}'$ patterns for the first subject are shown in the overlay plot (a), in blue $\bar{\mathbf{a}}$, and in orange $\bar{\mathbf{a}}'$. The difference of these patterns $\bar{\mathbf{a}} - \bar{\mathbf{a}}'$ is presented in plot (b). Red vertical lines separate the segments related to a given channel each containing $2J + 1 = 43$ values. Corresponding channel names are given in the upper plot (a).

target ones (defined by (3.10))

$$\Psi_{nt}(\delta) = \{k \mid |k - (k'_i + J \cdot \tau)| \leq \delta \cdot f_s, i = 1, 2, \dots, I_{nt}\}. \quad (3.19)$$

Here, k'_i stands for the moments when nontarget flashes occurred. The vectors located close to the expected nontarget responses $\{\mathbf{x}^{(k)} \mid k \in \Psi_{nt}(\delta)\}$ are gathered in matrix \mathbf{A}' . Performing averaging of this matrix (analogously as in (3.16)), the reference response pattern $\bar{\mathbf{a}}'$ is constructed. The objective function $Q(\mathbf{h})$ is changed

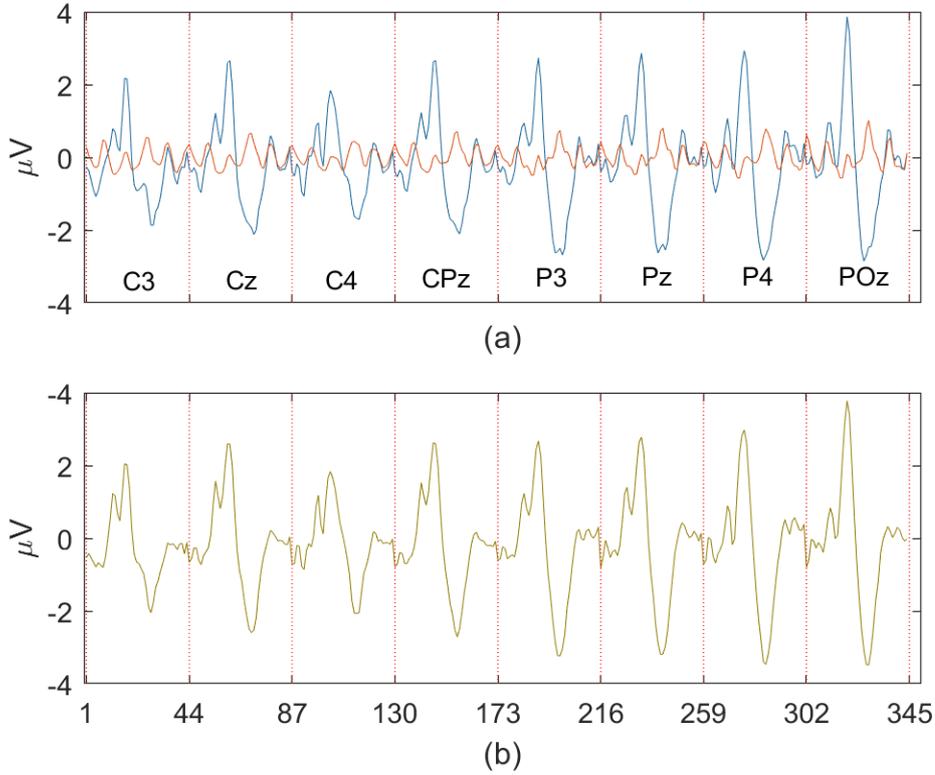


FIGURE 3.8: Averaged target ($\bar{\mathbf{a}}$) and nontarget $\bar{\mathbf{a}}'$ patterns for the second subject are shown in plot (a), in blue $\bar{\mathbf{a}}$, and in orange $\bar{\mathbf{a}}'$. The difference of these patterns $\bar{\mathbf{a}} - \bar{\mathbf{a}}'$ is presented in plot (b). Red vertical lines separate the segments related to a given channel each containing $2J + 1 = 43$ values.

accordingly

$$Q(\mathbf{h}) = \frac{\mathbf{h}^T (\bar{\mathbf{a}} - \bar{\mathbf{a}}') (\bar{\mathbf{a}} - \bar{\mathbf{a}}')^T \mathbf{h}}{\mathbf{h}^T \mathbf{C}_b \mathbf{h}}, \quad (3.20)$$

and its solution is given as

$$\mathbf{h} = \mathbf{C}_b^{-1} (\bar{\mathbf{a}} - \bar{\mathbf{a}}'). \quad (3.21)$$

The exemplary illustrations of the patterns $\bar{\mathbf{a}}$ and $\bar{\mathbf{a}}'$ (plots (a) and (c)) and their differences (plots (b) and (d)) are presented in Figures 3.7 and 3.8 for two different subjects.

3.6 The filter output interpretation

3.6.1 Classical decision rules

If the onset of the o^{th} object i^{th} flash is defined as ${}^o k_i$, its maximal response should occur at ${}^o k_i + J\tau$ (just in the middle of the filtered segment $\in [{}^o k_i, {}^o k_i + 2J\tau]$). Therefore, the following values are sufficient to be stored for the further analysis

$$q_{i,o} = y({}^o k_i + J\tau); \quad o = 1, 2, \dots, I_o; \quad i = 1, 2, \dots, I_r, \quad (3.22)$$

where I_o denotes the number of objects on the screen, and I_r the number of runs performed. This kind of results interpretation is applied also by the classifiers used in BCI; time segments equally distant from the flash onsets undergo classification.

Followingly, $\mathbf{Q}_{I_r \times I_o}$ matrix gathers all scores (filter output values given by $q_{i,o}$) obtained for every object in each of runs performed during the experiment. Therefore, in order to indicate the target object using one run only, it would suffice to read the first row of \mathbf{Q} and find the column with the highest score. However, typically in BCI experiments the results are checked for more runs ($I_r > 1$) in a manner that the flashes' scores for every object are averaged over the chosen number of runs [40]. Such an approach to results interpretation which involves classification of the segments equally distant from the stimuli onsets (contained in vectors $\mathbf{x}^{(k_i + J \cdot \tau)}$) will be called as the classical decision rules (CDR).

3.6.2 Modified decision rules

The described classical decision rules become less effective when the evoked potentials are slightly shifted in time. As a consequence the maximal filter response can occur in a wider time frame in proximity to the expected location. It is therefore justified to search for the maximum in this wider range, not only for one time index like in CDR. However, looking for a maximum, at the filter output obtained for a single brain response, can be risky because of the extremely low SNR (a peak produced by noise could be detected). Accordingly, to determine the target object for I_r runs one needs to average the responses for I_r flashes for every object as follows

$$\bar{y}_o(k) = \mathbf{h}^T \sum_{i=1}^{i=I_r} \mathbf{x}^{({}^o k_i + J \cdot \tau + k)} = \sum_{i=1}^{i=I_r} y({}^o k_i + J \cdot \tau + k). \quad (3.23)$$

If $k = 0$ is equivalent with the sample location $J\tau$ after the stimulus onset, the new range is defined as $k \in [-r_s, r_s]$. A new position of the response maximum k_0 is determined as

$$k_0 = \operatorname{argmax}_{|k| \leq r_s} \bar{y}_o(k), \quad (3.24)$$

and the score for the o^{th} object is given as $\bar{y}_o(k_0)$. The rules are illustrated in Fig. 3.9: one can see that a shift of the maximum (from $k = 0$) does not preclude its proper detection at position k_0 if it is located within $[-r_s, r_s]$.

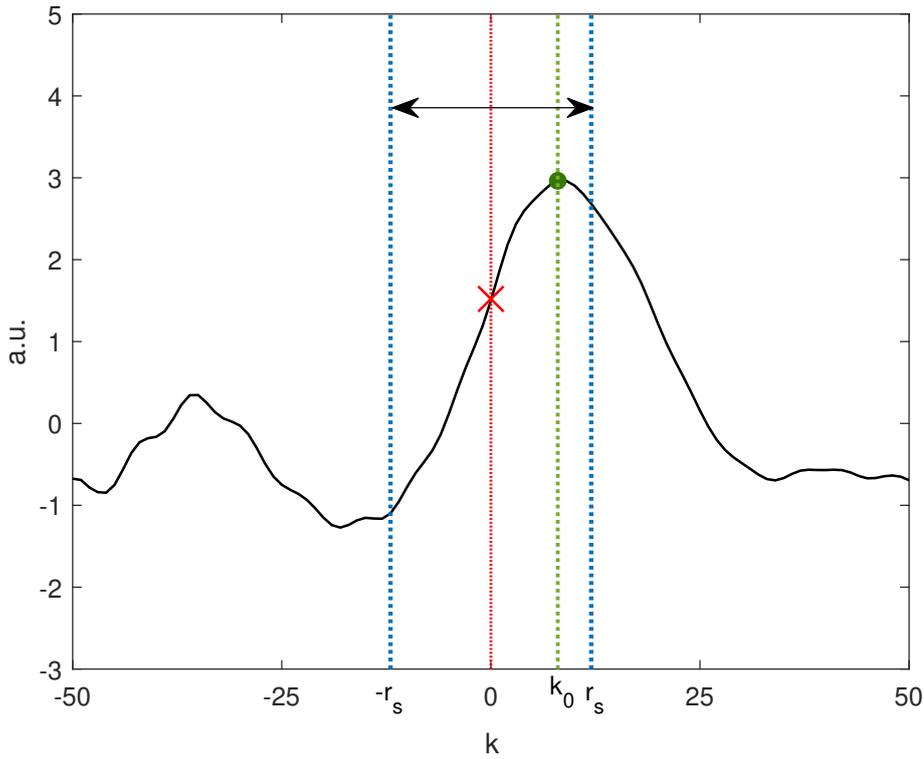


FIGURE 3.9: Visual explanation of the modified decision rules. The averaged filter output is plotted in black. Red, dotted line represents the position used by the classical decision rules ($k = 0$); the maximum of the response in such a case is marked with red X. Blue dotted lines indicate the larger range of search for the maximum of the averaged target response. The position found using the modified decision rules (k_0) is marked with a dotted green line, and the corresponding maximum with a green dot.

3.6.3 MSTMF responses as feature vectors for SVM classifier

The MSTMF output signals are interpreted based on the amplitude, obtained either in one expected position (CDR) or in a larger range around this position (MDR). However, to make use of other features like shape of the response, one can define the features vector for o^{th} object as $\bar{y}_o(k_0 + k), k = -r_s, -r_s + 1, \dots, r_s$, where r_s is the parameter introduced in Section 3.6.2, below (3.23). Therefore, the features are gathered from the segment whose length equals $T_s = 2 \cdot r_s$ with the center at location k_0 .

3.7 Nomenclature of the methods

The section introduces nomenclature of the various modifications of the original STF method, which is called as OSTF. Each method, which represents different accomplishment of the OSTF learning phase, can be recognized by its unique name

followed by + with the type of decision rules applied (CDR or MDR), and information on the denoising algorithm (DCT or ICA) if applied in a given method, in brackets along with information on a subset which underwent denoising, learning subset only (L), or both learning and testing subsets (L&T). As an example: *OSTF + CDR(DCT)* represents the OSTF method where results are interpreted using classical decision rules, and the selected denoising algorithm is based on DCT. Below, listed and explained are only the names of the methods which indicate specific modifications of the algorithm's learning phase:

- *OSTF* - Eigendecomposition based approach defined by (3.15).
- *GSTMF* - Involves averaging of the segments containing the target responses introduced in Section 3.4 by (3.18).
- *MSTMF* - In addition to *GSTMF* the nontarget responses subtraction, defined in Section 3.5.1, is applied.

The only exception from the nomenclature rules described above concerns the method which combines MSTMF based algorithm with an additional classifier e.g. SVM. Because the decision rules type is not applicable here, + is simply followed by the abbreviation of the classifier name. An exemplary method name is *MSTMF + SVM(ICA : L&T)* which applies to MSTMF algorithm using ICA for data denoising, and whose output signals are additionally classified using SVM.

Chapter 4

EEG signals preprocessing for artifacts removal

Problem of noise disturbing the EEG recordings has been investigated very regularly since an acquisition of the brain electrical signals was pioneered. Simple rejection of the affected signal segments is not recommended in BCI, mainly due to the mentioned earlier problem of the limited number of observations available in databases. Therefore, the attempts to suppress the artifacts without suppression of the underlying electrical activity of the brain are very often applied.

One of the basic solutions involves a search for the outlying samples, using the thresholds (positive and negative), and their trimming to the values of these thresholds. The thresholds calculation is based on the statistical information of the signal (e.g. 5th and 95th quantiles [40]). More advanced approaches are based on blind source separation (BSS) technique which presumes that the potential measured on each of the electrodes is a weighted sum of the source signals. Therefore, each of such source signals, including the ones related to noise, is transmitted to all electrodes, however, with different weights dependent on the distance between its source and the electrode on the skull. One of the most popular BSS techniques, commonly used to denoise the EEG signals, is independent component analysis (ICA) (see Appendix B for the details). Different of its implementations have been applied in numerous research to tackle the problem of noise in EEG signals, e.g. Joint Approximate Diagonalization of Eigen-matrices (JADE) algorithm [47], FastICA [67], [74], the algorithm for multiple unknown signals extraction (AMUSE) [4], second-order blind identification (SOBI) [97], Infomax [48]. Such various types of ICA implementations were compared in [19] with respect to ability to remove the myogenic artifacts from sleep EEG. Other BSS based approaches used to denoise EEG data involve canonical component analysis (CCA) [16], [102] and principal component analysis (PCA) [8]. EEG denoising algorithms incorporate also adaptive filtering [72], [23], [38], Wiener filtering (WF) [89], wavelet transform (WT) [56], multiresolution total variation (MTV) and multiresolution weighted total variation (MTV) filtering schemes [28], or empirical mode decomposition (EMD) [35]. Increasing number of noise removal solutions developed recently are based on the deep learning techniques e.g. a 5-layer neural network used to remove ocular artifacts [108], a novel 1D-ResCNN model [90],

DeepSeparator [109], or a novel convolutional neural network model designed to cope with a difficult task of myogenic noise cancellation [110].

Bioelectric signals, and particularly EEG, contain typically artifacts associated with the muscle activity, eye blinking, and electrode displacement or detachment. Their presence may have an adverse impact on the classifier construction, what is examined more closely in the numerical experiments part of this dissertation. Therefore, removal of these noisy components is an important step of the signal preprocessing. The algorithms presented in this chapter are founded on the two different kinds of techniques. First approach assumes a bioelectric signal is a linear mixture of the independent source signals which have their origin not only in the different parts of the brain but also in other organs, muscles etc. In such a case independent component analysis (ICA) [13] can be used to find the noisy source signals which contribute to the large artifacts and to reconstruct the signal without them. Second type of the methods involves blanking the outlying values and interpolation of the new ones. It can be performed in each channel separately or for the two-dimensional matrix formed by the consecutive channels. Such an approach typically applied to image processing is possible because of similarities of the EEG traces collected from the adjacent channels which form a kind of a surface. Therefore, this approach was applied and particularly the method based on the discrete cosine transform (DCT). The algorithm which is founded on the DCT is called in the sections below as DCTBS (discrete cosine transformation based smoothing) [30].

4.1 Types of noise artifacts disturbing the EEG signals

The artifacts disturbing the electroencephalogram can be classified depending on their sources in the body. Most common ones involve ocular, myogenic and cardiac artifacts. However, the noise can arise also from the surrounding conditions unrelated to the physiology of the human body. This chapter presents a characteristics of each type of the mentioned noise, defines its properties which can be helpful for development of the preprocessing steps aiming in the noise removal.

Myogenic artifacts are associated with the muscle activity of the human body. Besides the contractions of the muscles related to the movements of limbs, also activity of the cranial muscles can affect significantly the EEG recordings. The magnitude of electromyogram (EMG) measured from this source is typically much higher than from the neural brain activity, rising above the typical amplitude of EEG neural trace ($\sim 10\mu V$) to even $400\text{-}500\mu V$ and obscures wide range of the spectrum of EEG (up to 200Hz). Besides the coexistence in time of these two components (EMG and EEG), their spectral and also anatomical profiles are roughly similar. Therefore, this type of artifacts is considered as most difficult to be eliminated from the EEG [86]. The level of difficulty is raised when EMG contamination has its origin in the groups of muscles for which the generated noise can have substantially different properties

between the patients (versus other artifacts types like ocular or cardiac which do not exhibit large inter-subject variability) [33].

Artifacts related to eyes movement or blinking are known as ocular or electrooculographic (EOG) [20]. Movements of an eyeball, which can be regarded as a dipole [8], generate the potentials propagating to the skull surface where can disrupt the EEG recordings. Typical EOG artifact is characterized by low frequency, covering usually δ (0.5–4 Hz) and θ (4–8 Hz) waves [104]. Example of their presence in EEG traces is visible in Fig. 4.1.

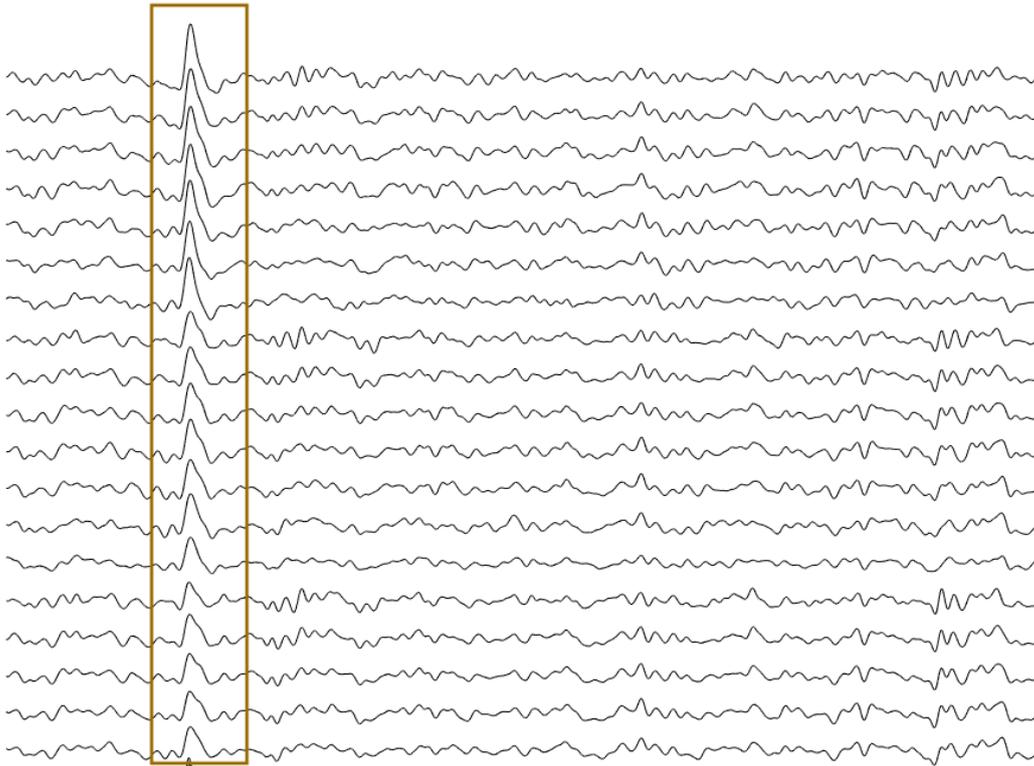


FIGURE 4.1: Exemplary occurrence of the ocular artifacts in EEG recording. The ocular artifact which spread to multiple channels is marked in a brown frame.

Group of cardiovascular artifacts, related to ECG, ballistocardiographic (BCG) or pulse (PA) artifacts, are caused by the cardiac activity of the subject [10], [2]. The physical explanation of transferring these artifacts from the heart to the scalp, where EEG electrodes are placed, involve pulse related slight local movements of the scalp associated with an expansion of the scalp blood vessels [55].

The artifacts whose source is unrelated with the human body electrical activity are known as extrinsic artifacts [49]. This type of noise can be produced in the aftermath of electrode or cable movements or electromagnetic interference passing the area of the EEG data acquisition. Whereas the latter can be rather easily removed by applying the notch filtering, the former is not always present in a form of an

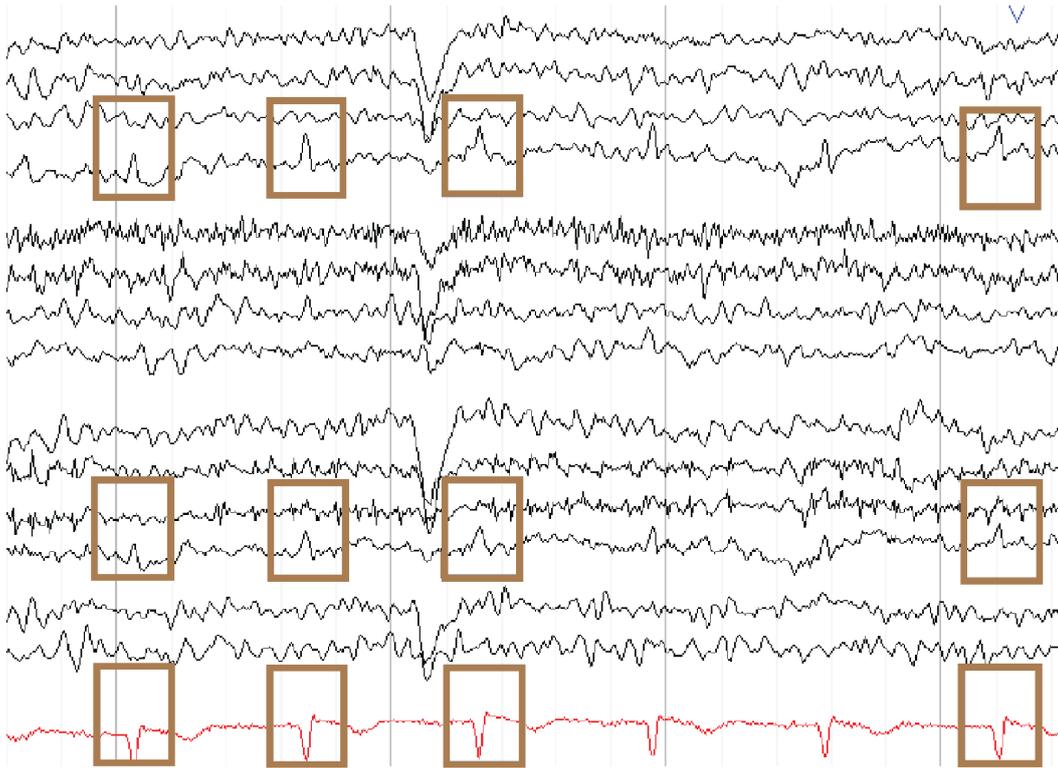


FIGURE 4.2: Examples of cardiac artifacts which are marked by the brown frames. Red plot is a reference ECG recording, the QRS complexes which have the greatest impact on the EEG traces are marked also by the brown frames.

abrupt voltage deflection but can manifest also in a rhythmical activity of a various frequencies [55]. In turn, low frequency artifacts can arise due to a poor contact of an electrode with the skin. Such occurrences can be detected by monitoring the impedance of electrodes in the measuring system, increased impedance suggests the contact with skin can be improper.

In this thesis, as mentioned earlier in Section 1.2, the emphasis is put on removal of the super-Gaussian artifacts characterized by a very high amplitude. In both databases used to verify the methods (see Chapter 2) these artifacts were predominant.

4.2 Application of DCTBS to noise removal

DCT based interpolation is the iterative procedure developed in [30] and facilitated as Matlab function *inpaintn* by R. Garcia. The method requires prior detection of the outliers and the samples belonging to the artifacts. Realization of this step is described in the subsection below. The further step of this method involves artifacts blanking, and finally the ensuing missing values interpolation.

