4-2016

Changes in the EEG Spectrum of a Child with Severe Disabilities in Response to Power Mobility Training

Nadina Zweifel
Grand Valley State University

Changes in the EEG Spectrum of a Child with Severe Disabilities in Response to Power Mobility Training

Nadina Zweifel

A Thesis Submitted to the Graduate Faculty of

GRAND VALLEY STATE UNIVERSITY

In

Partial Fulfillment of the Requirements

For the Degree of

Master of Science in Engineering

Padnos College of Engineering and Computing

April 2016

Acknowledgment

I would first like to thank my thesis advisor Dr. Samhita Rhodes of the School of Engineering in the Seymour and Esther Padnos College of Engineering and Computing of Grand Valley State University. Throughout my studies, she has been an amazing advisor and supporter, personally and academically. She consistently allowed this paper to be my own work, but helped me whenever I needed advice or guidance.

I would also like to thank the committee members and experts who were involved in this research project: Dr. Lisa K. Kenyon PT, DPT, PhD, PCS, Dr. John Farris, Dr. Paul Stephenson, and Dr. David Zeitler. Without their passionate participation and input, this study could not have been successfully conducted.

Finally, I would like to thank you my family and friends for providing me with support and continuous encouragement throughout my studies and through the process of researching and writing this thesis. This accomplishment would not have been possible without them.

Nadina Zweifel

Abstract

Literature suggests that self-generated locomotion in infancy and early childhood enhances the development of various cognitive processes such as spatial awareness, social interaction, language development and differential attentiveness. Thus, having access to a power mobility device may play a crucial role for the overall development, mental health, and quality of life of children with multiple, severe disabilities who have limited motor control. This study investigates the feasibility of using electroencephalography (EEG) as an objective measure to detect changes in brain activity in a child due to power mobility training. EEG data was collected with a modified wireless neuroheadset using a single-subject A-B-A-B design consisting of two baseline phases (A) and two intervention phases (B). One trial consisted of three different activities during baseline phase; resting condition at the beginning (Resting 1) and at the end (Resting 2) of the trial, interaction with adults, and passive mobility. The intervention phase included a forth activity, the use of power mobility, while power mobility training was performed on another day within the same week of data collection. The EEG spectrum between 2.0 and 12.0 Hz was analyzed for Resting 1 and Resting 2 condition in each phase. We found significant increase of theta power and decrease in alpha power during all three phases following the first baseline. In respect of previous findings, these observations may be related to an increase in alertness and/or anticipation. Analysis of the percentage change from Resting 1 to Resting 2 condition revealed decrease in theta and increasing alpha power during the first intervention phase, which could be associated with increasing cognitive capacity immediately after the use of power mobility. Overall, no significant difference between baseline phase and intervention phase was observed. Thus, whether the observed changes may have been influenced or enhanced by power mobility training remains unclear and warrants further investigation.

Table of Contents

List of Tables

List of Figures

1 INTRODUCTION

This pilot study was a collaboration between the Department of Physical Therapy (Dr. Lisa Kenyon), the Department of Psychology (Dr. Naomi Aldrich), the School of Engineering (Dr. John Farris and Dr. Samhita Rhodes), and the Statistics Department (Dr. Paul Stephenson) of Grand Valley State University (GVSU). This group has been working since 2008 with children who have multiple, severe disabilities; embracing and encouraging the use of power mobility devices in order to improve the mobility, independence, quality of life, and general well-being of children who have multiple, severe disabilities. To provide these children the opportunity to explore and practice power mobility, the group designed and built the Power Wheelchair Trainer (PWCT).[1,2] The PWCT is a motorized platform on which any manual wheelchair can be mounted temporarily such that individuals can practice using power mobility and explore their environment under safe and controlled conditions.

Preliminary data indicate that the self-generated locomotion effects not only the children's physical but also cognitive abilities positively.[3–5] Intensive power mobility training may result in positive changes in quality of life and qualitative improvements in areas related to psychosocial issues including behavior, temperament, motivation, and environmental awareness.[1–3,6,7] Due to the inability of the children to demonstrate motor and verbal responses, parental report measures are primarily used to assess the child's improvements during the training.[3,8,7] Using electroencephalography (EEG), this study sought to develop a method to support these observations with an objective report measure.

1.1 PURPOSE

The ability to move and explore the environment has immense impact on the development and psychology of children.[3–6,9] Self-generated locomotion in infancy and early childhood enhances the learning and maintaining of cognitive processes such as attention to distal events, spatial awareness, social interaction, language development and differential attentiveness.[5] The lack of independent and autonomous mobility is, therefore, a clear disadvantage for the overall development of children with multiple severe disabilities. Having access to a power mobility device may remedy this deficiency and provide these children with the opportunity to explore their environment voluntarily, and thus play a crucial role for their overall development, mental health and quality of life.[2,3,7] Typically developing children start to use self-generated locomotion between 8-16 months of age.[10] Similarly, appropriate power mobility training allows the introduction of power mobility to infants as young as 7 months of age successfully.[3] Although the benefit of power mobility for children with severe disabilities has been suggested,[2,3] the findings are primarily based on parent report measures. The present study attempted to overcome this problem by the employment of EEG.

1.2 SCOPE

This study attempted to identify a change in brain activity when children with multiple, severe disabilities are using power mobility. This change was assumed to represent a change of the children's emotional, and cognitive state due to the participation in power mobility trainings. Whether or not and how these changes can be detected is a question that has not been addressed in the literature. Therefore, we studied the feasibility of various techniques and methods to process the collected EEG data and to detect a change in brain activity. In order to address this problem we had identified the following specific aims.

1.2.1 Aim 1

Recording the brain activity of typically developing children and developing an appropriate signal processing framework for the EEG data analysis.

In order to complete this aim we collected EEG data using a modified version of the Emotiv EPOC© (Emotiv, Inc., San Francisco, CA) neuroheadset. Various techniques to remove artifacts and process the data were implemented in a MATLAB (The MathWorks, Inc., Natick, MA) program in order to find an appropriate method to identify a change in the EEG spectrum.

1.2.2 Aim 2

Investigating brain activity of one child with multiple, severe disabilities when using a PWCT.

In order to complete this aim, we conducted a single-subject study with an A-B-A-B design. The subject was a three year, 2-month old girl with spastic, quadriplegic cerebral palsy, microcephaly, and a cortical visual impairment. The study occurred over 16 week including two baseline and two intervention phases with each phase consisting of a four week period. The same EEG acquisition system as mentioned in Aim 1 was used for data collection, whereas the analysis was performed with the same MATLAB program.

Eventually, the findings of this pilot should provide insights into 1) whether the used EEG acquisition system and the applied signal processing algorithms can detect a change in the EEG spectrum of children with multiple, severe disabilities, and 2) whether power mobility training influences their brain activity in short-term or long-term.

1.3 ASSUMPTIONS

Based on previous research,[5] we believe that self-generated locomotion is an important factor in the development of typically developing children as well as in children with disabilities. This presupposes plasticity of the children's brain structure affected greatly by the experience of

locomotion during infancy and childhood. This implies that the impact of self-generated locomotion on the developing brain does not originate in the acquisition of motor control itself, but rather in the concomitant experience of exploring the environment. We assume that children with disabilities may be able to gain this experience through power mobility training, similar to the learning of crawling, walking, and running in typically developing children. Thus, power mobility training may also have an impact on the developing brain of children with severe disabilities.

1.4 HYPOTHESIS

Although parental report measures indicate a correlation between the learning of self-generated locomotion and child development, these findings lack support by physiological measurements. Since EEG has become a commonly used technique to measure electrophysiological changes in the adult and young brain, we believe that this measure is able to detect changes in brain activity related to locomotor experiences, if these changes are in fact present.

Hence, we hypothesize that intensive power mobility training has similar effect on the cognitive functions of children with multiple severe disabilities as it is observed in typically developing children when they start to explore their environment by crawling or walking. Since literature suggests that a person's EEG changes based on physical or mental activity,[11–15] mood or emotion,[15–17] we expect that the change in brain activity induced by self-generated locomotion is measurable using EEG. The change might be detected during or immediately after using the Power Wheelchair Trainer, or might manifest itself with regular practice over a prolonged period.

1.5 SIGNIFICANCE

Due to the great variety of factors that influence the developing brain, isolation of changes in brain activity originating from locomotion in typically developing children is very challenging. Hence, there is a great lack of EEG studies investigating the impact of locomotor experience on child development. In contrast, the effect of self-generated locomotion on brain activity of children with severe disabilities, who cannot move without assistance, can be investigated more easily since self-generated locomotion is not part of their daily activities. Positive results of the present study would give evidence for previous findings in development research and provide an objective report measure for the effect of power mobility training on brain activity in children with severe disabilities. More importantly, a positive outcome of our investigation would support the hypothesis that power mobility training benefits children with severe disabilities and their overall development, even if fully independent and autonomous use of power mobility cannot be achieved in the long term.

2 MANUSCRIPT

Abstract

Literature suggests that self-generated locomotion in infancy and early childhood enhances the development of various cognitive processes. Thus, having access to a power mobility device may play a crucial role for the overall development of children with multiple severe disabilities who have limited motor control. This study investigates the feasibility to use electroencephalography (EEG) as an objective measure for changes in brain activity in these children due to power mobility training. We collected EEG data with a modified wireless neuroheadset using a single-subject A-B-A-B design consisting of two baseline phases (A) and two intervention phases (B). One trial consisted of three different activities during baseline phase; resting condition at the beginning (Resting 1) and at the end (Resting 2) of the trial, interaction with adults, and passive mobility. The intervention phase included a forth activity, the use of power mobility, while power mobility training was performed within the same week of data collection. The EEG spectrum between 2.0 and 12.0 Hz was analyzed for Resting 1 and Resting 2 condition in each phase. We found significant increase of theta power and decrease in alpha power during all three phases following the first baseline. In comparison to previous findings, these observations may be related to an increase in alertness and/or anticipation. Analysis of the percentage change from Resting 1 to Resting 2 condition revealed decrease in theta and increasing alpha power during the first intervention phase, which could be associated with increasing cognitive performance immediately after the use of power mobility. Whether the observed changes may have been influenced or enhanced by power mobility training remains unclear and warrants further investigation.

2.1 INTRODUCTION

The most common cause of severe physical disability in childhood is cerebral palsy (CP),[18] of which the overall estimated prevalence in children of 3-10 years old is about 2.2 per 1000 children.[19] CP describes a range of non-progressive brain lesions with progressive syndromes of posture and motor impairment that is caused by irreversible damage to the brain, brainstem, or spinal cord, resulting from an injury in the developing central nervous system occurring within the first two years of life.[19] While 50% of CP patients are able to walk without any assistance, 25% cannot walk, and 30% have cognitive limitations.[20,21] Also neurological disorders such as seizures, sensory impairment of the arms, impairment of visual perception, and learning disabilities are common concomitants of CP.[22] Children that show severe manifestations of CP (categorized as Level V in the Gross Motor Function Classification System,[23] Communication Function Classification System,[24] and Eating and Drinking Ability Classification System[25]) are unable to walk or communicate even basic needs, and may or may not have cognitive limitations. However, the ability to move around and explore the environment has immense impact on the development and psychology of typically developing children.[3]

Self-generated locomotion in infancy and early childhood enhances the development of cognitive processes such as attention to distal events, spatial awareness, social interaction, language development and differential attentiveness.[5,4,6,9] Typically developing children start to use self-generated locomotion between 8-16 months of age.[10] The lack of independent and autonomous mobility is, therefore, a clear disadvantage for the overall development of children with multiple severe disabilities. Having access to a power mobility device may remedy this deficiency and provide these children with the opportunity to explore their environment

voluntarily, and thus play a crucial role for their overall development, mental health and quality of life.[3]

Although the benefit of power mobility for children with severe disabilities has been suggested in the limited literature on the topic,[3,2] the findings are primarily based on subjective measures. The present study attempts to overcome this problem using objective, frequency-domain measures from EEG recordings taken in the course of the experimental protocol.

This objective presents two main challenges: 1) Studies have shown that the individual alpha peak frequency (IAF), varies as a function of age, neurological diseases, memory performance, and task.[26,27] However, since the target population of the present study does not respond to verbal instructions, identifying the IAF is not easy. 2) Due to the limited number of studies that have looked at changes in EEG of children with cerebral palsy, very little is known about the characteristics of their brain activity.[28–32] General findings show differences in the EEG spectrum and in interhemispheric and intrahemispheric coherence in children with CP.[29,30,32,33] In general, children with CP showed increased delta and decreased alpha power and hypoconnectivity between left and right hemisphere compared to typically developing children.[28] Despite these significant challenges, we hypothesize that EEG spectral analysis will be able to provide objective measures of improvements in cognitive function with power mobility training in this particular population of children.

EEG signals have a fairly wide frequency spectrum, which ranges from 1 Hz to 40 Hz or higher and is usually split in five different frequency bands (rhythms); delta (1-4Hz), theta (5-7Hz), alpha (8-13Hz), beta (13-30Hz), and gamma (> 30Hz).[34,35] Lower frequencies indicate the less responsive states, whereas higher frequencies indicate increased alertness. For example, the delta rhythm is associated with sleep, while beta and gamma rhythm represent a waking state which is

associated with active thinking and attention.[36] Wave patterns vary not only as a function of state of consciousness but also according to where on the scalp they are recorded.[34] Thus, the alpha rhythm is typically observed in occipital, parietal, and posterior temporal areas, while theta oscillations occur predominantly in central-frontal areas.[34,37] In general, the amplitude of the EEG signal decreases with increasing frequency, which is also valid within the frequency band such that the lower alpha component has a higher amplitude than the higher alpha component.[34,38]

The most prominent rhythm is the alpha rhythm, best seen with eyes closed and during physical relaxation and relative mental inactivity in the occipital cortex. It has been acknowledged that the alpha rhythm reflects an idling state of primary cortical areas, which results in a decrease of alpha power when engaging in a task such as perceptual judgment or increased attentiveness.[39–42] Studies indicate involvement of alpha power in cognitive processes and memory, observing positive correlation of alpha power and awareness, which leads to the theory that non-essential processing is inhibited in order to facilitate performance of the actual task or support working memory processes.[39,43,44] In the context of cognitive and memory performance, a reciprocal relationship between alpha and the lower theta activity has been identified.[26] It was suggested that the power of the alpha rhythm is positively related to cognitive performance and brain maturity, whereas theta power is negatively related.[26,45] This theory was supported by the findings of Orekhova et al.[14] who examined the involvement of theta in cognitive and emotional processes of infants and preschool children. The observed phasic increase in theta power and simultaneous decrease in high alpha power in both age groups led to the conclusion that theta oscillations are strongly related to behavioral states with considerable attentional and emotional load and may reflect engagement of different brain networks in control and behavior.[14]

Based on these findings, a gradual increase in alpha power and simultaneous decrease in theta power would support our hypothesis that power mobility training enhances cognitive functions in children with multiple severe disabilities.

2.2 METHODS AND MATERIALS

2.2.1 Subject

The subject of the present study was a three year, 2 month-old child (female) from the Grand Rapids area. Parental permission was obtained. The subject was diagnosed with spastic quadriplegic cerebral palsy, microcephaly, cortical visual impairment, and a seizure disorder. The subject was classified as Level V according to the Gross Motor Function Classification System,[23] the Communication Function Classification System,[24] as well as the Eating and Drinking Ability Classification System[25] (feeding occurs through a gastrostomy-tube, nothing by mouth) and had significant limitations in the ability to manipulate objects even with assistance. During this study, the subject was taking Cytra K in order to prevent kidney stones, Sabril® for seizures and infantile spasms, and Omeprazole® for gastro-esophageal reflux, as well as a multi vitamin and a probiotic.

2.2.2 Power wheelchair trainer (PWCT)

The PWCT was developed as part of the Grand Valley Power Mobility Project and was designed and prototyped in the School of Engineering at Grand Valley State University in Grand Rapids, MI.[2,1] The current version of the PWCT is a motorized platform, on which any customized, manual wheelchair can be mounted and temporarily converted into a power wheelchair. This permits individuals with multiple, severe impairments to practice using power mobility and begin to explore their environment while optimally and safely positioned in their own

customized seating systems and under the supervision of a licensed physical therapist. The PWCT can be adapted to the individual trainee's abilities and power mobility can be controlled with one button, multiple buttons, or a joystick. For safety reasons, the PWCT is also equipped with a shared control that allows the supervising therapist monitoring the training to intervene with an additional joystick while the subject is using the device. The subject of the present study was using 1-3 buttons, each for a single direction (forward, left, right). Throughout this document, the use of the PWCT will be addressed as 'power mobility'.

2.2.3 Experimental Conditions

The present study investigates the impact of power mobility training on the EEG spectrum of children with multiple severe disabilities. This research problem is a typical case for a single-subject research design (SSRD), which is commonly used in developmental medicine and rehabilitation sciences.[46] SSRD attempts to ascertain if a change in the baseline target variable can be causally related to the intervention. In this study, the target variable is EEG activity at rest while the intervention is power mobility. This study represents the first, to our knowledge, that uses the SSRD protocol in combination with EEG spectral analysis.

There are several types of SSRD, of which the alternating treatment design, so called A-B-A-B design, was applied. 'A' represents the baseline phase, while 'B' represents the intervention phase. The A-B-A-B design represents a more rigorous design compared to the A-B design, but also allows the comparison of changes in the EEG spectrum in two baseline phases. The experiment was conducted once a week, while each phase lasted four weeks. The total duration of the study was 16 weeks.

Both baseline phases of the present study consisted of four experimental conditions during EEG data collection for one day per week– resting, interaction, and passive mobility, resting. During resting condition, the subject was sitting quietly in her wheelchair with minimal visual and auditory stimuli. Interaction involved close interaction with adults such as singing different songs and touching (holding hand or feet). During passive mobility, the subject sitting in the wheelchair was pushed in her own wheelchair in random directions, while playing and interacting with adults.

The baseline phase was alternated with the intervention phase. The intervention phase included two sessions on two different days each week. The first session was dedicated to power mobility training for 45 minutes, whereas the second session consisted of all the same conditions as in baseline phase, with addition of power mobility use. Passive mobility was followed by power mobility using the PWCT – resting, interaction, passive mobility, power mobility. The resting condition was repeated after passive (baseline) or power (intervention) mobility in order to capture the potential changes occurring immediately after using power mobility.

Table 1 summarizes the three conditions performed during one experimental trial. Each condition lasted 5 minutes during which EEG was recorded continuously along with simultaneous video recording to monitor the subject's facial expressions for non-verbal behavioral cues. The 5 minute time interval was chosen to balance out data analysis needs along with realistic constraints regarding the limited attention span of the subject.

Table 1. List of experimental conditions for one trial.

Number	*Condition*	*Description*
1	Resting 1	Sitting quietly in own wheelchair (no interaction or other stimuli)
2	Interaction	Sitting in own wheelchair, singing and interacting with adults
3	Passive Mobility	Sitting in own wheelchair, playing and interacting with adults using passive mobility
*4**	Power Mobility	Sitting in own wheelchair on PWCT, playing and interacting while using power mobility
1b	Resting 2	Sitting quietly in own wheelchair (no interaction or other stimuli)

**only during intervention phase*

2.2.4 EEG Recording

The EEG was recorded with a modified wireless Emotiv EPOC© (Emotiv, Inc., San Francisco, CA) neuroheadset. Our modified headset replaces the original system's 14 electrodes, including two reference electrodes, with gold cup disc electrodes (MVAP Medical Supplies Inc., Newbury Park, CA) mounted on a custom sized EasyCap (EASYCAP GmbH, Herrsching, Germany). The electrode placement of the original headset (AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4, referenced to CMS (P3) and DRM (P4)) according to the International 10-20 system as shown in Figure 2-1 was used.[47]

The data were recorded using the Emotiv's proprietary acquisition system at a sampling rate of 2048 Hz and notch filtered at 50 Hz and 60 Hz using a built-in digital 5^{th} order Sinc filter. The resulting signal bandwidth was 0.2-45.0 Hz. The electrode impedance was kept below 20 kΩ by applying Nuprep Skin Prep Gel at the electrode sites and dampening the subject's hair (the low

frequency artifacts induced by cross-bridges between the electrodes were filtered out as explained in the next section).

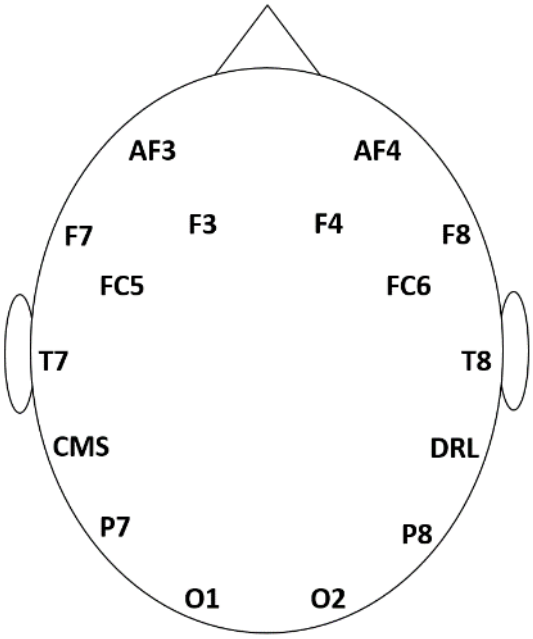


Figure 2-1. Electrode placement of the original and modified neuroheadset according to the International 10-20 system.[47] The 14 electrodes include anterior-frontal (AF), frontal (F), frontal-central (FC), temporal (T), parietal (P), and occipital (O) channels, while the two reference electrodes CMS and DRL are placed in the parietal positions equivalent to P3 and P4. Odd numbers refer to the left hemisphere, whereas even numbers refer to the right hemisphere.

The signal was transmitted with the Bluetooth® SMART 4.0 LE protocol from the EasyCap (EASYCAP GmbH, Herrsching, Germany) to a Lenovo X1 Carbon laptop (Lenovo, Morrisville, NC), where the data was acquired in TestBench™ software (Emotiv, Inc., San Francisco, CA). Prior to transmission, the signal is down sampled to 128 Hz. The low sampling rate poses a major challenge to subsequent signal analysis and limits spectral resolution.

Video was recorded with a Microsoft LifeCam (Microsoft Corporation, Redmond, WA) in order to capture the subject's movement and facial expression during data collection. EEG data were tagged to indicate the first frame of the video which allowed for data synchronization.

2.2.5 Preprocessing

The EEG data were referenced to average reference by subtracting the average over all electrodes from each electrode for each time point,[48] and was filtered using a high pass 7th order Butterworth filter with a cut-off at 2Hz. This was done to account for any cross-bridges that may formed between electrodes. Recall that the subject's hair was dampened to provide a good connection between electrode and scalp. However, this dampening results in electrode-electrode connections called cross-bridges that result in low-frequency artifacts (oscillations of 2 Hz or lower).[49,50] The data were visually inspected and periods of large amplitude artifacts due to electrode movement and were removed from the data.

A wavelet enhanced independent component analysis (wICA[51]) was applied using the InfoMax[52] algorithm in order to remove ocular artifacts. The advantages of this method are that it is automated and thereby more objective, and it avoids the complete removal of ICs containing artefactual data. The ICA algorithms InfoMax, JADE, and SOBI, as well as wavelet enhanced InfoMax and JADE, were tested for their performance of artifact removal while retaining relevant EEG signal, and the wICA algorithm using InfoMax achieved best results and stability. For each experimental condition an artifact-free record of 35 seconds was randomly selected from the pre-processed data.

2.2.6 Spectral Analysis

The data was Fourier[53] transformed using the Welch's method[54] with a smoothed Hanning-window of 2.5 seconds and 50% overlap, in order to obtain the power spectral density (PSD). Orekhova et al.[14] found that the frequency range of the brain rhythms in children is usually lower than the typical range as it is found in adults, and depends strongly on the children's age. Hence, the partition of the spectrum into narrower bands instead of conventional bands is

recommended.[26,14] In the present study, this was realized by dividing the frequency range into narrow sub-bands of 1 Hz in width (1 Hz = 4 samples with 512-point FFT). The mean (average) across the 4 samples in each sub-band was computed and normalized to the total mean power across channels and frequencies. Only the sub-bands falling into the range between 2.0 Hz and 12.0 Hz were considered for further analysis in respect of previous findings by Orekhova et al.[14] and for exclusion of myogenic artifacts with a typical frequency distribution above 11Hz,[55] which were abundant in several recordings.