4.2.1 Detection of outliers

Detection of the artifacts, performed individually in every channel, consists of two steps: search for the optimal thresholds separating the peaks (positive and negative) either to the accepted or the noisy ones, and determination of the artifacts' borders (onset and endset). The first step involves calculation of the 5th and 95th percentiles

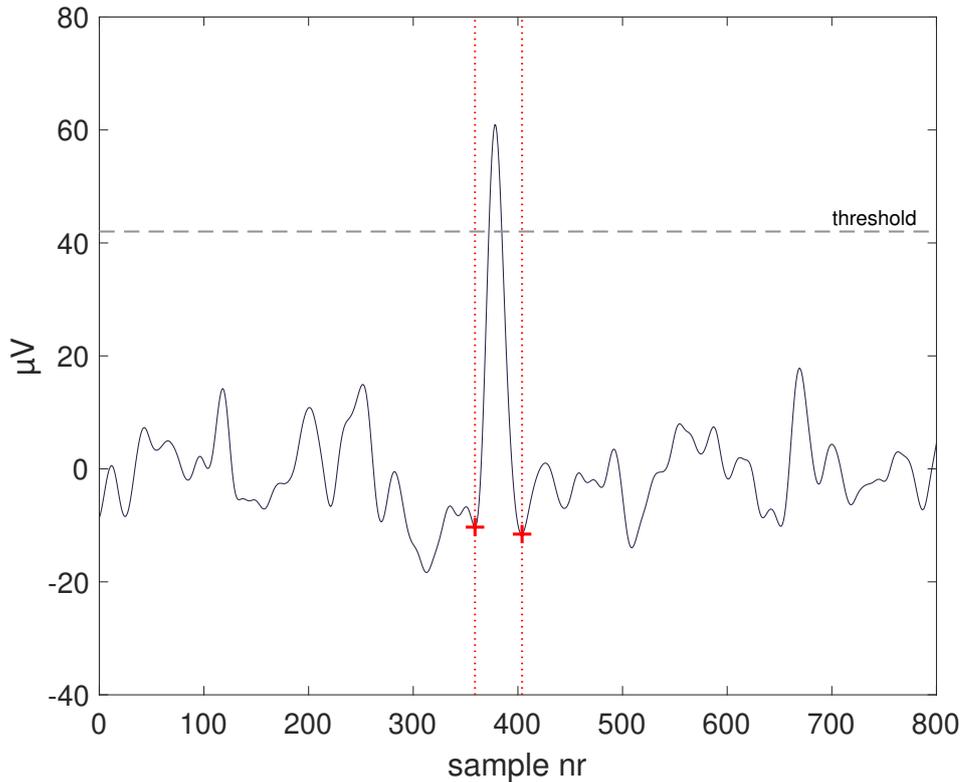


FIGURE 4.3: Exemplary artifact detected. Black dashed line indicates the threshold, boundaries of the artifact are marked with red + signs and the red dotted lines.

of all values in the channel. These percentiles, however, are not used directly as thresholds, otherwise some false outliers would be found even in the artifact free signals. In order to avoid that, these percentiles are multiplied by constant d and the resulting values constitute thresholds which allow to detect the outlying peaks. The initial experiments have shown that for $d \in [2, 5]$ performance of the EP classification was not changed. Therefore, $d = 3$ was applied for this algorithm. During the second step, each peak classified as an outlier (beyond the threshold) is a starting point from which the artifact onset and endset are searched. These border points are determined with the iterative procedure based on the signal derivative. From the starting point given in the peak extremum, the algorithm searches for the positions (to the left and to the right) where the derivative polarity changes. In the aftermath, all samples belonging to the found artifact can be marked as outlying. The exemplary outcome of this two-step operation is presented in Fig. 4.3. For a given signal

$\mathbf{x}(k)$ containing the samples of a single block (see Section 2.1 for an explanation), the matrix \mathbf{Z} is formed. It classifies the samples of $\mathbf{x}(k)$ as the outlying (0) or proper (1) ones according to

$$z_{i,j} = \begin{cases} 0, & \text{if } x_j(i) \text{ is outlying} \\ 1, & \text{if } x_j(i) \text{ is proper} \end{cases}. \quad (4.1)$$

4.2.2 Artifacts blanking and interpolation

As the DCT based interpolation procedure operates on the multichannel signal $\mathbf{x}(k)$, the values of signal $\mathbf{x}(k)$ are stored in matrix \mathbf{X} (subsequent channels in subsequent columns of \mathbf{X}). In turn, the matrix \mathbf{Z} defined by (4.1) is used to indicate the sam-

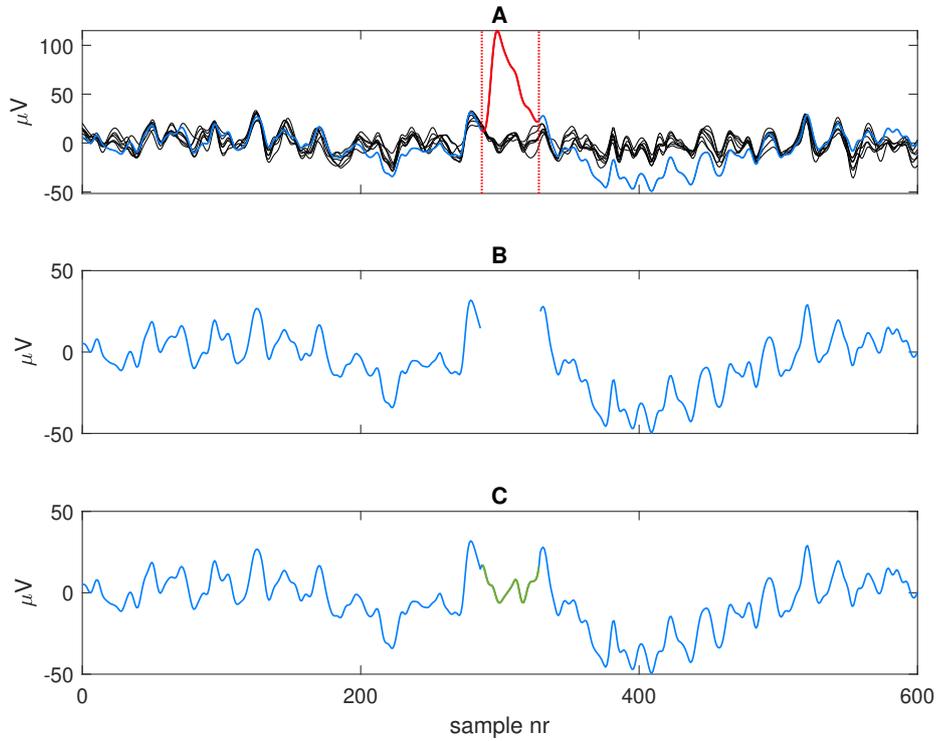


FIGURE 4.4: Plot A: segment of the 8-channel signal where the artifact has been detected (the samples belonging to the artifact, indicated by matrix \mathbf{Z} , are plotted in red); dotted, red lines mark the artifact's borders, the channel containing the artifact is plotted in blue beyond the artifact borders; plot B: The noisy (blue) channel with the artifact's samples blanked; plot C: the outcome of DCTBS interpolation of the blanked samples, the reconstructed part is plotted in green.

ples detected as noisy. Only these samples undergo modification in the process. The matrices \mathbf{X} and \mathbf{Z} are submitted to the *inpaintn* procedure. Initially, the noisy samples indicated by \mathbf{Z} are blanked (their values are changed to "not a number"(NaN)). On the basis of the output matrix $\hat{\mathbf{X}}$ ($\hat{y}_{\{k+1\}}$ defined by (20) in [30] where $\{k+1\}$ denotes the number of iterations), obtained is the signal with outlying artifacts suppressed $\hat{\mathbf{x}}(k)$. Consecutive steps of DCTBS: artifacts detection, blanking of the noisy

samples, and interpolation of the new samples are presented in Fig. 4.4 in plots A, B, and C, respectively.

4.3 Application of ICA and EMD to noise removal

ICA allows to identify the source signals which contribute to a presence of the artifacts in the EEG traces (see Appendix B). However, the initial experiments have shown that the computational time of ICA (*jader.m*) increases significantly when large number of channels is included. One of tested approaches to circumvent very high computational times of the *jader.m* algorithm, involved a use of fastICA algorithm (*fastICA.m*). However, it could not converge and properly estimate the source signals for all parts of 64-channel signal. Due to this inconveniences a direct application of ICA to all channels available, followed by a removal of the noisy source signals or their parts, can be considered only for the databases with rather small number of channels (e.g. the "Item selection" database which contains 8 channels). This assumption has been made with regard to the possible application of the presented algorithms in real BCI systems. Application of ICA to the databases with high number of channels (typically greater than 24, depending on the available computational resources, e.g. the "speller" database) requires a further effort.

Important limitation of ICA concerns possible distortions of the brain activity related power spectrum [103], in the aftermath of an exclusion of the noisy source signals which are the mixtures of noise and the desired, neural activity. Therefore an attempt to separate these elements (noise and neural activity) further seems to be beneficial. To this aim, the EMD (Empirical Mode Decomposition) method can be added as a refinement of ICA. EMD allows to decompose the signal into intrinsic mode functions (IMF) based on the local properties of the signal [45], [82]. By applying EMD to the noisy source signal and summing up only the noisy IMFs one can obtain the modified noisy source signal which more likely contains only the components contributing to the artifact, all the rest are treated as useful information whose deletion could have an adverse impact on the further classification.

This section is split on three subsections:

- first presents the technique of the artifacts detection which is used for the final versions of algorithms for both low and high number of channels in the dataset,
- second introduces 2 algorithms applicable only for the databases with limited number of channels,
- third, containing a description of 3 alternative algorithms which allow to operate on high number of available channels (e.g. 64 channels). All described algorithms process every block of flashes separately, their detailed steps are given below.

4.3.1 Artifacts detection

The procedure of artifacts detection begins with a search of the local ekstrema (minima and maxima) present in the analysed source signal. Then, the 5th and 95th percentiles of the component are calculated. Followingly, these percentiles are mul-

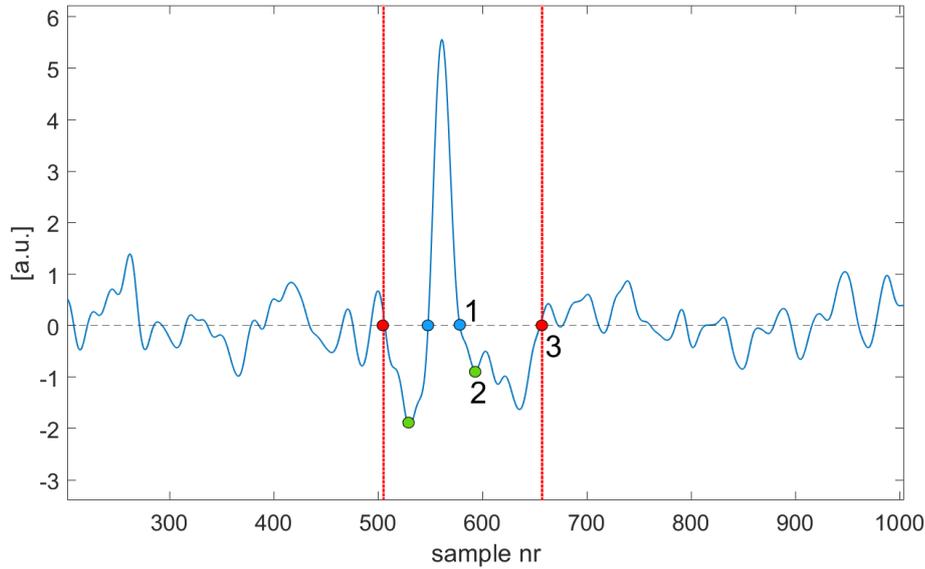


FIGURE 4.5: Exemplary artifact borders estimation. Blue dots represent the positions where 1st condition is met (first cross of the zero value), green dots: 2nd condition (change of the derivative polarity), and red dots: 3rd condition (second cross of the zero value). Red dots along with the dotted red vertical lines mark also the borders of the artifact.

tiplied by constant d resulting in the threshold which enables a distinction of the noisy extrema. Multiplication by d is introduced to avoid any modification of the artifact-free signals. Initial experiments proved that the values of d in range $[2, 5]$ do not have an impact on the performance of EP classification. Having found all local extrema classified as noisy, the borders of these artifacts are searched starting at the extrema positions (each extremum is processed separately). The preceding as well as succeeding signal parts are analysed to meet 3 conditions which need to occur consecutively one after another in the exactly stated order: first cross of the zero value, change of the derivative polarity, second cross of the zero value. All these conditions met are illustrated in Fig. 4.5 with the borders of the exemplary artifact.

4.3.2 Algorithms for low number of channels

Algorithm L-ICA1

Required: measured signals to be filtered $\mathbf{x}(k) = [x_m(k)]_{m=1}^M$.

1. Apply the JADE algorithm to the measured signals $\mathbf{x}(k)$ to estimate the mixing $\hat{\mathbf{A}}$ and separating $\hat{\mathbf{A}}^{-1}$ matrices (see Appendix B for details) and the ensuing M independent source signals

$$\mathbf{y}(k) = \hat{\mathbf{A}}^{-1}\mathbf{x}(k), \quad (4.2)$$

Example of the noisy signal and the outcome of ICA applied to it is presented in Fig. 4.6 and Fig. 4.7, respectively.

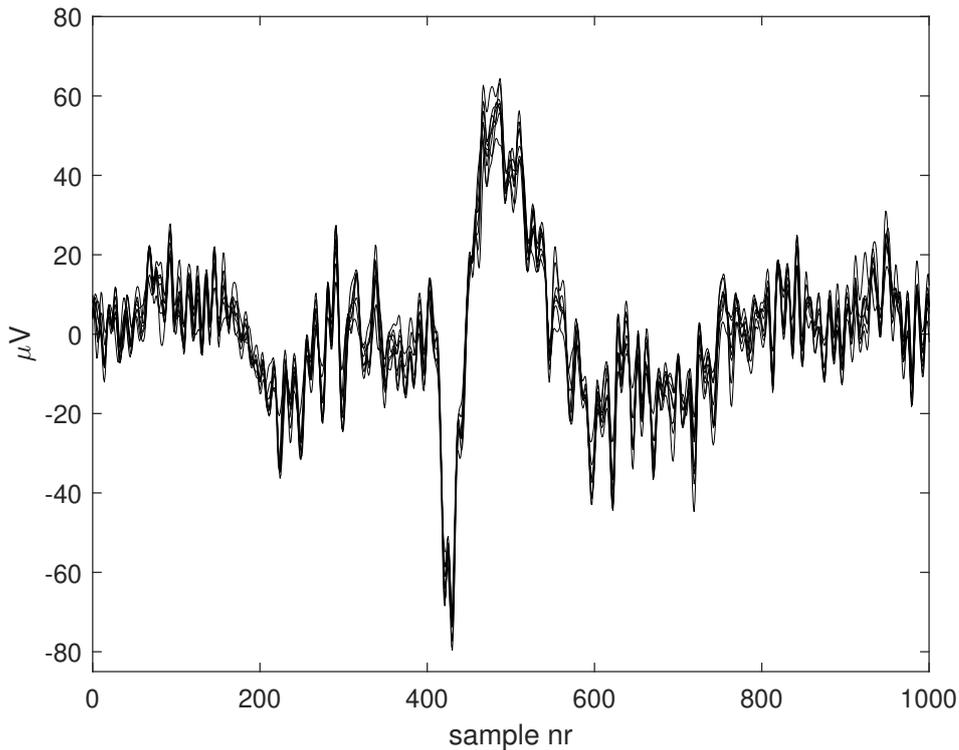


FIGURE 4.6: Part of the measured 8-channel signal containing the artifact. The signal is scaled in time to better illustrate details of the artifact

2. Search for the source signals which contribute to the presence of artifacts in the multichannel signal. Source signals discrimination as noisy or proper ones is based on the kurtosis measure. The ones whose kurtosis is above the predetermined threshold thr are regarded as noisy source signals according to (4.3). The independent source signal classified as noisy was plotted in red in Fig. 4.7.

$$y_i(k) \text{ is } \begin{cases} \text{noisy,} & \text{if } Kurt(y_i(k)) > thr \\ \text{proper,} & \text{if } Kurt(y_i(k)) < thr \end{cases}. \quad (4.3)$$

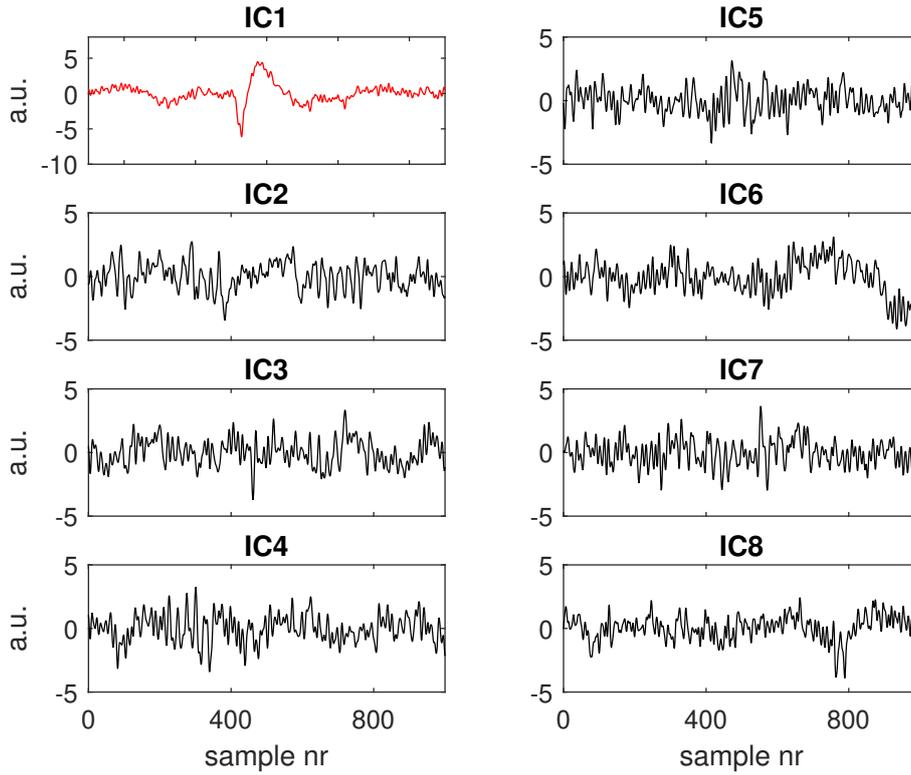


FIGURE 4.7: Outcome of ICA performed on the signal visible in Fig. 4.6. 8 subplots present 8 independent source signals, the one plotted in red was classified as noisy in procedure described in step 2.

3. Search for the time samples belonging to the artifacts, these samples indices will be added to the initially empty set Ω .

For every i^{th} noisy source signal in $\mathbf{y}(k)$ do

- (a) Find the high amplitude artifacts and the time segments they occupy using the procedure described in Section 4.3.1. Store the indices of these noisy time samples in the temporary set Ω_t . The outcome of artifact detection performed on the noisy source signal IC1 (plotted in red in Fig. 4.7) is illustrated in Fig. 4.8.
- (b) Zero the artifact whose indices are given by set Ω_t in $\mathbf{y}(k)$ (see Fig. 4.9)

$$y_i(k) = 0, k \in \Omega_t$$

- (c) Calculate the union of the sets Ω and Ω_t and store the result in Ω

$$\Omega = \Omega \cup \Omega_t$$

4. Reconstruct the original measured signal $\mathbf{x}'(k)$ using the estimated mixing matrix $\hat{\mathbf{A}}$ and the modified source signal vector $\mathbf{y}(k)$ only for the indices stored in

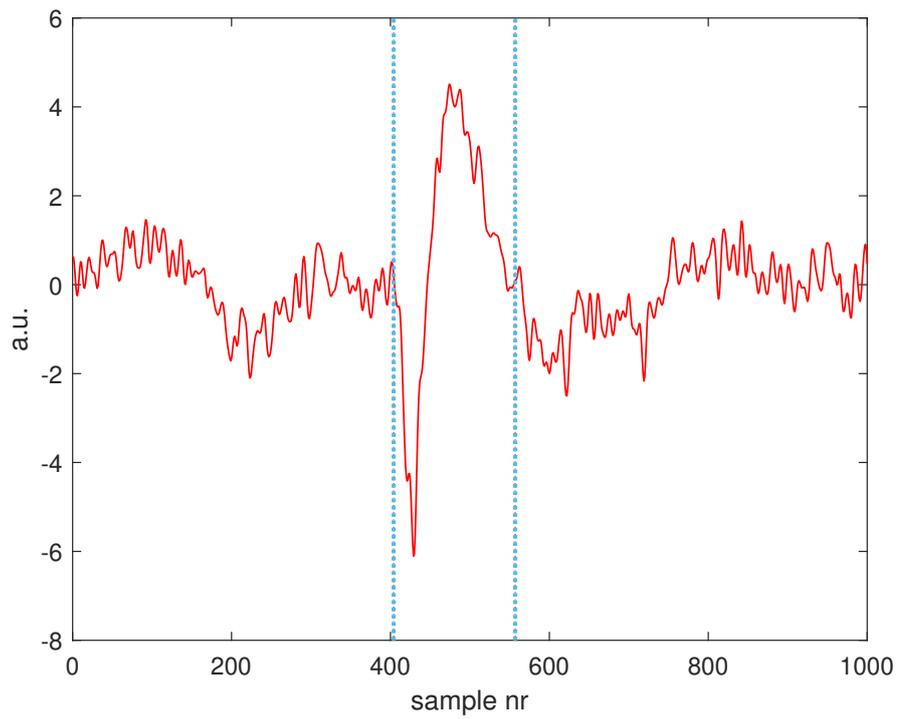


FIGURE 4.8: Illustration of the artifact borders (plotted by the vertical, blue, dotted lines) detected on the noisy IC (IC1).

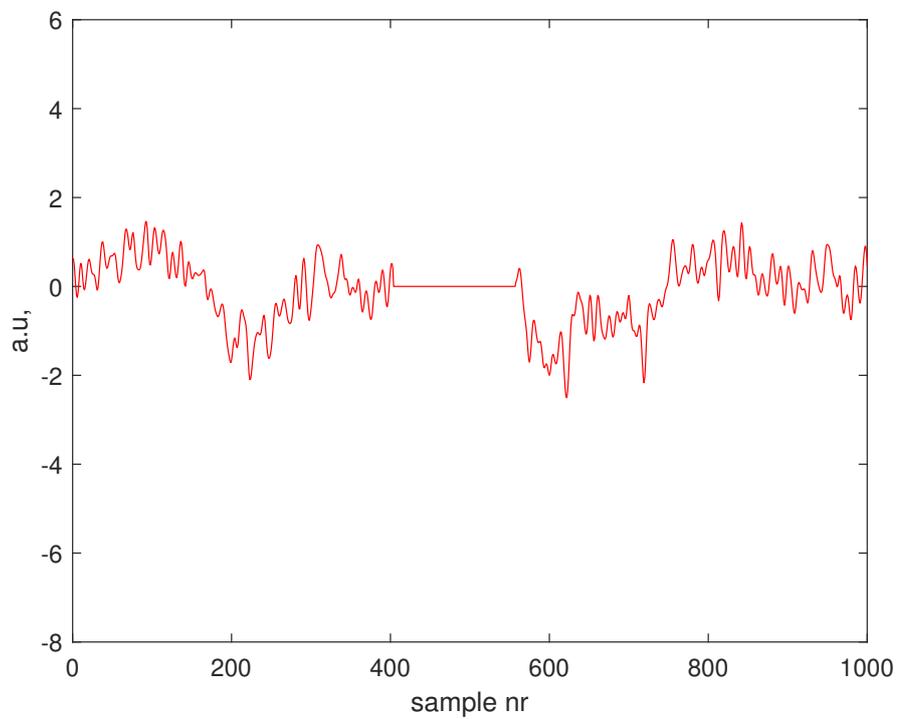


FIGURE 4.9: Noisy source signal with the artifact zeroed

set Ω (see plot A in Fig. 4.11)

$$\mathbf{x}'(k) = \hat{\mathbf{A}}\mathbf{y}(k), k \in \Omega \quad (4.4)$$

Algorithm L-ICA2

This version of algorithm is an extension of L-ICA1 where artifacts zeroing is replaced by their suppression using Empirical Mode Decomposition (EMD). To this aim the EMD2 procedure (see Appendix C for its detailed description) was applied to every noisy source signal determined in step 2 of L-ICA1:

$$\mathbf{y}'(k) = \text{EMD2}(\mathbf{y}(k))$$

During EMD2 procedure, the highly kurtotic IMFs for each noisy component are rejected.

Subsequently, the step 3 (b) is modified accordingly: instead of zeroing the samples (indicated by Ω_t) of the given source signal $y_i(k)$, they are replaced by the corresponding samples of $y'_i(k)$ (see Fig. 4.10). The rest of the algorithm remains unchanged.

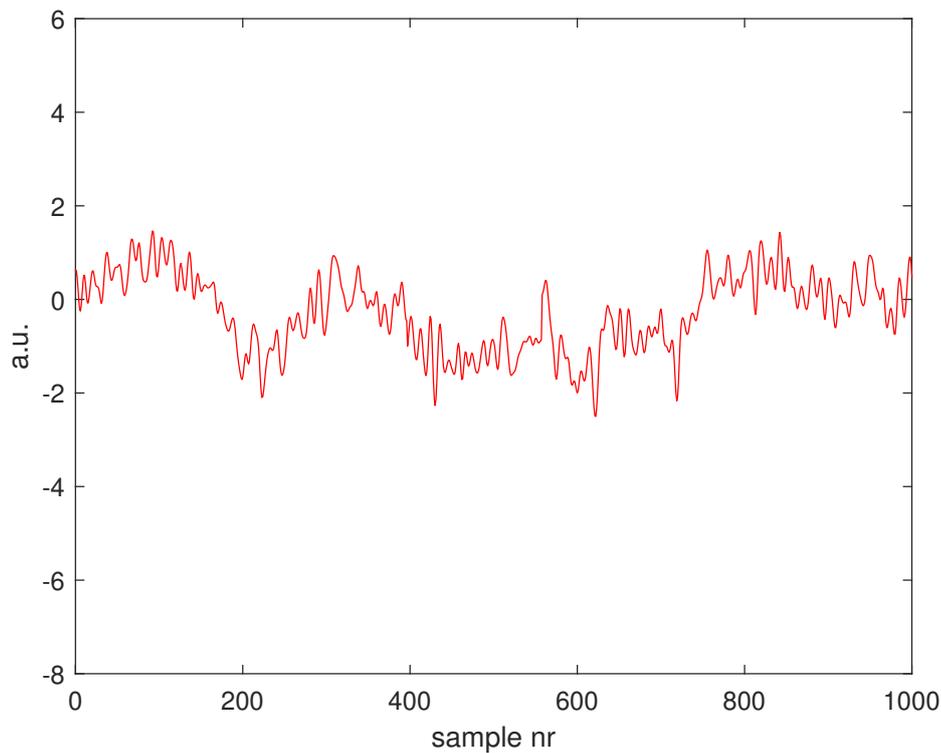


FIGURE 4.10: Noisy source signal with the artifact suppressed using EMD2 (which rejects the highly kurtotic IMFs).

The denoised signals obtained by applying either L-ICA1 or L-ICA2 are compared in Fig. 4.11, plots A and B, respectively.

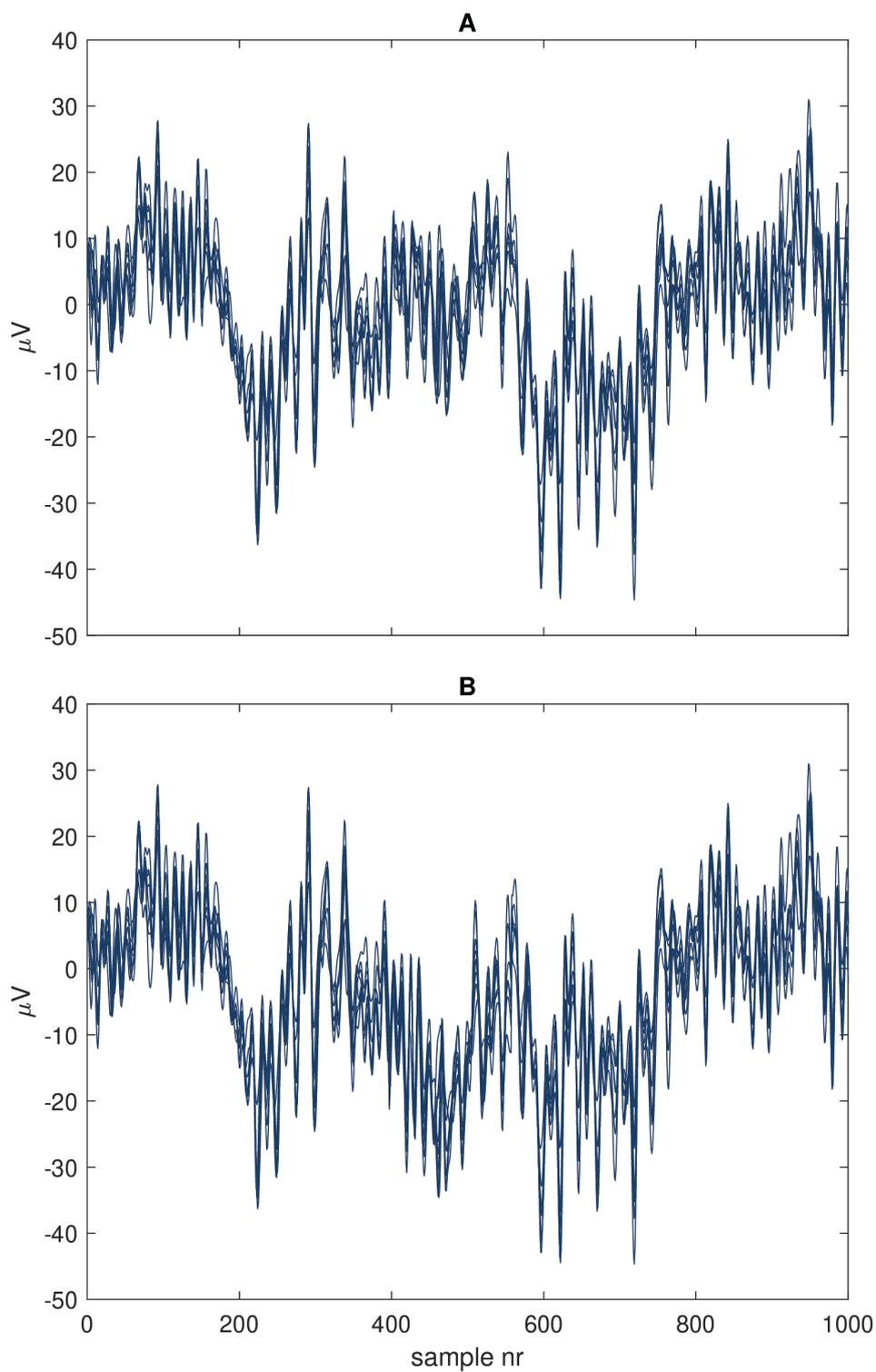


FIGURE 4.11: The effect of the original measured signal (presented in Fig. 4.6) denoising. Plot A: The signal reconstructed using L-ICA1 (ICA), plot B: The signal reconstructed by L-ICA2 (ICA+EMD2).

4.3.3 Algorithms for high number of channels

The problem of high dimensionality of EEG data, mentioned in the beginning of this section, was investigated in [42] where multichannel data were split on lower, randomly selected groups of channels and ICA was calculated for these limited groups. The approach proposed in this thesis, however, assumes that during the experiments only selected \hat{M} channels of the highest kurtosis are used to estimate the mixing $\hat{\mathbf{A}}$ and separating $\hat{\mathbf{A}}^{-1}$ matrices and the ensuing independent source signals (using the `jader.m` algorithm). Next, the extracted estimates of source signals are again ordered according to their kurtosis, and the selected highly kurtotic most spiky among them are mixed to approximate the artifacts occurring in the original measured signals. Since the estimated mixing matrix $\hat{\mathbf{A}}$ is associated with the selected highly kurtotic part of the channels, the new one is formed. The weights of the source signals combination are calculated using the mean square approach, i.e. by minimization of the following objective function

$$Q(\mathbf{v}) = \sum_{k \in \Gamma} \left(x(k) - \mathbf{v}^T \mathbf{y}(k) \right)^2 \quad (4.5)$$

where $x(k)$ is a signal approximated, $\mathbf{y}(k)$, a vector of the basis functions (signals used for approximation), and Γ is a set containing time indices of the entire signal. In the algorithms descriptions below as a default $k \in \Gamma$, if not specified otherwise.

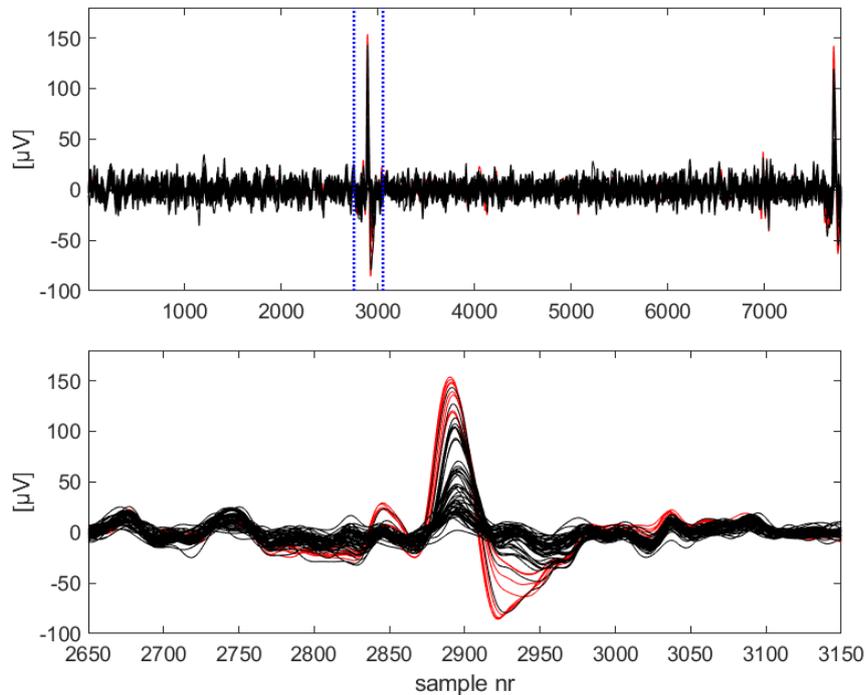


FIGURE 4.12: Upper plot: Measured signals containing the artifacts; In red: \hat{M} signals of the highest kurtosis determined in step 3, in black: remaining signals. Blue, dotted lines represent borders of the lower plot which magnifies view on one of the largest artifacts.

Algorithm H-ICA1

Required: measured signals to be filtered $\mathbf{x}(k) = [x_m(k)]_{m=1}^M$.

1. Calculate the kurtosis of all measured signals

$$q_m = \text{kurtosis}\{x_m(k)\}, m = 1, \dots, M$$

2. Find the permutation function $\psi : \{1, 2, \dots, M\} \rightarrow \{1, 2, \dots, M\}$ sorting the calculated q_m values in descending order $q_{\psi(1)} \geq q_{\psi(2)} \geq \dots \geq q_{\psi(M)}$.
3. Select \hat{M} signals of the highest kurtosis and remember them in vector $\hat{\mathbf{x}}(k)$:

$$\hat{x}_m(k) = x_{\psi(m)}(k), m = 1, 2, \dots, \hat{M}$$

The measured signals with \hat{M} signals of the highest kurtosis, plotted in red, are presented in Fig. 4.12

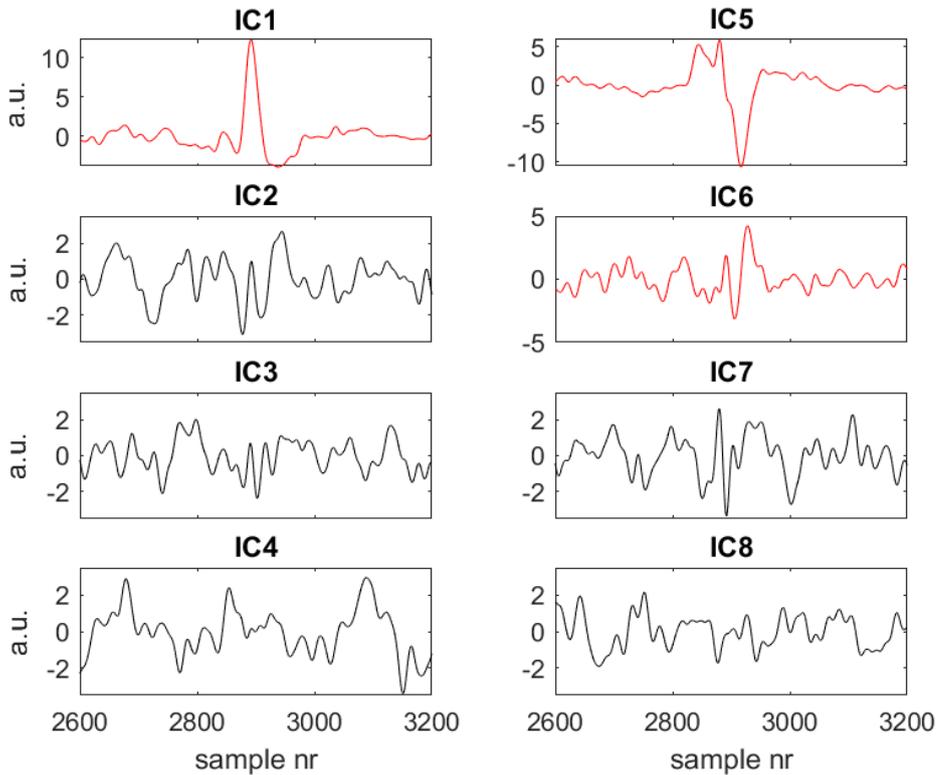


FIGURE 4.13: Source signals (independent components) estimated in step 4, in red: source signals detected as noisy (above the threshold thr) according to step 7. In black: remaining, unnoisy source signals.

4. Apply the JADE algorithm to estimate the mixing $\hat{\mathbf{A}}$ and separating $\hat{\mathbf{A}}^{-1}$ matrices and the ensuing independent source signals

$$\hat{\mathbf{y}}(k) = \hat{\mathbf{A}}^{-1} \hat{\mathbf{x}}(k)$$

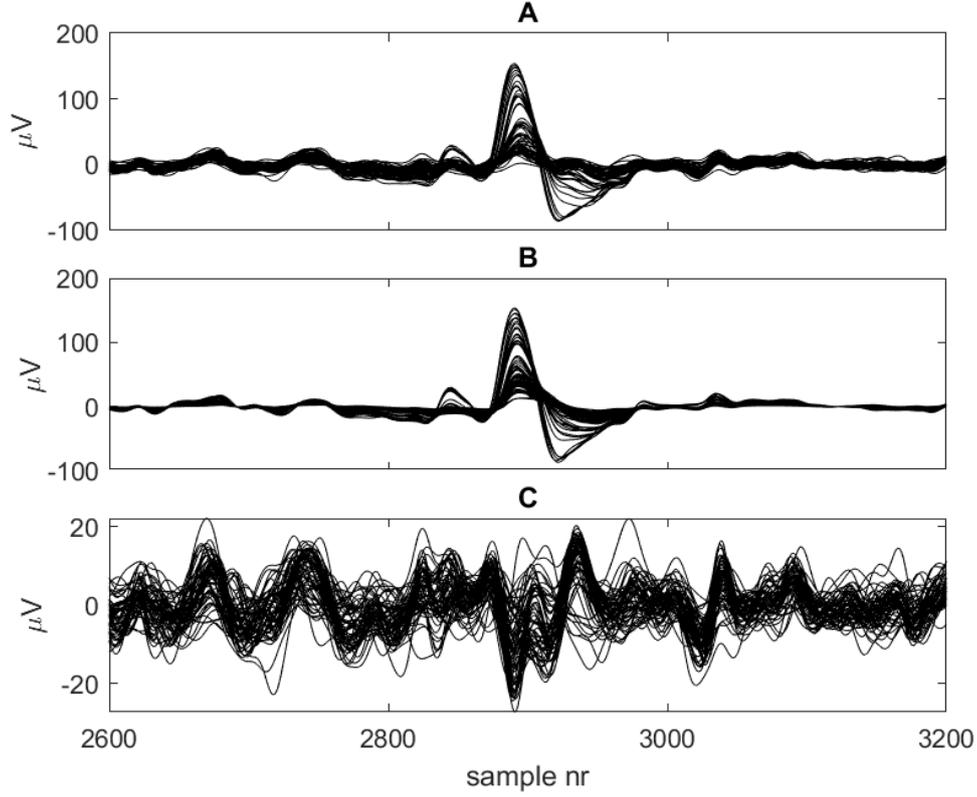


FIGURE 4.14: Plot A: Part of the measured signals containing the artifacts, plot B: artifacts approximated using H-ICA1 algorithm, plot C: the result of subtraction of the approximated artifacts from the corresponding measured signals.

5. Calculate the kurtosis of all source signals separated

$$\hat{q}_m = \text{kurtosis}\{\hat{y}_m(k)\}, m = 1, \dots, \hat{M}$$

6. Find the permutation function $\phi : \{1, 2, \dots, \hat{M}\} \rightarrow \{1, 2, \dots, \hat{M}\}$ sorting the calculated \hat{q}_m values in descending order $\hat{q}_{\phi(1)} \geq \hat{q}_{\phi(2)} \geq \dots \geq \hat{q}_{\phi(\hat{M})}$.
7. Find the number \tilde{M} of the source signals whose kurtosis is greater than the assumed threshold thr and remember \tilde{M} source signals of the highest kurtosis in vector $\tilde{\mathbf{y}}(k)$:

$$\tilde{y}_m(k) = \hat{y}_{\phi(m)}(k), m = 1, 2, \dots, \tilde{M}$$

The source signals are presented in Fig. 4.13, the ones whose kurtosis is greater than thr are plotted in red.

8. Calculate matrix $\tilde{\mathbf{A}}$ to mix the selected highly kurtotic source signals $\tilde{\mathbf{y}}$ for approximation of the measured channels artifacts

$$\tilde{\mathbf{x}} = \tilde{\mathbf{A}}\tilde{\mathbf{y}}$$

Each row of $\tilde{\mathbf{A}}$ is equal to vector \mathbf{v} calculated by minimizing (4.5) on the basis of the particular channel of the measured signal $x_m(k)$ and the source signals vector $\tilde{\mathbf{y}}$.

9. Suppress the artifacts by subtraction of the approximating them signals

$$\mathbf{x}'(k) = \mathbf{x}(k) - \tilde{\mathbf{x}}(k)$$

The approximated artifacts $\tilde{\mathbf{x}}(k)$, as well as the outcome of the artifacts suppression are illustrated in Fig. 4.14

Algorithm H-ICA2

This version of algorithm is an extension of H-ICA1. The changes in comparison to H-ICA1 apply to the step 7. Here, performed is an additional refinement of each noisy source signal, by means of the EMD1 procedure (see Appendix C for its detailed description):

$$\tilde{\tilde{\mathbf{y}}}(k) = EMD1(\tilde{\mathbf{y}}(k))$$

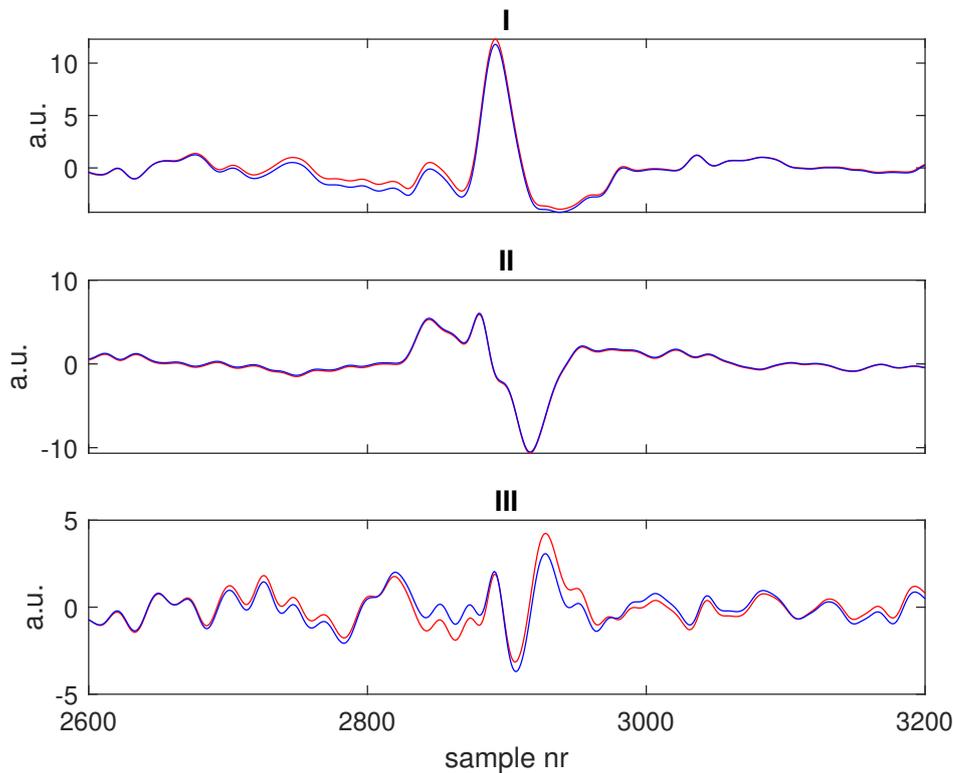


FIGURE 4.15: Plots I, II, and III present the comparison between the noisy source signals shown in Fig. 4.13 (red plots) with the modified source signals (blue plots) in the algorithm H-ICA2 (which adds the EMD1 procedure).

In the aftermath, every noisy source signal is a sum of the noisy intrinsic mode functions only. The proper IMFs, which should relate to the desired signal components rather than to the artifacts, are rejected. The changes of the noisy source signals that arise by application of EMD1 are presented in Fig. 4.15.

The effect of artifacts suppression is presented in Fig. 4.16.

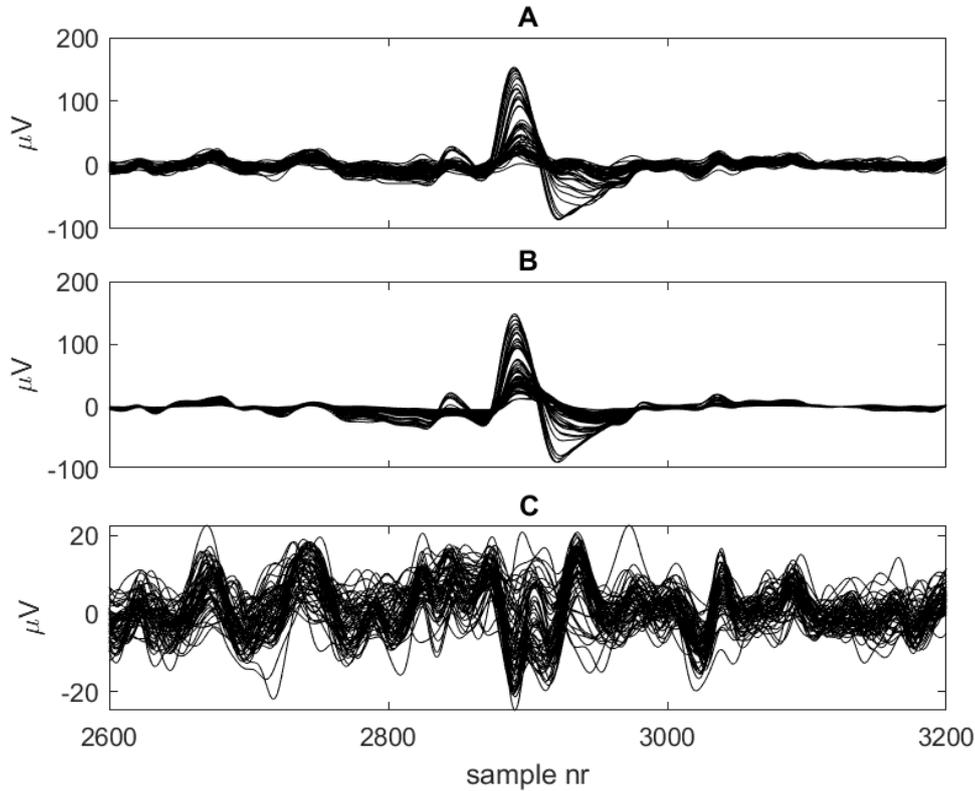


FIGURE 4.16: Plot A: Part of the measured signals containing the artifacts, plot B: artifacts approximated using H-ICA2 algorithm, plot C: the result of subtraction of the approximated artifacts from the corresponding measured signals.