2.2.6.1 Identification of Theta

The identification of the different rhythms (delta, theta, alpha, beta, and gamma) is crucial for the interpretation of their functional meaning. Therefore, the first step of the analysis was to find the frequency limits of the functional brain rhythms in the subject's frequency spectrum. Orekhova et al.[14] associated high theta power in frontal areas with exploratory behavior and posterior theta with attention to social stimulation. Based on these findings, one may expect an increase in theta power during social interaction in comparison to resting condition, which would allow us to identify the theta range of the subject in the present study. Thus, the comparison of the power in the EEG spectrum during Resting 1 condition and power in the EEG spectrum during Interaction in the first baseline phase was of special interest. Therefore, the percentage change in PSD magnitude was computed using following formula:

$$PSD_{change} = \frac{PSD_{interact} - PSD_{rest1}}{PSD_{rest1}} \cdot 100\%$$

where PSD_{rest1} refers to the PSD of the data recorded during Resting 1 condition and $PSD_{interact}$ refers to the PSD of data recorded in Interaction condition. The results were averaged across trials.

2.2.6.2 Changes in EEG Spectrum

For each channel, the mean PSD of each 1Hz-sub-band between 2.0 and 12.0 Hz for Resting 1 and Resting 2 were obtained. In addition, the percentage change in PSD magnitude from Resting 1 to Resting 2 was also computed using following formula:

$$PSD_{change} = \frac{PSD_{rest2} - PSD_{rest1}}{PSD_{rest1}} \cdot 100\%$$

where PSD_{rest1} refers to the PSD of the data recorded during Resting 1 condition and PSD_{rest2} refers to the PSD recorded during Resting 2 condition.

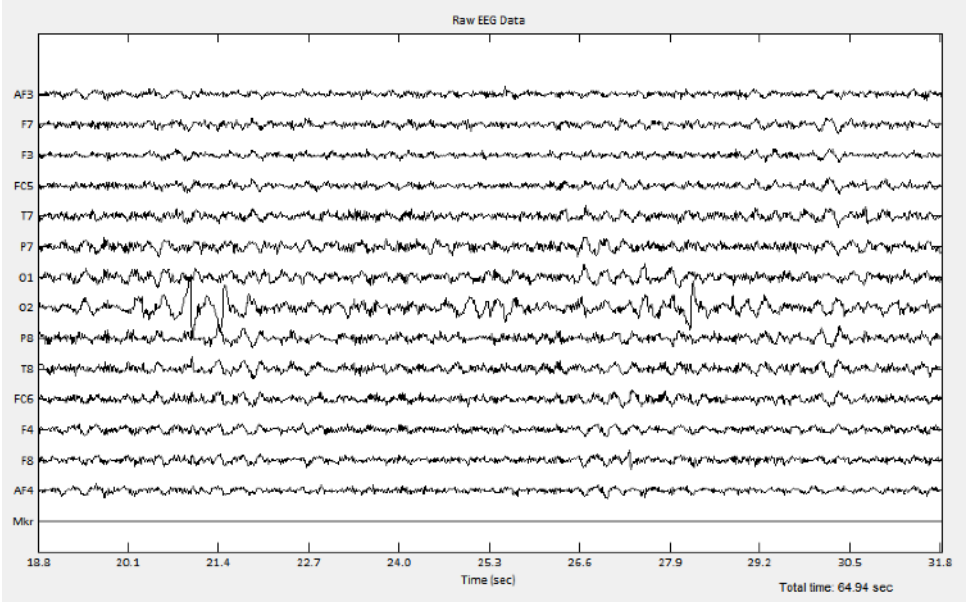
2.2.7 Statistical Analysis

Due to the serial dependency of data points in single-single subject studies, conventional statistical procedures such as ANOVA and t-test are rarely used.[56] An acknowledged method for analyzing single-subject data is the two-standard deviation band method.[57] The two-standard deviation band method compares the mean of the first baseline phase with subsequent data points.[57] Based on this technique, significance was measured by the number of data points falling outside of ±2 standard errors from the mean of the first baseline. The standard error was used due to the previous averaging in the Fourier transform.[54] With a maximum of 4 data points in each phase, the significance score (s-value) ranged from 0 to 4 for positive changes and from 0 to -4 for negative changes. An s-value greater or equal to 2 (positive change) and smaller or equal to -2 (negative change) was considered as significant ($p<0.05$).[57] This statistical procedure has the advantage of being sensitive to changes in variability across all phases of a single-subject design.[57]

2.3 RESULTS

Figure 2-2 (a) shows the representative raw data from all channels over 13 seconds during Resting 1 in the first baseline phase while Figure 2-2 (b) shows the same data after preprocessing to remove artifacts using the wICA[51] algorithm.

a)



b)

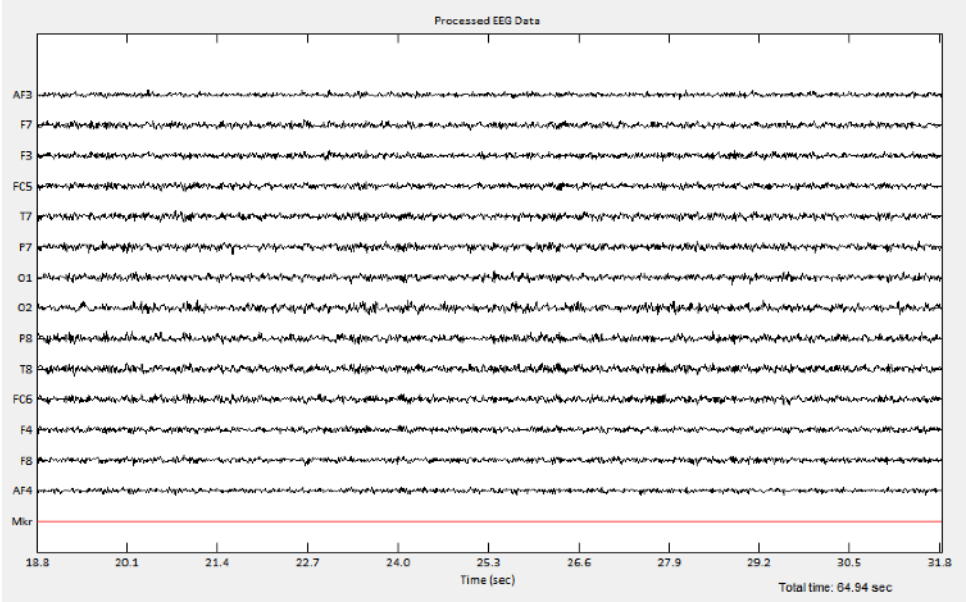


Figure 2-2. Representative raw (a) and preprocessed (b) EEG data from all channels after removing strong artifacts manually. The preprocessing using wICA to remove artifacts was done for all experimental conditions over the A-B-A-B phases.

Note that high amplitude artifacts with low frequency (e.g. channel O2 around 21.3 sec) as well as high (e.g. O2 around 25.3 sec) frequency character are removed after applying wICA. This data preprocessing step was done for all experimental conditions (Resting 1, Interaction, Passive Mobility, Power Mobility, and Resting 2) over the A-B-A-B phases.

2.4 IDENTIFICATION OF THETA

To identify the subject's theta band, EEG data of the first baseline phase (A1) during resting condition before the trial (Resting 1) and during social interaction (Interaction) was analyzed. The mean PSD in each sub-band was obtained for each channel and trial. The sub-band values were averaged across trials during A1, while the first trial was disregarded due to strong distortion.

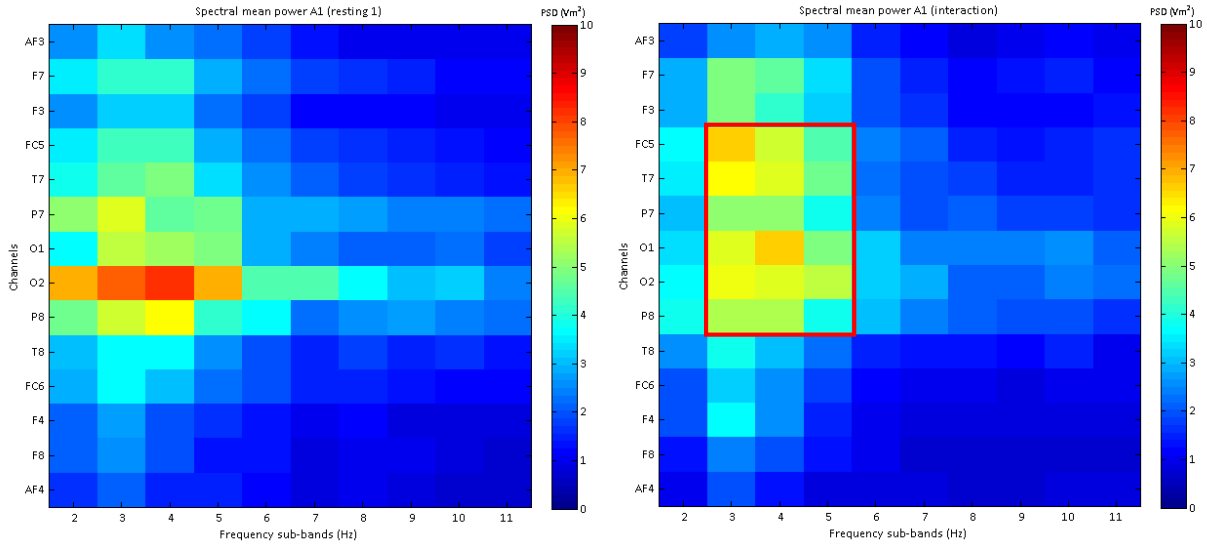


Figure 2-3. Spectral mean power for resting condition (left) and interaction condition (right) during first baseline (red=high mean power, blue=low mean power). The vertical axis represents the 14 different channels (odd numbers: left hemisphere, even numbers: right hemisphere), while the horizontal axis represents the 10 sub-bands between 2.0 and 12.0 Hz. The red square in the right panel marks the region of high power during Interaction condition.

Figure 2-3 illustrates the mean of the PSD in each sub-band during Resting 1 condition (left panel) and Interaction condition (right panel) over all channels. In resting condition, higher power is clustered around the peak in 4.0-5.0 Hz sub-band of the right occipital channel (O2). The peak shifts to the left hemisphere (O1) for Interaction condition, while higher power values spread primarily across a range from 3.0 to 6.0 Hz in left occipital, parietal, temporal, and frontal-central as well as right occipital and parietal regions. This region is marked in the right panel of Figure 2-3 with red square.

The computation of percentage change between the two conditions indicates a predominant increase in frontal-temporal regions in the left hemisphere from 3.0 to 6.0 Hz, while a predominant decrease in power is observed for the entire right hemisphere.

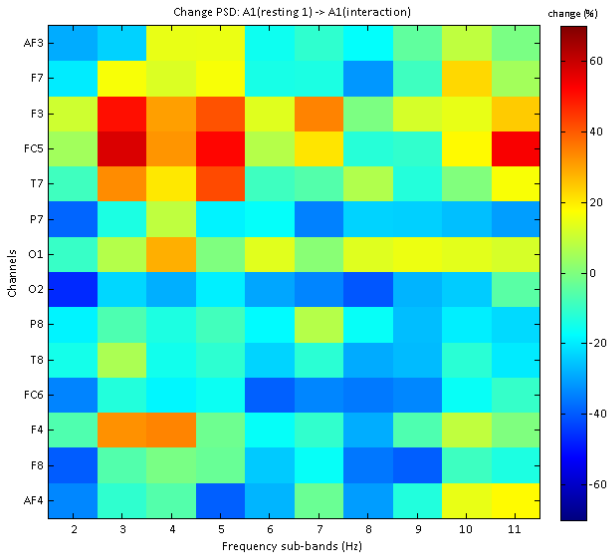


Figure 2-4. Average percentage change from Resting 1 condition to Interaction condition during the first baseline. The highest increase is noted between 3.0 and 6.0 Hz for left frontal and temporal channels (F3, FC5,T7).

The change from Resting 1 to interaction condition was also tested for significance ($p<0.05$) with the two-standard deviation band method. However, no significance was achieved for any of the

sub-bands and channels. However, these initial trend suggests the presence of a distinct theta band in the 3.0-6.0 Hz range for our subject.

2.5 CHANGE IN EEG SPECTRUM

For each channel, the PSD was computed for resting condition in the beginning of the trial (Resting 1) and at the end of the trial (Resting 2). In addition, the percentage change in PSD magnitude from Resting 1 to Resting 2 condition was also computed. Figure 2-5 (left hemisphere) and Figure 2-6 (right hemisphere) give an example of the computed PSD of each channel for the second week of each phase – first baseline (A1), first intervention (B1), second baseline (A2), and second intervention (B2).

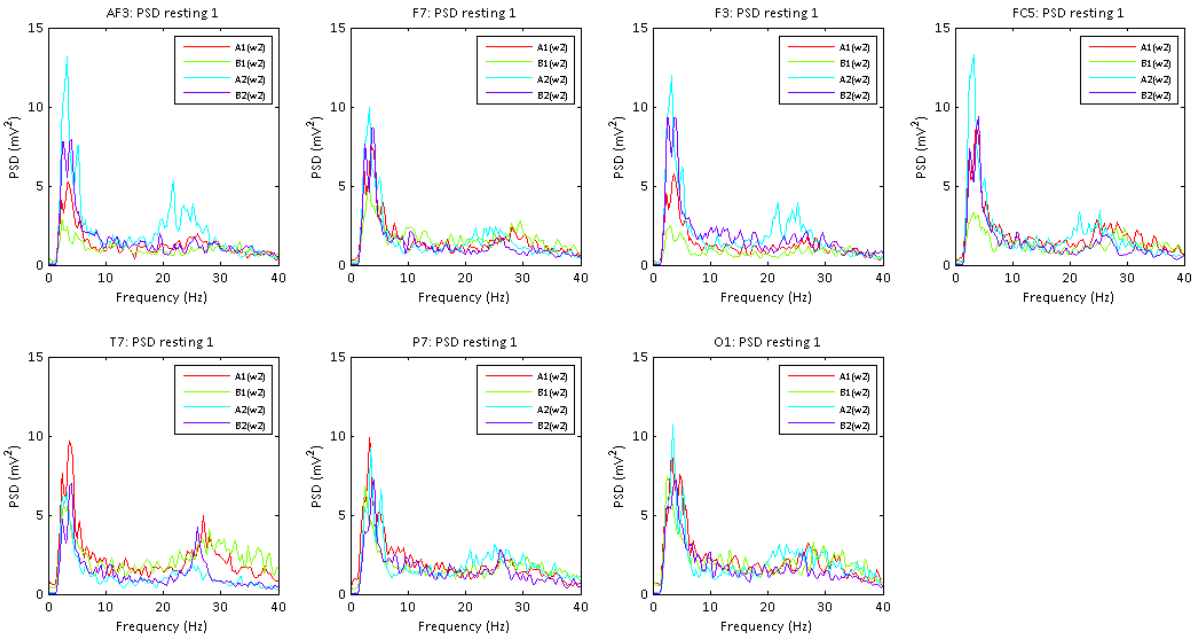


Figure 2-5. PSD of each channel of the left hemisphere during Resting 1 condition. Examples of the second week (w2) of each phase (A1, B1, A2, and B2) are shown.

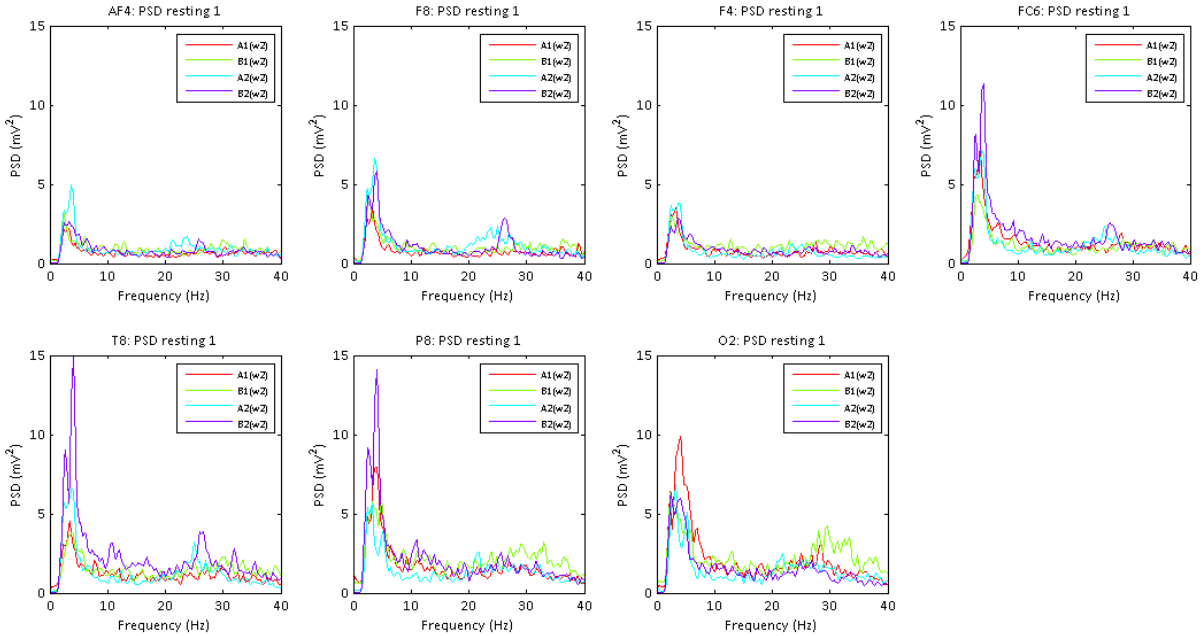


Figure 2-6. PSD of each channel of the right hemisphere during Resting 1 condition. Examples of the second week (w2) of each phase (A1, B1, A2, and B2) are shown.

An example of the PSD divided in sub-bands is given in Figure 2-7 corresponding with the data shown in Figure 2-5 and Figure 2-6. Each panel in Figure 2-7 illustrates the results of one trial in the frequency range from 2.0 to 12.0 Hz.

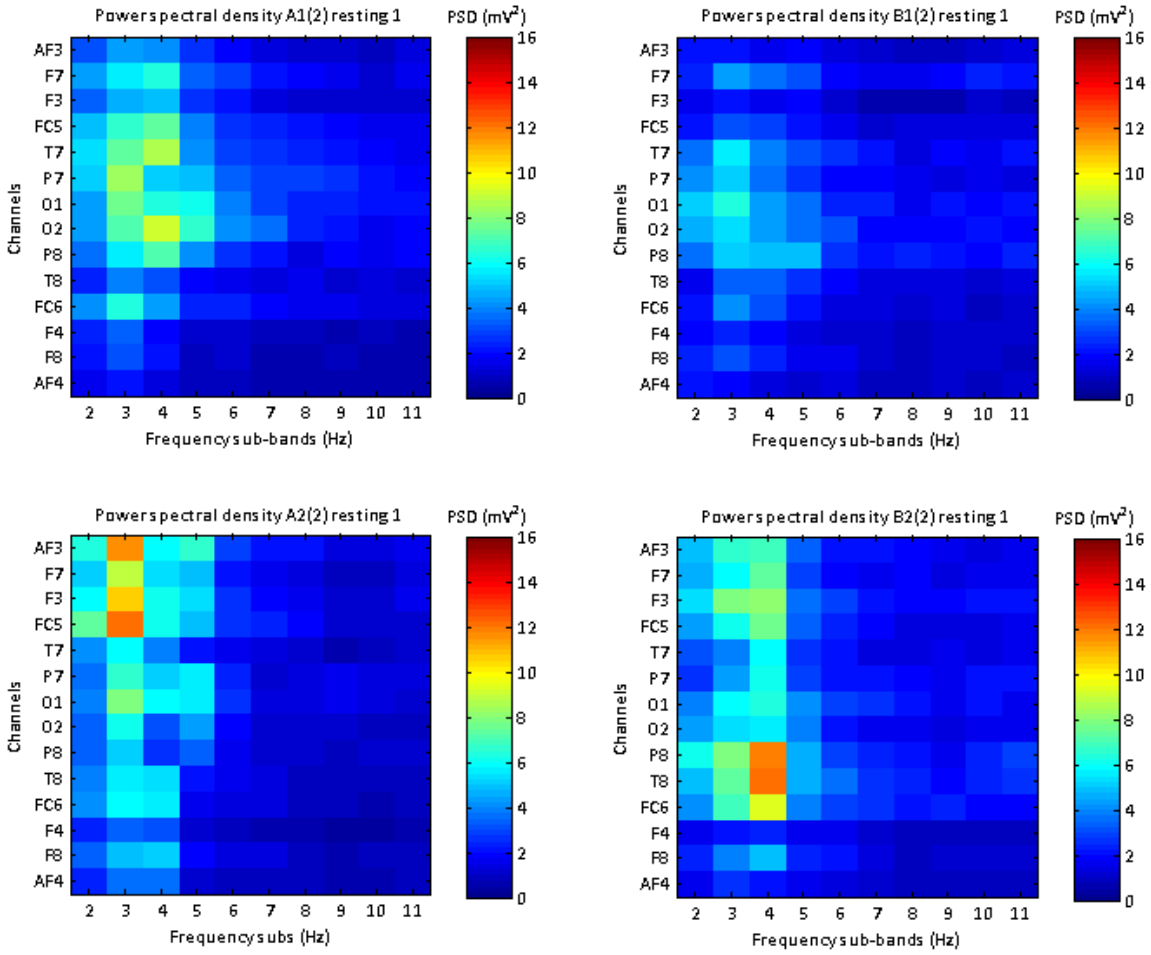


Figure 2-7. Mean PSD for each sub-band and channel represented by the color (red=high, blue=low). This is an example for the second week (w2) of each phase (A1, B1, A2, and B2).

In order to detect changes in the EEG spectra of the different trials, the means of each 1Hz-band between 2.0 and 12.0 Hz of each channel were compared. The comparison was tested for significance using the two-standard deviation band method. The analysis was performed for the three cases, Resting 1 condition, Resting 2 condition, and percentage change from Resting 1 to Resting 2 condition.

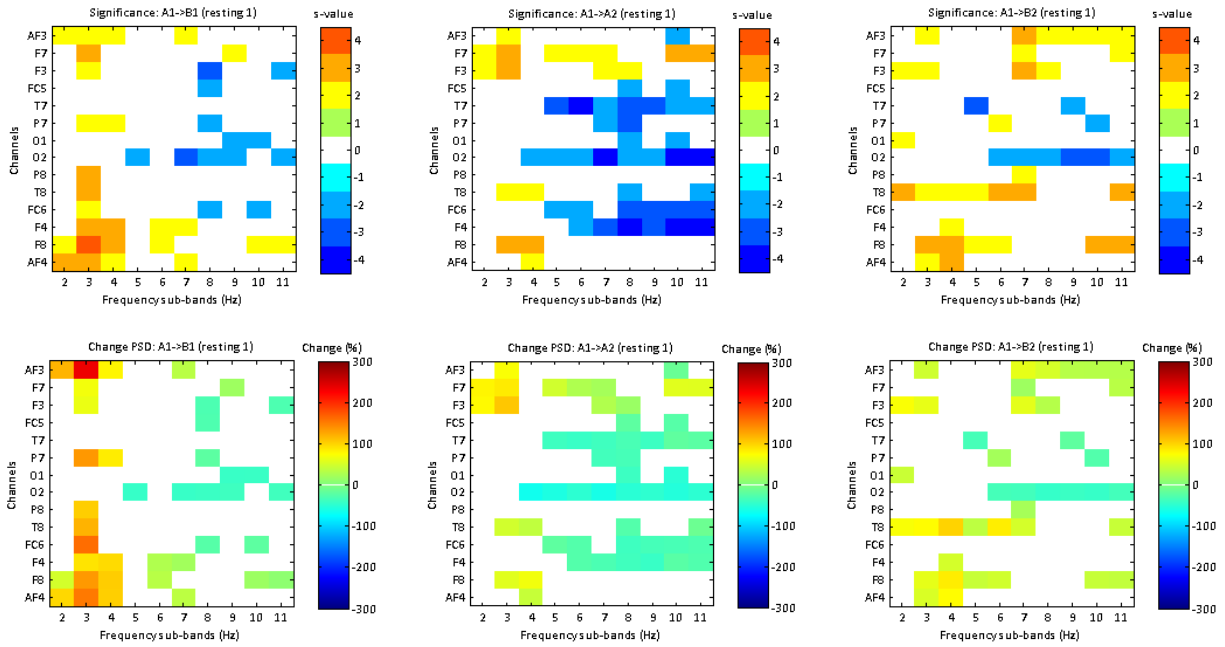


Figure 2-8. Significant increase (red) and decrease (blue) in spectral mean power for each sub-band in respect of the first baseline (A1) for each subsequent phase (B1, A2, and B2). The upper panels illustrate the significance score (s-value) for each sub-band and channel, while only s-values >= 2 or s-values <=-2 are displayed. The lower panels show the percentage change for significant changes in each sub-band and channel. The shown data was recorded during Resting 1 condition.