Algorithm H-ICA3

The algorithm H-ICA3 extends further the capabilities of the previously described H-ICA1 and H-ICA2. The major change relates to the ranges in which the original measured signals are modified. On the contrary to H-ICA1 and H-ICA2, only the time samples belonging to the detected artifacts are modified, here. The rest of the measured signals remains unchanged. The steps 1 – 7 of H-ICA3 are exactly the same as for H-ICA2 (including the EMD1 procedure used to refine the noisy source signals). The remaining steps are as follows:

8. Search for the time samples belonging to the artifacts, the indices of these samples will be stored in the initially empty set Ω
For every m^{th} noisy source signal stored in $\tilde{\mathbf{y}}(k)$ **do**

- (a) Determine the artifacts using the procedure given in Section 4.3.1 and store the indices of the samples belonging to these artifacts in the temporary set Ω_t
- (b) Calculate the union of the sets Ω and Ω_t and store the result in Ω

$$\Omega = \Omega \cup \Omega_t$$

9. Determine the number F of artifacts in Ω which are separate in a time scale and store each artifact's indices in a different set: Indices of the f^{th} artifact are stored in the set Ω_f
10. **For every f^{th} artifact do**
 - (a) Calculate matrix $\tilde{\mathbf{A}}_f$ to mix the selected highly kurtotic source signals, limited in length to the length of the f^{th} artifact, ($k \in \Omega_f$) for approximation of the measured channels artifacts

$$\tilde{\mathbf{x}}'(k) = \tilde{\mathbf{A}}_f \tilde{\mathbf{y}}(k), k \in \Omega_f$$

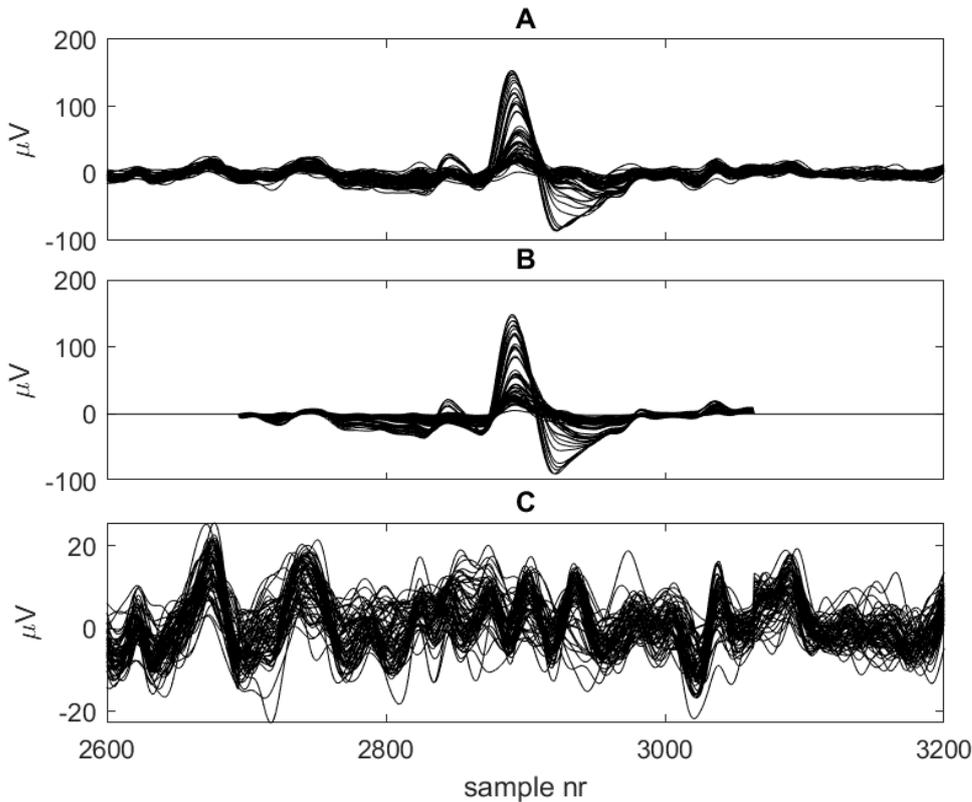


FIGURE 4.17: Plot A: Part of the 64-channel signal, shown in Fig. 4.12, containing the artifact, plot B: modeled artifact elements using H-ICA3 algorithm, plot C: the result of subtraction the artifact elements from the noisy signal channels.

Each row of $\tilde{\mathbf{A}}_f$ is equal to vector \mathbf{v} calculated by minimizing (4.5) on the basis of the particular channel of the measured signal $x_m(k)$ and the source signals vector $\tilde{\mathbf{y}}(k)$ for $k \in \Omega_f$.

- (b) Suppress the f^{th} artifact by subtraction of the approximating it signals

$$\mathbf{x}'(k) = \mathbf{x}(k) - \tilde{\mathbf{x}}'(k), \quad k \in \Omega_f$$

An exemplary effect of such a subtraction is presented in Fig. 4.17.

Visual comparison of the results achievable using all 3 algorithms (H-ICA1, H-ICA2, and H-ICA3) is presented in Fig. 4.18. It seems evident that H-ICA3 algorithm allows for visually the best results.

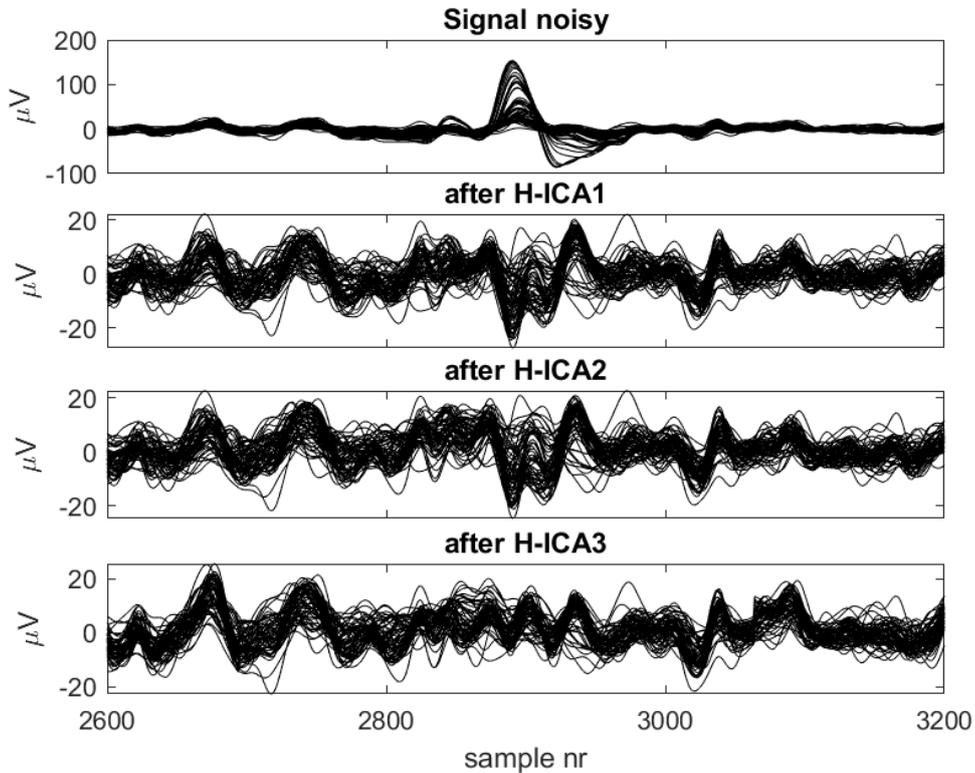


FIGURE 4.18: Visual comparison of the algorithms dedicated for databases with a high number of channels. Plot A: Noisy 64-channel signal, Plots B, C, and D: Signals denoised using H-ICA1, H-ICA2, and H-ICA3, respectively.

Chapter 5

Numerical experiments

5.1 Experiments with the "Item selection" database

5.1.1 Signal preprocessing and artifacts removal

"Item selection" dataset has been originally band-pass filtered with cut-off frequencies of 2Hz and 30Hz and notch-filtered at 50Hz. Sampling frequency equals 250Hz and no further downsampling was applied during the experiments. Because the

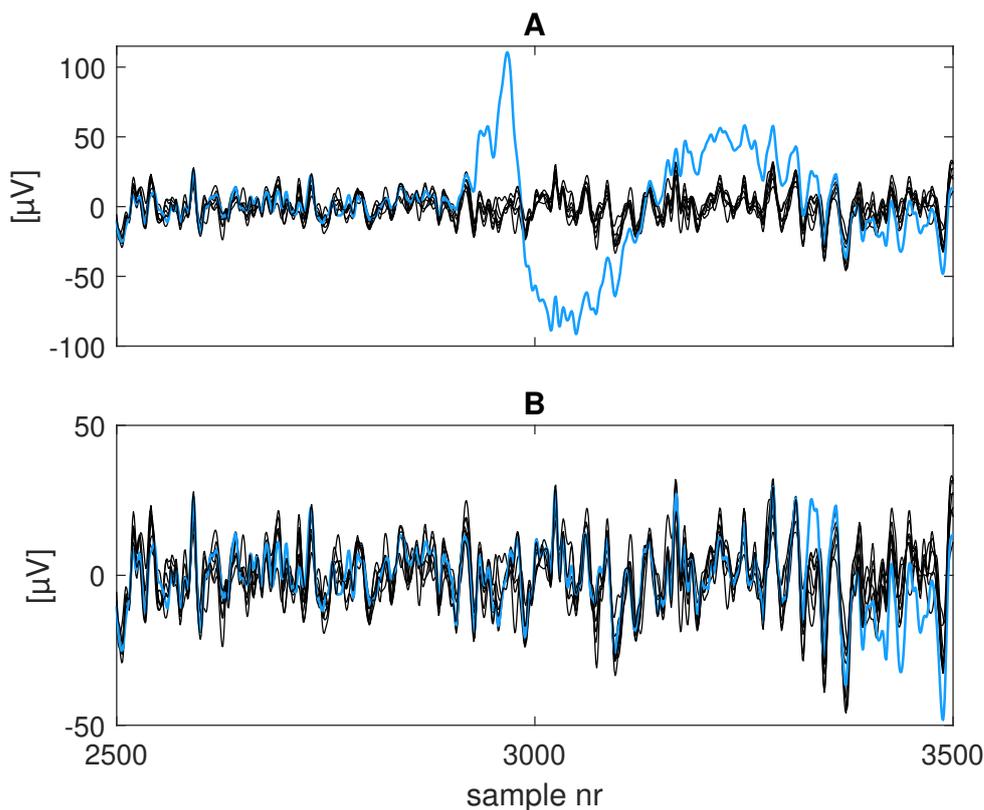


FIGURE 5.1: Exemplary performance of DCT based denoising applied to "Item selection" database. Plot A: the signal containing a single channel artifact (plotted in blue), plot B: the signal with artifacts rejected using DCT (the modified channel is plotted in blue).

dataset is split on the short segments associated with one flash, each lasting 1.4s, the length of such a single segment is insufficient to properly detect the artifacts and to find their borders. Therefore, prior to applying a denoising procedure these shorts

parts of the signals should be merged. However, because the adjacent 1.4s segments contain partly the same information (which is a result of rather low inter-stimulus interval (200 ms), so that the responses to more than one stimulus are typically included in a single 1.4s segment), prior to application of the merging procedure these signal parts are converted into the continuous longer signal (without mentioned redundant information). Such a continuous signal which gathers responses to flashes within one block is used to detect the artifacts and remove them according to the denoising algorithms described in Sections 4.2 and 4.3 (DCTBS, L-ICA1 or L-IC2) implemented for low number of channels available in the dataset). Finally, after denoising the signals are reshaped back to the previous form. This section presents the visual examples of how the mentioned algorithms perform what can give the insight about type of the artifacts present in the dataset, and the quality of output denoised signals.

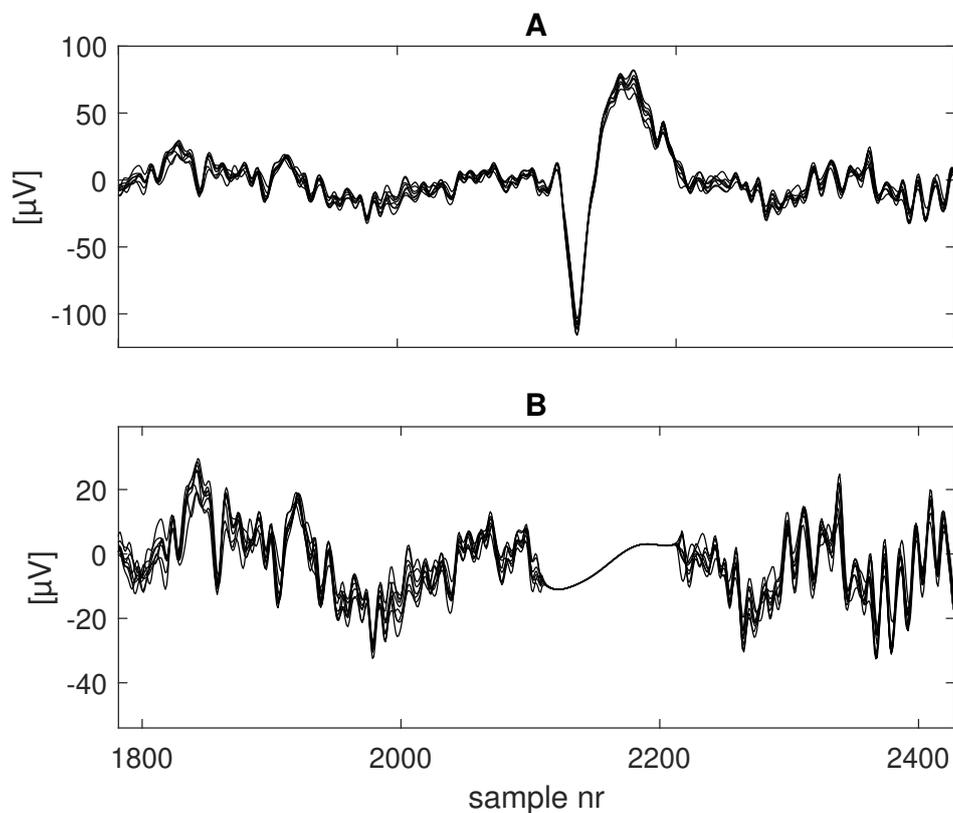


FIGURE 5.2: Exemplary performance of DCT based denoising applied to the "Item selection" database. Plot A: the signal containing a multichannel artifact, plot B: the signal with artifacts rejected using DCT based procedure.

Performance of DCT based artifacts rejection algorithm is presented in Figures 5.1 and 5.2. Whereas it managed to reject a single channel artifact (see Fig. 5.1) in an acceptable way, it was not that efficient when the artifact spreads to all available channels (see Fig. 5.2). In the latter case the denoised segments do not contain a high amplitude spike anymore, however, the fluctuations of a higher frequency, typical for EEG waves, are lost. Besides, interpolation of a noisy signal segment is almost

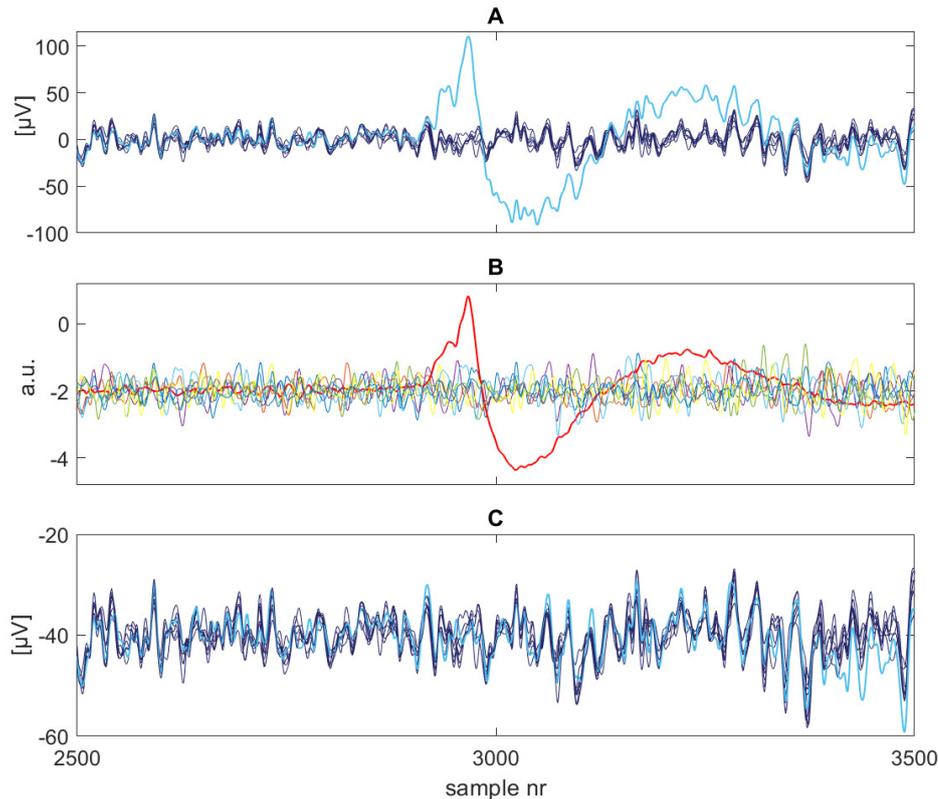


FIGURE 5.3: Exemplary performance of L-ICA based denoising applied to "Item selection" database. Plot A: the signal containing a single channel artifact (affected channel in light blue), plot B: estimated source signals (in red the noisy source signal), plot C: the signal with artifacts rejected using ICA; the modified channel is plotted in blue.

identical for all channels. On the other hand, when only one channel had to be interpolated the algorithm performed well what was caused by additional spatial information from the neighboring channels.

The same noisy parts as visible in Figures 5.1 and 5.2 are presented in Figures 5.3 and 5.4. Here, the outputs of L-ICA1 denoising algorithm are visualized. Watching Fig. 5.3 one can discern that the efficiency of noise reduction is very similar to that obtained by the algorithm founded on DCT. There are only minor differences in a shape of the reconstructed channel. Definitely greater difference can be observed comparing these two methods when applied to the multichannel artifact removal. Here, L-ICA1's advantage is apparent in plot C of Fig. 5.4; the output does not contain the artifact but also the EEG baseline seems to be unimpaired significantly. Such an example confirms L-ICA1's superiority to handle multichannel artifacts, particularly when all channels available in the signal are affected. Worthy of note is that the ICA based method managed to properly isolate the noisy source signal (plots B in Figures 5.3 and 5.4) in both scenarios (single channel and multichannel artifact) and reconstruct these signals efficiently excluding these noisy components. This implicates an expectation that L-ICA should be helpful also in denoising the EEG data

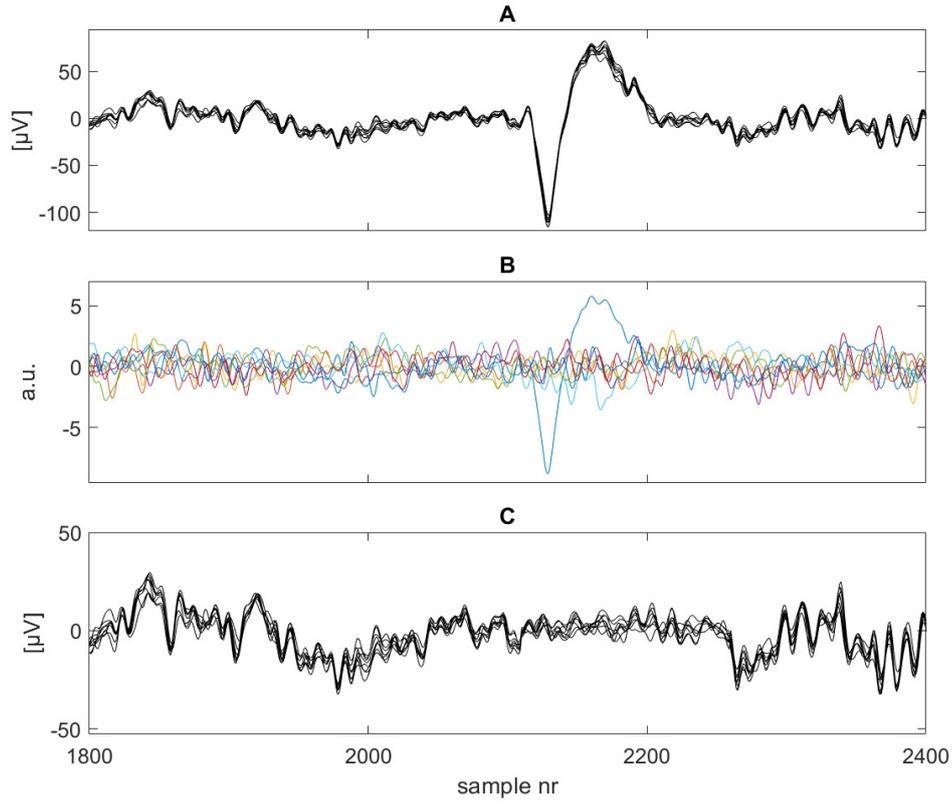


FIGURE 5.4: Exemplary performance of L-ICA based denoising applied to "Item selection" database. Plot A: the signal containing multichannel artifact, plot B: estimated source signals (in blue the noisy independent component (IC)), plot C: the signal with artifacts rejected using L-ICA.

when only one channel recording has been affected by e.g. electrode detachment or worsened contact with a skull.

5.1.2 Parameters selection and filters construction

With respect to the formulas defined in Section 3.3 the parameters of all STF based methods were set to values: $J = 21$ and $\tau = 6$. Primarily, such a selection, which defines the filter length ($2 \cdot J \cdot \tau / f_s$) stemmed from the presumed length of the analyzed signal segment (1s) that covers a typical region where the whole group of evoked potentials should occur. Furthermore, the value of parameter τ , as explained in Section 3.3, should not decrease the sampling frequency below 40Hz, therefore is calculated using formula given below (3.9) ($\tau = \lfloor \frac{f_s[\text{Hz}]}{40} \rfloor$). Its assumed value should have guaranteed rather fast computational speed while not deteriorating the classification accuracy. More detailed information showing the influence of J and τ values on the algorithm performance is given in Section 5.1.5. For each subject all learning data subsets were used during the filters construction. Samples containing the desired components to be enhanced were gathered in matrix **A**; the noise in turn in

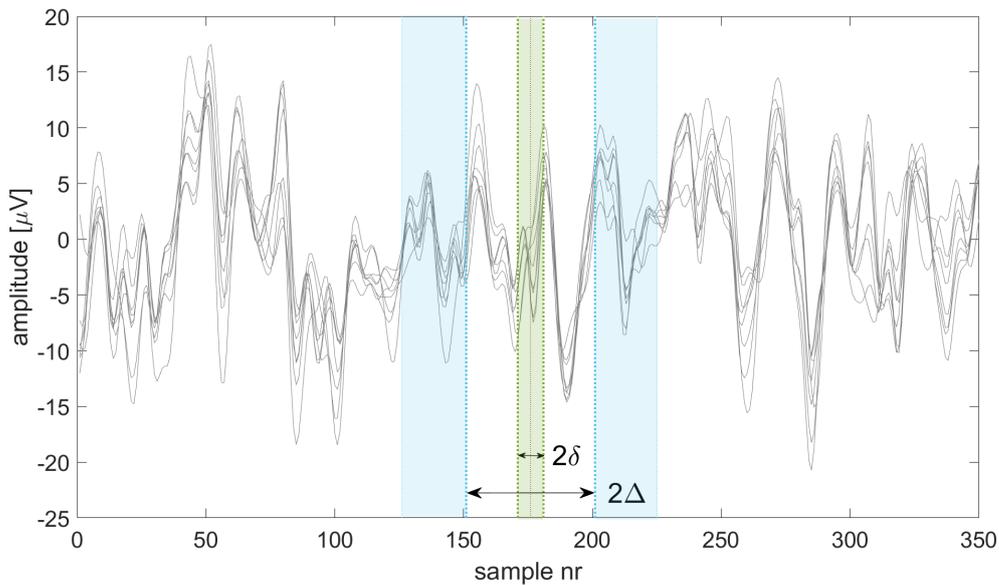


FIGURE 5.5: Explanation of parameters δ and Δ on the picture showing 1.4s segments from all 8 channels. The sets given in Fig. 3.3 were readjusted to the purpose of experiments on this database. 2δ , marked in green, defines the width of the window containing time indices of the set of vectors $(\mathbf{x}^{(k)}, k \in \Psi_t(\delta))$ to be enhanced. Blue area on both sides shows the indices of vectors $(\mathbf{x}^{(k)}, k \in \Psi_s(\Delta))$ to be suppressed (these areas were additionally reduced on both sides by $J\tau$ so that the spatio-temporal vectors stored in \mathbf{B} could be properly constructed).

matrix \mathbf{B} . The set Ψ_s in its form presented in Fig. 3.3 could be used only for continuous signal (recovered form of the signals was used for the denoising purposes only). For the purpose of processing this database, which consists of 1.4s segments: each related to a single response, the set had to be modified to take into account the finite length of the examined signal segment (each lasting 1.4s). The corrected indices which are shown in Fig. 5.5 in blue were reduced on both sides by $J\tau$ so that the spatio-temporal vectors could be properly constructed. The samples gathered in matrix \mathbf{B} , whose indices were given by such reduced Ψ_s , were used to compute the covariance matrix \mathbf{C}_b estimating the noise properties in the EEG signals. For the purpose of OSTF application, similarly \mathbf{C}_a was calculated using \mathbf{A} (see the description below (3.13)). For GSTMF, the target vector $\bar{\mathbf{a}}$ was computed by averaging the data stored in \mathbf{A} (see (3.16)). Finally, verification of MSTMF required a construction of vector $\bar{\mathbf{a}}'$, analogical to $\bar{\mathbf{a}}$ but related to nontarget responses. The visual example of the difference $(\bar{\mathbf{a}} - \bar{\mathbf{a}}')$ is presented in Fig. 5.6 in plots A (for $\delta > 0$) and in plot C (for $\delta = 0$). For reference, also the analogical construct is presented for LDA in plot E. Plot A is slightly smoother than plots C and E, what is the result of kind of low-pass filtering introduced when $\delta > 0$ (in this case more successive spatio-temporal vectors are gathered in matrix \mathbf{A} whose averaging results in less noisy patterns $\bar{\mathbf{a}}$ and $\bar{\mathbf{a}}'$).