For the Resting 1 condition, significant increase in power was found for 3.0-5.0 Hz sub-bands of parietal, temporal and frontal channels (AF3, F7, F3, P7, P8, T8, FC6, F4, F8, AF4) with highest s-values in the right hemisphere during phase B1. The increase is most pronounced in the 3.0-4.0 Hz band as seen in the lower panel showing the percentage change from phase A1 to B1. A decrease in power is most significant in occipital channel O2 and frontal channel F3 between 7.0 and 9.0 Hz, along with a cluster that spans over occipital, parietal, temporal and frontal regions from 7.0 to 12.0 Hz.

Referring to the left lower panel, occipital regions decrease most in magnitude; this decrease is blunted towards the frontal areas. In phase A2, temporal and frontal channels (AF3, F7, F3, T8, F8, AF4) show most significant increase between 3.0 and 5.0 Hz. Decreasing power with highest s-values are found in left temporal, right occipital, and right frontal-central channels between 5.0

and 12.0 Hz, while a cluster of high power spanning over occipital, parietal, temporal, and frontal regions is noted with predominance in the right hemisphere.

The percentage change plot in the middle lower panel shows similar results as found in phase B1. The power increase is most pronounced between 3.0 and 4.0 Hz for frontal channels, while the decrease is marked in occipital regions getting successively smaller towards frontal regions between 5.0 and 12.0 Hz. A pattern did not emerge in the increased power during B2 which can occur in arbitrary channels and sub-bands in the observed frequency range from 2.0 to 12.0 Hz. Decreasing power, however, occurs only in left parietal and temporal channels and right occipital channel O2 in the range from 5.0 to 12.0 Hz.

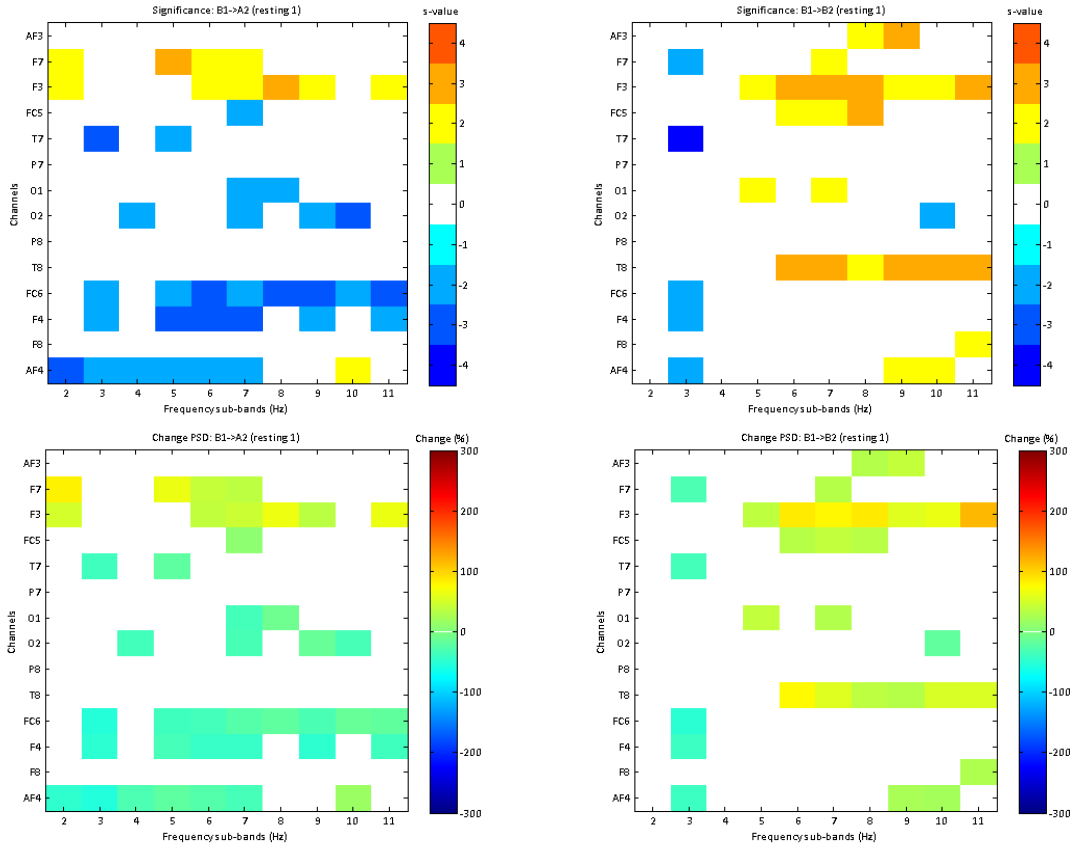


Figure 2-9. Significant increase (red) and decrease (blue) in spectral mean power for each sub-band in respect to the first intervention phase (B1) for each subsequent phase (A2, and B2). The upper panels illustrate the significance score (s-value) for each sub-band and channel. The lower panels show the percentage change for significant changes in each sub-band and channel. The shown data was recorded during Resting 1.

Figure 2-9 shows changes in phase A2 and B2 in comparison to intervention phase B1 for Resting 1 condition. In phase A2, The s-value indicates an increase in power in the left frontal channels F7 and F3 which spans from 2.0 Hz up to 12.0 Hz, while a decrease in power is denoted predominantly in the occipital and right frontal channels across the entire range from 2.0 to 12.0 Hz. Phase B2, however, shows a power decrease in the 2.0-3.0 Hz sub-band for left frontal (F7), temporal (T7), and the right frontal (FC6, F4, AF4) channels. Between 6.0 and 12.0 Hz the power increases in left frontal (AF3, F7, F3, FC5) and right temporal (T8) regions.

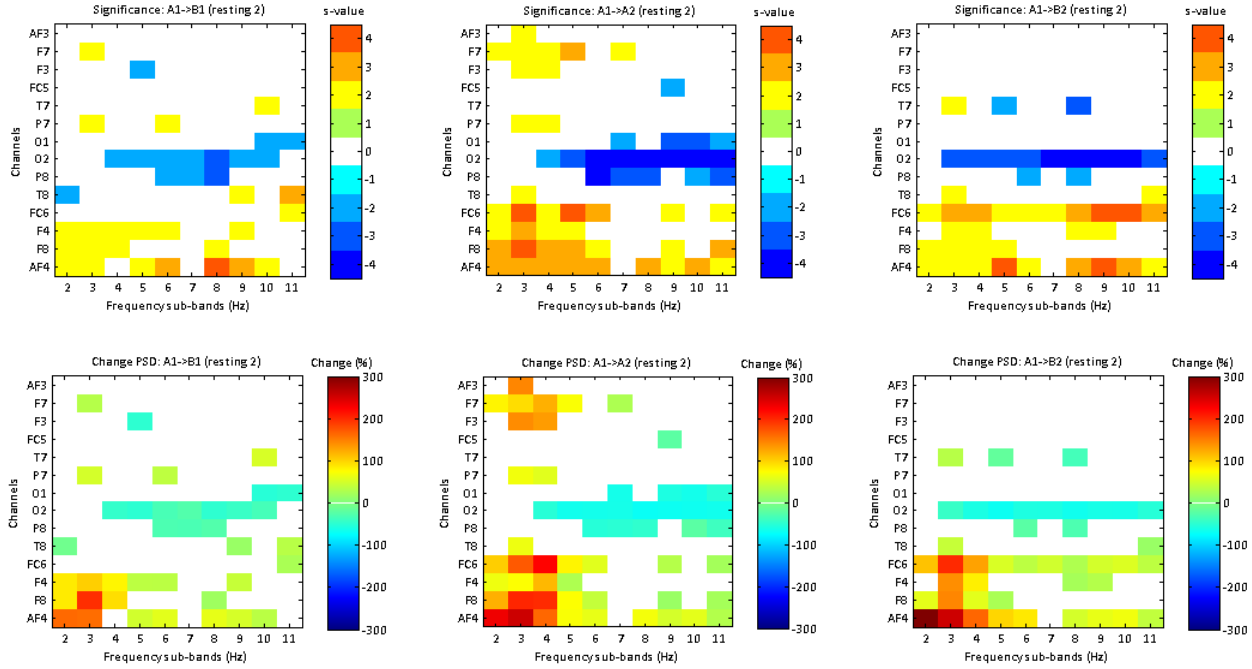


Figure 2-10. Significant increase (warm colors) and decrease (blue colors) in spectral mean power for each sub-band with respect to the first baseline (A1) for each subsequent phase (B1, A2, and B2). The upper panels illustrate the significance score (s-value) for each sub-band and channel, while only s-values >= 2 or s-values <=-2 are displayed. The lower panels show the percentage change for significant changes in each sub-band and channel. The shown data was recorded during Resting 2.

Analysis of Resting 2 data revealed a significant broadband (2.0-12.0 Hz) power increase for temporal and frontal channels, primarily in the right hemisphere (T8, FC6, F4, F8, AF4) for all phases. Highest percentage change occurs in right frontal channels (F8, AF4) between 2.0 and 5.0 Hz. In phase B1, decreasing power is observed in the left occipital channel (O1) between

10.0 and 12.0 Hz and in the right occipital channel (O2) between 3.0 and11.0 Hz. Additionally, power decrease is found right parietal channel P8 between 6.0 and 9.0Hz. The decrease is most significant for right parietal and occipital channels in the 8.0-9.0 Hz band. Phase A2 shows considerable decrease between 4.0 and 12.0 Hz for occipital and right parietal regions (O1, O2, P8), while predominant power decrease in phase B2 is limited to the right occipital channel O2, but spans from 3.0 to 12.0 Hz.

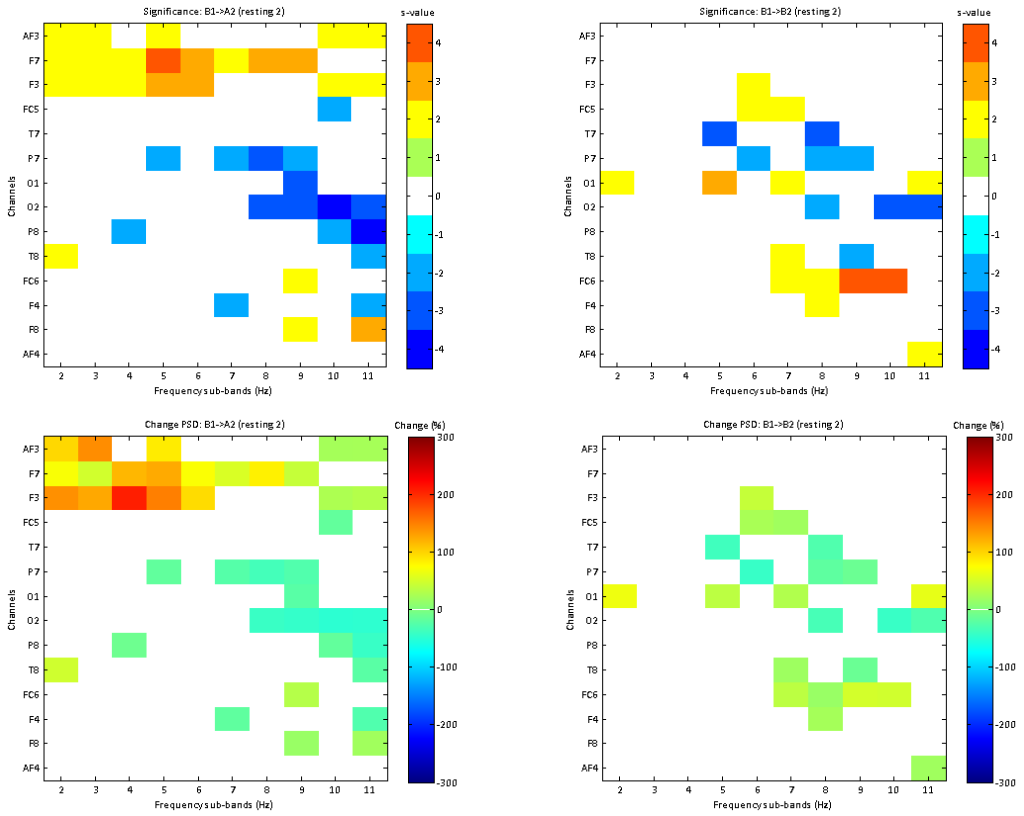


Figure 2-11. Significant increase (warm colors) and decrease (blue colors) in spectral mean power for each sub-band with respect to the first intervention phase (B1) for each subsequent phase (A2, and B2). The upper panels illustrate the significance score (s-value) for each sub-band and channel. The lower panels show the percentage change for significant changes in each sub-band and channel. The shown data was recorded during Resting 2.

Also for Resting 2, phase A2 and B2 were compared to intervention phase B1. As shown in Figure 2-11, the power for the left frontal channels increases significantly from 2.0 to 12.0 Hz in phase A2. The percentage change of power (shown in left lower panel), however, is most

prominent between 2.0 and 6.0 Hz. Additionally, the power decreases between 5.0 and 12.0 Hz for left parietal, occipital, right temporal and frontal channels (P7, O1, O2, P8, T8, F4). The power changes in phase B2 in respect to B1 do not have any specific clusters but low percentage increases and decreases are concentrated between 5.0 and 12.0 Hz for the channels F3, FC5, T7, P7,O1, O2, T7, FC6, F4.

During phase A1, frontal and left temporal channels (F3, T7, FC6, F4, F8, AF4) show a power decrease from Resting 1 to Resting 2 condition, while occipital channels (O1, O2) and P8 have slightly increasing power as is noted in Figure 2-12 and Figure 2-13.

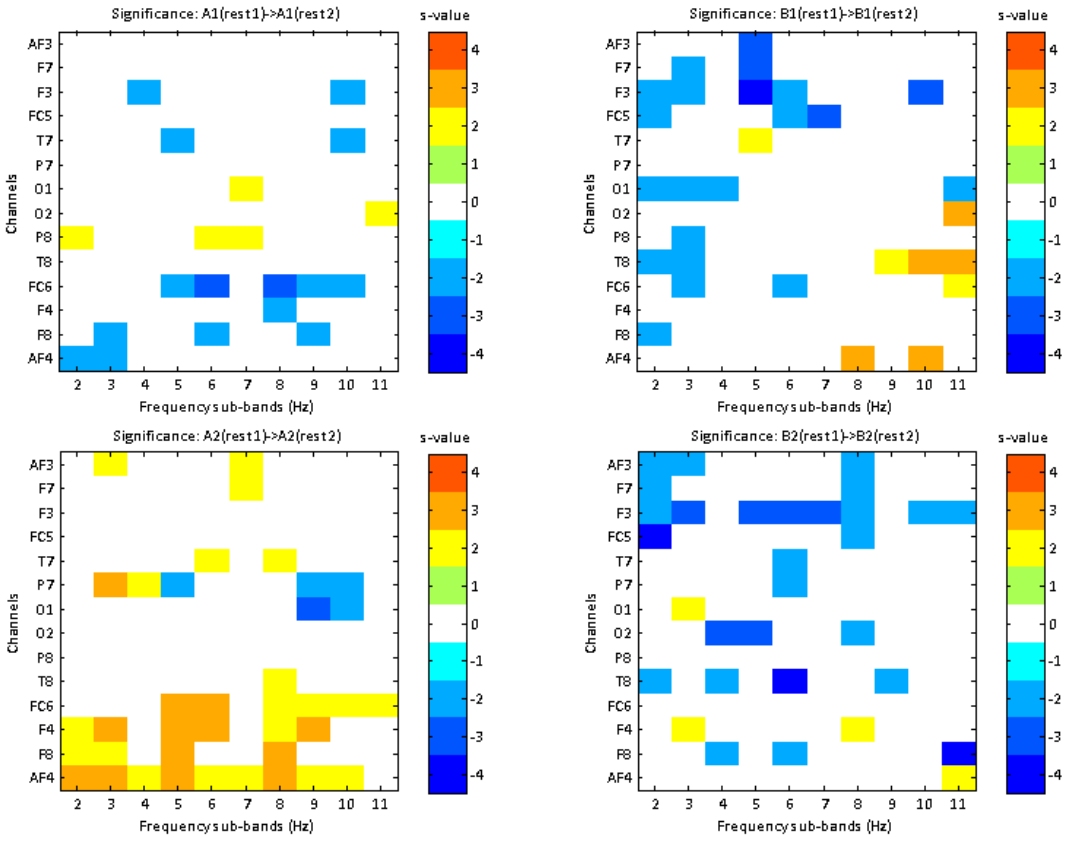


Figure 2-12. Significant increase (warm colors) and decrease (blue colors) in spectral mean power for each sub-band and channel of Resting 2 with respect to Resting 1 in each phase (A1, B1, A2, B2). The colors illustrate the significance score (s-value) for each sub-band and channel, while only s-values >= 2 or s-values <=-2 are displayed.

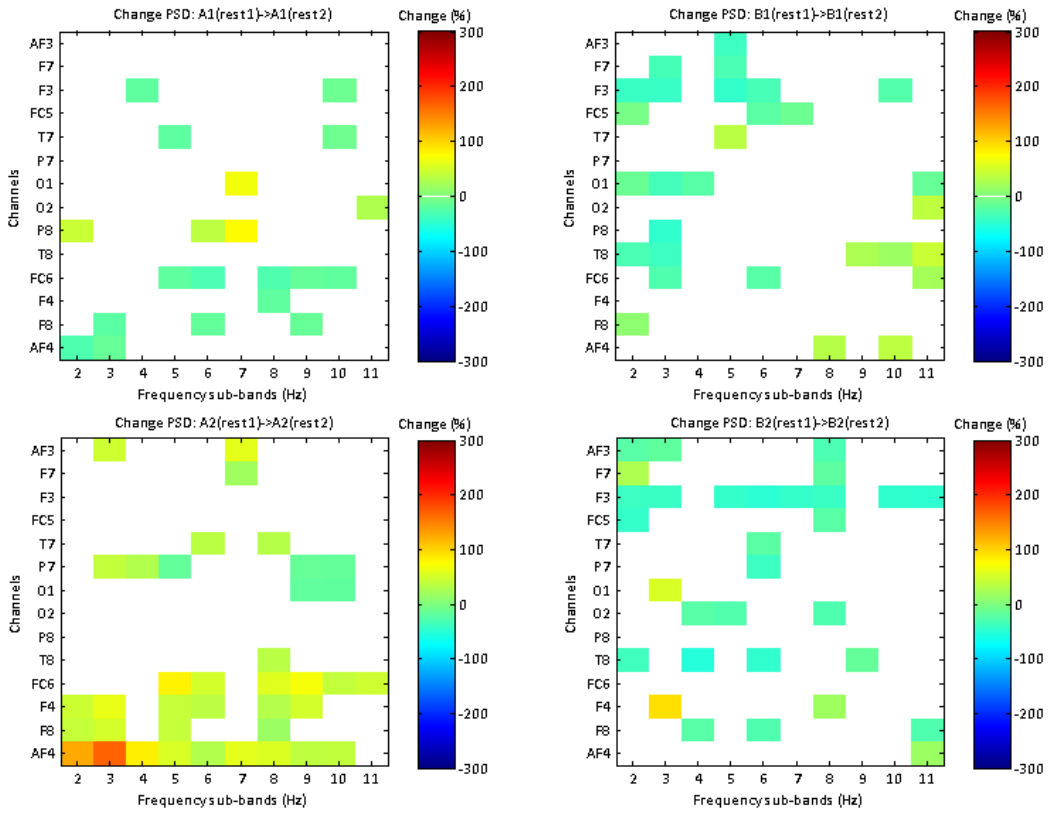


Figure 2-13. Significant increase and decrease in spectral mean power for each sub-band and channel of Resting 2 with respect to Resting 1 in each phase (A1, B1, A2, B2). The colors illustrate the percentage change for significant changes in each sub-band and channel.

Positive changes were observed in phase B1 for frequencies above 8.0 Hz in the right hemisphere for occipital (O2), temporal (T8), and frontal channels (FC6, AF4), while power decreased in channel O1, P8, T8, FC6, and F8 from 2.0 to 4.0Hz, and in left frontal region (AF3, F7, F3, FC5) from 2.0 to 8.0 Hz. In phase A2, broadband positive changes occurred for frontal areas in the right hemisphere, along with some power increase in left frontal, temporal, parietal regions below 9.0 Hz. Left parietal channel P7 and the left occipital channel O1 also show a negative change in power between 9.0 and 11.0 Hz. Significant negative power change is predominant in left frontal and temporal (AF3, F7, F3, FC5, T7) as well as in right occipital, temporal, and frontal regions (O2, T8, F8) between 2.0 and 12.0 Hz.

2.6 DISCUSSION

2.6.1 Identification of Theta

The goal of this part of the analysis was to identify a cluster of adjacent frequency bins that could be associated with the properties of the theta band.[14] Previous research has defined the theta band for typically developing infants (8-12.4 months) from 3.6 to 5.6 Hz and for preschool children (three years 8 months to six years 11 months) from 4.0 to 7.6 Hz.[14] For the subject of the present study (3 years and 2 months) with microcephaly and CP, an increase in spectral power from Resting 1 condition to Interaction condition was primarily observed in frontal-temporal regions in the left hemisphere from 3.0 to 6.0 Hz. The frequency range as well as topography of increased brain activity are in accordance with the results of Orekhova et al.[14] for infants during social stimulation. The inconsistency in age may be attributed to the abnormal brain development of the subject. Additionally, the subject's cognitive abilities are unknown and limited communication makes classification of her stage of maturity difficult.

The lack of statistical significance in change of brain activity due to social interaction may be attributed to the limited number trials. Additionally, the level of distortion in the analyzed signals, which is unknown but regarded as high even after artifact removal, may also contribute to higher variability in the signal and lead to distortion of the results.

Despite these confounding factors, our data suggest that the theta frequency band for our subject is 3-6 Hz and the alpha band, which typically has a bandwidth of 5 Hz, is 6 – 11 Hz.

2.6.2 Functional Meaning of the EEG Activity

First, data are compared to A1, the first baseline phase. For Resting 1 condition in phase B1, significant increase in power was found particularly in the 3-4 Hz band in left frontal and right frontal, temporal and parietal regions. Theta activity is typically associated with emotional and

attentional processes during a specific task.[26,14] Specifically, infants and children have shown increased theta power during exploration of toys and social stimulation.[14] However, the subject is not performing a specific task during Resting 1 condition, so the observed increase in theta power contradicts the findings of previous studies. In addition, the spectral power in the 7.0 - 12.0 Hz band decreases significantly in the occipital, parietal, and frontal regions. The subject's alpha band lies in a range from 6.0 to 11.0 Hz, which implies a decrease in alpha power for Resting 1 condition.

This opposing trend in theta and the alpha rhythms has been observed before.[26] Klimesch[26] related spontaneous increase in theta and decreasing alpha power in adults with decrease of cognitive performance or the transition from wakefulness to sleep. However, very little is known about tonic changes of theta in children.[58] Klimesch[26] also found alpha suppression during relaxed but alert wakefulness in an anticipatory situation. Hence, one could argue that the observed decrease in alpha power may not be related to diminishment of cognitive performance but rather reflect anticipation of the upcoming trial. Additionally, theta has been associated with highly focused attention and engagement of neural networks in a highly focused mode of processing.[14] Even if the subject was not involved in a specific task during resting condition, the theta increase may be related to recruitment of additional cortical resources and enhancement of cognitive and attentional processes during rest.

Similar patterns were observed for phase A2 and B2, however, with diminished theta activity only in frontal regions. The power increase in the left frontal channels F7 and F3 during phase A2, and in the frontal channels AF3, F3, F8 and temporal channel T8 during phase B2 is not limited to the theta band and spreads across the entire frequency window from 2.0 to 12.0 Hz. A broadband increase does not comply with any known functional meaning and is rather attributed

to artefactual data, imbalances of the data acquisition system, or daily condition of the subject, to which the applied statistical procedure in combination with the small sample size of four trials may not be sufficiently robust.

Phase A2 shows a more pronounced alpha activity that spreads also over temporal regions in both hemispheres. Again, these observations could be related to diminished cognitive performance or anticipation. The latter explanation is more plausible after eight weeks of repeated trials. Although less pronounced in phase B2, some of the sub-bands assigned to the alpha band show a decrease in spectral power in the left parietal and temporal as well as right occipital regions. After twelve weeks, the subject may have become familiar with the weekly procedure which could lead to a 'thinning' of the anticipatory response before trials. The consistency of theta power, and the clear demarcation of the alpha rhythm across all three phases imply once more that the broadband increase in frontal channels during phase A2 and B2 may not have neural but rather artefactual origin.

Phases A2 and B2 were also compared to the first intervention phase B1. The changes from B1 to A2 range from 2.0 12.0 Hz and denote an increase of power in the left hemisphere and a broadly distributed decrease in power of the right hemisphere. However, the changes do not seem to be related to the subject's theta or alpha activity, and therefore inferences about the subject's underlying cognitive brain activity are not possible. The change from phase B1 to B2, however, shows a power decrease in the range 3.0-4.0 Hz, which is associated with the subject's theta band, accompanied by an increase in the subject's alpha band from 6.0 to 12.0 Hz in left frontal and right temporal regions. Decreasing theta and simultaneous increase in alpha power has been observed in previous studies and has been associated with an increase in cognitive performance.[26] From these results, one could infer that phase A2 did not have an effect on

cognitive performance, but phase B2 may enhance the subject's cognitive functions in comparison to phase B1. However, it remains unclear whether any changes in cognitive brain function in phase A2 are indeed missing or just masked by artefactual noise.

In all three phases of Resting 2 condition, a predominant power increase from 2.0 to 12.0 Hz appears in frontal-temporal channels of the right hemisphere. The origin of this broadband power increase is unknown and literature suggests that this pattern may be caused by ocular artifacts.[59] Focusing on the presumably unaffected channels of the posterior areas and left hemisphere, the results of Resting 2 condition seem more arbitrary than in Resting 1 condition, but might still be worth an attempt for interpretation. Phase B1 shows a power increase in the theta band (3.0-4.0 Hz) in frontal and parietal regions and also for the upper alpha band (8.0-11.0 Hz) in the temporal and right frontal channel. Power increase in the upper alpha band has been associated with an increase in cognitive performance, but was found to bring along a decrease in the theta band.[60] A decrease in the theta range appeared in the left frontal channel F3 in the 5.0-6.0 Hz-bin. Due to the lack of topographically clustered changes, however, these observations do not support any association to a functional meaning of these result. A more typical pattern is found during the second baseline phase A2, where a cluster of decreasing alpha power appears in occipital and parietal channels. Ranging predominantly from 6.0 to 12.0 Hz, the power decrease leaks into the theta band down to 4.0 Hz for the right occipital channel. Since the bands may overlap, this should not be of any concern for further interpretation of the alpha rhythm. Increased theta power is observed in frontal regions from 3.0 to 5.0 Hz, along with presumably artefactual activity in channel F7 from 2.0 to 8.0 Hz. As stated earlier, these observations could be associated with increased alertness and expectancy.[26] Since the data was recorded at the end

of the trial, it seems more plausible to relate the power changes to increased alertness than anticipation.

In phase B2, increased theta power of the left hemisphere is reduced to only the temporal channel T7, while alpha activity occurs in left temporal channel T7 and right parietal channel P8 with broadband decrease from 3.0 to 12.0 Hz in the right occipital channel (O2). As mentioned earlier, this 'thinning' of alpha activity may be attributed to increasing familiarity with the performed tasks and environment and diminish the cognitive load to process new information.[26,14]

Comparison of phase A2 and B2 to first intervention phase B1 reveals a broadband power increase in the left frontal channels, while the percentage change is most pronounced in the theta band. The power decreases in the alpha range for parietal and occipital regions as well as frontal areas in the right hemisphere. These observations are consistent with the results obtained in comparison with the first baseline A1. The changes from phase B1 to B2 do not show any association with the subject's functional rhythms and therefore no plausible inferences can be drawn from these results.

The change from Resting 1 to Resting 2 did not reveal any obvious patterns that might be associated with any functional meaning in phase A1 and B2. The significant changes are scattered arbitrarily in a wide topographical and frequency range. This measure is also influenced by the noted artefactual properties of the Resting 2 data of the right hemisphere, which may explain the broad band power increase of the right frontal channels in phase A2. In contrary to the lack of clear demarcation of theta and alpha during phase A1 and B2, the intervention phases B1 and A2 show interesting clusters. In phase B1, left frontal and occipital channels as well as right parietal, temporal, frontal channels denote a power decrease in the theta band, while the

alpha power increases for right occipital, temporal, and frontal channels. The phenomenon of decreasing theta and increasing alpha power was found to be associated with an increase in cognitive performance in typically developing children and adults.[26] This may imply that using power mobility induced short term activation of cortical resources that allows enhancement of cognitive processing.

In phase A2, the theta activity increases, while alpha decreases in the upper alpha band (9.0-11.0 Hz) in the left parietal and occipital channels. Alpha power also increases between 6.0 -9.0 Hz in left frontal and temporal channels. The pronounced broadband increase in power in the right hemisphere is likely the result of artifacts in Resting 2 data. The clusters of increasing and decreasing power in phase A2 are localized in a small topographical range. Decreased alpha power in parietal and occipital regions implies enhancement of visual processing, while increase in theta power in temporal regions may be attributed to recruitment of cortical resources for emotion, memory, and audition.[61] However, due to the subject's brain damage, the topography of the different brain functions is unknown.

In summary:

- The theta rhythm of the subject was identified in the range from 3.0 to 6.0 Hz, while the alpha rhythms was assumed to fall into the band from 6.0 to 11.0 Hz.
- Significant increase of theta power and decrease in alpha power were observed for Resting 1 condition during all three phases following the first baseline A1. Comparison to the first intervention phase B1 revealed changes in the second intervention phase B2, which may derive from enhancement in cognitive performance.[26]
- The results of Resting 2 condition showed similar characteristics, while considerable distortion by ocular artifacts or other factors are assumed in right frontal-temporal

channels due to the broadband distribution of power increase. With respect of previous findings,[26,14] these observations may be related to an increase in alertness and/or anticipation. Since there is no significant difference between baseline phase and intervention phase, there is no evidence for a long term impact of power mobility training. The observed changes in the subject's EEG spectrum could also be explained by the unusual engagement in specific tasks and social interaction as well as the progress of the overall development of the subject during the study. Whether these changes may have been influenced or enhanced by power mobility training is unknown and warrants further investigation.

- Analysis of the percentage change from Resting 1 to Resting 2 condition did not reveal a clear difference between baseline and intervention phase. The observed decrease in theta and increasing alpha power during the first intervention phase could be associated with increasing cognitive performance immediately after the use of power mobility.[26] This implies a short term impact of power mobility training on the subject's EEG spectrum, while it remains unclear whether the observed changes are caused by the use of power mobility or by general learning of a new task.

2.6.3 Limitations of the study

2.6.3.1 Equipment and Artifacts

The EEG was recorded with a modified wireless Emotiv EPOC© neuroheadset that has been successfully used before.[62–64]. The original headset was found to be inappropriate for children, which was addressed by using a custom sized cap instead of the rigid headset frame and replacing the original electrodes with gold cup disc electrodes. Despite previous validation of this modification,[65] different electrodes and the use of skin prep gel instead of the moistened felt

pads could have effected signal strength and quality. A short set-up time was required in order to be tolerated by the subject. Thus, the subject's hair was moistened before administering the cap to achieve better connection from the electrodes through the hair to the scalp. This, however, formed contact bridges between electrodes, similar to so called 'sweat bridges' which are known to induce low frequency artifacts below 1 Hz. Although a high pass filter with cut-off at 2 Hz was applied, the contact bridges may have caused imbalances in signal strength among the different channels.

The original electrode placement of the Emotiv EPOC© system was adopted, but might still might not have been ideal. At the periphery, brain activity is less but muscular artifacts are stronger and more frequent,[55] whereas an even electrode distribution would also increase the spatial resolution of central and posterior regions of the scalp.

It was established that the major problem with the used data acquisition system was artifact contamination due to electrode and cable movement, which was induced primarily by head motion of the subject but also by only light vibration of the subject's wheelchair. These artifacts had to be removed manually from the recording, and reduced, in some cases, the amount of analyzable data considerably from 5 min to a minimum of 35 seconds. Head motion and facial muscle activity such as laughing and coughing contaminated the data additionally with muscular artifacts that are not easy to distinguish from the neural signal. Although a wavelet enhanced independent component analysis was applied to reduce muscular and ocular artifacts, there are not any known reliable measures to assess the efficacy of this method. Hence, residual artifacts in the data must be considered in the analysis. Visual inspection and rejection of the independent components instead of the use of an automated method might have led to better results and may be considered for future work. Since ocular artifacts were quite abundant and distorted a

considerable part of the data, additional processing may be needed in order to achieve an artifact free signal. Various techniques have been developed for removal of ocular artifacts such as eye blinks and eye movement over the past years.[66–69]

The EEG data was acquired with a sampling frequency of 2048 Hz but down sampled to 128 Hz for wireless transmission, although modern EEG acquisition systems usually use a sampling rate of 500 Hz or higher.[70] Besides the low temporal resolution, data quality was also limited by the spatial resolution of only 14 channels, which is lower than the typically used minimum of 32 channels. The low spatial resolution limited the application of certain filtering and source localization techniques that are based on signal interpolation between electrode sites.[43,71–74] Moreover, the low spatial resolution does not give a good sample of the measured electrophysiological signal. Except for the frontal channels, the used system provides only one single channel for the temporal, parietal, or occipital region of each hemisphere. Statistically, this is equivalent to only one sample of each of these regions and might not give a reliable measure of the underlying neural activity.

2.6.3.2 Study design and Statistics

The findings of the present study are based on a very small number of trials, only four trials or less during each phase. Since EEG is very sensitive to variations in the experimental setup and the subject's cognitive and emotional state during the trial, the small sample size may play a significant role in the outcome of the analysis. Cohen[75] recommends 50 trials or more for one condition in order to guarantee replication standards for time-frequency analysis of cognitive electrophysiological data. Such a high number of trials may not be feasible for the target population in this study, however, 4 trials are objectively insufficient.

In respect of the single-subject A-B-A-B design, however, an increase of the number of trials, would require to either shorten the inter-trial intervals or prolong the phases. Shorter inter-trial intervals could mean a great strain to the subjects and their parents or guardians, while extension of the phases could bring about other problems such as undesired changes in EEG activity due to the plasticity of the developing brain. Therefore, the single-subject A-B-A-B design must be further assessed in terms of its suitability for the analysis of electrophysiological data.

2.6.3.3 EEG Analysis

One crucial factor of EEG data analysis is the identification of the boundaries of the different brain rhythms, which show considerable inter- and intra-individual differences, especially related to age.[26,41,14] Little is known about the characteristics of the EEG spectrum in children, which makes identification of individual brain rhythms in each subject indispensable before any inferences can be drawn. The alpha peak frequency is typically used as a reference for the alpha rhythm in adults, which requires the subject to be in relaxed wakefulness while the eyes are closed;[27] a state that cannot possibly demanded from infants and very young children. However, Orekhova et al.[14] showed that social stimulation and manipulation of toys can be reliably associated with the theta rhythm. It may be good practice to incorporate this knowledge into the study design and use the theta rhythm as reference for the interpretation of the EEG spectrum in children. Consequently, the application of the conventional bands has no value for EEG spectrum analysis of children, and the use of narrow frequency bands is recommended.

2.6.4 Future Work

In conclusion, we found that power mobility training does not have any observable effect on the EEG spectrum in the within the four month of data collection, but may have an impact in the short term. The observed changes during the intervention phases suggest that power mobility

training may bring about enhancement of cognitive processes. However, further research is necessary in order to verify these findings. The present study was considerably limited by various technical problems and the study design used. For future work, improvement of the EEG equipment and pre-processing techniques as well as revision of the study design are indispensable. Higher temporal and spatial resolution of the data acquisition and more reliable and controlled artifact removal must be achieved. The applied single-subject design was found to be inappropriate for an EEG study due to the low number of subjects and trials, and thus, it is highly recommended to revise the current design. An inclusive study design might be able to consider the special characteristics of EEG data as well as the challenges that come along with the needs and limitations of the target population. Also, adjustments of the study design may allow the exclusion of influencing factors which currently hinder conclusive interpretation of the results.

Although the results of the present study does not support the hypothesis that EEG can be used as an objective measure for the impact of power mobility training on child development, our findings suggest that there are spectral changes with power mobility training that cannot be accounted for in the literature. The use of better equipment and an appropriate study design may elucidate these spectral changes.

3 EXTENDED REVIEW OF LITERATURE AND EXTENDED METHODOLOGY

3.1 EXTENDED LITERATURE REVIEW

3.1.1 The Brain and its Functions

The brain is composed of three parts: the cerebrum, the cerebellum, and the brainstem.[14] Functions such as coordinating movements, maintaining posture, and balance are processed in the cerebellum, which is located under the cerebrum.[15] The brainstem consisting of the midbrain, pons, and medulla, acts as a relay center connecting the cerebrum and cerebellum to the spinal cord and performs many automatic body functions such as breathing, heart rate, body temperature, wake and sleep cycles, and digestion.[15]

The cerebrum, the largest and most important part of the brain, comprises the left and the right hemisphere and performs higher functions like interpreting touch, vision and hearing, speech, reasoning, emotions, memory, and fine control of movement.[14] The surface of the cerebrum, the so called cortex, is folded into ridges, so called gyri, and grooves called sulci or fissures. The two hemispheres are divided into four lobes (frontal, temporal, parietal, and occipital), of which the frontal lobe is the largest and most anterior. The frontal lobe accommodates the part of the motor system and is involved in the production of language, motivation, comportment, and executive function. The temporal lobes bring about comprehension of language, promote memory and emotion, and receive primary auditory input, while the parietal lobes receive tactile input, and are involved in visuospatial processing as well as reading and calculation skills. The most posterior and smallest lobe is the occipital lobe, which receives primary visual input, and processes visual perception.[14]

Table 2. Brain regions and their functions.[14]

Lobe	*Function*
Frontal	Motor system
	Language production (left)
	Motor prosody (right)
	Comportment
	Executive function
	Motivation
Temporal	Audition
	Language comprehension (left)
	Sensory prosody (right)
	Memory
	Emotion
Parietal	Tactile sensation
	Visuospatial function (right)
	Reading (left)
	Calculation (left)
Occipital	Vision
	Visual perception

The cortex contains about 70% of the 100 billion nerve cells, called neurons, in the human brain, which form the grey matter.[15] Beneath the cortex and between the hemisphere are about 300 billion axons connecting to these neurons, which allow interhemispheric exchange of information and communication to structures found deep in the brain.[14,15] The next section explains the structure of neurons and the physiological processes, which allow neurons to interconnect and pass information packages to each other.

3.1.2 Neurons and Brain Signals

A neuron consists of a cell body, a dendrite, an axon, and a synapse.[16] The cell body contains organelles that are responsible for the maintenance of the cell structure and cell function, such as the nucleus. On one end of the cell body (soma), the dendrites are located, which increase the receptive surface of the neuron. The axon emerges from the opposite side and terminates in the synapse, which connects the neuron to the dendrite of another neuron.[77]

The main function of a neuron is to receive neural information with the receptors of the dendrites and transmit it through the axon to the synapse.[78] Neural information is encoded in electrical impulses, so called action potentials, which is a rapid change in the membrane potential of a neuron. The magnitude of the resting potential of a neuron depends on the differences in specific ion concentrations in the intracellular and extracellular fluid, as well as on the differences in membrane permeability to the different ions. The ions which contribute most to the resting potential are sodium (Na^+) and potassium (K^+). The sodium and potassium concentrations in the intracellular and extracellular fluid define a resting potential of -70mV. Voltage-gated channels inlayed in the membrane of neuron allow the change from the resting potential to an action potential. The action potential is induced by an initial depolarization in the dendrites, which stimulates the opening of voltage-gated Na^+ channels. The influx of sodium ions into the nerve cell leads to an increase of the membrane potential. As soon as a critical threshold potential is reached, a positive feedback loop is induced, which causes a rapid depolarization of the membrane potential such that the membrane becomes positive on the inside and negative on the outside. The cycle of the positive feedback loop is broken by the inactivation of the Na^+ channels. At this point the action potential reaches its peak at +30mV and the influx of potassium through voltage-gated K^+ channels, which have opened in the meantime, brings the membrane potential back to its resting potential. This process is called repolarization and it is terminated by the closing of the voltage-gated K^+ channels through a negative feedback. The nerve cells require a so called refractory period of approximately 2 ms before a stimulus can generate another action potential. The action potential in the initial segment of the neuron depolarizes the adjacent region of the membrane of the axon, which brings the membrane to the threshold potential and induces another action potential. This causes the action potential to travel from the dendrite through the

axon to the axon terminal. The movement of the action potential through a brain cell (neuron) produces a current, which generates a magnetic field measurable by magnetoencephalography (MEG) and a secondary electrical field over the scalp measurable by electroencephalogram (EEG).[78]

3.1.3 Electroencephalography (EEG)

EEG is the record of electrical activity generated in the brain and is nowadays a commonly used tool in neuroscience to study cognitive processes.[79] An EEG acquisition system consists of multiple electrodes attached to the head, which pick up the electrical signals of the neurons in the brain and transmit them to a machine that records those signals as the EEG.[78] In many studies the electrodes are positioned according to the international 10-20 electrode setting.[47] Whereas the original 10-20 electrode setting considers 21 electrodes, other systems are used for higher electrode density, such as the 10-10 system for 74 electrodes and the 10-5 system for up to 345 electrodes.[36]

Since the current of a single neuron is too weak to be measured with EEG or MEG, the detected signal rather represents the synchronous activity of a neuronal population.[36] The location of the firing neurons is crucial for the interpretation of these signals.[80] However, the estimation of these locations is a basic problem in EEG and MEG. The estimation of neuronal sources based on the distribution of electrical potentials or magnetic fields recorded on the scalp is called the inverse problem.[79] Since there is not a unique solution for this problem, several methods are used to estimate the location of the current source.[74,81–85] One simple way to circumvent the inverse problem is using a Surface Laplacian filter.[71] This method interpolates the distribution of electrical activity between the measured potentials and estimates the current source density by

differentiation.[71] An advantage of surface Laplacian filtering is the effect of noise reduction in the processed EEG data.[73]

3.1.4 Analysis of Brain Activity

3.1.4.1 Event-related and spontaneous brain activity

Brain activity can be studied with a variety of experimental approaches, which can be divided in event-related and continuous approaches.[30] Event-related experiments typically study changes in activity, which are induced by events such as sensory stimuli.[30] Studied exclusively in the time domain, these changes are called event-related potentials (ERPs) and are assumed to occur with a relatively fixed time-delay to the stimulus.[31] Averaging methods are used in order to detect the ERPs and enhance the signal-to-noise ratio.[87] Additionally, stimuli can also evoke a reorganization of the phases of the ongoing EEG signal, which require frequency analysis for their extraction.[31] These changes in the frequency domain become manifest in decrease and increase of power in given frequency bands, which can be measured by computing the event-related spectral perturbations (ERSPs).[88]

In contrast to event-related experiments, continuous experiments study changes in brain activity which are not time-locked.[86,89] There are several reasons to choose such an experiment design. Specific tasks or controlled stimuli might be absent, which leads to the investigation of the "resting-state" of the brain.[30] The identification of a general resting-state has been subject of numerous studies in the past.[83,90–92] Another reason for neglecting the time domain might be the investigation of one or more continuous tasks that can last over a longer time period.[30] The time domain may also be irrelevant for studies that use repeated short stimuli, which entrain brain oscillations.[30] The analysis of continuous paradigms typically tries to identify dominant patterns in the frequency domain or connectivity between brain regions.[30]

Spontaneous brain activity can reveal important properties of the current brain state such as attention, alertness, drowsiness, etc..[93] For example, a previous study showed that the power spectrum of EEG data recorded with only two electrodes provides information about the level of alertness, whereas the power increases with decreasing alertness.[94] Spontaneous brain activity at rest with eyes closed and/or eyes open can be considered as resting state and can be used as baseline for investigating changes in brain activity.[42]

3.1.4.2 EEG Spectrum Analysis

EEG signals have a fairly wide frequency spectrum which ranges from 0.5 Hz to 40 Hz or higher.[38] This range can be split in five different frequency bands, the so called delta (0.5-4Hz), theta (4-7.5Hz), alpha (8-13Hz), beta (14-26Hz), and gamma (> 30Hz) rhythms. Lower frequencies indicate the less responsive states, whereas higher frequencies indicate increased alertness. Thus, the delta rhythm is primarily associated with deep sleep, but can also be present during waking state. Delta waves are difficult to distinguish to from myogenic artifacts originating in the activation of neck and jaw muscles. The theta rhythm is presumed of thalamic origin and appears during the transition from drowsiness to consciousness. Theta oscillations are typically associated with access to unconscious material, creative inspiration, and deep mediation, while they also seem to be related to level of arousal. Furthermore, theta waves play an important role in infancy and childhood, whereas predominant appearance in adults is considered as pathological. Alpha oscillations appear in the posterior half of the brain, mainly in occipital regions. The alpha wave is the most prominent rhythm and has been thought of an indicator for relaxed awareness without any attention or for concentration.[38] The alpha rhythm is of special interest for various reasons which is why the next section will discuss its special meaning more in detail. The beta rhythm, typically found in frontal and central regions, indicates

usual waking state.[38] It is associated with active thinking, active attention, focus on the outside world, or solving concrete problems. Gamma oscillation rarely occur and are of very low amplitude, but have been proved to indicate event-related synchronization of the brain.[38]

In general, the amplitude decreases with increasing frequency, i.e. for example the alpha rhythm has a higher amplitude than the beta or gamma rhythm[34]. This is also valid within the frequency band such that the lower alpha component has a higher amplitude than the higher alpha component[38].