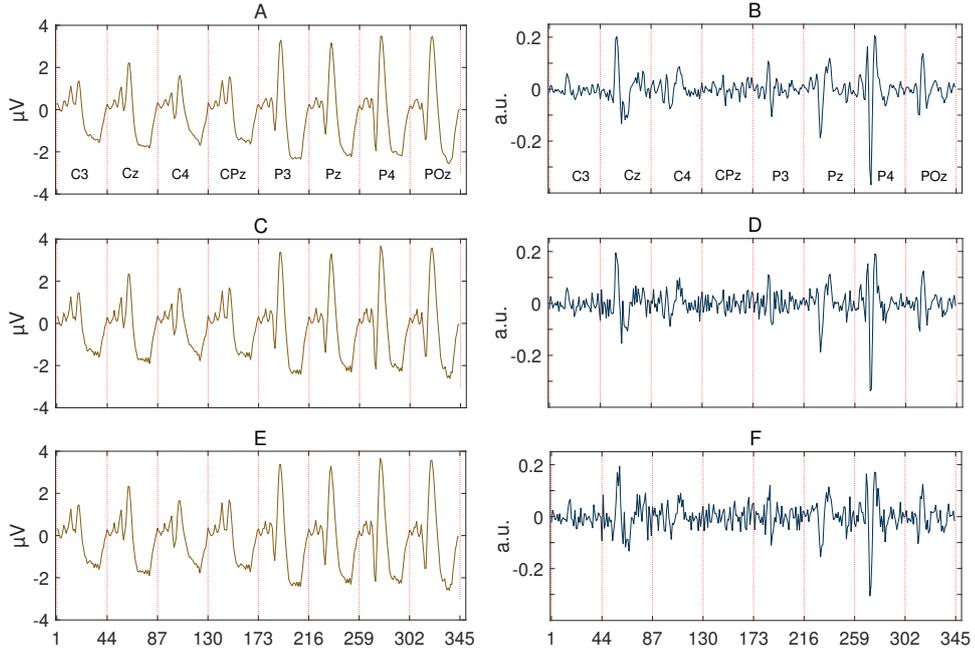


FIGURE 5.6: Visual comparison of the MSTMF and LDA performance. The plots on the left illustrate the differences $\bar{\mathbf{a}} - \bar{\mathbf{a}}'$ in MSTMF (plot A for $\delta > 0$ and plot C for $\delta = 0$). Plot E shows the analogical difference of classes means used in LDA. The plots in blue on the right present the filter templates of MSTMF with $\delta > 0$ (plot B), MSTMF with $\delta = 0$ (plot D), and LDA (plot F). These filter templates are calculated using formula (3.21) where the difference $\bar{\mathbf{a}} - \bar{\mathbf{a}}'$ is multiplied by the inverse of the noise covariance matrix: \mathbf{C}_b^{-1} . The vertical dotted lines in red separate the values belonging to the different channels (whose names are given in the upper plots); each channel contains $2J + 1 = 43$ values.

Inclusion of such numerous samples in matrix \mathbf{B} , given by Ψ_s , was particularly important to overcome a previously mentioned problem of singularity of the noise covariance matrix \mathbf{C}_b (equivalent to the LDA's within-class scatter matrix \mathbf{S}_w) which undergoes inversion (see Sections 1.2 and 1.3). In case the data has been facilitated in a form so that it contains only separate, cut segments with single responses for the events, like in the currently analysed database, still there was a way to take benefit of the short segments (before and after the target events or those related to nontarget events) and construct a filter without singularity limitation. A greater number of these samples implicated better information on the noise properties, thus the noise covariance matrix \mathbf{C}_b (defined below (3.13)) could be estimated more effectively. Effect of this is visible when the filter templates constructed for MSTMF in Fig. 5.6 (plot B) and for LDA (plot F) are compared. Whereas the shape of the templates is similar, the former is of a better quality. It can be explained by the fact that whereas LDA allows to create only one vector based on the single response, STF, due to its capability to extend such a vector to its spatio-temporal representation, allows to use nearly 100 vectors (taking into account $F_s = 250\text{Hz}$ and assigned values of parameters J

and τ). Similarly, the vectors used to create the covariance matrix C_b , containing noise, are formed on the basis of nearly 100 vectors from the nontarget segments and adequately less vectors from the target ones (their number is dependent on the assumed value of Δ parameter).

5.1.3 Tests of the methods performance

First, the section presents a procedure of search for favorable values of parameters δ and Δ introduced in description of spatio-temporal filtering (Section 3.3) and discussed further in Section 5.1.2. Subsequently the results obtained by each method are presented along with the favorable for a given method parameters values.

5.1.3.1 Search for favorable values of parameters δ and Δ

Parameters, δ and Δ which have an impact on the number of constructed vectors to be magnified and suppressed (described in Section 3.3.1 and shown in Fig. 5.5) were determined, separately for each method, in an automated procedure utilizing a crossvalidation on the learning subsets only. Among 7 learning sessions for a given subject, one was selected as a validation session, 6 remaining were used to construct the filter. The procedure was repeated for every learning session as a validation session and the results of EPs classification were summed up for every set of parameters. The most advantageous set that allowed to obtain the highest score for a given method was saved. The initial experiments indicated that the change of Δ had a minor effect on the final scores, therefore, its value was set by default to 0.1s. On the contrary, optimal δ varied depending on the type of the method applied; the exact results of the parameter value selection are gathered in the next subsection along with the results obtained by each method.

5.1.3.2 Verification tests

The results of verification of all the methods described in Chapter 3 when applied to the "Item selection" database are presented in Table I. OSTF, whose capabilities to suppress the noise in the multichannel signals were efficacious to detect low amplitude fetal QRS complexes [61], fails when applied to BCI signals with the average accuracy for all subjects falling below 50%. Even though this eigendecomposition based procedure allows to attenuate the fluctuations associated with normal activity of the brain, it cannot sufficiently enhance the desired component of such low amplitude. High value of standard error of mean (SEM, given in Table 1 next to the percentage accuracy) for OSTF confirms the method performance was relevantly different for individual subjects. For ones whose brain evoked responses were greater, it managed to create quite efficient filter, for those with more challenging, tiny brain responses the filtering went rather wrong. For *GSTMF*, in which the averaged model of target responses was introduced, the accuracy results go up notably, to above 82%. Moreover, the inter-subject performance inequalities, measured

by SEM, are diminished clearly as well. Only slight improvement of accuracy was obtained when *MSTMF* was applied. The refinements implemented in this method: averaged nontarget model construction which was subtracted from the averaged target model, as well as removal of the super Gaussian artifacts using the DCT based algorithm allowed to raise the accuracy to 83.52%. However, another major breakthrough in the efficiency is obtained when decision rules are modified, so that the maximal response of the filter output is searched in a wider time range, not only at one location (see Section 3.6.2 for details). In that, the limit of 90% properly recognized blocks for all subjects is reached. The reason of such significant improvement is visually explained in the example shown in Fig. 5.7.

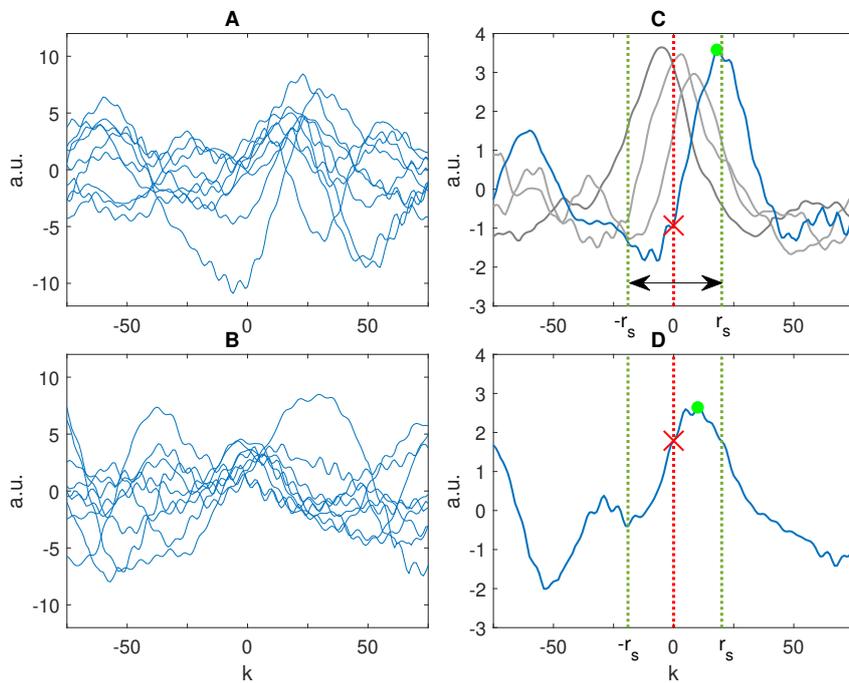


FIGURE 5.7: Visualization of the decision rules (modified versus classical ones): A) MSTMF responses to target flashes within one block ($I_T = 9$ consecutive runs of flashes, k on the horizontal axis is introduced in (3.23)), B) MSTMF responses to I_T flashes of a single nontarget object within the same block as in plot A, C) average target MSTMF responses (the one in blue is the mean of the plots in A; the gray ones correspond to different blocks of flashes), D) the average nontarget MSTMF response (the mean of the plots in B). The double arrow shows the extended region of the search for the maximal filter output. The red crosses mark the values of the scores obtained for $k = 0$ (expected position according to CDR), and the green points, the ones obtained using MDR (the maxima in the specified range of the search).

Like in this figure it has been observed during the experiments that maximal responses at the filter output were frequently slightly shifted with regard to their expected positions. Because these response peaks had rather high slope, the values at positions checked using classical decision rules could decrease abruptly, like in plot C in Fig. 5.7 where the blue average target response drops below zero. Wider range of search allowed to capture properly more target responses. As a result of preliminary experiments the search range T_s was set to 0.1s. Almost all latency delays, having its source in intra-subject response time variation as well as shifts introduced by the signal acquisition equipment, were included within this period of time. It must be noted that increasing T_s even further can lead to deterioration of the results, particularly when low number of runs are available in a block. In such a case, there is a risk that some high noise peaks can appear in the segment analysed by the classifier.

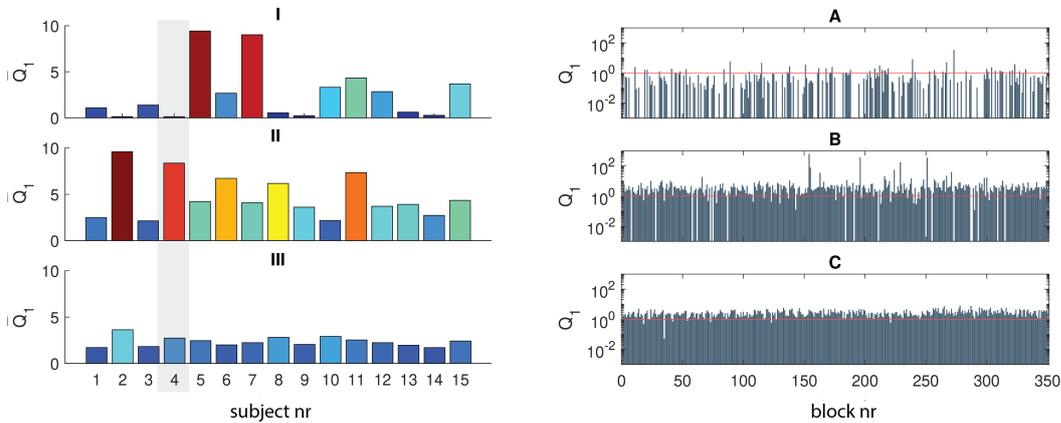


FIGURE 5.8: Left plots: histograms of the blocks averaged qualities coefficients \bar{Q}_1 for all 15 subjects for the methods: *OSTF* (plot I), *GSTMF* (plot II), and *MSTMF + MDR* (plot III). With the grayed frame marked is the subject 4 which is presented in more details (Q_1 coefficients for 350 single blocks) in the plots on the right side of the figure. Red horizontal lines indicate the detection threshold, the blocks whose Q_1 is above this threshold (equal to 1) have properly recognized target responses.

Besides the percentage accuracies collected for the methods, the differences between *OSTF* and the methods applying the two breakthrough changes (*GSTMF* and *MSTMF + MDR*) has been studied more thoroughly. Worthy of further investigation was how the methods performed for an individual block (the part of the experiment where the flashes of all objects were repeated multiple times, up to times 10 in this database, and the responses to them averaged). To this aim a quality coefficient Q_1 was introduced as a quotient of the averaged target response amplitude (in a given block) and the highest nontarget averaged response amplitude (in the same block); a correct target detection in a block is accomplished when $Q_1 > 1$. By thus, extended information, other than just a binary decision about a success or

a failure in detecting the EP, is presented in Fig. 5.8. Upper most histogram, on the left (I), which gathers the averaged coefficient \bar{Q}_1 over all blocks for each subject for OSTF, confirms very significant inter-subject discrepancy in this algorithm performance. For subjects 2, 4, 9 and 12 the results are very poor, on the contrary for 5 or 7 the results are rather good. Whereas the middle histogram (II), for GSTMF, shows expected, overall improvement, for MSTMF + MDR (plot III) the averaged coefficients \bar{Q}_1 are lower almost for all the subjects. It does not preclude, however, this method to achieve the highest accuracies because of its consistency over all subjects. Such a consistency is visible particularly watching the right plots in Fig. 5.8 which present the coefficients Q_1 for single blocks (all 350 blocks analyzed for the 4th subject in the testing subset). Both OSTF and GSTMF (histograms A and B, respectively) show an outstanding performance for some single blocks (to properly present such large bars, the logarithmic scale was selected). However, there are also many blocks which are below the detection threshold (equal to 1 and marked with the red horizontal lines). Particularly, for OSTF (histogram C) the number of such blocks below the threshold is massive. On the contrary, MSTMF + MDR shows rather small values of coefficients Q_1 but only a few of them dropping below the threshold. Stability and consistency of this method, over all analyzed blocks, seems to be crucial in terms of the achieved accuracy of EP detection.

Slightly higher results in comparison to MSTMF + MDR(DCT) are obtained when additional SVM classifier is involved. As described in Section 3.6.3 the output signal of MSTMF contained in a segment of size T_s forms an input feature vector to be classified by SVM. During tests of this method the length of the feature vector was set to 0.1s, what corresponded to the search range T_s applied for the modified decision rules e.g. in MSTMF + MDR(DCT). In order to collect the feature vectors to learn the SVM classifier, the learning subset was split on 6 learning sessions and one validation session. The MSTMF filter was constructed for these 6 sessions, and the output filter responses gathered from a validation session were saved. The operation was repeated so that each session was used as a validation session. Therefore, the chosen segments of the collected filter responses from 7 validation sessions were used to learn the SVM classifier. Accuracy of EPs detection exceeding 91% obtained using this method confirmed that the analysis of the filter responses in a wider range, not only at one position, was essential.

It must be noted that the accuracies presented in Table I surpass slightly the results published in [60]. It is caused by application of the denoising algorithm for both learning and testing subsets (and not only for the learning subset). More detailed analysis of the influence of such a change is described in the next subsection. To indicate which subsets underwent artifact suppression, the algorithm descriptions (see Section 3.7 for methods nomenclature details) are followed by L or $L\&T$ addition which represents learning subset denoising only and learning + testing denoising, respectively.

Table I. Accuracy (\pm standard error of mean (SEM)) of target stimuli detection using the methods proposed (the methods were described in [60], at the time only DCT based deoising was introduced, results which concern the denoising algorithms based on ICA will be presented later in the section). L and T letters next to the artifacts removal method (DCT) inform whether only learning (L), or both learning and testing subsets (L&T) underwent artifacts removal.

Method	Accuracy[%] \pm SEM
OSTF+CDR $\delta = 44ms$	47.07 \pm 8.01
GSTMF+CDR $\delta = 68ms$	82.90 \pm 1.56
MSTMF+CDR (DCT: L&T) $\delta = 60ms$	83.52 \pm 1.41
MSTMF+MDR (DCT: L&T) $\delta = 20ms$	90.97 \pm 1.61
MSTMF+SVM (DCT: L&T) $\delta = 20ms$	91.05 \pm 1.70

5.1.4 Investigation of the new ICA based approach to artifacts removal

The artifact removal added to the methods described in Table I involved application of DCT based technique. This method, which is usually well-suited to interpolation of the missing values in images, emerged to be a good alternative also for similar operations on 2-dimensional representation of the multichannel signal. EEG signal traces of neighboring channels form kind of a surface which is caused by similarity of the potentials detected by the electrodes closely placed on a head. Such property makes the DCT based method useful to replace the outlying artifacts particularly when they are surrounded by artifact-free regions. The method, however, performs worse when the artifacts spreads to a larger area (like for the multichannel artifact present in the measured signal). In such a case the spatial information is insufficient, what was shown in Section 5.1.1. Due to that a search for the alternative denoising methods was desirable.

The second proposed group of algorithms of artifacts removal is based on a more traditional approach utilizing ICA and a kurtosis measure to distinguish the noisy source signals from the proper ones. The detailed description of these algorithms can be found in Section 4.3 and particular solutions (L-ICA1 and L-ICA2) developed for the currently analysed database, containing low number of channels, are explained in Section 4.3.2. The ICA based algorithm is particularly useful when the source of artifacts spreads for a greater number of channels. The reason of this is that the procedure of zeroing (L-ICA1) or suppression (L-ICA2) of the artifacts detected in noisy source signals has an impact on all original measured signal channels. Their reconstruction using such modified source signals (with artifacts rejected) allows for more efficient removal of noisy peaks, even those of rather low amplitude, in the lowest kurtotic channels.

Kurtosis threshold thr , which limits the accepted source signals in L-ICA algorithms, was determined using the same crossvalidation procedure as described in Section 5.1.2. The learning subset was split on 6 learning sessions and a validation one. For the selected value of thr the filter was constructed and the results obtained

Table II. Accuracy (\pm standard error of mean (SEM)) of target stimuli detection using the methods proposed (applied along with the denoising algorithms based on DCT and ICA (L-ICA1 and L-ICA2)). In brackets there is a note of the denosing algorithm used; L and T letters next to the artifacts removal methods inform whether only learning (L), or both learning and testing subsets (L&T) underwent the artifacts removal procedure.

MSTMF+MDR (DCT: L only) $\delta = 20ms$	90.55 \pm 1.70
MSTMF+MDR (DCT: L&T) $\delta = 20ms$	90.97 \pm 1.61
MSTMF+SVM (DCT: L only) $\delta = 20ms$	90.93 \pm 1.67
MSTMF+SVM (DCT: L&T) $\delta = 20ms$	91.05 \pm 1.70
MSTMF+MDR (L-ICA1: L only) $\delta = 20ms$	90.25 \pm 1.72
MSTMF+MDR (L-ICA1: L&T) $\delta = 20ms$	91.41 \pm 1.71
MSTMF+MDR (L-ICA2: L only) $\delta = 20ms$	90.19 \pm 1.72
MSTMF+MDR (L-ICA2: L&T) $\delta = 20ms$	91.31 \pm 1.71
MSTMF+SVM (L-ICA1: L only) $\delta = 20ms$	90.55 \pm 1.80
MSTMF+SVM (L-ICA1: L&T) $\delta = 20ms$	91.45 \pm 1.73
MSTMF+SVM (L-ICA2: L&T) $\delta = 20ms$	91.33 \pm 1.74

on the validation session were stored. The procedure was repeated for all sessions used as a validation one. The overall results were used to determine which value was most favorable. The values of kurtosis threshold from the set $\in [3, 4...10]$ were checked. The most favorable value of the parameter *thr* found in this procedure was 5, and this value was applied in the experiments.

The results given in Table II indicate that the ICA based approach performs better when both learning and testing subsets are included into denoising procedure. Both MSTMF+MDR (L-ICA: L&T) and MSTMF+SVM (L-ICA: L&T) surpass slightly the best solution founded on DCT (*MSTMF + SVM(DCT : L&T)*). However, when only learning subset is denoised the advantage of the algorithms based on DCT is visible for both MSTMF+MDR as well as MSTMF+SVM.

5.1.5 Illustration of the influence of parameters J and τ on the accuracy and computational time

In order to evaluate the influence of J and τ parameters on the classification accuracy, the verification procedure was repeated for τ values in range $[2, 18]$. For every value in this range J was calculated according to instructions given in Section 3.3, so that the filter length was maintained. Fig. 5.9 presents how changes of τ influence the classification accuracy when MSTMF+MDR is applied. For values $\tau > 6$ the accuracy starts to worsen slightly. More significant decrease is observed for $\tau = 10$ what is equivalent to the sampling frequency 25Hz. Such an observations confirms the validity of the assumption made in Section 3.3 that the sampling frequency should not be decreased below 40Hz. Selection of τ has also an influence on the computational time. Very low value of τ implies high sampling frequency, what leads to increased number of samples being processed both in the learning and test subsets.

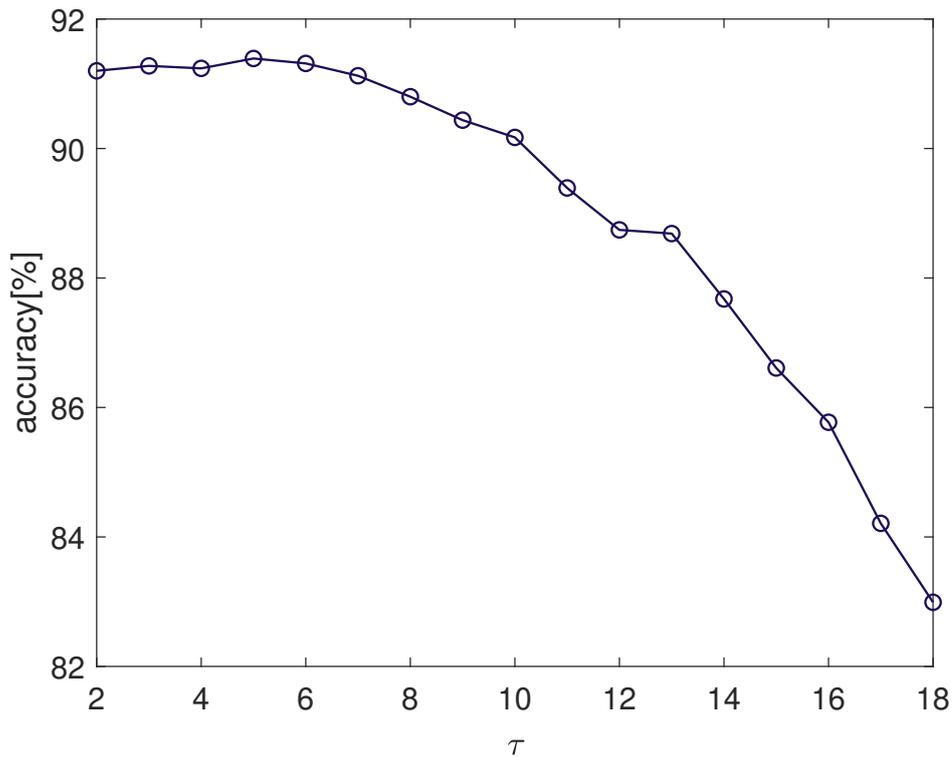


FIGURE 5.9: Illustration of the dependency of MSTMF+MDR accuracy on the τ parameter value.

Fig. 5.10 illustrates these dependencies: for $\tau < 4$ the computational times (for both learning and testing subsets showed on plots A and B, respectively) increase abruptly, on the contrary for $\tau \in [4, 18]$ these times are not significantly changed.

5.1.6 On the execution times of the methods

Execution time is a crucial factor when decided whether the algorithm can be applied in real-life applications. Therefore, the effort was made to estimate execution times of all the methods alone and along with the different denoising algorithms. In order to determine the inter-method differences in computational times, the value of τ equal to 6 was used, this value was applied also to verify the methods performance (see Section 5.1.2). The experiments which aimed in estimating these times were performed on the computer: Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz 3.19 GHz with 16 GB RAM, and the results are gathered in Table III. These are the times averaged over the subjects and involve analysis of the complete signal for a given subject; the lengths of the learning and testing data for one subject are around 37.3 min and 56 min, respectively.