Table 3. The five distinguished brain rhythms identifying different states of consciousness and their properties.[38]

	Delta	*Theta*	*Alpha*	*Beta*	*Gamma*
Frequency	0.5-4 Hz	4-7.5 Hz	8-13 Hz	14-26 Hz	> 30 Hz
State of Consciousness	Deep Sleep	Consciousness / Drowsiness	Relaxed awareness without attention or concentration	Waking state	Waking state
Association	Sleep	Access to unconscious material, creative inspiration, deep meditation	Is eliminated by opening eyes, hearing unfamiliar sounds, anxiety, mental concentration or attention	Active thinking, active attention, focus on the outside world, problem solving	Active thinking, active attention, focus on the outside world, problem solving
Brain Region		Frontal regions	Posterior half of the head, occipital region	Frontal and central regions	Frontal and central regions
Other Factors	Likely to be confused with artefact signals due to muscle activity	Important role in infancy and childhood, may be pathological in adults	most prominent rhythm	Found in normal adults, can be blocked by motor activity or tactile stimulation	Rare occurrence, confirmation of certain brain diseases, well-localized activity

3.1.4.3 Special Meaning of Alpha Rhythm

The most prominent rhythm is the alpha rhythm, which occurs at 8-13 Hz during wakefulness, best seen with eyes closed and during physical relaxation and relative mental inactivity in the occipital cortex.[34] When the eyes are opened, the amplitude of the alpha peak observed with closed eyes, is reduced, which is known as "Berger effect" or reaction of activation.[95] It has been acknowledged that the alpha rhythm reflects an idling state of primary cortical areas.[39] This results in a decrease of alpha power when engaging in a task such as perceptual judgment or increased attentiveness.[41,42] There are several different "alpha rhythms" associated with different meanings within the alpha band.[40] Alpha rhythms related to motor control are called mu rhythms of which two types can be distinguished.[40] Both are blocked before and during movement and occur either at lower frequency (8-10 Hz) in a widespread manner, or at higher frequency (10-13 Hz) showing more focused patterns.[40] The former is associated with unspecific movements, i.e. the responses to finger and foot movements are similar, while the latter is specific and different for different limbs.[40] Lower alpha desynchronization is also observed in response to various non-task-specific factors and may be understood as general "attention".[40] Studies[39,43,44] indicate involvement of alpha power also in cognitive processes and memory. The observation of positive correlation of alpha power and awareness leads to the theory that non-essential processing is inhibited in order to facilitate performance of the actual task or support working memory processes.[39]

Although the alpha band typically ranges from 8 to 13 Hz, differences between subjects should be considered.[41,60] For example, whereas the individual alpha peak frequency increases from early childhood to puberty, it is decreasing again with increasing age.[60]

3.1.4.4 Brain Activity and Brain Performance

Klimesch has examined and discussed the alpha rhythm and its functional meaning thoroughly in the last decade.[12,26,39,96] His findings identify a reciprocal relationship between alpha and the lower theta activity in the context of cognitive and memory performance.[26]

Two types of EEG phenomena were linked to good performance whereas a tonic increase in alpha and a decrease in theta power and a large phasic (event-related) decrease in alpha and increase in theta were observed. Also, a decrease in alpha power and simultaneous increase of theta power was noticed during the transition from waking to sleeping state. Klimesch suggests that the power of the alpha rhythm is positively related to cognitive performance and brain maturity, whereas theta power is negatively related. However, during actual task demands alpha power is negatively related to cognitive (and memory) performance, whereas again the opposite holds true for the theta range. Klimesch also mentions that alpha reactivity can be an indicator for performance. Alpha or theta reactivity is usually thought as the comparison of a resting period (eyes open or closed) with a test period.[26] Klimesch also references studies that showed that learning disabled children and Alzheimer patients showed less task-related alpha attenuation than the control group.[96]

Mathewson et al.[97] demonstrated a correlation between resting alpha power and video game learning rate. The higher the individual alpha power before the training, the steeper the improvement in score during the training. The same study showed also that in addition to resting alpha oscillations, alpha and delta ERSPs can be used as predictors for the amount of skill improvement in tasks.[97]

Orekhova et al.[14] who examined the involvement of theta in cognitive and emotional processes of infants and preschool children supporting the previous findings by Klimesch[26]. The observed

increase in theta power and simultaneous decrease in high alpha power in both age groups led to the conclusion that theta oscillation are strongly related to behavioral states with considerable attentional and emotional load and may reflect engagement of different brain networks in control and behavior.[14]

3.1.5 Signal Processing of EEG Data

3.1.5.1 Artifacts

There is a variety of different factors, which might contaminate or imitate EEG recording, and might lead to misinterpretation of the underlying brain activity. There are anatomical and physiological factors such as the conductivity of the head tissue (bone, brain, skin, etc.), cerebral blood flow, hormones, and muscle activity which influence EEG recordings.[55,98,99] Also, topographical factors such as montage choice of the electrodes and the spatial resolution can influence subsequent analysis of the EEG data.[41,100] However, the major sources of contamination of EEG data are muscular or electromyogram (EMG) artifacts, and electrooculogram (EOG) artifacts because they cause a change in the electric fields over the scalp.[55,69] EMG of skeletal muscle, which is recorded from the skin, has a broad frequency distribution reaching from 0 Hz to over more than 200 Hz with distinct frequency components at 0-5 Hz, and at 10 Hz.[55] Facial muscle shows a broad frequency response with two peaks at 20-30 Hz and 35-60 Hz. Due to its broad frequency distribution EMG contaminates alpha, beta, and delta rhythms.[55] Eye blinks and eye movement are expected to cancel by averaging methods, but they will increase the variability in the averaged data.[101] It is also known that EOG artifacts increase the spectral power of EEG data, particularly for the slow bands delta and theta.[67] There exist several less or more sophisticated methods to detect contaminated EEG data or remove the artifacts from the data.[100,102–104] While in the past, artifacts were identified by eye and

then manually rejected, nowadays more and more automated methods are available to detect artifacts in EEG data more efficiently.[105–107] One simple way to do this is to measure the amplitude within a certain range of data points.[56] For the EEG, amplitudes above 100 µV are considered as noise resulting from movement and electrode artifacts, which can be detected and removed by simple thresholding.[55] The disadvantage of this method is that it results in data loss of the affected region or channel.[55]

The employment of spectral filtering can be very useful when the artifact manifests itself in a specific frequency range, but this is only recommended when the contaminated frequency range is narrow since it will attenuate also EEG activity in this range.[108] Spatial and temporal filtering has been used successfully for EEG-based communication, but this method is only applicable in situations of voluntary motor control by the subject.[109]

Jung et al.[104] demonstrated that Independent Component Analysis (ICA) can detect EMG contamination effectively. ICA identifies the components of a signal, which are statistically independent of each other, such that artifacts are represented as a separate component of the signal.[110] The component which is associated with muscle activity such as eye blinks can be completely removed or processed before the signal is composed again to a signal without those artifacts.[52,110–112] This method has been expanded and improved in various ways over the last decade combining ICA with other methods such as wavelet transform and blind source separation.[51,104,106]

3.1.5.2 Transformation to Frequency Domain

Various methods are available in order to transform the recorded EEG data to frequency domain for spectral analysis.[86] The most common are non-parametric methods such as Fourier transform,

wavelet transform, and Hilbert transform. Parametric spectral estimation is typically based on autoregressive modelling such as the Yule-Walker method.[86]

The Fourier Transform typically uses the short-time Fourier method, which computes the power spectrum considering shorter overlapping time windows.[86] Tapering, such as the use of a 'Hanning' window, is used in order to reduce leakage. The absolute value of the obtained complex frequency spectrum can be averaged over several trials for noise reduction. The disadvantage of the Fourier transform is the dependency of frequency resolution on time resolution, which is a reciprocal relation, i.e. higher time resolution results in lower frequency resolution. The use of wavelet functions can address this problem by constructing the oscillatory basis of the transform from a prototypical 'mother wavelet'. The most commonly used wavelet function is the Morlet or Gabor wavelet. Since the wavelet length decreases for higher frequencies, the wavelet spectral estimates have better temporal resolution for higher frequencies.[86,113,114] Another useful tool for spectral analysis is the Hilbert transform which allows computation of instantaneous phase and amplitude.[86] However, band-pass filtering is required since the phase and amplitude estimates are only interpretable for narrow-band signals. Combined with Empirical Mode Decomposition (EMD), the so called Hilbert-Huang transform, high spectral resolution for arbitrary frequencies is obtained. For EEG however, clinical mode decomposition (CMD) is more useful since the spectrum is decomposed in components corresponding to partitions of the clinical spectrum. HT and CMD allow high temporal resolution of rapid changes in frequency, phase, and amplitude of a signal.[86,115]

3.2 EXTENDED METHODOLOGY

3.2.1 Data Acquisition System

The EEG was recorded with a modified wireless Emotiv EPOC© (Emotiv, Inc., San Francisco, CA) neuroheadset. The original Emotiv EPOC© system consists of a headset with 14 electrodes with moistened felt pads for good connection to the scalp and an integrated acquisition unit containing amplifier, pre-processer, and Bluethooth® transmitter. Internally, the data are sampled at 2048Hz and notch filtered at 50 Hz and 60 Hz using a built in digital 5th order Sinc filter with a resulting signal bandwidth is 0.2-45 Hz. The data is streamed with a sampling rate of 128 Hz to a USB receiver via Bluethooth® SMART 4.0 LE protocol. The data were acquired in TestBench™ software (Emotiv, Inc., San Francisco, CA).

Figure 3-1. Original Emotiv EPOC© neuroheadset with 14 electrodes and acquisition unit.[116]

The Emotiv EPOC© neuroheadset was tested beforehand with three children of about 3 years of age with multiple severe disabilities. All three participants were wheelchair users. Since the Emotiv EPOC© neuroheadset is designed for adolescents and adults, the size of headset was not appropriate for the head circumferences of the participating children. The headset was too lose

which caused uncontrolled shifting and moving of the electrodes. Additionally, the arc, which enclosed the wireless acquisition unit, was disturbing for the children due to its position at the backside of the head. In order to overcome these problems, the Emotiv EPOC© neuroheadset was modified for the present study. The 14 electrodes and the two reference electrodes of the original system were replaced with gold cup disc electrodes mounted on a custom sized EasyCap (EASYCAP GmbH, Herrsching, Germany), which was delivered with pre-cut holes for the electrode placement. The electrode placement of the original headset (AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4 referenced to CMS (P3) and DRM (P4)) according to the International 10-20 system was retained.[47] Plastic adapters for holding the electrodes in position were designed and 3D-printed at Grand Valley State University. The electrode adapters consist of a bottom part and a lid. A groove in the bottom part allows the attachment to the cap, while a slot holds the electrode. The electrode is kept in place by a lid that is fixated through a screw mechanism.

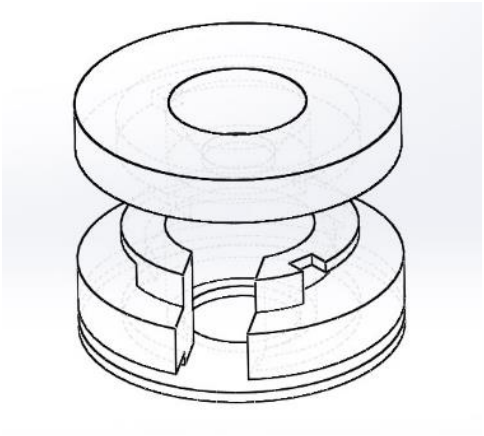


Figure 3-2. Solidworks drawing of the designed electrode adapter.

The wireless acquisition unit consisting of amplifier, pre-processer, wireless transmitter, and battery, was removed from the original frame and placed into a plastic enclosure. The enclosure

was also designed and 3D-printed at Grand Valley State University. It provides three slots, one for the power button, one for the USB connector, and one for the 25-pin D-sub connector for the electrode bundle.

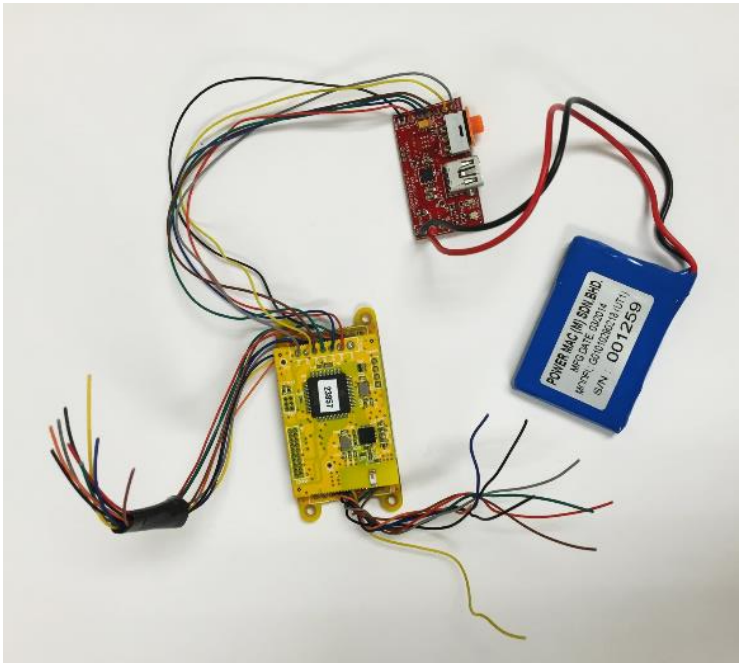


Figure 3-3. The wireless acquisition unit of the Emotiv EPOC© neuroheadset with amplifier, pre-processer, transmitter, and battery.

Figure 3-4. Plastic enclosure containing the wireless acquisition unit. A 25-pin D-sub connector allows the connection to the gold cup electrodes. The black USB dongle serves as the receiver of the streamed data.

Figure 3-5. The 16 gold cup electrodes with 25-pin D-sub connector to the wireless acquisition unit. The prototype version of the electrode adapter is also shown.

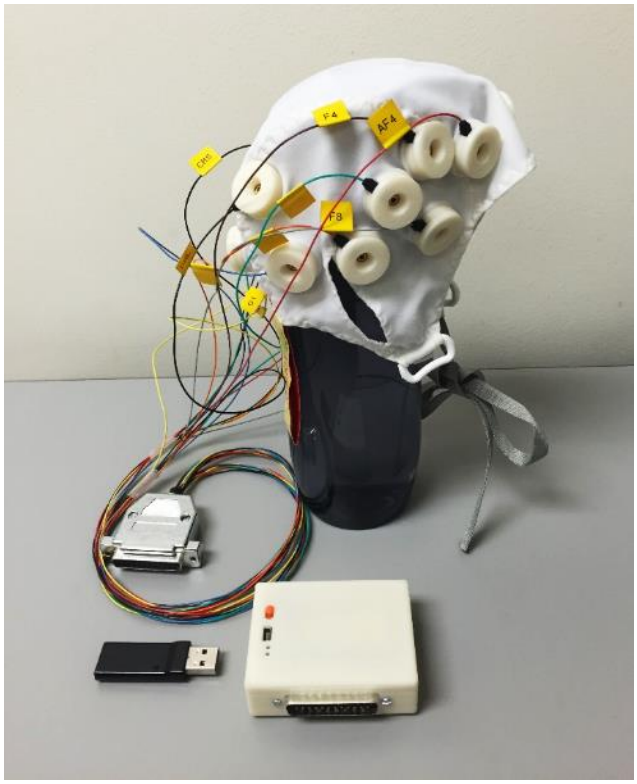


Figure 3-6. The final version of the modified Emotiv EPOC© system showing the cap with the mounted electrodes, the wireless acquisition unit and USB receiver.

3.2.2 Artifact Removal

Urigüen et al.[117] discusses the most recent and most commonly used methods for artifact removal from EEG data. In conclusion of the extensive review, independent component analysis (ICA) is regarded as the safest approach to correct EEG recordings without prior knowledge of the EEG signals and its contamination.[75] ICA methods attempt to find a linear representation of non-Gaussian data so that the resulting components are as independent as possible.[110] ICA was originally proposed for blind source separation in order to recover independent source signals **s** (e.g. different speakers, different EEG channels, noise sources, etc.) after they are mixed by an unknown matrix **A**.[104] This can be expressed as **x = As**, where **x** are *N* mixtures (e.g. recordings of voices, EEG recordings, etc.) of the original *N* sources. Except the *N* different mixtures **x**, nothing is known about the sources or the mixing process. The goal of ICA is to solve the equation **x = As** for **s** by finding the unmixing matrix **W**.[104] The recovered sources are known as independent components.

In the last two decades, various algorithms were developed to solve this problem, while nowadays the most commonly used are the second-order blind identification (SOBI[118]), (extended) InfoMax[119], fastICA[120], and adaptive mixture of independent component analyzers (AMICA[121]).[117]

Regarding EEG recordings, the application of ICA is used to separate the multichannel EEG signal from artefactual signal.[51] Typically, ICA components are inspected visually and the artefactual components are removed manually by zeroing those components. Since this procedure is very time consuming, a lot of research has been done with the goal to automate the removal of the contaminated components.[107,122–125] Nonetheless, none of them has gained

acceptance and the manual rejection of artefactual components is still the most common technique.

For the present study, two different approaches for an automatic selection of the relevant independent components were pursuit and evaluated. The first approach is based on a practice used in principle component analysis (PCA[126]). In contrary to ICA, PCA attempts to reduce the dimensionality of a data set by finding the most uncorrelated representation consisting of the principle component. By selecting only the first few components, which contain most of the information in the data, the dimensionality of the original data can be considerably reduced. A common way to identify these components, is to select a cumulative percentage, e.g. 95%, of total variation which one desires that the selected principle components contribute.[126] We used a similar method in order to identify the independent components containing most of the neural signal. However, assuming that artefacts have a higher variability than the neural signal, we considered the components that add least to the total variance of the data. Thus, the variance of each component is measured using the mixing matrix **W** (inverse of unmixing matrix) and the original data **X**.

$$\sigma_k{}^2 = \frac{\sum(w_k)^2(\sum x_k)^2}{NT-1}, \quad for\ k = 1,2,\dots N \tag{3-1}$$

$\sigma_k{}^2$ is the mean variance of the kth component and w_k is the kth column of the mixing matrix **W**, while x_k denotes the kth channel of data **X**. *N* is the number of channels, while *T* represents the number samples.

Sorted from highest to lowest mean variance, the components with a cumulative percentage of 95% of the total variance are zeroed before back-projection. Hence, only the components which contribute less than 5% of the total variance are retained.

For the second approach, a method proposed by Castellanos and Makarov[51] was used, which recovers EEG signals by using wavelet transform for artifact detection. The proposed wavelet enhanced ICA (wICA) has the advantage to be fully automated and preserves also EEG signals leaking into components which are considered as artefactual. The method makes use of the differences in temporal and spectral properties between EEG signals and artifacts. While EEG signals are of low amplitude and have a broad band frequency spectrum, artifacts are of high magnitude (power) and are localized in the time and/or frequency domain. In order to identify these features, Castellanos and Makarov use discrete wavelet transform of the independent components. Expressed in mathematical form the wavelet transform would be

$$W^s(d,b) = \frac{1}{\sqrt{d}} \int s(t)\psi_{d,b}(t)dt, \quad \psi_{d,b} = \psi\left(\frac{t-b}{d}\right), \tag{3-2}$$

where $W^s(d,b)$ is the wavelet representation of the independent component $s(t)$ and ψ is the mother wavelet with discrete values b and d defining the time localization and scale. $W^s(d,b)$ is assumed to be the sum of artefactual and neural signal which can be expressed as

$$W^s(d,b) = W^n(d,b) + W^a(d,b)\,, \tag{3-3}$$

where $W^n(d,b)$ and $W^a(d,b)$ are the wavelet coefficients of neural and artefactual parts of the independent component.

Due to the spectral properties, the neural signal spreads almost homogenously over the whole spectrum of scales and localization, while the artifacts appear with high amplitude in long enough scales and localized in time windows. The neural signal is separated from the artifacts by thresholding and setting parts that exceed the threshold value in amplitude are set to zero. The

selection of the threshold value *K* is a crucial element of the algorithm. For removal of ocular and heartbeat artifacts, a fixed form threshold has provided good performance and is set to

$$K = \sqrt{2\log(N\sigma)}\,, \quad \sigma^2 = \frac{median(|W(d,b)|)}{0.6745}, \tag{3-4}$$

where *N* is the length of the data segment to be processed and σ^2 is the estimator of the magnitude of the neural signal part. After thresholding, the inverse wavelet transform is applied and the cleaned components are composed to the original signal. The wICA algorithm was validated with semi-simulated and real EEG recordings and shows significant improvement of artifact suppression in comparison to the conventional ICA method.[51] The wICA method uses the ICA algorithm InfoMax by default, but Castellano and Makarov[51] state that a different algorithm might be able to achieve better results.

Variations in success for blind source separation depending on the signal content were also observed by Urigüen et al..[117] The review examines the differences and suitability of the most popular algorithms for different types of signal contaminants. For ocular artifacts, muscular artifacts, and cardiac artifacts, SOBI was identified as the most reliable and successful algorithm, while also AMICA showed satisfactory results for muscular and cardiac artifacts. If a combination of different types of artifacts is present or the type of contamination is unknown, AMICA and InfoMax seem to be the best choices, while also SOBI performs successfully.[117]

In order to identify the best algorithm for artifact removal of the EEG data in the present study, different types of ICA algorithms and artifact detection in the independent components were tested. Test subjects were the InfoMax, wavelet enhanced InfoMax with threshold of 1.00 (wIM100), wavelet enhanced InfoMax with threshold of 1.25 (wIM125), the JADE[127] algorithm, wavelet enhanced JADE with threshold of 1.00 (wJ100), wavelet enhanced JADE with threshold

of 1.25 (wJ125), and the SOBI algorithm. The SOBI algorithm and the wavelet enhanced method were not compatible due to the low amplitude of the SOBI components.

Two measures for evaluating the performance of the different algorithms were used as proposed by Sweeney et al.[128] Firstly, the signal-to-noise ratio (SNR) was measured in decibel using the following formula:

$$SNR = 10\ \log_{10}\left(\frac{{\sigma_s}^2}{{\sigma_a}^2}\right) \tag{3-5}$$

where ${\sigma_s}^2$ is the variance of the noise-free (neural) signal and ${\sigma_e}^2$ is the variance of the error (artefactual) signal. The noise-free signal is the estimated desired signal after application of the artifact removal algorithm, while the error signal was obtained by subtracting the noise-free signal from the noisy signal. For comparison, the percentage increase of SNR was computed.

$$SNR_{diff} = \frac{SNR_{after} - SNR_{before}}{SNR_{clean} - SNR_{before}} \cdot 100\% \tag{3-6}$$

The second measurement for performance assessment was the percent correlation increase between artifact-free and estimated signal using following formula:

$$R_{diff} = \frac{R_{after} - R_{before}}{R_{clean} - R_{before}} \cdot 100\% \tag{3-7}$$

R_{before} is the correlation calculated over the epochs of contaminated data, while R_{after} is the correlation of the same data after artifact removal. R_{clean} is the correlation of known artefact-free signal.

In the first test run, 10 different artificial test signals were used. Each test signal consisted of 14 channels, which were divided into two groups, namely, 7 clean channels and 7 noisy channels. The group of noisy channels differed from the groups of clean channels only by the addition of different levels and duration of random noise in order to simulate artefact contamination. The

percentage increase of SNR (SNR_{diff}) and correlation (R_{diff}) were calculated for the seven contaminated channels of each file. The mean was computed across channels and files to obtain an overall score of the method. The analysis revealed that all the different algorithms achieve a minimum score of SNR_{diff} = +20.0 (±9.2) % and R_{diff} = +19.1 (±10.8) %. The SOBI algorithm achieved the highest scores with SNR_{diff} = +69.6 (±5.8) % and R_{diff} = +129.6 (±60.4) %.

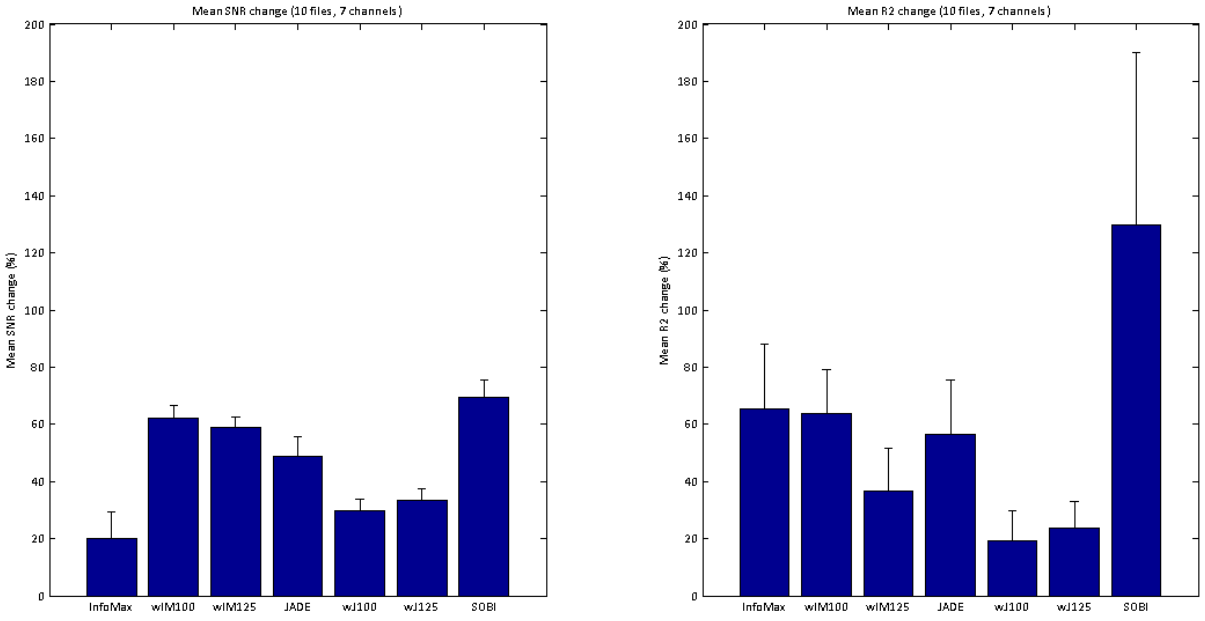


Figure 3-7. Mean percentage increase of SNR and correlation R2 after applying the seven different ICA algorithms to artificial test signal. Ten different files with a 14-channel test signal were tested.

In the second test run, the algorithm were fed with real EEG data which consisted of 6 different files with 14 channels (AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4). The underlying EEG signal was unknown, however, the strongest artifacts were identified in channel O1 and O2 (representing the artefactual signal), while the channels F7, T7, T8, and F8 were the cleanest ones (representing the signal of interest). For the SNR, the algorithms InfoMax, JADE, and SOBI show considerable decrease in SNR, and thus seem to add noise instead of removing it. The two wavelet enhanced algorithms wJ100 and wJ125 achieve the highest increase in SNR

with SNR_{diff} = +19.0 (±7.8) % and SNR_{diff} = +19.0 (±7.9) %. The same algorithms, however, do not perform well regarding the percentage increase of correlation. Only the two wavelet enhanced InfoMax algorithms, wIM100 and wIM125, achieve positive results in SNR as well as correlation. With SNR_{diff} = +12.9 (±11.9) % and R_{diff} = +24.3 (±21.9) %, wIM125 performs slightly better than wIM100 with SNR_{diff} = +5.5 (±8.4) % and R_{diff} = +21.5 (±20.5) %.

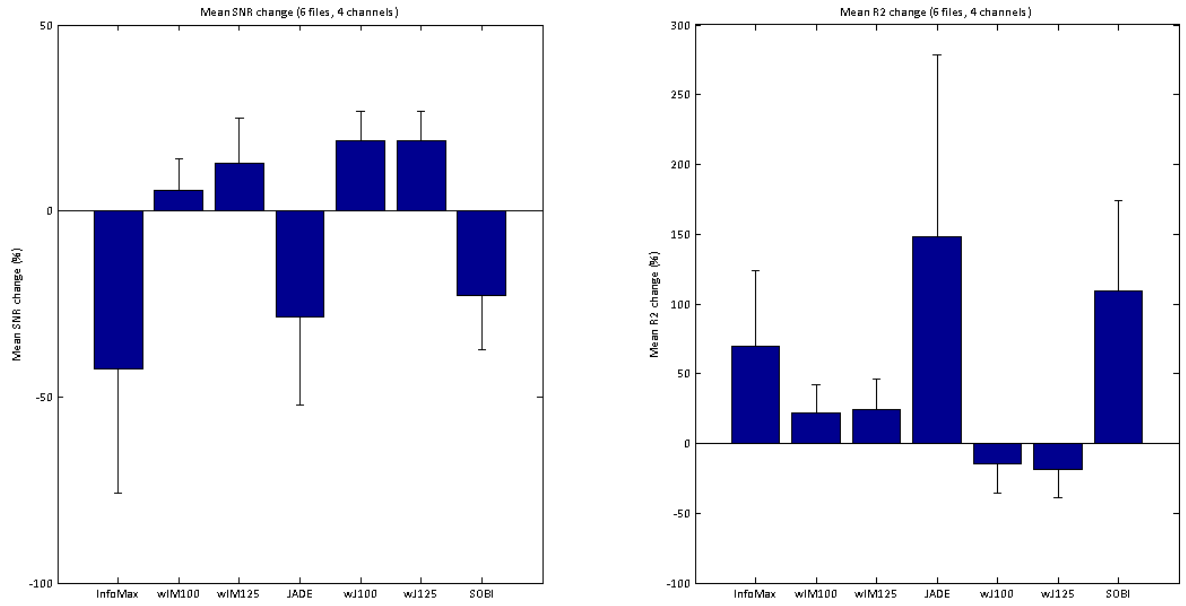


Figure 3-8. Mean percentage increase of SNR and correlation R2 after applying the seven different ICA algorithms to real EEG data. Six different files, each containing real 14-channel EEG data.

Due to these results, the wavelet enhanced InfoMax algorithm wIM125 was considered as the most stable one with good performance on artificial and real data. This was also visually confirmed, comparing the time series after artifact removal.

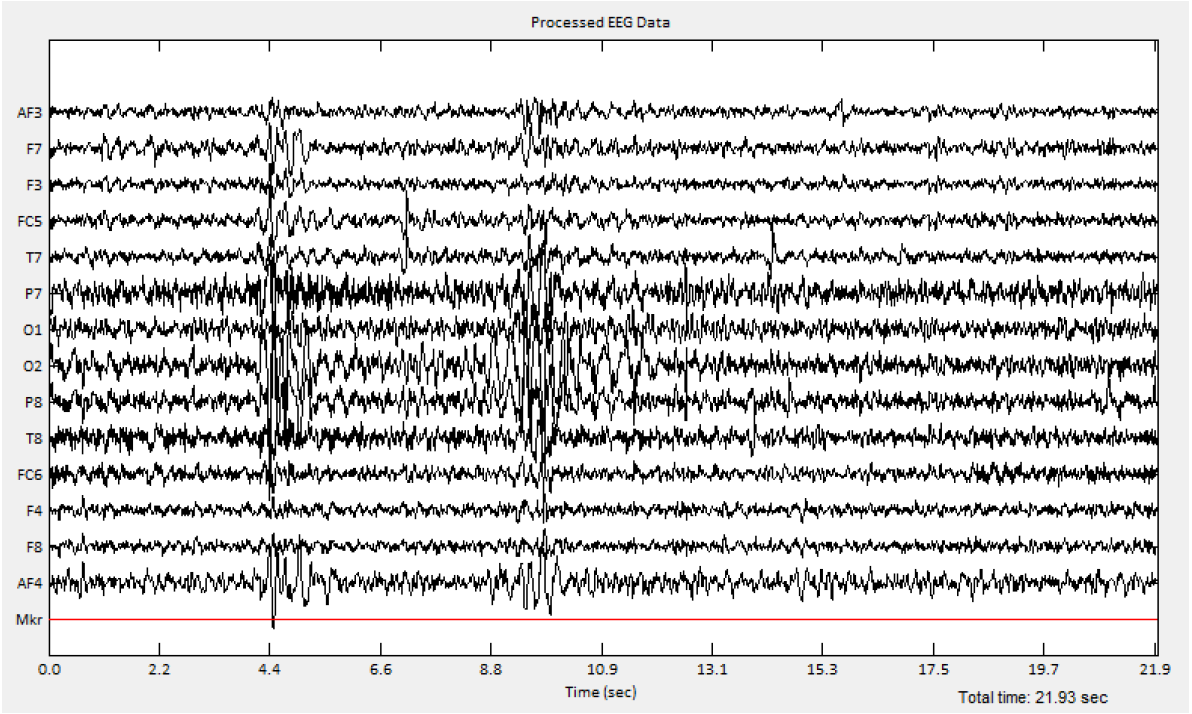


Figure 3-9. An example of real EEG data used for the evaluation of the algorithms.

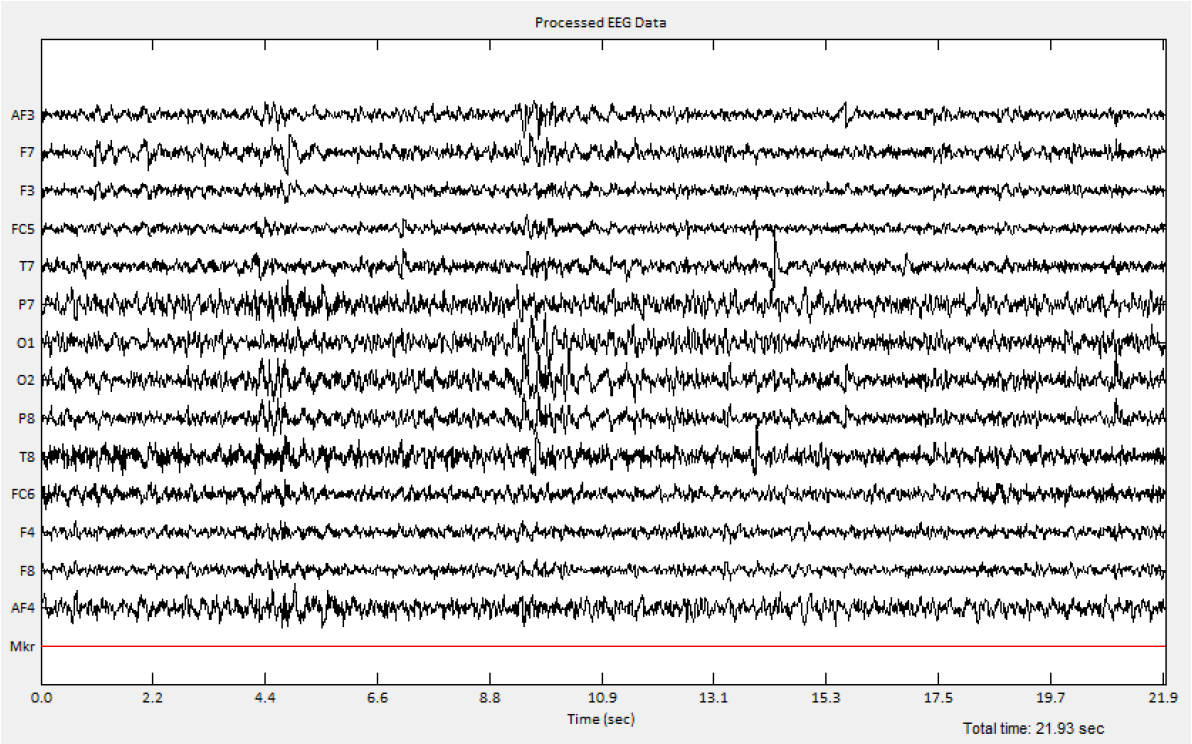


Figure 3-10. The EEG data shown above after applying the JADE algorithm. As already implied by the test measures, the JADE algorithm seems to reduce the artifacts only slightly.

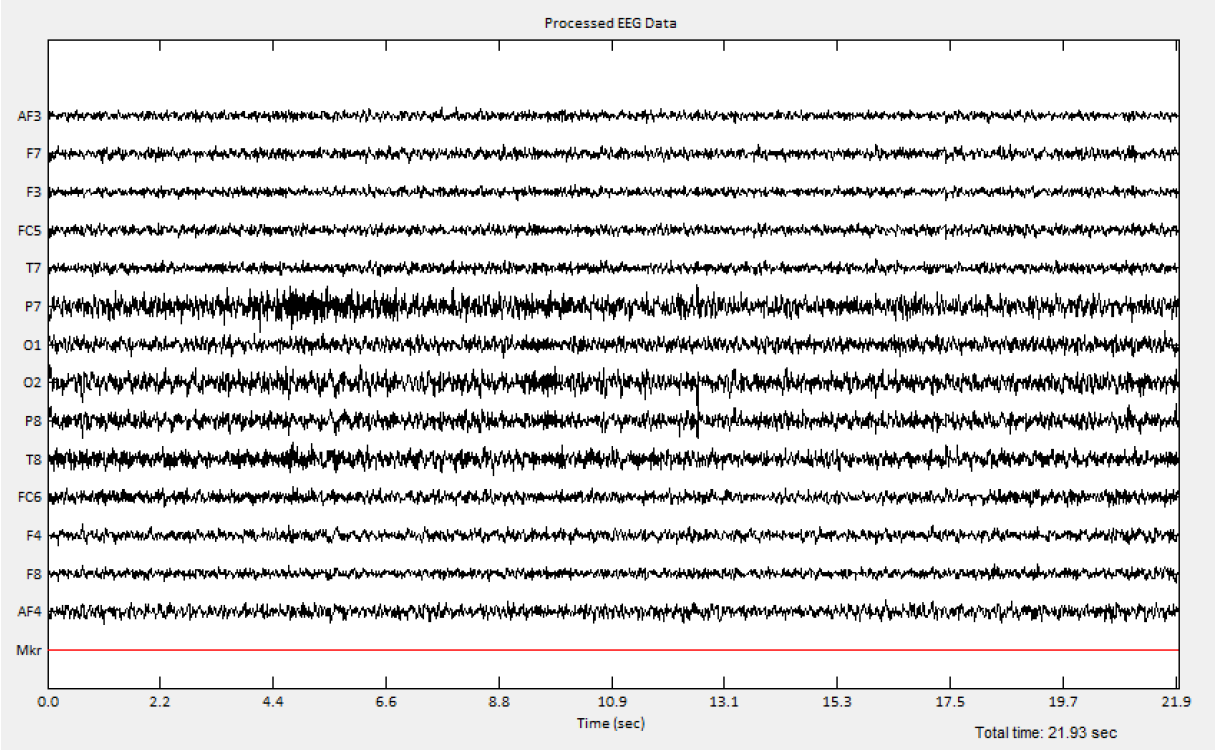


Figure 3-11. The EEG data shown above after applying the wavelet enhanced InfoMax (wIM125) algorithm. wIM125 obviously reduces the artefacts more than the JADE algorithm, which was identified as unstable applying it to real EEG data regarding the test measures.

3.2.3 Spectral Analysis

After artifact removal the recorded EEG signal was Fourier transformed to frequency domain for further analysis using the Welch's method.[54] The Welch's method uses *K* sequences of length *L* with typically an overlap *D* of 50% (i.e. L/2). The periodograms

$$P_k\left(e^{i\omega}\right) = \frac{1}{L}\left|\sum_{l=0}^{L-1} w(l) * x_k(l)e^{-i\omega n}\right|^2 \tag{3-8}$$

of the subsequence x_k of L samples formed with a window function $w(l)$ are averaged resulting in the power spectral density (PSD) with following equation:

$$P_x\left(e^{i\omega}\right) = \frac{1}{KLU}\sum_{k}^{K-1}\left|\sum_{l=0}^{L-1} w(l) * x_k(l + kD)e^{-i\omega n}\right|^2, \tag{3-9}$$

where

$$U = \frac{1}{L}\sum_{l=0}^{L-1} w^2(l). \tag{3-10}$$

Through averaging over multiple periodograms, the variance, and thus the noise, of Welch's PSD estimate is considerably reduced.[54]