The core methods of STF have rather similar times of learning stage execution after the averaging of spatio-temporal vectors in matrix \mathbf{A} is introduced beginning with GSTMF (see Section 3.4 for details). OSTF stores very large matrix \mathbf{A} whose

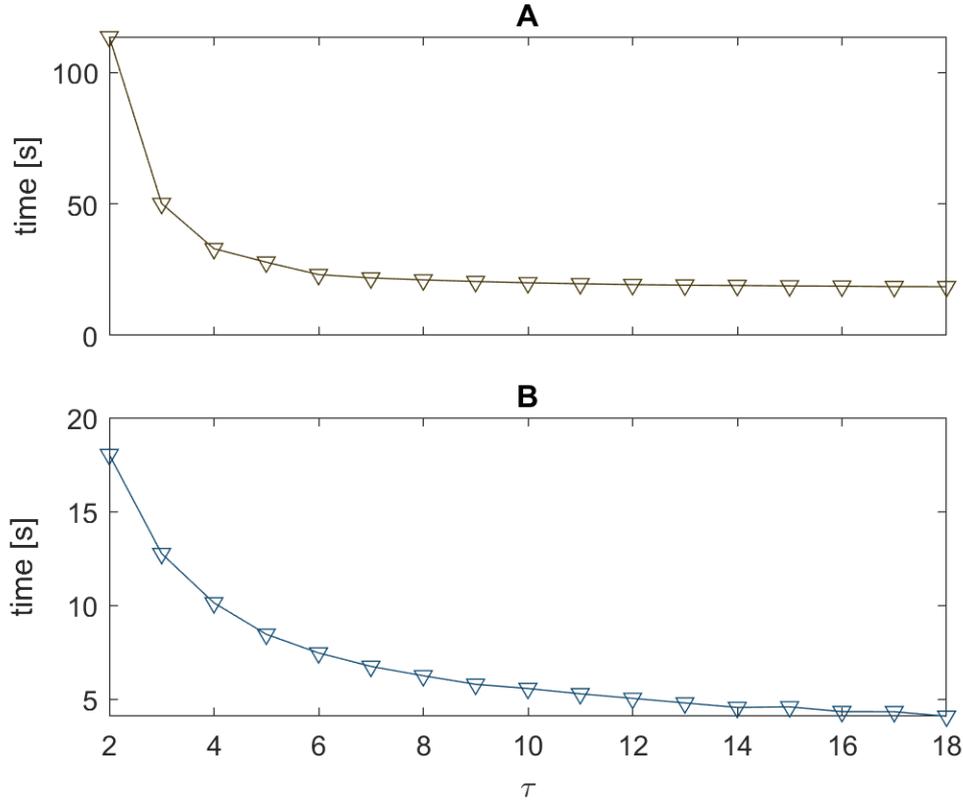


FIGURE 5.10: Illustration of the dependency of the computational time on the τ parameter value. Plot A applies to the learning subset, plot B to the testing subset.

size is not reduced until the covariance matrix \mathbf{C}_b is calculated. It leads to significantly increased computational costs thus the method computes longer. On the contrary, testing execution times are almost the same for *OSTF*, *GSTMF*, and *MSTMF*. When artifact removal methods are included, duration of both the learning and testing phases goes up. *MSTMF + MDR(DCT : L&T)** concerns execution times (learning and testing subsets), described in [60], when all complete 1.4s signal parts were included in denoising procedure. The new procedure of merging these segments mentioned in Section 5.1.1, which removes redundant information, allowed to reduce the execution times significantly: by around 60% for the learning phase, and almost 80% for the testing phase. In [60] the DCT as a method of artifact removal was considered only for a learning phase due to such a long average computational time (161.20s) measured for the testing phase. Such an enormous change in algorithm execution time makes this method more possible to be used in practical applications. Nevertheless, according to Table III the time efficiency of the algorithms when ICA is involved into denoising is even greater. Worth noting is that for DCT artifact removal, the execution time is highly dependent on the artifacts number present in the signal. In case of noisier signals it would grow much faster in comparison to the ICA based algorithm. Such a statement can be legitimated by observation of the standard deviations for both methods. Whereas for L-ICA it is rather low, it is increased significantly for DCT up to nearly 2.9 and 7 for learning

Table III. Computational times (\pm STD).

Method	learning time[s] \pm STD	testing time[s] \pm STD
OSTF	39.28 \pm 0.26	7.19 \pm 0.85
GSTMF	23.26 \pm 0.12	7.19 \pm 0.83
MSTMF+MDR	25.00 \pm 0.12	7.66 \pm 0.82
MSTMF+MDR(DCT: L)*	99.03 \pm 3.66	7.17 \pm 0.85
MSTMF+MDR(DCT: L & T)*	99.03 \pm 3.66	161.20 \pm 28.35
MSTMF+MDR(DCT: L & T)	39.94 \pm 2.89	33.48 \pm 6.99
MSTMF+MDR(L-ICA1: L)	30.97 \pm 0.37	7.66 \pm 0.82
MSTMF+MDR(L-ICA1: L & T)	30.97 \pm 0.37	19.74 \pm 1.98

and testing phases, respectively. It confirms the inter-subject differences when artifacts number is considered and resulting large differences of DCT execution times. Based on the aforementioned calculations, L-ICA seem to be more adequate to be applied both in the learning and testing phase.

5.1.7 Final results and comparison to the participants of the 2019 IFMBE Scientific Challenge

Two best methods, along with the different denoising algorithms, described in the earlier section are collated in Table IV with the methods proposed by the participants of the 2019 IFMBE Scientific Challenge. Additionally, the results obtained using BLDA [40] (the method which applies Bayesian regularization to overcome limitations of LDA) were added; the results were gathered using a software provided by the authors. In comparison to the challenge results, all best *MSTMF* based methods would place on the second position with just a slight loss to the winning algorithm, accuracy of *MSTMF + MDR(L - ICA1)* is lower by 0.85 percent point. Rather significant improvement is observed in comparison to the second best competition algorithm (accuracy higher by more than 7 percent points) and to the reference method founded on LDA: BLDA (accuracy is higher by more than 8 percent points in this case). Analyzing the results given in Table IV, one can assume the methods which make use of the neural networks have a great potential as the efficient BCI classifiers. Nevertheless, the results obtained by the methods described in this thesis prove that some additional operations, which modify and adjust the linear filter to the problem of detection very tiny brain potentials, can make it very competitive in this field as well.

5.2 Experiments with the "speller" database

"Speller" is a second database which has been used to verify performance of the methods presented in the thesis. The experiment design is slightly different, in comparison to the "Item selection" database, due to the change of the user interface (which contains 36 characters), and the way the stimuli are presented (flashes

Table IV. Accuracy (\pm standard error of mean (SEM)) of target stimuli detection using the methods proposed, the Bayesian LDA, and the methods participating in 2019 IFMBE Scientific Challenge (for their precise description see [88]).

Method	Accuracy[%] \pm SEM
MSTMF+MDR (DCT: L&T) $\delta = 20ms$	90.97 \pm 1.61
MSTMF+SVM (DCT: L&T) $\delta = 20ms$	91.05 \pm 1.70
MSTMF+MDR (L-ICA1: L&T) $\delta = 20ms$	91.41 \pm 1.71
MSTMF+SVM (L-ICA1: L&T) $\delta = 20ms$	91.45 \pm 1.73
BLDA	82.55 \pm 1.26
CNN based on EEGNet	92.3 \pm 1.8
CNN-BLSTM	84.0 \pm 3.2
LDA, CWT, and PCA	82.0 \pm 2.5
SVM	81.5 \pm 2.6
400 LDA classifiers and FGDA	81.2 \pm 2.1
VB-ARD	80.3 \pm 2.2
BLDA, RUSBoost, and CNN	76.3 \pm 2.9
MLP	70.0 \pm 3.8
LDA	67.2 \pm 3.3

of the whole column or row) (see Section 2.4 for more details). For the purpose of the BCI contest (BCI Competition III Challenge), the data related to each subject has been originally split to the learning and testing subsets containing 85 and 100 blocks, respectively. This dataset is composed of 64 channel recordings and all were used during testing for the purpose of this thesis. The participants of the contest had a choice to use the whole amount or reduce the number of channels. Their solutions were examined on the testing datasets, containing 100 blocks for each of the subjects (A and B). The results were gathered for both 5 and all available 15 averaged runs. In this thesis, however, the results are gathered only for all 15 available runs, what is consistent with the rules applied for the "Item selection".

5.2.1 Signal preprocessing and artifacts removal

"Speller" dataset was band-pass filtered using the third order Butterworth filter with cut-off frequencies of 1Hz and 12Hz, respectively. The original sampling frequency was 240Hz and remained unchanged during the experiments. The signals used to construct the proper filter underwent the secondary pre-processing operation. Likewise in Section 5.1.1 it involved artifacts detection and their removal by means of the methods based on DCT or ICA (here, versions of the algorithms for high number of channels are applied: H-ICA1, H-ICA2, and H-ICA3).

Figures 5.11 and 5.12 show the exemplary results of DCTBS. Fig. 5.11 contains an original 64-channel EEG recording with the noise artifacts (channels detected as noisy are marked in bold black). Exactly the same time frame of the signal is included in Fig. 5.12; here the artifacts are reduced (in bold black are the plots of the

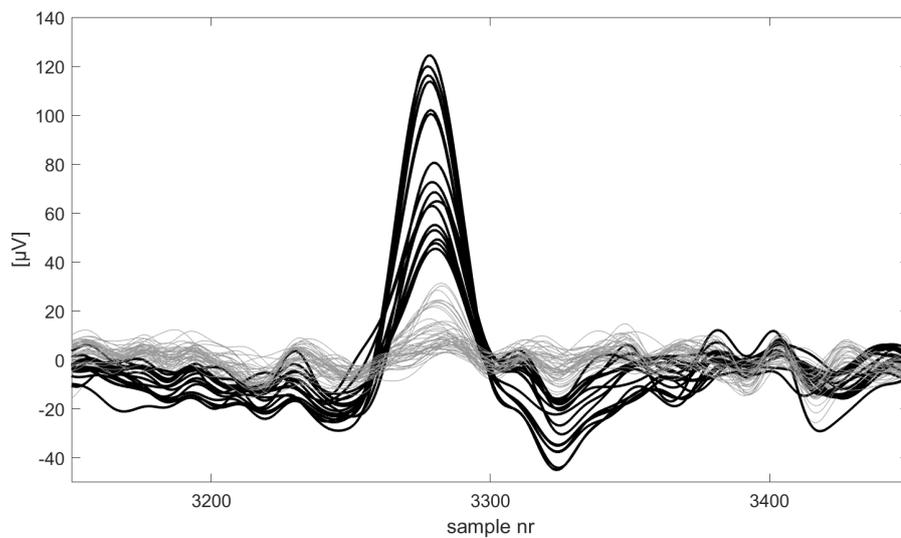


FIGURE 5.11: Results of outlying artifacts suppression using DCTBS: original 64-channel EEG recording; in bold black: noisy channels, in grey: acceptable channels

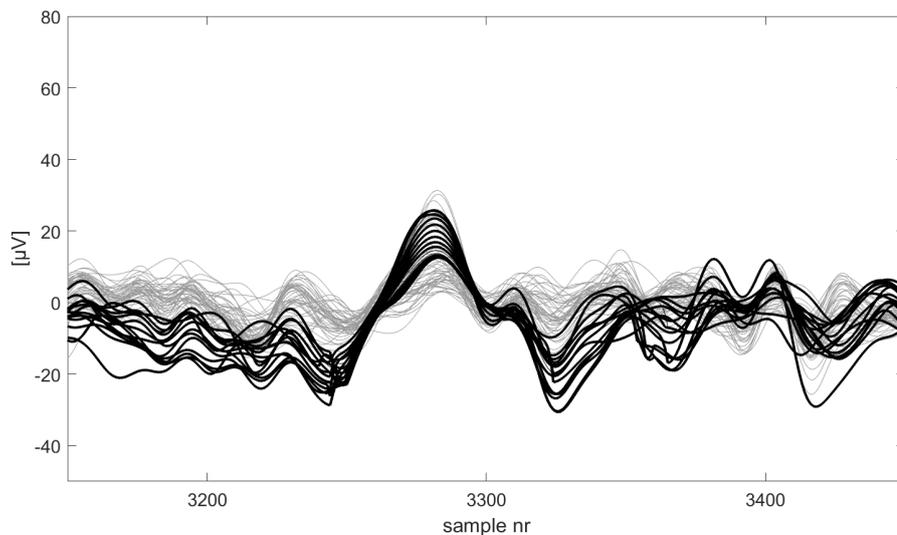


FIGURE 5.12: Results of outlying artifacts suppression using DCTBS: the modified 64-channel EEG recording; in bold black: modified channels, in grey: unchanged channels. The amplitude scale has been changed with respect to Fig. 5.11

channels modified). Watching the plots on the output of DCTBS, one can observe high similarities between the reconstructed channels. It is caused by the way the DCT based algorithm fills the gaps (created by blanking the artifact samples), where the neighboring proper channels are used to reconstruct the ones containing artifacts. The procedure is similar to filling the empty pixels or larger gaps in pictures. In this case not all available channels were classified as noisy (above the threshold). Therefore, the interpolated channels segments (replacing the noisy ones) can be reconstructed using not only preceding and subsequent time samples but also samples

of other channels within this time frame. Obviously, this leads to better reconstruction of the signal channels, visible in Fig. 5.12.

Quite different visual effects of denoising are obtained when ICA based methods are applied. In such cases each individual channel is changed in the aftermath of subtracting the approximated artifacts. Therefore, the similarities between channels (see. Fig. 5.13), particularly within the segments where artifacts were removed, are smaller, however still resemble the inter-channel similarity typical for EEG recordings. Depending on the type of the algorithm applied small visual differences of the resulting plots can be noticed. As shown in Fig. 5.13, the output of H-ICA1 (plot B) and H-ICA2 (plot C) are similar. Some of the noisy channels are more effectively attenuated after H-ICA3 (plot D).

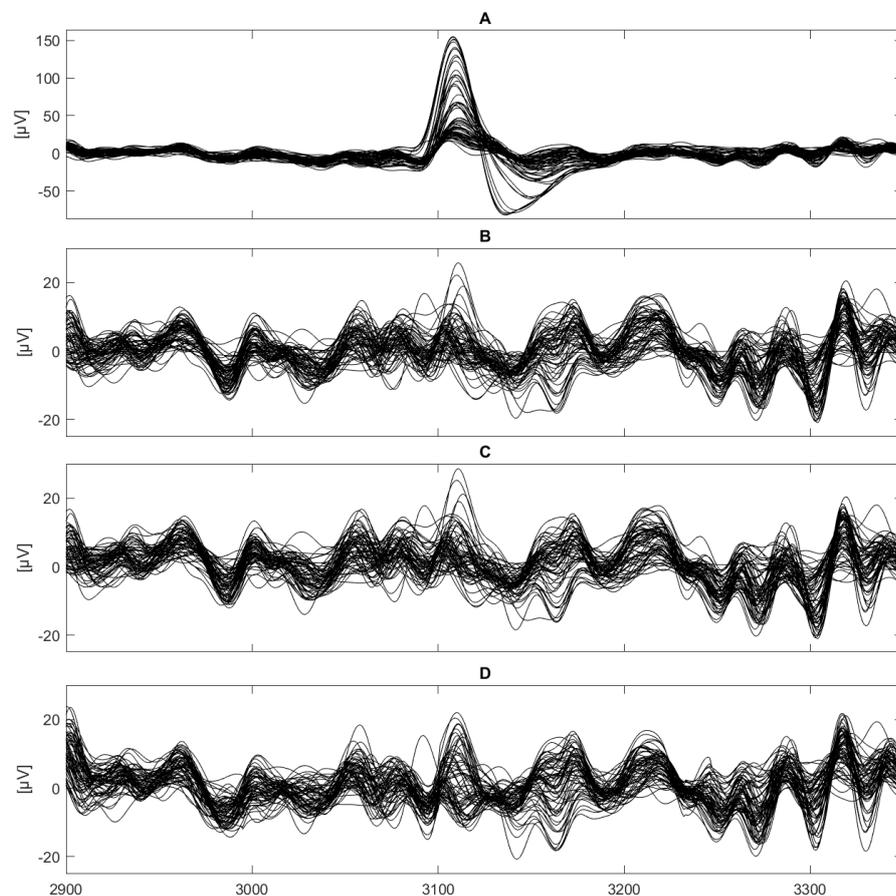


FIGURE 5.13: Results of outlying artifacts suppression using the ICA based algorithms. Plot A: original 64-channel EEG recording, plots B, C, and D are the same 64-channel recording with the artifacts suppressed using algorithms H-ICA1, H-ICA2, and H-ICA3, respectively.

Another example of H-ICA3 performance is illustrated in Fig. 5.14, in plot B. Here, in addition to the multichannel artifacts, presented is a single artifact which occurs in one channel only (marked with a blue ellipse in plot A). Such an artifact

may be the consequence of an electrode movement, detachment, and is rather challenging for ICA, particularly when the ICA is applied to limited number of channels. Moreover, usually, ICA is dedicated to eliminate the artifacts which spread to more channels (one source signal affects signals acquired by many electrodes). In this particular case the single artifact seem to have too low amplitude to place this channel among \hat{M} most kurtotic ones. As the source signals are estimated using these \hat{M} channels only, there is no representation of the single artifact in any of the source signals. On the other hand, for the multichannel noise there is a better chance to remove even low amplitude artifacts in low kurtosis channels due to the fact the given artifact has typically higher amplitudes in other channels so that is included in ICA procedure (the ICA based algorithm modifies all channels associated with a given time segment of the found artifacts). Therefore, H-ICA3 succeeds at removal of mul-

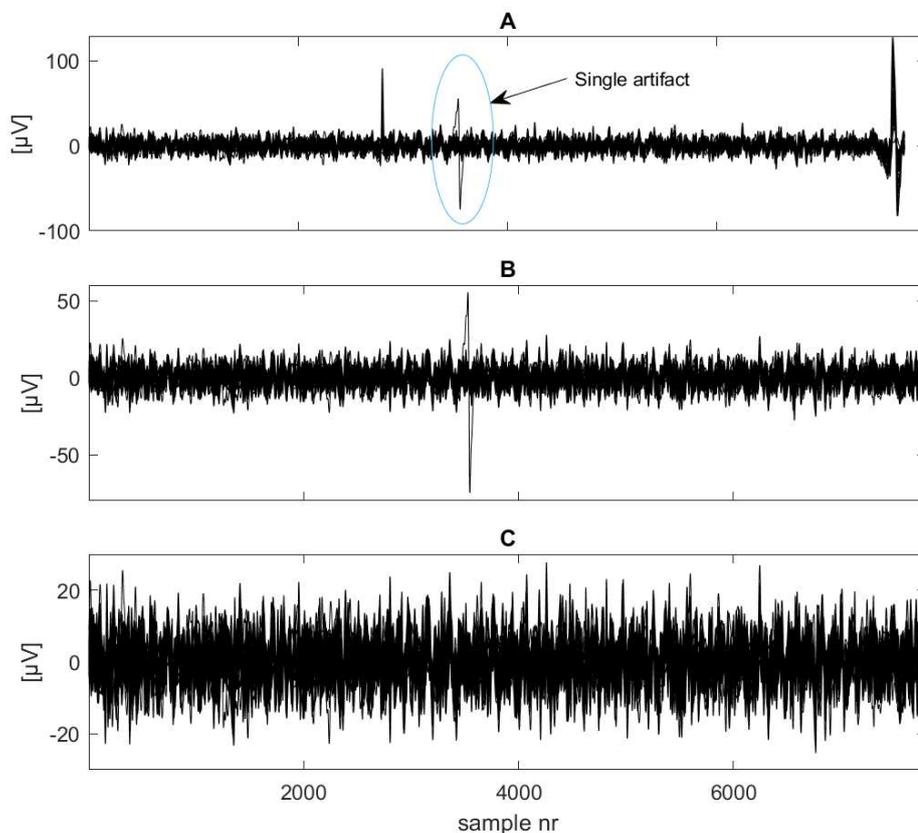


FIGURE 5.14: Visualization of H-ICA3 performance. Plot A: original 64-channel EEG recording containing both multichannel artifacts and single artifact (in one channel only, marked in blue ellipse), plot B presents the outcome of H-ICA3 denoising, and plot C is a results of H-ICA3 applied twice.

tichannel artifacts and fails in case of a single one. A possible remedy is to apply the H-ICA3 twice. As shown in plot C, such an approach allows to remove the single channel artifacts because this time the affected channel is more easily classified as one of \hat{M} most kurtotic. Second use of H-ICA3 does not change the rest of the signal

channels, which were already denoised in the first step. It is assured by the way the thresholds were calculated in Section 4.3.3. This solution can be useful also to process very noisy signal parts which contain multiple spikes at full length of these recordings. In the aftermath, the calculations which aim in determining the thresholds (based on the 5th and 95th percentiles) can be affected and by thus there is a risk that some of the lower amplitude spikes remain. The spike which were skipped in the first step can be more effectively attenuated in the second turn of H-ICA3.

5.2.2 Adjustment of the developed method's parameters

The GSTMF filter parameters J and τ used in the "Item selection" database (Section 5.1) were recalculated in order to fit the lower sampling frequency applied in "speller" database (240 Hz). To this aim the same formulas, defined in Section 3.3, were used. The parameters values set during the experiments were: $J = 20$ and $\tau = 6$. Value of Δ was set to 0.1s (the same as for the experiments with the previous database). However, on the contrary to the "Item selection" database, here, the value of δ was less crucial to the methods performance. The experiments have shown that the best, stable performance of the methods, presented in this section, was observed for $\delta \in [20, 30]ms$. Finally, the parameter associated with the range of search for the maximum of the filter response (T_s) was set to the value of 0.1s (the same as for the previous database).

5.2.3 Verification of the selected methods

The experiments on this database involved use of the best performing methods, which were developed during the tests on the "Item selection" database, described in Section 5.1. The final version of algorithm *MSTMF* was checked along with the denoising algorithms (based on DCT or ICA) as well as with different decision rules (CDR or MDR). Likewise in "Item selection" database, here, the focus was made on the results obtained using all available runs within a block. Therefore, the Tables V and VI contain only the results of classification where all 15 available runs were analyzed.

Table V contains the classification results obtained by *MSTMF* applied using different types of decision rules and specific algorithms of denoising based on ICA (H-ICA1, H-ICA2, H-ICA3, see Section 4.3.3 for details) or DCT (see Section 4.2 for details). The basic version of *MSTMF* applied with classical decision rules (*MSTMF* + CDR), only, allows to obtain quite competitive results to those described in literature. However, the accuracy drops slightly when the DCT based method of denoising is added, the performance is even worse after applying primary versions of ICA based denoising algorithms (H-ICA1, H-ICA2). As mentioned in the Section 4.3 such a database containing 64 channels was more difficult when ICA was considered as a tool to remove the artifacts. ICA could not be applied directly to all available channels. Additional procedure, which uses only selected highly kurtotic

Table V. The influence of different modifications of *MSTMF* and denoising algorithms on the classification results.

Method	Accuracy[%]±SEM
MSTMF+CDR $\delta \in [30, 40]ms$	95.5%
MSTMF+CDR (DCT) $\delta \in [30, 40]ms$	95.0%
MSTMF+CDR (H-ICA1) $\delta \in [30, 40]ms$	94.0%
MSTMF+CDR (H-ICA2) $\delta \in [30, 40]ms$	94.5%
MSTMF+CDR (H-ICA3) $\delta \in [30, 40]ms$	97.0%
MSTMF+MDR $\delta \in [30, 40]ms$	98.0%
MSTMF+MDR (DCT) $\delta \in [30, 40]ms$	97.5%
MSTMF+MDR (H-ICA1) $\delta \in [30, 40]ms$	97.5%
MSTMF+MDR (H-ICA2) $\delta \in [30, 40]ms$	98.0%
MSTMF+MDR (H-ICA3) $\delta \in [30, 40]ms$	99.0%
MSTMF+SVM (H-ICA3) $\delta \in [30, 40]ms$	99.0%

channels to estimate the source signals (from which the noisy ones are then used to approximate the artifacts in all channels), needed to be implemented in order to process such a large number of available channels. Nevertheless, these two algorithms (H-ICA1 and H-ICA2) in their core constitute a very common approach of ICA application to the artifacts removal where modified are the samples along the entire length of the signals (either by reconstructing the signal without the noisy source signals or by subtracting the approximated artifacts like in the mentioned algorithms). Such a solution seems to be unfavourable when applied to the "speller" database. On the contrary, application of H-ICA3 with *MSTMF + CDR* allows to improve the final accuracy to 97%. This is the only denoising algorithm proposed which made a positive difference in classification performance of *MSTMF + CDR*. The reason of this could be the fact H-ICA3 modifies only the samples detected as parts of the artifacts. Such an observation is valid also for *MSTMF + MDR*.

Whereas this method alone allows to overcome *MSTMF + CDR(H - ICA3)*, still application of DCT, H-ICA1 or H-ICA2 has rather adverse or insignificant influence on the classification accuracies. The advantage of *MSTMF + MDR* can arise from the way the output filter responses are analyzed. Firstly, averaging of the whole responses, lasting 1s, (according to description in Section 3.6.2) facilitates the evoked potentials enhancement rather than noisy peaks which are more randomly located in the analyzed time frame. Moreover, MDR assures detection of the averaged response maximum even though it is slightly shifted from its expected position. The effectiveness of DCT or ICA based algorithms used to remove the artifacts is similar for *MSTMF + MDR* and *MSTMF + CDR*. Neither of the methods including H-ICA1, H-ICA2 and DCTBS allowed to improve the accuracies obtained by *MSTMF + MDR* applied alone, the accuracies are rather diminished slightly. As previously, only H-ICA3 guarantees further progression, it enabled to obtain 99% accuracy. It confirms again, the algorithm which operates selectively on

the noisy signal channel by modifying only the samples of detected artifacts is advantageous when compared to other ICA or DCT based solutions presented in this thesis. Equally high efficiency (99%) was obtained by application of MSTMF+SVM (H-ICA3) algorithm.

5.2.4 Results and comparison to the winners of BCI challenge

Table VI gathers accuracy results on the "speller" dataset obtained by the contestants of the BCI Competition III, and below the results of the methods found in literature, acknowledged as the best performing ones on the given database. Finally, the performances of the methods which are the subject of this thesis are listed below the second double line, in bold. Presented are only the algorithms which are either competitive to the competition winning method or the ones which overcome it. *MSTMF + CDR* applied alone could be placed on the second position with a slight loss to the winning, ensemble support vector machine (ESVM) based, algorithm. However, when the proper preprocessing step is added (*MSTMF + CDR(H-ICA3)*) the accuracy gets higher by one percentage point and surpasses the outcome of the challenge winner. Further enhancements, introduced in *MSTMF + MDR*, allow this method to place on the first position with 2 percentage points of advantage to the competition winners. Whereas, the accuracy results for *MSTMF + MDR* are equal to those obtained by the best reference method found in literature (ensemble of weighted SVMs (EWSVM)), addition of H-ICA3 preprocessing once or twice refines the accuracies so that only 1% and 0.5%, respectively, are not properly classified. Therefore, both *MSTMF + MDR* and *MSTMF + SVM* applied with H-ICA3 denoising algorithm outperform all best known solutions applied to this dataset.

Table VI. Results of BCI Competition III Challenge 2004 [9]. Below the first double line listed are the results achieved by the best performing methods found in the literature (with the proper reference in brackets). The accuracy values obtained using *MSTMF* combined with different denoising algorithms are presented in bold below the second double line.

Method	Accuracy[%]±SEM
SVM (17 SVM based classifiers)	96.5%
SVM (multiple classifiers)	90.5%
PCA and t-statistics based classifier	90.0%
GBM	89.5%
LDA	87.5%
SVM	83.0%
PCA and SVM	78.5%
KFD	75.0%
ICA and Bayesian network	33.5%
SVM	7.5%
PCA + WELM [59]	97.0%
MCL + LDA [69]	97.5%
EWSVM [64]	98.0%
MSTMF+CDR $\delta \in [30, 40]ms$	95.5%
MSTMF+CDR (H-ICA3) $\delta \in [30, 40]ms$	97.0%
MSTMF+MDR $\delta \in [30, 40]ms$	98.0%
MSTMF+MDR (H-ICA3) $\delta \in [30, 40]ms$	99.0%
MSTMF+SVM (H-ICA3) $\delta \in [30, 40]ms$	99.0%
MSTMF+MDR (2x H-ICA3) $\delta \in [30, 40]ms$	99.5%

Chapter 6

Summary

Aim of the thesis was to apply spatio-temporal filtering to classification of evoked brain potentials for development of a new and effective tool that could improve the operation of a selected type of brain-computer interfaces.

In order to assure the filter proper operation, several difficulties and limitations needed to be reconsidered and overcome. The most demanding one involved very low energy of the evoked potentials which are typically hidden within the routine activity brain waves. Analysis of the former was even more difficult in presence of the high amplitude artifacts, whose rejection was of high importance for construction of a proper filter. Another difficulty referred to a large number of features, which represent a single brain response to a stimulus, and consequently, to an insufficient number of learning data, i.e. the responses recorded during the appropriate learning sessions. Finally, the difficulties introduced by different delays of the brain responses, influencing adversely the interpretation of the results, should have been reduced.

Firstly, the problem of high dimensionality of the feature vectors was solved by inclusion of the signal segments other than those containing the target and nontarget responses, only. This allowed to avoid the singularity of the noise scatter matrix to be inverted. In turn, the presence of the high amplitude (super-Gaussian) artifacts in EEG recordings was overcome by application of the two alternative denoising techniques used to their rejection: the first based on discrete cosine transform and the second on independent component analysis (and ICA combined with empirical mode decomposition).

Because spatio-temporal filtering in its original form appeared to be ineffective to tackle the problem of immensely low signal-to-noise ratio, its objective function, used to construct the filter, was changed. The first breakthrough towards a higher immunity to noise was introduced in generalized spatio-temporal matched filtering (GSTMF). The method initiated the averaging of the spatio-temporal vectors constructed from the gathered brain responses to target stimuli, and by thus allowed to raise the classification accuracy on the first database used ("Item selection"), from around 50% to above 80%. Inclusion of the nontarget responses averaging, added in modified spatio-temporal matched filtering (MSTMF), improved the accuracy only slightly. Another crucial enhancement involved the change of the way the results

were interpreted; modified decision rules (MDR) replacing the classical ones (CDR) allowed to reduce the influence of the brain responses varying delays and by thus to elevate the results obtained for this database above 90%, to the level unattainable by most of the best known classifiers. The final improvement, incorporating a use of the SVM classifier to interpret MSTMF output signals, emerged the method achieving the highest accuracy, only slightly better however than MSTMF applied with the modified decision rules.

The best methods proposed: MSTMF+MDR, and MSTMF+SVM allowed to exceed the results obtained by most of the reference methods for the first database considered. Only the winning algorithm, based on the convolutional neural networks (CNN), was slightly more accurate. Worthy a note is that the database contained only the target and nontarget responses related segments what was unfavorable for the methods proposed in this dissertation, which benefit if more signal segments (even related to noise only) are included. The proposed methods applied to the second database ("Speller") enabled to achieve outstanding results, better than those obtained by any of the reference methods. The obtained results seem to substantiate the validity of the thesis:

"It is possible to apply the method of spatio-temporal filtering to brain responses enhancement and classification, and by the proper modification of its learning phase and the applied interpretation rules a significant growth of the classification accuracy can be achieved."

In the author's opinion the most important accomplishments presented in the thesis involve:

1. Demonstration that the linear algorithm based on the spatio-temporal filtering can be competitive or better than most known classifiers used in the BCI, on condition that all difficulties inherent to the analysis of extremely low amplitude evoked potentials are adequately investigated and handled.
2. Proof that the proper inclusion of other signal segments e.g. routine brain activity waves (unrelated with the target and nontarget responses) allows to solve the problem of small sample size (and ensuing singularity of the noise scatter matrix) without any additional regularization techniques. Better estimation of the noise scatter matrix enables construction of the classifiers whose capabilities to suppress noise are improved.
3. Introduction of the modified decision rules (MDR). This way of the results interpretation can easily be added to many existing classifiers to increase their effectiveness in detecting the evoked potentials. MDR applied in this thesis elevated significantly the accuracies of the MSTMF method, which had already incorporated all highly important modifications of the learning phase.
4. Development of the algorithms of artifacts detection and removal based on ICA (and ICA+EMD) which can be applied before the classifier learning as well

as during the BCI operation stage. The proposed most advanced solutions: L-ICA1, and H-ICA3 contribute to increase of the overall accuracies obtained by the proposed classifiers applied to databases with low and high number of channels, respectively. The algorithm dedicated for the latter databases circumvents the problem of very long computational times of the JADE algorithm (when processing a high number of channels).

5. Finally, development of the highly effective tool for BCI data analysis.

It seems justified to claim that the aims of the thesis, namely application of spatio-temporal filtering to work efficiently as a BCI classifier which is immune to the mentioned inconveniences of EEG signals processing, have been fulfilled. The proposed solutions may contribute to the development and improvement of the efficiency of currently used brain-computer interfaces.

Possible future extension of this study can include construction of an ensemble of the spatio-temporal filters whose outputs are combined during the evoked potentials classification. An interesting direction of the future research can also be related with development of rules for the filter adaptation to varying properties of the processed signals.

Appendix A

Discrete cosine transform based smoothing (DCTBS)

The regarded smoothing procedure [30], applied to a one-dimensional signal, stored in n samples long vector \mathbf{x} relies on the following model

$$\mathbf{x} = \hat{\mathbf{x}} + \boldsymbol{\varepsilon}, \quad (\text{A.1})$$

assuming that the processed signal \mathbf{x} contains some smooth component $\hat{\mathbf{x}}$ plus noise $\boldsymbol{\varepsilon}$. To determine the smooth signal, there is a need to find the right balance between its fidelity to the original signal and the degree of its roughness. This balance is expressed by the following objective function [30]

$$F(\hat{\mathbf{x}}) = \|(\hat{\mathbf{x}} - \mathbf{x})\|^2 + \rho \|\mathbf{D}\hat{\mathbf{x}}\|^2 \quad (\text{A.2})$$

where

$$\mathbf{D} = \begin{bmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1/ \\ & & & & \end{bmatrix}$$

is a difference operator used for numerical approximation of the signal derivative and $\rho \in R^+$ denotes a smoothing parameter. Thus the roughness penalty condition is defined as the energy of the $\hat{\mathbf{x}}$ signal derivative.

Putting the derivative of (A.2) to zero leads to [12]

$$\hat{\mathbf{x}} = (\mathbf{I}_n + \rho \mathbf{R})^{-1} \mathbf{x} \quad (\text{A.3})$$

where \mathbf{I}_n is the identity matrix of size $n \times n$ and

$$\mathbf{R} = \mathbf{D}^T \mathbf{D} = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix}.$$

The result of eigendecomposition of \mathbf{R} gives

$$\mathbf{R} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}. \quad (\text{A.4})$$

The consecutive columns of \mathbf{U} contain coefficients of the discrete cosine transform and the diagonal elements of $\mathbf{\Lambda}$ are equal $\lambda_i = -2 + 2\cos((i-1)\pi/n)$ [30].

Since $\mathbf{U}^{-1} = \mathbf{U}^T$ and $\mathbf{U}\mathbf{U}^T = \mathbf{I}_n$, (A.3) can be expressed as

$$\hat{\mathbf{x}} = \mathbf{U}(\mathbf{I}_n + \rho\mathbf{\Lambda}^2)^{-1}\mathbf{U}^T\mathbf{x} \quad (\text{A.5})$$

and further shortened using a diagonal matrix $\mathbf{\Gamma}$

$$\hat{\mathbf{x}} = \mathbf{U}\mathbf{\Gamma}\mathbf{U}^T\mathbf{x} \quad (\text{A.6})$$

where $\gamma_i = [1 + \rho(2 - 2\cos((i-1)\pi/n))^2]^{-1}$.

Since $\mathbf{U}^T\mathbf{x}$ and $\mathbf{U}\mathbf{x}$ define the discrete cosine transform and its inverse (IDCT) [30], the final formula for one-dimensional signal smoothing is as follows

$$\hat{\mathbf{x}} = \text{IDCT}\{\mathbf{\Gamma}\text{DCT}\{\mathbf{x}\}\}. \quad (\text{A.7})$$

In case of smoothing a two-dimensional signal \mathbf{X} of size $n \times m$, it is stored row-wise in vector $\hat{\mathbf{x}}$ of the dimension $nm \times 1$

$$\hat{\mathbf{x}} = [x_{1,1}, x_{1,2}, \dots, x_{1,m}, \dots, x_{n,m}]^T. \quad (\text{A.8})$$

For such data formula (A.7) has its equivalent form given as follows [12]

$$\hat{\mathbf{x}} = \text{IDCT2}\{^2\mathbf{\Gamma}\text{DCT2}\{\hat{\mathbf{x}}\}\} \quad (\text{A.9})$$

where DCT2 and IDCT2 stand for bivariate versions of DCT and IDCT, respectively [30]; $^2\mathbf{\Gamma}$ is the diagonal matrix of the size $nm \times nm$ and $^2\gamma_{m(i-1)+j} = [1 + 4\rho(2 - \cos((i-1)\pi/n) - \cos((j-1)\pi/m))^2]^{-1}$ with i and j corresponding to the row and column index of the processed matrix \mathbf{X} , respectively.

Application of the outlined procedure to the signal with some values missing requires an iterative procedure [30]. Prior to this, the missing values are initially replaced with their nearest neighbors (for every unknown sample $x_{i,j}$, the known one $x_{i',j'}$ with $(i-i')^2 + (j-j')^2$ minimal is searched).

Defining $\mathbf{W}_{nm \times nm}$ as the matrix whose diagonal elements $\in \{1, 0\}$ indicate the status (missing or not) of the corresponding entries of the vector $\hat{\mathbf{x}}$, the iterative formula is given as

$$\hat{\mathbf{x}}_{(k+1)} = \text{IDCT2}(^2\mathbf{\Gamma}\text{DCT2}(\mathbf{W}(\hat{\mathbf{x}} - \hat{\mathbf{x}}_k) + \hat{\mathbf{x}}_k)) \quad (\text{A.10})$$

where $\hat{\mathbf{x}}_k$ represents the smooth signal after k iterations. The method is implemented

and facilitated as Matlab function: inpaintn by D. Garcia [30]. The default number of iterations accomplished by this function equals 100 and the value of the smoothing parameter ρ is logarithmically changing with every iteration from the high one (10^3) to low (10^{-6}) at the end. High values of ρ in the first iterations promote minimizing of the derivative. As the procedure progresses and lower values of ρ are used, fidelity to the original data is gaining more importance.

The obtained vector $\hat{\mathbf{x}}$ can then be written to the resulting output matrix $\hat{\mathbf{X}}$: $\hat{x}_{i,j} = \hat{\mathbf{x}}_{j+m(i-1)}$, $i = 1, 2, \dots, n$; $j = 1, 2, \dots, m$.

Appendix B

Independent Component Analysis (ICA) for noise reduction

Independent Component Analysis (ICA) is one of the methods used to find a solution of the blind source separation (BSS) problem. The method was originally named as such and defined mathematically in [39], [52], and [17], and assumes estimation of the independent sources (e.g. voices of different people speaking in a room, noises, music etc.) after they were linearly mixed by unknown matrix \mathbf{A} . The example of this kind of signal mixing is an audio recording of various, multiple sounds performed by an array of microphones located in different locations. Each of the microphones records a similar mixture of sounds, however each of the source sound contributes to this mixture with a different coefficient. Similarly, for a multichannel bioelectric signal, each of the channels is a linear combination of the different electric source signals (e.g. electric activity of the brain, heart or muscles). Such a multichannel signal can be represented according to the equation

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) + \eta(k) \quad (\text{B.1})$$

where $\mathbf{x}(k)$ represents the measured signal vector, $\mathbf{s}(k)$ is the vector which contains the source signals, \mathbf{A} is the previously mentioned unknown mixing matrix, and $\eta(k)$ is the noise vector. Separation of the source signals is practical provided that these signals are statistically independent (at most one of them can have a Gaussian distribution). Having the mixing matrix $\hat{\mathbf{A}}$ and the separating one $\hat{\mathbf{A}}^{-1}$ estimated, the independent source signals can be estimated using

$$\mathbf{y}(k) = \hat{\mathbf{s}}(k) = \hat{\mathbf{A}}^{-1}\mathbf{x}(k) \quad (\text{B.2})$$

In terms of bioelectric signals denosing, the analysis of the ICA output allows to identify the source signals which contribute most to the presence of artifacts in the signal trace. In order to neutralize the noisy source signals and remove their contribution to the measured signal vector $\mathbf{x}(k)$, the noisy source signals in the source signal vector are zeroed (this operation is equivalent to zeroing the corresponding column of the separating matrix $\hat{\mathbf{A}}^{-1}$). Reconstruction of the measured signal vector $\mathbf{x}'(k)$ on the basis of the modified source signal vector $\mathbf{y}'(k)$ (with noisy source

signals zeroed) enables removal of the unwanted signal components and can be considered as a noise reduction operation.

$$\mathbf{x}'(k) = \hat{\mathbf{A}}\mathbf{y}'(k) \quad (\text{B.3})$$

In the experiments performed as part of this thesis, the JADE algorithm (*jader.m*) was used. JADE exploits the second and the fourth moments: covariances and cumulants in order to determine the separating and mixing matrices [14]. Other ICA implementations involve: Second order blind identification (SOBI) which is based on the second order statistics, in this algorithm the calculated covariance matrices undergo diagonalization [7], FastICA [46], Infomax which estimates the independent components by minimizing mutual information between the sources [6], algorithm for multiple unknown signals extraction (AMUSE) [96].

Appendix C

Empirical Mode Decomposition for artifacts rejection

Empirical mode decomposition (EMD) allows to decompose a signal $y(k)$ into more simple elements called as intrinsic mode functions (IMFs) $d_n(k)$ and residuals $r(k)$ according to the equation

$$y(k) = \sum_{n=1}^N d_n(k) + r(k) \quad (\text{C.1})$$

The method was introduced in [45] and further explained and developed in [43], [44], and [83]. EMD is fully "data driven" algorithm which does not require any initial information about the analyzed signal, it is adjusted to its local time and frequency properties. Therefore the method is suitable also for analysis of the challenging nonlinear or nonstationary signals [70].

The algorithm which enables a decomposition of the signal $y(k)$ into IMFs consist of the following steps [81], [73]:

1. Search for the locations of the extrema in $y(k)$
2. Estimation of the lower and upper "envelopes" ($e_{min}(k)$ and $e_{max}(k)$, respectively) by interpolating (using the cubic spline interpolation) between the minima and maxima found in step 1.
3. Calculation of the local mean $m(k)$ using the signal envelopes determined in step 2 as follows $m(k) = [e_{min}(k) + e_{max}(k)]/2$.
4. Subtraction of the mean $m(k)$ from the signal as follows $s(k) = y(k) - m(k)$, where $s(k)$ is called as a single oscillatory mode. If at this point $s(k)$ satisfies the stopping criteria it is saved as a new IMF $d_n(k)$. Otherwise, the procedure is repeated from the step 1 for $y(k) = s(k)$. It must be noted the IMFs are any functions which have the same number of (or different at most by one) zero crossings and extrema.

Once the $d_1(k)$ is determined, the residuals $r(k)$ are calculated as $r(k) = y(k) - d_1(k)$ and the whole procedure (beginning in step 1 is repeated for $y(k) = r(k)$.

Applying such an iterative procedure, the subsequent IMFs are estimated: $d_1(k)$, $d_2(k) \dots d_n(k)$.

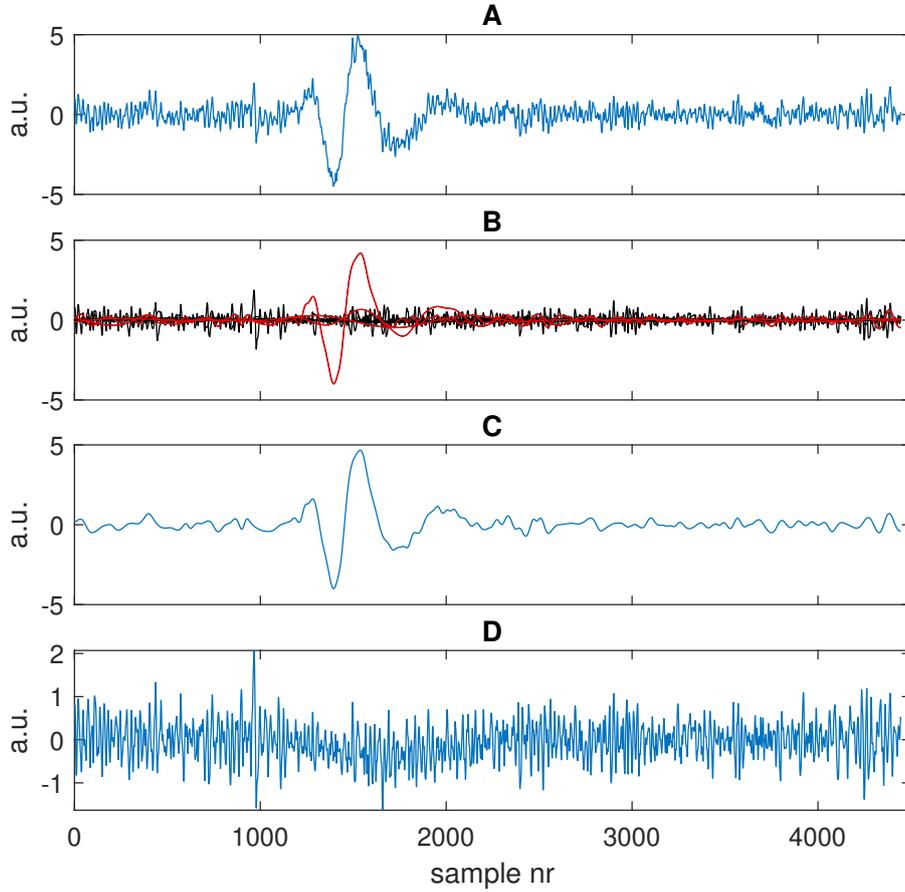


FIGURE C.1: Plot A: The exemplary noisy source signal, plot B: the outcome of its decomposition using EMD into Intrinsic mode functions (the noisy IMFs are plotted in red), plot C: the reconstructed noisy source signal based on the noisy IMFs only, plot D: the reconstructed source signal which is a sum of the unnoisy IMFs only.

Because in this thesis EMD is applied to the noisy independent source signals in order to separate the noise (super-Gaussian artifacts) from the desired signal of neural activity, the initial signal $\mathbf{y}(k)$ is indeed one of the noisy source signals. Two algorithms of EMD were implemented: first EMD1 reconstructs the source signal using the noisy IMFs only, second EMD2 reconstructs this source signal using the proper IMFs only:

The algorithm EMD1:

For every i^{th} noisy signal source signal stored in $\mathbf{y}(k)$ do

- (a) Apply the EMD procedure to calculate N intrinsic mode functions: $\mathbf{d}(k) = [d_1(k), d_2(k), \dots, d_N(k)]$

- (b) Calculate the kurtosis of all intrinsic mode functions

$$q_n = \text{kurtosis}\{d_n(k)\}, n = 1, \dots, N$$

- (c) Find the permutation function $\xi : \{1, 2, \dots, N\} \rightarrow \{1, 2, \dots, N\}$ sorting the calculated q_n values in descending order $q_{\xi(1)} \geq q_{\xi(2)} \geq \dots \geq q_{\xi(N)}$.
- (d) Find the number \tilde{N} of the intrinsic mode functions whose kurtosis is greater than the assumed threshold thr and remember \tilde{N} intrinsic mode functions of the highest kurtosis in vector $\tilde{\mathbf{d}}(k)$:

$$\tilde{d}_n(k) = d_{\xi(n)}(k), n = 1, 2, \dots, \tilde{N}$$

- (e) Reconstruct a given i^{th} noisy source signal in $\mathbf{y}(k)$ by summing all the intrinsic mode functions stored in $\tilde{\mathbf{d}}(k)$:

$$y'_i(k) = \sum_{n=1}^{\tilde{N}} \tilde{d}_n(k)$$

The algorithm EMD2:

The steps (a) and (b) of EMD1 are exactly the same for EMD2. The remaining part of the algorithm is as follows:

- c) Find the permutation function $\xi : \{1, 2, \dots, N\} \rightarrow \{1, 2, \dots, N\}$ sorting the calculated q_n values in ascending order $q_{\xi(1)} \leq q_{\xi(2)} \leq \dots \leq q_{\xi(N)}$.
- d) Find the number \tilde{N} of the intrinsic mode functions whose kurtosis is lower or equal than the assumed threshold thr and remember \tilde{N} intrinsic mode functions of the lowest kurtosis in vector $\tilde{\mathbf{d}}(k)$:

$$\tilde{d}_n(k) = d_{\xi(n)}(k), n = 1, 2, \dots, \tilde{N}$$

- e) Reconstruct a given i^{th} noisy source signal in $\mathbf{y}(k)$ by summing all the intrinsic mode functions stored in $\tilde{\mathbf{d}}(k)$:

$$y'_i(k) = \sum_{n=1}^{\tilde{N}} \tilde{d}_n(k)$$

Visual example of EMD operation on the exemplary noisy source signal is illustrated in Fig. C.1. The source signal (plot A) is decomposed into IMFs (plot B) and then reconstructed using only the IMFs classified as noisy (based on the kurtosis criterion), plotted in red. The outcome of such a reconstruction is given in plot C, the modified source signal has significantly lower amplitude in range beyond the artifact. In turn, the modified source signal calculated by summing up only the unnoisy

IMFs, is presented in plot D. Details of the IMFs, plotted in the separate subplots,

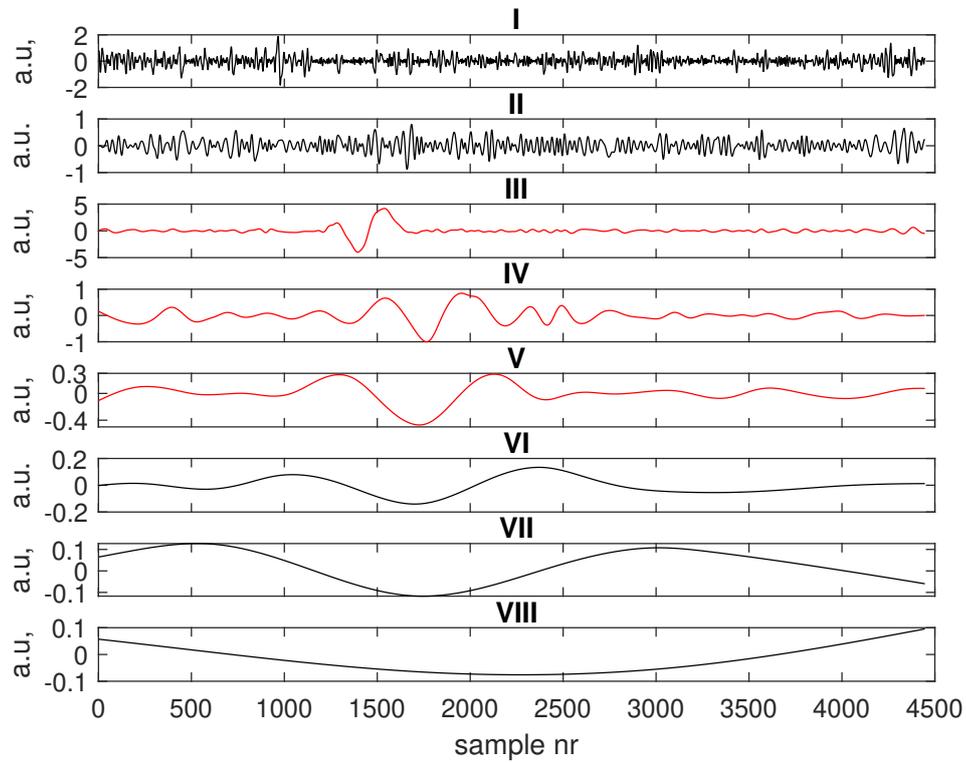


FIGURE C.2: Details of the EMD outcome. IMFs are plotted on separate subplots, the noisy IMFs are plotted in red.

are shown better in Fig. C.2. In this example 3 IMFs were detected as noisy ones.