4 APPENDIX

4.1 CODE: PREPROCESSING FOR ARTIFACT REMOVAL (GUI)

```
function varargout = EEG_GUI_ArtRem(varargin)
% EEG_GUI_ARTREM MATLAB code for EEG_GUI_ArtRem.fig
%      EEG_GUI_ARTREM, by itself, creates a new EEG_GUI_ARTREM or raises the existing
%      singleton*.
%
%      H = EEG_GUI_ARTREM returns the handle to a new EEG_GUI_ARTREM or the handle to
%      the existing singleton*.
%
%      EEG_GUI_ARTREM('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in EEG_GUI_ARTREM.M with the given input arguments.
%
%      EEG_GUI_ARTREM('Property','Value',...) creates a new EEG_GUI_ARTREM or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before EEG_GUI_ArtRem_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to EEG_GUI_ArtRem_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help EEG_GUI_ArtRem

% Last Modified by GUIDE v2.5 21-Jan-2016 14:06:29

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @EEG_GUI_ArtRem_OpeningFcn, ...
                   'gui_OutputFcn',  @EEG_GUI_ArtRem_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before EEG_GUI_ArtRem is made visible.
function EEG_GUI_ArtRem_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to EEG_GUI_ArtRem (see VARARGIN)

% gabby's head radius is 6.266 cm
```

```
clc;
initializeGUI(handles);

% Choose default command line output for EEG_GUI_ArtRem
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes EEG_GUI_ArtRem wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = EEG_GUI_ArtRem_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes when figure1 is resized.
function figure1_ResizeFcn(hObject, eventdata, handles)
% hObject    handle to figure1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% ============================================================================
% SLIDERS
% ============================================================================

% --- Executes on slider movement.
function slider_display_Callback(hObject, eventdata, handles)
% hObject    handle to slider_display (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of slider

updateGUI(handles);

% --- Executes during object creation, after setting all properties.
function slider_display_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider_display (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% --- Executes on slider movement.
function slider_epoch_Callback(hObject, eventdata, handles)
% hObject    handle to slider_epoch (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of slider

global Fs

% get epoch length from slider
eLength = floor(str2double(get(handles.slider_epoch,'String')));

% set edit value for epoch to epoch length
set(handles.edit_length,'String',sprintf('%.1f',eLength/Fs));
```

```
% refresh GUI
updateGUI(handles);

% --- Executes during object creation, after setting all properties.
function slider_epoch_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider_epoch (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% ========================================================================
% PUSHBUTTONS
% ========================================================================

% --- Executes on button press in pushbutton_load.
function pushbutton_load_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_load (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global rawdata cleandata file icaEEG wicaEEG artdata filename resetdata ...
    icaComp Fs

% indicate loading status
set(handles.text_loading,'String','Loading...');
dataLoaded = 0; % reset flag for loaded data

% load file selected by user
[filename,filepath,filteridx] = uigetfile({'*.edf','EDF-files (*.edf)';...
    '*.csv','CSV-files (*.csv)';...
    '*.mat','MAT-files (*.mat)';},'Select Data File');
file = strcat(filepath,filename); % concatenate file path and name

% load data depending on file format
switch filteridx
    case 0
    case 1 % loading edf file
        % reset raw data
        rawdata = [];

        % read edf file
        rawdata = readedf(file);    % read edf file
        rawdata = rawdata([3:16 36],Fs+1:end-Fs); % extract channels of interest
        % 3-16 eeg channels, 36 marker, remove first and last second from data

        % set flag to data loaded
        dataLoaded = 1;

        % display file name on GUI
        set(handles.text_file,'String',filename);

        % indicate end of loading process
        set(handles.text_loading,'String','Done.');
    case 2 % load csv file
        % reset rawdata
        rawdata = [];

        % read csv file
        rawdata = xlsread(file);
        rawdata = rawdata(:,[3:16 36]); % extract channels of interest
        % 3-16 eeg channels, 36 marker, remove first and last second from data

        % set flag to data loaded
        dataLoaded = 1;
```

```
        % display file name on GUI
        set(handles.text_file,'String',filename);

        % indicate end of loading process
        set(handles.text_loading,'String','Done.');
    case 3
        % reset raw data
        rawdata = [];

        % load variable
        vars = load(file);
        varname = fieldnames(vars); % get variable name
        load(file);

        % read data from variable
        rawdata = eval(char(varname{1}));

        % add marker channel if channel # smaller than 15
        rawdata = rawdata(1:14,:);
        if size(rawdata,1)<15
            n = 15-size(rawdata,1); % calculate missing channels
            rawdata = [rawdata; zeros(n,size(rawdata,2))];
            % add zeros for missing channles
        end

        % set flag to data loaded
        dataLoaded = 1;

        % display file name on GUI
        set(handles.text_file,'String',varname);

        % indicate end of loading process
        set(handles.text_loading,'String','Done.');
end


if (dataLoaded)
    % search for first marker in data
    ind = find(rawdata(15,:)>0,1,'first');
    if isempty(ind) % if no marker set indices to one
        ind = 1;
    end
    % trim data to first marker
    rawdata = rawdata(:,ind:end);

    % get only eeg data
    data = rawdata(1:14,:)';
    % remove DC value
    data = detrend(data,'constant');
    % highpass at 2Hz
    [b,a] = butter(7,[2/64],'high');
    filtdata = filtfilt(b,a,data);

    % average reference
    set(handles.text_process,'String','status: AvgRef processing...');
    avgRefEEG = avgReference(filtdata');

    % add markers
    rawdata = [avgRefEEG; rawdata(15,:)];
    % save data in variables
    cleandata = rawdata;
    resetdata = cleandata; % save this data as reset data
    % reset variables
    artdata = nan(size(rawdata));
    icaEEG = zeros(size(rawdata));
    wicaEEG = zeros(size(rawdata));

    % indicate statuse on GUI
    set(handles.text_process,'String','status: AvgRef done.');
```

```
    % switch to clean data view
    set(handles.popupmenu_view,'Value',2);

    % refresh GUI
    updateGUI(handles);
else
    % if no data loaded just refresh GUI with current data
    updateGUI(handles);
end

% --- Executes on button press in pushbutton_video.
function pushbutton_video_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_video (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global vidFile vidLoaded videoIdx

% ask user for directory of video file
[filename,filepath,filteridx] = uigetfile({'*.avi','AVI' },'Select Data File');

% concatenate file path and file name
file = strcat(filepath,filename);

% read video file
vidFile = VideoReader(file);

% get number of frames
nFrames = vidFile.NumberOfFrames;

% set video frame index
videoIdx = 1:nFrames;

% set flag that video loaded
vidLoaded = 1;

% refresh GUI
updateGUI(handles);

% --- Executes on button press in pushbutton_save.
function pushbutton_save_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_save (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global cleandata filedir comp icaComp Fs wlength

% ask user for directory
filedirnew = uigetdir();

% save path in global variable
if(~isempty(filedirnew))
    filedir = filedirnew;
end

% check if folder for spectra exists, create one if  not
if ~exist([filedir '\spectra'],'dir')
    mkdir(filedir, 'spectra');
end

% check if folder for data exists, create one if  not
if ~exist([filedir '\data'],'dir')
    mkdir(filedir, 'data');
end

% check if folder for components exists, create one if  not
if ~exist([filedir '\components'],'dir')
    mkdir(filedir, 'components');
end
```

```
% calculate optimal FFT length
NFFT = 2^nextpow2(wlength+1);

% initalize frequency vector
freq=[];

% get clean data for saving in time domain
savedata = cleandata;

% initalize variable for fft spectrum
savedata_fft=[];

% comput power spectral density with Welch's method, sliding window with
% length of wlength and 50% overlap
for c = 1:size(savedata,1)-1
    [savedata_fft(c,:),freq] = pwelch(savedata(c,:),wlength,wlength/2,NFFT,Fs);
end
% get onesided spectrum
savedata_fft = savedata_fft(:,1:NFFT/2+1);
% normalize spectrum to total mean power
normval = mean(sum(savedata_fft,2));
savedata_fft = savedata_fft/normval;

% get file name for saving
filenameSave = get(handles.edit_id,'String');

% save timeseries with file name
filedata = [filedir '\data\' filenameSave '_data'];
[pathstr,namedata,ext] = fileparts(filedata);
subsetdata = genvarname(namedata);
eval([subsetdata '= savedata;'])
save(filedata,namedata);

% save PSD with file name
filespec = [filedir '\spectra\' filenameSave '_spec'];
[pathstr,namespec,ext] = fileparts(filespec);
subsetspec = genvarname(namespec);
eval([subsetspec '= savedata_fft;'])
save(filespec,namespec);


% plot spectra for control purposes
figure
plotPSD(savedata_fft,freq)
title('Power Spectrum Density')

% compute spectra of components if ica components available
% resest variable for spectra of ica components
components_fft=[];

if ~isempty(icaComp)
    freq = [];
    k = 1;
    % comput spectra of ica components with Welch's method and window
    % length of wlength and 50% overlap
    for c = comp(end)+1:size(icaComp,1) % iterate of components of interest
        [components_fft(k,:),freq] = pwelch(icaComp(c,:),wlength,wlength/2,NFFT,Fs);
        k = k+1;
    end
    % obtain onesided spectrum
    components_fft = components_fft(:,1:NFFT/2+1);
    freq = freq(1:NFFT/2+1)';
    % normalize to total mean power
    normval = mean(sum(components_fft,2));
    components_fft = components_fft/normval;

    % plot spectrum for control purposes
    figure
    plotPSD(components_fft,freq)
    title('Power Spectrum Density (ICs)')
```

```
    % save spectra of ica components
    filecompspec = [filedir '\components\' filenameSave '_comp_spec'];
    [pathstr,namecompspec,ext] = fileparts(filecompspec);
    subsetcompspec = genvarname(namecompspec);
    eval([subsetcompspec '= components_fft;'])
    save(filecompspec,namecompspec);
end


% --- Executes on button press in pushbutton_remove.
function pushbutton_remove_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_remove (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global x Fs rawdata cleandata videoIdx vidFile vidLoaded resetdata

% calculate samples of time selection
xSmpl = round(x*Fs);
xleft = min(xSmpl); % get left border of selection
xright = max(xSmpl); % get right border of selection

% cut and remove selection from raw and clean data
if xright>size(rawdata,2)   % remove data from end
    rawdata = rawdata(:,1:xleft);
    cleandata = cleandata(:,1:xleft);
elseif xleft<1              % remove data from start
    rawdata = rawdata(:,xright:end);
    cleandata = cleandata(:,xright:end);
else
    % remove data in the middle
    rawdata = [rawdata(:,1:xleft) rawdata(:,xright:end)];
    cleandata = [cleandata(:,1:xleft) cleandata(:,xright:end)];

end

resetdata = rawdata;
% cut and remove selection from video if available
if vidLoaded
    xFrames = round(x*vidFile.FrameRate); % calculate frames of selection
    xvidleft = min(xFrames);    % get left border of selection
    xvidright = max(xFrames);   % get right border of selection
    if xvidright>length(videoIdx)   % remove frames at the end
        videoIdx = videoIdx(1:xvidleft);
    elseif xvidleft<1
        videoIdx = videoIdx(xright:end); % remove frames at the beginning
    else
        % remove frames in the middle
        videoIdx = [videoIdx(1:xvidleft+2) videoIdx(xvidright-2:end)];
    end
end

% refresh GUI
updateGUI(handles);

% --- Executes on button press in pushbutton_reset.
function pushbutton_reset_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_reset (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global resetdata cleandata artdata

% set clean data to reset data
cleandata = resetdata;

% reset detected artifacts
artdata = nan(size(cleandata));

% set display to clean data
set(handles.popupmenu_view,'Value',2);
```

```
% refresh GUI
updateGUI(handles);

% --- Executes on button press in pushbutton_test.
function pushbutton_test_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_test (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global rawdata cleandata Fs filename

rawdata = zeros(15,100*Fs);
cleandata = zeros(15,100*Fs);

a = -0.5;
b = 0.5;

rawdata(1:14,:) = a + (b-a).*randn(14,100*Fs);
cleandata(1:14,:) = a + (b-a).*randn(14,100*Fs);

rawdata(1:14,:)  = detrend(rawdata(1:14,:),'constant');
cleandata(1:14,:)  = detrend(cleandata(1:14,:),'constant');

filename = 'test-signal-randn';
updateGUI(handles);


% --- Executes on button press in pushbutton_autoDetect.
function pushbutton_autoDetect_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_autoDetect (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global rawdata cleandata artdata Fs concatdata

% get selected datatype
datatype = get(handles.popupmenu_view,'Value');

if datatype==1
    % get rawdata
    data = rawdata(1:14,:);
else
    % get cleandata
    data = cleandata(1:14,:);
end

% initialize variable for detected artifacts
artdata = nan(size(cleandata));

% indicate process status
set(handles.text_process,'String','auto detect: processing...');

% get number of samples
N = size(data,2);

% define length of extension for detection
L = Fs/2;

% calculate standard deviation
sd = std(data,[],2);

% define threshold for detection
thr1 = repmat(4*sd,[1, N]);
% find indices of artifacts based on threshold
[row,col] = find(abs(data) > thr1);
% find indices of clean data
idx = setdiff(1:N,col);
% find indices of artifacts
idx = setdiff(1:N,idx);
```

```
% extend regions of artifacts by length L left and right
idx_ext = zeros(1,length(col)*(2*L+1));
for k=1:length(idx)
    idx_ext((2*L+1)*(k-1)+1:(2*L+1)*k) = [idx(k)-L:idx(k)+L];
end
% final indices of signal
idx_sig = setdiff((1:N), idx_ext);
% final indices of artifacts
idx_art = setdiff((1:N), idx_sig);

% remove artifacts - set to zero
artchannel = data;
data(:,idx_art) = 0;
% mark artifacts for display
artchannel(:,idx_sig) = NaN;
artdata(1:14,:) = artchannel;

% prepare timeseries without artifacts for rejection
concatdata = [];
for ch = 1:size(data,1);
    % get snippets of good data
    autosnips = struct2cell(regionprops(data(ch,:)~=0,...
        data(ch,:), 'PixelValues'));
    [elength] = cellfun('size', autosnips,2) ;
    k=1;
    concatchn = [];
    % concatenate the good data
    while k <= length(elength);
            concatchn = [concatchn autosnips{k}];
            k = k+1;
    end
    concatdata = [concatdata; concatchn];
end
concatdata = [concatdata; zeros(1,size(concatdata,2))]; % set marker channel

% indicate process status
set(handles.text_process,'String','auto detect: done.');

% set display to clean data
set(handles.popupmenu_view,'Value',2);

% refresh GUI
updateGUI(handles);

% --- Executes on button press in pushbutton_reject.
function pushbutton_reject_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_reject (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global concatdata rawdata cleandata resetdata artdata

rawdata = concatdata; % replace clean data with data without artifacts
cleandata = rawdata; % replace clean data with data without artifacts
resetdata = rawdata;

artdata = nan(size(cleandata)); % reset detected artifacts

% refresh GUI
updateGUI(handles);

% --- Executes on button press in pushbutton_runICA.
function pushbutton_runICA_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_runICA (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global cleandata Fs icaEEG wicaEEG meanvar comp icaComp

% get clean data without markers
data = cleandata(1:14,:);
```

```
% run wavelet enhanced ICA
icaMethod = get(handles.popupmenu_base,'Value'); % get ICA method

% run wavelet enhanced ICA
icaAlg = get(handles.popupmenu_algorithm,'Value'); % get ICA algorithm

% indicate method and status of process
nameMethod = {'status: ICA ','status: wICA '};
set(handles.text_process,'String', [nameMethod{icaMethod} 'processing...']);

% run ICA
[artRemEEG,icaEEG,wicaEEG,meanvar,comp,icaComp] = ...
    runICA(data,Fs,icaMethod,icaAlg);

% indicate end of process
set(handles.text_process,'String',[nameMethod{icaMethod} 'done.']);

% update clean data
cleandata(1:14,:) = artRemEEG;

% set display to clean data
set(handles.popupmenu_view,'Value',2);

% refresh GUI
updateGUI(handles);

% ========================================================================
% POPUPMENUES
% ========================================================================

% --- Executes on selection change in popupmenu_scale.
function popupmenu_scale_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu_scale (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns popupmenu_scale contents as cell array
%        contents{get(hObject,'Value')} returns selected item from popupmenu_scale

% refresh GUI
updateGUI(handles);

% --- Executes during object creation, after setting all properties.
function popupmenu_scale_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu_scale (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in popupmenu_view.
function popupmenu_view_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu_view (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns popupmenu_view contents as cell array
%        contents{get(hObject,'Value')} returns selected item from popupmenu_view

% refresh GUI
updateGUI(handles);
```

```
% --- Executes during object creation, after setting all properties.
function popupmenu_view_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu_view (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in popupmenu_base.
function popupmenu_base_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu_base (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns popupmenu_base contents as cell array
%        contents{get(hObject,'Value')} returns selected item from popupmenu_base


% --- Executes during object creation, after setting all properties.
function popupmenu_base_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu_base (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), ...
        get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on selection change in popupmenu_algorithm.
function popupmenu_algorithm_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu_algorithm (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns popupmenu_algorithm contents as cell
array
%        contents{get(hObject,'Value')} returns selected item from popupmenu_algorithm


% --- Executes during object creation, after setting all properties.
function popupmenu_algorithm_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu_algorithm (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
% ========================================================================
% EDIT TEXTS
% ========================================================================

function edit_length_Callback(hObject, eventdata, handles)
% hObject    handle to edit_length (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_length as text
%        str2double(get(hObject,'String')) returns contents of edit_length as a double
global Fs

% get epoch length from edit
eLength = floor(str2double(get(handles.edit_length,'String'))*Fs);

% set corresponding slider to value
set(handles.slider_epoch,'Value',eLength);
% refresh GUI
updateGUI(handles);

% --- Executes during object creation, after setting all properties.
function edit_length_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_length (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit_id_Callback(hObject, eventdata, handles)
% hObject    handle to edit_id (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_id as text
%        str2double(get(hObject,'String')) returns contents of edit_id as a double


% --- Executes during object creation, after setting all properties.
function edit_id_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_id (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% ========================================================================
% TEXT
% ========================================================================

% --- Executes during object creation, after setting all properties.
function text_file_CreateFcn(hObject, eventdata, handles)
% hObject    handle to text_file (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
% ========================================================================
% INTERACTIVE FEATURES
% ========================================================================

% --- Executes on mouse press over axes background.
function axes_display_ButtonDownFcn(hObject, eventdata, handles)
% hObject    handle to axes_display (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global x

% get coordinates from mouse position
[x,y,button] = ginput(2);


% mark selection on data
selection = patch([x(1) x(2) x(2) x(1)],[0 0 1600 1600],'r','EdgeColor','none');
alpha(selection,0.2); % adjust transparency

% ========================================================================
% FUNCTIONS
% ========================================================================

% function for initialize all GUI components
% ------------------------------------------------------------------------
function initializeGUI(handles)

global Fs rawdata  cleandata vidLoaded  artdata wlength

% initialize variables
Fs = 128;       % sampling frequency of 128 Hz
wlength = 384;  % window length of 3 seconds (3*128 = 384 samples)
vidLoaded = 0;  % reset flag for video loaded

rawdata = zeros(15,100*Fs);     % initialize rawdata
cleandata = zeros(15,100*Fs);   % initialize cleandata
artdata = nan(15,100*Fs);       % reset dected artifacts

% initialize GUI components
set(handles.slider_display,'Value',1);      % set slider to fist data sample
set(handles.slider_epoch,'Value',10*Fs);    % show epoch of 10 sec
set(handles.popupmenu_scale,'Value',8)      % set scale to 100
set(handles.popupmenu_view,'Value',1)       % show raw data
set(handles.popupmenu_algorithm,'Value',1)       % show raw data
set(handles.popupmenu_base,'Value',2)       % show raw data
set(handles.axes_video, 'xtick',[],'ytick',[]);   % remove ticks from video axes
set(handles.axes_video, 'xticklabel',[],'yticklabel',[]); % remove labels from video axes

% update entire GUI
updateGUI(handles);

% function for updating all GUI components
% ------------------------------------------------------------------------
function updateGUI(handles)

global rawdata cleandata icaComp Fs vidFile vidLoaded videoIdx wicaEEG ...
    meanvar comp  artdata

% get length of data
xmax = size(cleandata,2);

% set slider maxima
set(handles.slider_display,'Max',xmax);
set(handles.slider_epoch,'Max',xmax);

% get epoch length
eLength = floor(get(handles.slider_epoch,'Value'));
if eLength < 0.5
    eLength = 0.5;
end
```

```
if(eLength>xmax)
    eLength = xmax;
    set(handles.slider_epoch,'Value',xmax);
end
set(handles.text_eLength,'String',sprintf('%.1f sec',eLength/Fs));
set(handles.edit_length,'String',sprintf('%.1f', eLength/Fs));

% get epoch start in samples
smpStart = round(get(handles.slider_display,'Value'));

% prevent exceeding index
if(smpStart < 1)
    smpStart = 1;
end
if(smpStart>xmax)
    smpStart = xmax-eLength-1;
end
% set slider to epoch start
set(handles.slider_display,'Value',smpStart);

% update textfield for time
set(handles.text_time,'String',sprintf('Total time: %.2f sec',xmax/Fs));

% get scaling parameter
contents = cellstr(get(handles.popupmenu_scale,'String'));
scale = str2num(contents{get(handles.popupmenu_scale,'Value')});

% get display mode for view
mode = get(handles.popupmenu_view,'Value');

switch(mode)
    case 1 % rawdata
        cla reset
        data = rawdata; % get rawdata

        % extract epoch from data
        epoch = extractEpoch(data,smpStart,eLength);
        % plot epoch
        ChanTitles = {'AF3' 'F7' 'F3' 'FC5' 'T7' 'P7' 'O1' ...
            'O2' 'P8' 'T8' 'FC6' 'F4' 'F8' 'AF4' 'Mkr'};
        Title = 'Raw EEG Data';
        set(handles.axes_display,'HitTest','off');
        PlotEEG(handles,epoch,Fs,smpStart,scale,ChanTitles,Title,'k');
        % reactivate mouse selection
        set(handles.axes_display, 'ButtonDownFcn', ...
            {@axes_display_ButtonDownFcn,handles});
    case 2 % clean data
        cla reset
        data = cleandata; % get clean data
        art = artdata;    % get detected artifacts
        % extract epoch from data
        epoch = extractEpoch(data,smpStart,eLength);
        epochart = extractEpoch(art,smpStart,eLength);

        % plot epoch
        ChanTitles = {'AF3' 'F7' 'F3' 'FC5' 'T7' 'P7' 'O1' ...
            'O2' 'P8' 'T8' 'FC6' 'F4' 'F8' 'AF4' 'Mkr'};
        Title = 'Processed EEG Data';
        set(handles.axes_display,'HitTest','off');
        % plot data
        PlotEEG(handles,epoch,Fs,smpStart,scale,ChanTitles,Title,'k');
        hold on
        % mark detected artifacts
        PlotEEG(handles,epochart,Fs,smpStart,scale,ChanTitles,Title,'r');
        hold off

        % reactivate mouse selection
        set(handles.axes_display, 'ButtonDownFcn', ...
            {@axes_display_ButtonDownFcn,handles});
```

```
    case 3 % ICA components
        cla reset
        [N,M] = size(icaComp);  % get number of components
        dispComp = zeros(N+1,M); % initialize displayed components (no markers)
        dispComp(1:N,:) = icaComp;  % get ica components
        dispComp(end,:) = rawdata(end,:); % insert markers
        % extract epoch from compoonents
        data = dispComp;
        epoch = extractEpoch(data,smpStart,eLength);
        % plot epoch
        ChanTitles = {'1' '2' '3' '4' '5' '6' '7' ...
            '8' '9' '10' '11' '12' '13' '14' 'Mkr'};
        Title = 'Independent Components (ICA)';
        set(handles.axes_display,'HitTest','off');
        PlotEEG(handles,epoch,Fs,smpStart,scale,ChanTitles,Title,'k');
        set(handles.axes_display, 'ButtonDownFcn', ...
            {@axes_display_ButtonDownFcn,handles});
    case 4 % wavelet based ICA components
        cla reset
        [N,M] = size(wicaEEG);
        dispComp = zeros(size(rawdata)); % initialize displayed components
        dispComp(1:N,:) = wicaEEG;       % get wavelet based ica components
        dispComp(end,:) = rawdata(end,1:M); % get markers
        % extract epoch from data
        data = dispComp;
        epoch = extractEpoch(data,smpStart,eLength);
        % plot epoch
        ChanTitles = {'1' '2' '3' '4' '5' '6' '7' ...
            '8' '9' '10' '11' '12' '13' '14' 'Mkr'};
        Title = 'Independent Components (wICA)';
        set(handles.axes_display,'HitTest','off');
        PlotEEG(handles,epoch,Fs,smpStart,scale,ChanTitles,Title,'k');
        set(handles.axes_display, 'ButtonDownFcn', ...
            {@axes_display_ButtonDownFcn,handles});
    case 5 % screen plot
        cla reset
        plot([1:length(meanvar)],meanvar,'b'),hold on
        plot([1:length(comp)],meanvar(comp),'ok');
        title('SVD')
        xlabel('components')
        ylabel('singular values')
end

% show video if available
if (vidLoaded)
    % calculate frame for video display
    vidFrame = floor(smpStart/Fs*vidFile.FrameRate)+1;
    showVideo(handles,vidFile,vidFrame,videoIdx);
end

% function for plotting EEG data
% --------------------------------------------------------------------
 function PlotEEG(handles,data,Fs,start,Scale,ChanTitles,Title,col)

% make display current axes
axes(handles.axes_display);

% get number of channels and samples of data
[nChs,nPts] = size(data);

% generate time vector
t = (start:start+nPts-1)/Fs;

% initialize variables
Ct = {};
tt = {};
dt = floor(nPts/10); % define interval of time labels
```

```
% define time lables
for tlabel = ((start:dt:start+nPts)/Fs)
    tt = [tt; {sprintf('%.1f',tlabel)}];
end

% plot EEG channels
for k = 1:nChs-1,
    plot(t, real((nChs - k)*Scale + data(k,:)),col);
    hold on
    if isempty(ChanTitles),
        Ct = [Ct; {num2str(nChs - k + 1)}];
    else
        Ct = [Ct; ChanTitles(nChs - k + 1)];
    end
end

% plot markers
Marker = data(15,:);
Marker(Marker>0) = 1500;
mrkIdx = find(Marker>0);
stem(t,Marker,'r','Linewidth',2,'Marker','none');
Ct = [Ct; ChanTitles(1)];
b = num2str(round(data(nChs,mrkIdx))); c = cellstr(b);
dx = 0.1; dy = 0.1; % displacement so the text does not overlay the data points
text(t(mrkIdx)+dx, Marker(mrkIdx)+dy, c,'color','r','FontWeight','bold')

% adjust axes
hold off;
xlim([start start+nPts]/Fs);
ylim([-1 nChs+1]*Scale);
set(gca,'XTick',(start:dt:start+nPts)/Fs);
set(gca,'XTickLabel',tt);
set(gca,'YTick', (0:1:nChs-1)*Scale);
set(gca,'YTickLabel',Ct);

% set title and x-label
title(Title);
xlabel('Time (sec)')

% function to show video
% --------------------------------------------------------------------
function showVideo(handles,vidfile,vidFrame,videoIdx)

% get frame index
frameIdx = videoIdx(vidFrame);
% read video file at frame index
vid = read(vidfile,frameIdx);
% display frame
image(vid,'parent',handles.axes_video);
% remove axes for video
set(handles.axes_video, 'xtick',[],'ytick',[]);
set(handles.axes_video, 'xticklabel',[],'yticklabel',[]);

% function to extract displayed epoch from data stream
% --------------------------------------------------------------------
function epoch = extractEpoch(data,start,length)

% prevent over/underflow
if ((start+length-1) > size(data,2))
    stop = size(data,2);
else
    stop = start+length-1;
end
% extract data from start to stop
epoch = data(:,start:stop);
```

```
% function to re-reference data to average
% --------------------------------------------------------------------
function avgRefEEG = avgReference(data)

avgCh = mean(data);
for ch = 1:14
    avgRefEEG(ch,:) = data(ch,:) - avgCh;
end

% function to remove artifacts using ICA (normal and wavelet based
% --------------------------------------------------------------------
function [artRemEEG,icaEEG,wicaEEG,meanvar,comp,icaComp] = ...
        runICA(data,Fs,icaMethod,icaAlg)

% initialize variables
icaEEG = [];
wicaEEG = [];
icaComp = [];
comp = [];
meanvar = [];

% obtain unmixing matrix
switch(icaAlg)
    case 1
        [weight, sphere] = runica(data); % run InfoMax algorithm
        W = weight*sphere;    % obtain unmixing matrix
        Winv = inv(W); % obtain mixing matrix
        icaEEG = W*data; % compute components

        % set thrshold for wavelet artifact detection
        threshold = 1.25;

        % clean components with thresholding of wavelets
        [wicaEEG, opt]= RemoveStrongArtifacts(icaEEG, 1:size(data,1),...
            threshold, Fs);

    case 2
        W = jader(data); % run JADE algorithm
        Winv = inv(W); % obtain mixing matrix
        icaEEG = W*data; % compute components

        % set thrshold for wavelet artifact detection
        threshold = 1.25;

        % clean components with thresholding of wavelets
        [wicaEEG, opt]= RemoveStrongArtifacts(icaEEG, 1:size(data,1),...
            threshold, Fs);

    case 3
        [Winv,icaEEG] = sobi(data); % run SOBI algorithm

        % set wicaEEG variableto ica EEG (wavelet based removal not
        % applicable)
        wicaEEG = icaEEG;
end

% compute variances without backprojecting to save time and memory -sm 7/05
[chans,frames] = size(data);
meanvar = sum(Winv.^2).*sum((data').^2)/((chans*frames)-1); % from Rey Ramirez 8/07

% sort variances from large to small
[sortvar, windex] = sort(meanvar);
windex = windex(chans:-1:1); % order large to small
meanvar = meanvar(windex);


% calculate number of components that contribute less than 95% to variance
var95 = 0.95*sum(meanvar);
svsum = meanvar(1);
oldsvsum = 1;
k = 1;
```

```
while (svsum<var95) && (k<length(meanvar) && (svsum-oldsvsum)/oldsvsum>0.2)
    k=k+1;
    oldsvsum = svsum;
    svsum = sum(meanvar(1:k));
end
comp = 1:k; % save "artifact" components

% save sorted components
icaComp = icaEEG(windex,:);

% set "artifact" components to zero
icaEEG(windex(1:k),:) = 0;

% remix components to eeg channels using mixing matrix
if icaMethod == 1
    artRemEEG = real(Winv*icaEEG);
elseif icaMethod == 2
    artRemEEG = real(Winv*wicaEEG);
end


% function to plot spectrum (PSD)
% ------------------------------------------------------------------------
function plotPSD(spec,f)

hold off
% define limits of plot
xlim = [0 40];
ylim = [0 1.1*max(spec(:))];

% get number of channels
num = size(spec,1);

% define color vector
cc = hsv(num);

% plot spectra with changing colors
for i=1:num
    plot(f,spec(i,:),'color',cc(i,:)),hold on
end
axis([xlim ylim])
xlabel('Frequency [Hz]')
ylabel('Spectral Density (Hz^{-1})');

% adjust labels on Y axis
if num < 14
    legendnr = {'1','2','3','4','5','6','7','8','9','10','11','12','13'};
    legend(legendnr(1:num),'fontsize',8);
else
    legend({'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8);
end
```

4.1.1 Function: selectData(…)

```
% Grand Valley State University, Master of Science in Engineering
% ==========================================================================
% author: Nadina Zweifel
% date: 3/16/2016
% advisor: Dr. Samhita Rhodes
% --------------------------------------------------------------------------
% function: data = selectData(data,scale,Fs)
% this function reads the user's mouse cursor position when a selection in
% the data is made.
% input: data, scale, Fs
% data is a matrix with the time series where the rows are the channels and
% the columns the samples. scale is the vertical scaling factor of the
% displayed data plot. Fs is the sampling frequency of the data.
% output: data
% data returns the indexes for the selected region.
% --------------------------------------------------------------------------

function data = selectData(data,scale,Fs)

% get the mouse cursor position from the user
[x,y] = ginput(2);
% mark the user's selection in the data with a colored rectangle
selection = patch([x(1) x(2) x(2) x(1)],[0 0 15 15]*scale,'r','EdgeColor','none');
% set the rectangle transparent
alpha(selection,0.2);
% convert the curser position to samples
xSmpl = round(x*Fs);
% determin left and right boundery
xleft = min(xSmpl);
xright = max(xSmpl);

% return the indexes of the selected regions preventing that the indexes
% exceed the dimensions
if xright>size(data,2)
    data = data(:,xleft:end);
elseif xleft<1
    data = data(:,1:xright);
else
    data = data(:,xleft:xright);
end
end
```