Bibliography

- [1] F. Akram, S. M. Han, and T.-S. Kim. “An efficient word typing P300-BCI system using a modified T9 interface and random forest classifier”. In: *Computers in Biology and Medicine* 56 (2015), pp. 30–36. ISSN: 0010-4825. DOI: <https://doi.org/10.1016/j.compbiomed.2014.10.021>.
- [2] P. J. Allen et al. “Identification of EEG Events in the MR Scanner: The Problem of Pulse Artifact and a Method for Its Subtraction”. In: *NeuroImage* 8.3 (1998), pp. 229–239.
- [3] M. Arvaneh, I. H. Robertson, and T. E. Ward. “P300-Based Brain-Computer Interface for Improving Attention”. In: *Frontiers in Human Neuroscience* 12 (2019).
- [4] Q. Barthélemy et al. “Online denoising of eye-blinks in electroencephalography”. In: *Neurophysiologie Clinique* 47.5-6 (2017), pp. 371–391.
- [5] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. “Eigenfaces vs. Fisherfaces: recognition using class specific linear projection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19.7 (1997), pp. 711–720. DOI: [10.1109/34.598228](https://doi.org/10.1109/34.598228).
- [6] A. J. Bell and T. J. Sejnowski. “An information-maximization approach to blind separation and blind deconvolution”. In: *Neural computation* 7.6 (1995), pp. 1129–1159.
- [7] A. Belouchrani et al. “A blind source separation technique using second-order statistics”. In: *IEEE Transactions on Signal Processing* 45.2 (1997), pp. 434–444. DOI: [10.1109/78.554307](https://doi.org/10.1109/78.554307).
- [8] P. Berg and M. Scherg. “Dipole models of eye movements and blinks”. In: *Electroencephalography and Clinical Neurophysiology* 79.1 (1991), pp. 36–44.
- [9] B. Blankertz. “BCI Competition 2003 results (web page)”. In: (2020). URL: <http://www.bbci.de/competition/iii/results>.
- [10] J. W. Britton. “Adult EEG”. In: *Clinical Neurophysiology* (2021), p. 91.
- [11] R. Bro, E. Acar, and Tamara G. Kolda. “Resolving the sign ambiguity in the singular value decomposition”. In: *Journal of Chemometrics* 22.2 (2008), pp. 135–140.
- [12] M. J. Buckley. “Fast computation of a discretized thin-plate smoothing spline for image data”. In: *Biometrika* 81.2 (1994), pp. 247–258.

- [13] J. F. Cardoso. "Multidimensional independent component analysis". In: *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing 4.0* (2006), pp. 1941–1944.
- [14] J.-F. Cardoso and A. Souloumiac. "Blind beamforming for non-Gaussian signals". In: *IEE proceedings F (radar and signal processing)*. Vol. 140. 6. IET. 1993, pp. 362–370.
- [15] Soekadar S. R. Ushiba J. Millán J. Liu M. Birbaumer N. Cervera M. A. and G. Garipelli. "Brain-computer interfaces for post-stroke motor rehabilitation: a meta-analysis". In: *Annals of clinical and translational neurology* 5.5 (2018), 651–663.
- [16] W. De Clercq et al. "Canonical Correlation Analysis Applied to Remove Muscle Artifacts From the Electroencephalogram". In: *IEEE Transactions on Biomedical Engineering* 53.12 (2006), pp. 2583–2587. DOI: [10.1109/TBME.2006.879459](https://doi.org/10.1109/TBME.2006.879459).
- [17] P. Comon, Ch. Jutten, and J. Herault. "Blind separation of sources, Part II: Problems statement". In: *Signal processing* 24.1 (1991), pp. 11–20.
- [18] D. Coyle, G. Prasad, and T. M. McGinnity. "Extracting features for a brain-computer interface by self-organising fuzzy neural network-based time series prediction". In: *Annual International Conference of the IEEE Engineering in Medicine and Biology Society* 2 (2004), pp. 4371–4374.
- [19] M. Crespo-Garcia, M. Atienza, and J. L. Cantero. "Muscle Artifact Removal from Human Sleep EEG by Using Independent Component Analysis". In: *Annals of Biomedical Engineering* 36 (2008), pp. 467–475.
- [20] R.J. Croft and R.J. Barry. "Removal of ocular artifact from the EEG: a review". In: *Clinical Neurophysiology* 30.1 (2000), pp. 5–19.
- [21] J. Delgado. *Physical Control of the Mind: Toward a Psychocivilized Society*. New York: NY: Harper and Row, 1969.
- [22] S. Dutta et al. "Development of a BCI-Based Application Using EEG to Assess Attentional Control". In: *Proceedings of the Global AI Congress 2019* 1112 (2020), pp. 659–670.
- [23] A. S. Eltrass and N. H. Ghanem. "A new automated multi-stage system of non-local means and multi-kernel adaptive filtering techniques for EEG noise and artifacts suppression". In: *Journal of Neural Engineering* 18 (2021).
- [24] J. Qu Q. Xie Q. Lin X. Ni Y. Chen J. Pan S. Laureys R. Yu F. Wang Y. He and Y. Li. "Enhancing clinical communication assessments using an audiovisual BCI for patients with disorders of consciousness". In: *Journal of Neural Engineering* 14.4 (2017), p. 046024.
- [25] L. A. Farwell and E. Donchin. "Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials". In: *Electroencephalography and Clinical Neurophysiology* 70.6 (1998), pp. 510–523.

- [26] Jerome H. Friedman. "Regularized Discriminant Analysis". In: *Journal of the American Statistical Association* 84.405 (1989), pp. 165–175.
- [27] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. New York: Academic Press, 1990.
- [28] P. Gajbhiye, R. K. Tripathy, and A. Bhattacharyya R. B. Pachori. "Novel Approaches for the Removal of Motion Artifact From EEG Recordings". In: *IEEE Sensors Journal* 19.22 (2019), pp. 10600–10608. DOI: [10.1109/JSEN.2019.2931727](https://doi.org/10.1109/JSEN.2019.2931727).
- [29] Z. Gao et al. "EEG-Based Spatio-Temporal Convolutional Neural Network for Driver Fatigue Evaluation". In: *IEEE Transactions on Neural Networks and Learning Systems* 30.9 (2019), pp. 2755–2763.
- [30] D. Garcia. "Robust smoothing of gridded data in one and higher dimensions with missing values". In: *Computational Statistics and Data Analysis* 54.4 (2010), pp. 1167–1178.
- [31] G. B. Giannakis and M. K. Tsatsanis. "Signal detection and classification using matched filtering and higher order statistics". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 38.7 (1990), pp. 1284–1296.
- [32] J. Giraldo-Guzmán et al. "Application of Spatio-Temporal Filtering for Atrial Activity Waveforms Enhancement". In: *Proceedings of the 2019 2nd International Conference on Sensors, Signal and Image Processing* (2019), pp. 67–72.
- [33] I. I. Goncharova et al. "EMG contamination of EEG: spectral and topographical characteristics". In: *Clinical Neurophysiology* 114.9 (2003), pp. 1580–1593.
- [34] C. Guger et al. "How many people are able to control a P300-based brain-computer interface (BCI)?" In: *Neuroscience Letters* 462 (2009), pp. 94–98.
- [35] S. Hadiyoso and I. Wijayanto. "Noise Removal in Mild Cognitive Impairment EEG Recording using Empirical Mode Decomposition". In: *2019 IEEE International Conference on Signals and Systems (ICSigSys)*. 2019, pp. 74–78. DOI: [10.1109/ICSIGSYS.2019.8811086](https://doi.org/10.1109/ICSIGSYS.2019.8811086).
- [36] T. Hastie, A. Buja, and R. Tibshirani. "Penalized Discriminant Analysis". In: *The Annals of Statistics* 23.1 (1995), pp. 73–102.
- [37] G. He, X. Dong, and Min Qi. "From the perspective of material science: a review of flexible electrodes for brain-computer interface". In: *Materials Research Express* 7.10 (2020).
- [38] P. He, G. Wilson, and C. Russell. "Removal of ocular artifacts from electroencephalogram by adaptive filtering". In: *Medical and Biological Engineering and Computing* 42.3 (2004), pp. 407–412.
- [39] J. Herault and Ch. Jutten. "Space or time adaptive signal processing by neural network models". In: *AIP conference proceedings*. Vol. 151. 1. American Institute of Physics. 1986, pp. 206–211.

- [40] U. Hoffmann et al. "An efficient P300-based brain computer interface for disabled subjects". In: *Journal of Neuroscience* 167.1 (2008), pp. 115–125.
- [41] M.R. Hohmann et al. "Chapter 8 - A cognitive brain–computer interface for patients with amyotrophic lateral sclerosis". In: *Brain-Computer Interfaces: Lab Experiments to Real-World Applications*. Ed. by Damien Coyle. Vol. 228. Progress in Brain Research. Elsevier, 2016, pp. 221–239. DOI: <https://doi.org/10.1016/bs.pbr.2016.04.022>.
- [42] J. Hou et al. "An improved artifacts removal method for high dimensional EEG". In: *Journal of Neuroscience Methods* 268 (2016), pp. 31–42. DOI: <https://doi.org/10.1016/j.jneumeth.2016.05.003>.
- [43] N. E. Huang, Z. Shen, and S. R. Long. "A new view of nonlinear water waves: the Hilbert spectrum". In: *Annual review of fluid mechanics* 31 (1999), pp. 417–457.
- [44] N. E. Huang et al. "A confidence limit for the Empirical Mode Decomposition and Hilbert Spectral Analysis". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 459 (2003), pp. 2317–2345.
- [45] N. E. Huang et al. "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 454.1971 (1998), pp. 903–995.
- [46] Aapo Hyvarinen. "Fast and robust fixed-point algorithms for independent component analysis". In: *IEEE transactions on Neural Networks* 10.3 (1999), pp. 626–634.
- [47] J. Iriarte et al. "Independent Component Analysis as a Tool to Eliminate Artifacts in EEG: A Quantitative Study." In: *Journal of Clinical Neurophysiology* 20.4 (2003), pp. 249–257.
- [48] M. S. Islam et al. "EEG mobility artifact removal for ambulatory epileptic seizure prediction applications". In: *Biomedical Signal Processing and Control* 55 (2020), p. 101638. ISSN: 1746-8094. DOI: <https://doi.org/10.1016/j.bspc.2019.101638>.
- [49] X. Jiang, G-B. Bian, and Z. Tian. "Removal of artifacts from EEG signals: a review". In: *Sensors* 19.5 (2019), p. 987.
- [50] Z. Jin et al. "EEG classification using sparse Bayesian extreme learning machine for brain–computer interface". In: *Neural Computing and Applications* 32 (2020), 6601–6609.
- [51] T-P. Jung et al. "Analysis and visualization of single-trial event-related potentials". In: *Human Brain Mapping* 14.3 (2001), 166–185. DOI: [10.1002/hbm.1050](https://doi.org/10.1002/hbm.1050).

- [52] Ch. Jutten and J. Herault. "Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture". In: *Signal processing* 24.1 (1991), pp. 1–10.
- [53] J. Vidal. "Toward direct brain-computer communication". In: *Annual Review of Biophysics and Bioengineering* 2 (1973), pp. 157–180.
- [54] S. M. Kay. "Fundamentals of Statistical Signal Processing. Detection Theory". In: *Prentice Hall Signal Processing Series 2* (1998).
- [55] D. W. Klass. "The Continuing Challenge of Artifacts in the EEG". In: *American Journal of EEG Technology* 35.4 (1995), pp. 239–269.
- [56] J. E. Kline et al. "Isolating gait-related movement artifacts in electroencephalography during human walking". In: *Journal of Neural Engineering* 12.4 (2015), p. 046022. DOI: [10.1088/1741-2560/12/4/046022](https://doi.org/10.1088/1741-2560/12/4/046022).
- [57] V. Kohli et al. "A review on Virtual Reality and Augmented Reality use-cases of Brain Computer Interface based applications for smart cities". In: *Microprocessors and Microsystems* 88 (2022).
- [58] W. Kong et al. "Weighted extreme learning machine for P300 detection with application to brain computer interface". In: *Journal of Ambient Intelligence and Humanized Computing* (2018).
- [59] W. Kong et al. "Weighted extreme learning machine for P300 detection with application to brain computer interface". In: *Journal of Ambient Intelligence and Humanized Computing* (2018), pp. 1–11.
- [60] M. Kotas, M. Piela, and S. H. Contreras-Ortiz. "Modified Spatio-Temporal Matched Filtering for Brain Responses Classification". In: *IEEE Transactions on Human-Machine Systems* 52.4 (2022), pp. 677–686.
- [61] M. Kotas et al. "Application of spatio-temporal filtering to fetal electrocardiogram enhancement". In: *Computer Methods and Programs in Biomedicine* 104 (Jan. 2010), pp. 1–9.
- [62] D. J. Krusienski et al. "A comparison of classification techniques for the P300 Speller". In: *Journal of Neural Engineering* 3.4 (2006), pp. 299–305.
- [63] W. J. Krzanowski et al. "Discriminant analysis with singular covariance matrices: methods and applications to spectroscopic data". In: *Applied Statistics* 44 (1995), pp. 101–115.
- [64] S. Kundu and S. Ari. "P300 detection with brain-computer interface application using PCA and ensemble of weighted SVMs". In: *IETE Journal of Research* 64.3 (2018), pp. 406–416.
- [65] O. Y. Kwon et al. "Subject-independent brain-computer interfaces based on deep convolutional neural networks". In: *IEEE Transactions on Neural Networks and Learning Systems* 31.10 (2020), pp. 3839–3852.

- [66] Y-R. Lee and H-N. Kim. "A data partitioning method for increasing ensemble diversity of an eSVM-based P300 speller". In: *Biomedical Signal Processing and Control* 39 (2018), pp. 53–63.
- [67] P. LeVan, E. Urrestarazu, and J. Gotman. "A system for automatic artifact removal in ictal scalp EEG based on independent component analysis and Bayesian classification". In: *Clinical Neurophysiology* 117.4 (2006), pp. 912–927. DOI: <https://doi.org/10.1016/j.clinph.2005.12.013>.
- [68] M. Li et al. "The MindGomoku: An Online P300 BCI Game Based on Bayesian Deep Learning". In: *Sensors* 21.5 (2021).
- [69] S. Li et al. "Feature selection method based on Menger curvature and LDA theory for a P300 brain-computer interface". In: *Journal of Neural Engineering* 18.6 (2022), p. 066050.
- [70] L. Lin and J. Hongbing. "Signal feature extraction based on an improved EMD method". In: *Measurement* 42.5 (2009), pp. 796–803. ISSN: 0263-2241. DOI: <https://doi.org/10.1016/j.measurement.2009.01.001>.
- [71] Ronghua Ma et al. "Capsule Network for ERP Detection in Brain-Computer Interface". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 29 (2021), pp. 718–730.
- [72] J. G. Madhale, J. Lerga, and I. Štajduhar. "Adaptive filtering and analysis of EEG signals in the time-frequency domain based on the local entropy". In: *EURASIP Journal on Advances in Signal Processing volume 7* (2020).
- [73] T. Tanaka D. P. Mandic. "Complex empirical mode decomposition". In: *IEEE Signal Processing Letters* 14.2 (2007), pp. 101–104.
- [74] D. Mantini et al. "Complete artifact removal for EEG recorded during continuous fMRI using independent component analysis". In: *Neuroimage* 34.2 (2007), pp. 598–607.
- [75] C. G. Y. Ngan, R. M. I. Kapsa, and P. F.M. Choong. "Strategies for neural control of prosthetic limbs: from electrode interfacing to 3D printing". In: *Materials* 12.12 (2019).
- [76] T. Nguyen et al. "Classification of Multi-Class BCI Data by Common Spatial Pattern and Fuzzy System". In: *IEEE Access* 6 (2018), pp. 27873–27884. DOI: [10.1109/ACCESS.2018.2841051](https://doi.org/10.1109/ACCESS.2018.2841051).
- [77] M. Jeon P. Howland and H. Park. "Structure preserving dimension reduction for clustered text data based on the generalized singular value decomposition". In: *SIAM Journal on Matrix Analysis and Applications* 25.1 (2003), pp. 165–179.
- [78] K. Puthuveetil and D. J. Krusienski. "Brain-controlled Assistive Robotics and Prosthetics". In: (2021).

- [79] Y. H. Hu Q. Xue and W. J. Tompkins. "Neural-network-based adaptive matched filtering for QRS detection". In: *IEEE Transactions on Biomedical Engineering* 39.4 (1992), pp. 317–329.
- [80] A. Rakotomamonjy and V. Guigue. "BCI Competition III: Dataset II - Ensemble of SVMs for BCI P300 Speller". In: *IEEE Transactions on Biomedical Engineering* 55.3 (2008), pp. 1147–1154.
- [81] N. Rehman and D. P. Mandic. "Multivariate empirical mode decomposition". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 466.2117 (2010), pp. 1291–1302.
- [82] G. Rilling, P. Flandrin, and P. Goncalves. "On empirical mode decomposition and its algorithms". In: *In IEEE-EURASIP workshop on nonlinear signal and image processing by the kurtosis* 3.3 (2003), pp. 8–11.
- [83] G. Rilling, P. Flandrin, P. Goncalves, et al. "On empirical mode decomposition and its algorithms". In: *IEEE-EURASIP workshop on nonlinear signal and image processing*. Vol. 3. 3. IEEEER Grado. 2003, pp. 8–11.
- [84] B. Rivet et al. "xDAWN algorithm to enhance evoked potentials: application to brain-computer interface". In: *IEEE Transactions on Biomedical Engineering* 56.8 (2009), pp. 2035–2043.
- [85] H. Serby, E. Yom-Tov, and G. F. Inbar. "An improved P300-based brain-computer interface". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 13 (2005), pp. 89–98.
- [86] A. J. Shackman et al. "Electromyogenic Artifacts and Electroencephalographic Inferences". In: *Brain Topography* 22 (2009), pp. 7–12.
- [87] M. Shim et al. "Machine-learning-based diagnosis of schizophrenia using combined sensor-level and source-level EEG features". In: *Schizophrenia Research* 176.2 (2016), pp. 314–319.
- [88] M. Simões et al. "BCIAUT-P300: A multi-session and multi-subject benchmark dataset on autism for P300-based brain-computer-interfaces". In: *Frontiers in Neuroscience* 14 (2020), p. 568104.
- [89] B. Somers, T. Francart, and A. Bertrand. "A generic EEG artifact removal algorithm based on the multi-channel Wiener filter". In: *Journal of Neural Engineering* 15 (2018).
- [90] W. Sun et al. "A novel end-to-end 1D-ResCNN model to remove artifact from EEG signals". In: *Neurocomputing* 404 (2020), pp. 108–121. ISSN: 0925-2312.
- [91] A. Sundaresan et al. "Evaluating deep learning EEG-based mental stress classification in adolescents with autism for breathing entrainment BCI". In: *Brain Informatics* 8.13 (2021), pp. 659–670.

- [92] A. F. Szymanska et al. "Accurate detection of low signal-to-noise ratio neuronal calcium transient waves using a matched filter". In: *Journal of neuroscience methods* 259 (2016), pp. 1–12.
- [93] T.J. Sejnowski T-W. Lee. "Independent component analysis for mixed sub-Gaussian and super-Gaussian sources". In: *4th Joint Symposium on Neural Computation Proceedings*. Citeseer. 1997, pp. 132–139.
- [94] A. Tharwat et al. "Linear discriminant analysis: A detailed tutorial". In: *AI Communications* 30.2 (2017), pp. 169–190.
- [95] M. Thulasidas, C. Guan, and J. Wu. "Robust classification of EEG signal for brain-computer interfaced". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 14.1 (2006), pp. 24–29.
- [96] L. Tong et al. "Indeterminacy and identifiability of blind identification". In: *IEEE Transactions on circuits and systems* 38.5 (1991), pp. 499–509.
- [97] A. Turnip. "Comparison of ICA-Based JADE and SOBI Methods EOG Artifacts Removal". In: *Journal of Medical and Bioengineering* 4.6 (2015).
- [98] A. T. Tzallas et al. "EEG Classification and Short-Term Epilepsy Prognosis Using Brain Computer Interface Software". In: *2017 IEEE 30th International Symposium on Computer-Based Medical Systems (CBMS)* 30.9 (2017), pp. 349–353.
- [99] L. Vareka. "Evaluation of convolutional neural networks using a large multi-subject P300 dataset". In: *Biomedical Signal Processing and Control* 58 (2020).
- [100] G. A. M. Vasiljevic and L. C. de Miranda. "Brain–Computer Interface Games Based on Consumer-Grade EEG Devices: A Systematic Literature Review". In: *International Journal of Human–Computer Interaction* 36.2 (2020), pp. 105–142.
- [101] C. Verbaarschot et al. "A visual brain-computer interface as communication aid for patients with amyotrophic lateral sclerosis". In: *Clinical Neurophysiology* 132.10 (2021), pp. 2404–2415. ISSN: 1388-2457. DOI: <https://doi.org/10.1016/j.clinph.2021.07.012>.
- [102] D. Vos et al. "Removal of muscle artifacts from EEG recordings of spoken language production". In: *Neuroinformatics* 8 (2010), pp. 135–150.
- [103] G. L. Wallstrom et al. "Automatic correction of ocular artifacts in the EEG: a comparison of regression-based and component-based methods". In: *International Journal of Psychophysiology* 53.2 (2004), pp. 105–119.
- [104] G. Wang et al. "The Removal of EOG Artifacts From EEG Signals Using Independent Component Analysis and Multivariate Empirical Mode Decomposition". In: *IEEE Journal of Biomedical and Health Informatics* 20.5 (2016), pp. 1301–1308.

- [105] X. Wang, A. Hassanien, and M. G. Amin. "Sparse transmit array design for dual-function radar communications by antenna selection". In: *Digital Signal Processing* 83 (2018), pp. 223–234.
- [106] C. D. Woody. "Characterization of an adaptive filter for the analysis of variable latency neuroelectric signals". In: *Medical and biological engineering* 5.6 (1967), pp. 539–554.
- [107] T. Xiong and V. Cherkassky. "A combined SVM and LDA approach for classification". In: *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005*. Vol. 3. 2005, pp. 1455–1459. DOI: [10.1109/IJCNN.2005.1556089](https://doi.org/10.1109/IJCNN.2005.1556089).
- [108] B. Yang et al. "Automatic ocular artifacts removal in EEG using deep learning". In: *Biomedical Signal Processing and Control* 43 (2018), pp. 148–158.
- [109] J. Yu et al. "Embedding decomposition for artifacts removal in EEG signals". In: *Journal of Neural Engineering* 19.2 (2022).
- [110] H. Zhang et al. "A Novel Convolutional Neural Network Model to Remove Muscle Artifacts from EEG". In: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, pp. 1265–1269. DOI: [10.1109/ICASSP39728.2021.9414228](https://doi.org/10.1109/ICASSP39728.2021.9414228).