4.1.2 Function: ploteeg(…)

```
% Grand Valley State University, Master of Science in Engineering
% ==========================================================================
% author: Nadina Zweifel
% date: 3/16/2016
% advisor: Dr. Samhita Rhodes
% --------------------------------------------------------------------------
% function: ploteeg(data,Fs,start,scale,chanTitles,title,col)
% function to plot the time series of EEG data. The function is based on
% the code for wavelet enhanced ICA by Castellanos & Makarov 2006
% input: data, Fs, start, scale, chanTitles, title, col
% data is a matrix with the time series where the rows are the channels and
% the columns the samples using the sampling frequency Fs. start is the
% index of the first sample displayed in the figure. scale defines the
% vertical scaling factor of the EEG data. chanTitles consists of cell
% array with strings for labeling the channels on the vertical axis. col
% contains the values for defining the color of the plot.
% --------------------------------------------------------------------------

function ploteeg(data,Fs,start,scale,chanTitles,title,col)

% check if at least data variable is an input
if nargin < 1,
    disp('No arguments');
    help PlotEEG
    return;
end
% get dimensions of the data - nChns: # channels, nPts: # samples
[nChs,nPts] = size(data);

% check number of channels, if too high it's assumed to be time samples
if nChs > 20,
    disp('Too many channels (try to transpose the data)');
    return
end

% define time vector
t = (start:start+nPts-1)/Fs;

% initialize variables
Ct = {};
tt = {};

% define time resolution on plot
dt = floor(nPts/10);

% define lables for time axis
for tlabel = ((start:dt:start+nPts)/Fs)
tt = [tt; {sprintf('%.1f',tlabel)}];
end

% plot the data in the correct scaling defined by variable scale
for k = 1:nChs,
    plot(t, real((nChs - k)*scale + data(k,:)),col);
    hold on
    % add labels for channels
    if isempty(chanTitles),
        Ct = [Ct; {num2str(nChs - k + 1)}];
    else
        Ct = [Ct; chanTitles(nChs - k + 1)];
    end
end

% add marker label to last row of data
Ct = [Ct; chanTitles(1)];
hold off;
```

```
% define plot limits
xlim([start start+nPts]/Fs);
ylim([-1 nChs+1]*scale);
% add title to plot
title(title);

% set time labels
set(gca,'XTick',(start:dt:start+nPts)/Fs);
set(gca,'XTickLabel',tt);
% set channels labels
set(gca,'YTick', (0:1:nChs-1)*scale);
set(gca,'YTickLabel',Ct);
% add label for x-axis
xlabel('Time (sec)')
```

4.1.3 Function: wICA(…)

```
%% For wICA the algorithm of Valeri A. Makarov was used:
% This code is for illustration of the method described in:
% N.P. Castellanos, and V.A. Makarov (2006). "Recovering EEG brain signals: Artifact
% suppression with wavelet enhanced independent component analysis"
% J. Neurosci. Methods, 158, 300-312.
%
% Requirements: runica from EEGLAB toolbox, and rwt - Rice Wavelet Toolbox
% (both freely available in internet).
%
% This code is copyright © by the authors, and we hope you acknowledge our
% work. We distribute it in the hope that it will be useful, but without any warranty.
%
% Author: Valeri A. Makarov
% e-mail: vmakarov@opt.ucm.es
%
% 2006
%
% Find independent components
% EEGLAB is required!!! You can also use other algorithms, e.g. fICA.
% Note, the use of long (in time) data sets reduces the algorithm performance
% see for details the abovementioned paper.

function Data_wICA = wICA(Data,Fs)

Fnyq  = Fs/2;
% F_notch = 50; % Notch at 50 Hz
% [b,a] = iirnotch(F_notch/Fnyq, F_notch/Fnyq/20);
% Data  = filtfilt(b,a, Data);

%% Conventional High Pass Filter
% This is an optional step (suppress low <4Hz frequency noise)
F_cut = 4;
[b,a] = ellip(1, 0.5, 20, F_cut/Fnyq, 'high');
Data  = filtfilt(b,a, Data);

%% Remove mean values from the channels and plot raw data
Data  = detrend(Data,'constant');
% Transpose the data matrix to get (channel x time) orientation
Data = Data';

%% wICA
[weight, sphere] = runica(Data, 'verbose', 'off');

W = weight*sphere;    % EEGLAB --> W unmixing matrix
icaEEG = W*Data;      % EEGLAB --> U = W.X activations

[icaEEG2, opt]= RemoveStrongArtifacts(icaEEG, (1:14), 1.25, Fs);
Data_wICA = inv(W)*icaEEG2;
```

4.1.4 Function: RemoveStrongArtifacts(…)

```
% This function denoise high amplitude artifacts (e.g. ocular) and remove them from the
% Independent Components (ICs).
%
% INPUT:
%
% icaEEG - matrix of ICA components (Nchanel x Nobservations)
%
% Comp   - # of ICs to be denoised and cleaned (can be a vector)
%
% Kthr   - threshold (multiplayer) for denoising of artifacts
%          (default Kthr = 1.15)
%
% F      - acquisition frequency
%          (default F = 256 Hz)
%
% OUTPUT:
%
% icaEEG - matrix of cleaned indenpendent components
%
% opt    - vector of threshold values used for filtering of corresponding
%          ICs
%
% NOTE: If a component has no artifacts of a relatively high amplitude
%       the function will skip this component (no action), dispaly a
%       warning and the corresponding output "opt" will be set to zero.
%
% Valeri A. Makarov, vmakarov@opt.ucm.es
% ver 0.1 Sept. 2005
% ver 0.2 May   2006

function [icaEEG, opt] = RemoveStrongArtifacts(icaEEG, Comp, Kthr, F)

if nargin < 2,
    disp('At least two arguments are required!');
    help RemoveStrongArtifacts
    return;
end
if nargin < 3 || isempty(Kthr), Kthr = 1.15; end
if nargin < 4 || isempty(F), F = 256; end
L = round(F*0.1);
[Nchan, Nobser] = size(icaEEG);
if Nchan > Nobser,
    error('Problem with data orientation, try to transpose the matrix!');
end

N = 2^floor(log2(Nobser));
h = daubcqf(6);
opt = zeros(1,length(Comp));
for c=1:length(Comp),
    Y = icaEEG(Comp(c),1:N);     % cth component from 1 to N
    Sig = median(abs(Y)/0.6745); % get measure of data amplitude
    Thr = 4*Sig;                 % determine threshold
    idx = find(abs(Y) > Thr);    % find indices above threshold
    idx_ext = zeros(1,length(idx)*(2*L+1)); % extend artifact part
    for k=1:length(idx),
        idx_ext((2*L+1)*(k-1)+1:(2*L+1)*k) = [idx(k)-L:idx(k)+L];
    end
    id_noise=setdiff((1:N), idx_ext);   % indices of signal of interest
    id_artef=setdiff((1:N), id_noise);  % indices of artifacts
    if isempty(id_artef),
        %disp(['The component #' num2str(Comp(c)) ' has passed unchanged']);
        continue;
    end
    thld = 3.6;     % threshold for artifact removal
    KK = 100;
    LL = floor(log2(length(Y)));
    [xl, xh] = mrdwt(Y, h, LL); % wavelet transform (without downsampling)
```

```
    while KK > Kthr,            % test vs. input parameter Kthr = 1.25
        thld = thld + 0.5;      % adjust treshold
        xh = HardTh(xh, thld);  % hard thresholding of detail coefficients
        xd = mirdwt(xl,xh,h,LL); % backprojection of artifacts with inverse wt
        xn = Y - xd;             % subtract artifacts
        cn=corrcoef(Y(id_noise),xn(id_noise)); % calculate ratios
        cd=corrcoef(Y(id_noise),xd(id_noise));
        ca=corrcoef(Y(id_artef),xd(id_artef));

        KK = ca(1,2)/cn(1,2);
        KKnew = ca(1,2)/cd(1,2);
    end
    opt(c) = thld;          % save optimal threshold
    Y = icaEEG(Comp(c),end-N+1:end);  % apply wavelet thresholding to rest of data
    icaEEG(Comp(c),1:N) = xn;
    LL = floor(log2(length(Y)));
    [xl, xh] = mrdwt(Y, h, LL);
    xh = HardTh(xh, thld);
    xd = mirdwt(xl,xh,h,LL);
    xn = Y - xd;
    icaEEG(Comp(c),N+1:end) = xn(end-(Nobser-N)+1:end);
%     disp(['The component #' num2str(Comp(c)) ' has been filtered']);
end
```

4.2 CODE: THETA RHYTHM

4.2.1 Main Script: thetaRhythm.m

```
% Grand Valley State University, Master of Science in Engineering
% ==========================================================================
% author: Nadina Zweifel
% date: 3/16/2016
% advisor: Dr. Samhita Rhodes
% --------------------------------------------------------------------------
% Master Thesis: Main script for finding the theta rhythm of the subject
% --------------------------------------------------------------------------

clear all, clc, close all

% load data from a file
load('data.mat')

% define frequency range limits
lf = 2;
hf = 12;

% calculate sample points for limits
lfsmp = floor(lf*4);
hfsmp = ceil(hf*4);

% define frequency subband width
wsub = 4;

% get number of sessions
ntrials = size(data_rest1,2);

% define normalization method: norm=1 normalizes spectrum with overall mean
% power across channels and frequencies
norm = 1;

% compute spectra and sub the frequencies across wsub (=4) points
for trial = 1:ntrials
    % resting 1
    [c_psd_rest1{trial},freq,psd_subs,err_subs,freq_subs] = ...
        psdsubswitherr(data_rest1{trial},lf,hf,wsub,norm);
    c_psd_subs_rest1{trial} = psd_subs;
    c_err_subs_rest1{trial} = err_subs;
    % interaction
    [c_psd_interact{trial},freq,psd_subs,err_subs,freq_subs] = ...
        psdsubswitherr(data_interact{trial},lf,hf,wsub,norm);
    c_psd_subs_interact{trial} = psd_subs;
    c_err_subs_interact{trial} = err_subs;
end

% transform the cell arrays to matrices
psd_subs_rest1 = (cat(3,c_psd_subs_rest1{:}));
psd_subs_interact = (cat(3,c_psd_subs_interact{:}));
err_subs_rest1 = (cat(3,c_err_subs_rest1{:}));
err_subs_interact = (cat(3,c_err_subs_interact{:}));

% calculate the percentage change from resting 1 to interaction
psd_change = 100*(psd_subs_interact-psd_subs_rest1)./psd_subs_rest1;

% remove 1st trial due to artifactual data
trials_a1 = [2 3 4];
```

```
% calculate mean for first baseline phase over the 3 points
mean_a1_interact = mean(psd_subs_interact(:,:,trials_a1),3);
mean_a1_rest1 = mean(psd_subs_rest1(:,:,trials_a1),3);

% calculate the percentage change of the means
mean_a1_change = 100*(mean_a1_interact-mean_a1_rest1)./mean_a1_rest1;

% test the change for significance using one standard error
[psd_sig,psd_sig_mag,psd_mean,psd_change] = ...
    test22SD(psd_subs_rest1,psd_subs_interact,err_subs_interact);


%% display results

% define limits of colorbar
zmin = 0;
zmax = 10;
% define resolution colormap: continuous -> ncol=0
ncol = 0;
% define colorbar label
cb_name = 'PSD (Vm^2)';
% define frequency sub vector
freq_subs = (lfsmp:wsub:hfsmp)/4;

% get number of channels and subs
[nchannels,nsubs] = size(mean_a1_rest1(:,:,1));

% resting 1
figure
imagesc(mean_a1_rest1)
setColorbar(zmin,zmax,cb_name,0,ncol);
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',10)
title('Spectral mean power A1 (resting 1)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% interaction
figure
imagesc(mean_a1_interact)
setColorbar(zmin,zmax,cb_name,0,ncol);
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',10)
title('Spectral mean power A1 (interaction)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

%% display results for percentage change
% define colorbar limits
zmin = -70;
zmax = 70;
% define colorbar label
cb_name = 'change (%)';
% define resolution colormap: continuous -> ncol=0
ncol = 0;
```

```
% display figure
figure
imagesc(mean_a1_change)
setColorbar(zmin,zmax,cb_name,1,ncol);
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',10)
title('Change PSD: A1(resting 1) -> A1(interaction)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for significant percentage change
figure
imagesc(psd_sig_mag(:,:,1))
setColorbar(zmin,zmax,cb_name,1,ncol);
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',10)
title('Significant change (rest1/interact) of mean PSD in 1st baseline (A1)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])
```

4.3 CODE: SPECTRAL ANALYSIS

4.3.1 Main Script: spectralAnalysis.m

```
% Grand Valley State University, Master of Science in Engineering
% ======================================================================
% author: Nadina Zweifel
% date: 3/16/2016
% advisor: Dr. Samhita Rhodes
% ----------------------------------------------------------------------
% Master Thesis: Main script for spectral analysis of EEG data. The script
% loads the files and transforms the time series to frequency domain for
% spectral analysis.
% ----------------------------------------------------------------------

clear all, clc, close all

% load data from a file
load('data.mat')

% define frequency range limits
lf = 2;
hf = 12;

% calculate sample points for limits
lfsmp = floor(lf*4);
hfsmp = ceil(hf*4);

% define frequency sub width
wsub = 4;

% get number of sessions
ntrials = size(data_rest1,2);
```

```
% define normalization method: norm=1 normalizes spectrum with overall mean
% power across channels and frequencies
norm = 1;

% compute spectra and sub the frequencies across wsub (=4) points
for trial = 1:ntrials
    % resting 1
    [c_psd_rest1{trial},freq,psd_subs,err_subs,freq_subs] = ...
        psdsubswitherr(data_rest1{trial},lf,hf,wsub,norm);
    c_psd_subs_rest1{trial} = psd_subs;
    c_err_subs_rest1{trial} = err_subs;
    % resting 2
    [c_psd_rest2{trial},freq,psd_subs,err_subs,freq_subs] = ...
        psdsubswitherr(data_rest2{trial},lf,hf,wsub,norm);
    c_psd_subs_rest2{trial} = psd_subs;
    c_err_subs_rest2{trial} = err_subs;
    % interaction
    [c_psd_interact{trial},freq,psd_subs,err_subs,freq_subs] = ...
        psdsubswitherr(data_interact{trial},lf,hf,wsub,norm);
    c_psd_subs_interact{trial} = psd_subs;
    c_err_subs_interact{trial} = err_subs;
end

% transform the cell arrays to matrices
psd_rest1 = (cat(3,c_psd_rest1{:}));
psd_rest2 = (cat(3,c_psd_rest2{:}));
psd_interact = (cat(3,c_psd_interact{:}));

psd_subs_rest1 = (cat(3,c_psd_subs_rest1{:}));
psd_subs_rest2 = (cat(3,c_psd_subs_rest2{:}));
psd_subs_interact = (cat(3,c_psd_subs_interact{:}));

err_subs_rest1 = (cat(3,c_err_subs_rest1{:}));
err_subs_rest2 = (cat(3,c_err_subs_rest2{:}));
err_subs_interact = (cat(3,c_err_subs_interact{:}));

% calculate the percentage change from resting 1 to interaction
psd_subs_change = (psd_subs_rest2 - psd_subs_rest1)./psd_subs_rest1*100;

% save the workspace to a file
% save('vars_6.mat')


%% display results

% define channel labels
chnlabels = {'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
    'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'};

%% resting 1

% apply two-standard deviation band method
[psd_sig,psd_sig_mag,psd_mean] = test2SD(psd_subs_rest1,err_subs_rest1);

% get number of channels and subs
[nchannels,nsubs] = size(psd_sig(:,:,1));

% display results with significant score

% define frequency vector
freq_subs = (lfsmp:wsub:hfsmp)/4;

% define margin around subplot
margin = 0.06;
```

```
% define paramter for colorbar
zmin = -4.5;
zmax = 4.5;
cb_name = 's-value';
% define resolution colormap: continuous -> ncol=0
ncol = 8;

figure
% display change A1-B1
h=subplot(131);
% adjust margins of subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig(:,:,2))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% adjust graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Significance: A1->B1 (resting 1)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display change A1-A2
h=subplot(132);
% adjust margins of subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig(:,:,3))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% adjust graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Significance: A1->A2 (resting 1)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display change A1-B2
h=subplot(133);
% adjust margins of subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig(:,:,4))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% adjust graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Significance: A1->B2 (resting 1)')
```

```
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results with significant magnitude
% set colorbar properties
zmin = -300;
zmax = 300;
cb_name = 'Change (%)';
% define resolution colormap: continuous -> ncol=0
ncol = 0;

figure
% display results for A1-B1
h=subplot(131);
% adjust margin of subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig_mag(:,:,2))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Change PSD: A1->B1 (resting 1)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for A1-A2
h=subplot(132);
% adjust margins of subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% dislay result matrix
imagesc(psd_sig_mag(:,:,3))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Change PSD: A1->A2 (resting 1)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for A1-B2
h=subplot(133);
% adjust margins of subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig_mag(:,:,4))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
```

```
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Change PSD: A1->B2 (resting 1)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])


%% display results for resting 2

% apply two-standard deviation band method
[psd_sig,psd_sig_mag,psd_mean] = test2SD(psd_subs_rest2,err_subs_rest2);

% set colorbar properties
zmin = -4.5;
zmax = 4.5;
cb_name = 's-value';
% define resolution colormap: continuous -> ncol=0
ncol = 8;

figure
% display results with significant score

% display results for A1-B1
h=subplot(131);
% adjust margins for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig(:,:,2))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Significance: A1->B1 (resting 2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for A1-A2
h=subplot(132);
% adjust margins for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig(:,:,3))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Significance: A1->A2 (resting 2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])
```

```
% display results for A2-B2
h=subplot(133);
% adjust margins for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig(:,:,4))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Significance: A1->B2 (resting 2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for significant change (magnitude)
% set colorbar properties
zmin = -300;
zmax = 300;
cb_name = 'Change (%)';
% define resolution colormap: continuous -> ncol=0
ncol = 0;

figure
% display results for A1-B1
h=subplot(131);
% adjust margin for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig_mag(:,:,2))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Change PSD: A1->B1 (resting 2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for A1-B2
h=subplot(132);
% adjust margins for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig_mag(:,:,3))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
```

```
title('Change PSD: A1->A2 (resting 2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for A2-B2
h=subplot(133);
% adjust margins for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig_mag(:,:,4))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Change PSD: A1->B2 (resting 2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

%% display results for rest1/resting 2

% apply two-standard deviation band method to detect change from rest1- to
% resting 2
[psd_sig,psd_sig_mag,psd_mean,psd_change] = ...
    test22SD(psd_subs_rest1,psd_subs_rest2,err_subs_rest2);

% set margin parameter
margin = 0.04;

% display results for significance
% set colorbar parameters
zmin = -4.5;
zmax = 4.5;
cb_name = 's-value';
% define resolution colormap: continuous -> ncol=0
ncol = 8;

figure
% display results for A1(rest1) - A2(rest2)
h=subplot(221);
% adjust margins for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig(:,:,1))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Significance: A1(rest1)->A1(rest2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])
```

```
% display results for B1(rest1) - B1(rest2)
h=subplot(222);
% adjust margins for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig(:,:,2))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Significance: B1(rest1)->B1(rest2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for A2(rest1) - A2(rest2)
h=subplot(223);
% set margins for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig(:,:,3))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Significance: A2(rest1)->A2(rest2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for B2(rest1) - B2(rest2)
h=subplot(224);
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
imagesc(psd_sig(:,:,4))
setColorbar(zmin,zmax,cb_name,1,ncol);
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Significance: B2(rest1)->B2(rest2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display significant change in magnitude
% adjust colorbar parameters
zmin = -300;
zmax = 300;
cb_name = 'Change (%)';
% define resolution colormap: continuous -> ncol=0
ncol = 0;
```

```
figure
% display results for A1(rest1) - A1(rest2)
h=subplot(221);
% adjust margins for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig_mag(:,:,1))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Change PSD: A1(rest1)->A1(rest2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for B1(rest1) - B1(rest2)
h=subplot(222);
% adjust margins for subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display results matrix
imagesc(psd_sig_mag(:,:,2))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Change PSD: B1(rest1)->B1(rest2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])

% display results for A2(rest1) - A2(rest2)
h=subplot(223);
% adjust margins of subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display results matrix
imagesc(psd_sig_mag(:,:,3))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Change PSD: A2(rest1)->A2(rest2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])
```

```
% display results for B2(rest1) -> B2(rest2)
h=subplot(224);
% adjust margins of subplot
p = get(h, 'pos');
p([3 4]) = p([3 4]) + margin;
set(h, 'pos', p);
% display result matrix
imagesc(psd_sig_mag(:,:,4))
% set colorbar
setColorbar(zmin,zmax,cb_name,1,ncol);
% set graph properties
set(gca,'XTick',1:nsubs)
xticklabel = cellstr(num2str(freq_subs',2));
set(gca,'XTickLabel',xticklabel,'fontsize',8)
set(gca,'YTick',1:nchannels)
set(gca,'YTickLabel',{'AF3', 'F7', 'F3', 'FC5', 'T7', 'P7', 'O1',...
        'O2', 'P8', 'T8', 'FC6', 'F4', 'F8', 'AF4'},'fontsize',8)
title('Change PSD: B2(rest1)->B2(rest2)')
xlabel('Frequency sub-bands (Hz)')
ylabel('Channels')
pbaspect([1 1 1])
```

4.3.2 Function: test2SD(…)

```
% Grand Valley State University, Master of Science in Engineering
% ========================================================================
% author: Nadina Zweifel
% date: 3/16/2016
% advisor: Dr. Samhita Rhodes
% ------------------------------------------------------------------------
% function: [psd_sig,psd_sig_mag,psd_mean] = test2SD(psd,substd)
% this function applies the two-standard deviation method to the input
% data which contains data of different phases (baseline and intervention).
% The function assumes that the phases consists of 4 data points and
% consideres the first 4 points as the baseline. Significance is tested by
% counting the data points exceeding the 2*SE band (SE=standard error).
% Cases that exceed the band 2 times or more are considered as significant.
% input: psd,substd
% psd represents the power spectral density of the EEG data under
% consideration. The input data is subned by taking the average across
% data points in a particular frequency range. Hence, psd contains the
% means of the subs while substd contains the standard deviation of each
% sub.
% output: psd_sig, psd_sig_mag, psd_mean
% psd_sig is a matrix containing the significance score (s-value) for each
% sub and channel in respect of the baseline. Unsignificant changes are
% represented with a zero. The s-value is 2 or higher which is considered
% as significant. psd_sig_mag is of the same type as psdsig but instead of
% s-values it contains the percentage change score for signifcant changes.
% psd_mean returns just the mean value for each sub and channel for each
% phase.
% ------------------------------------------------------------------------

function [psd_sig,psd_sig_mag,psd_mean] = test2SD(psd,substd)
    [nchannels,nsubs,nsessions] = size(psd);

    % calculate the baseline values in the first phase of the input data
    psd_a1_mean = mean(psd(:,:,1:4),3);  % average across 4 trials
    psd_a1_SE = std(psd(:,:,1:4),0,3); % standard deviation across 4 trials
    psd_a1_SE = psd_a1_SE/sqrt(4); % standard error across 4 trials

    % calculate bounderies for significance testing
    lowlim = psd_a1_mean-2*psd_a1_SE;
    highlim = psd_a1_mean+2*psd_a1_SE;

    % iteration over channels and subs to determine significant changes
    for ch = 1:nchannels
        for b = 1:nsubs
            p=1;
            for ph = 1:4:nsessions
                psd_tempcopy = psd(ch,b,ph:ph+3); % copy data of phase p
                error = substd(ch,b,ph:ph+3);   % copy st.dev. of phase p
                % test negative changes for significance
                psd_temp = psd_tempcopy+error; % add st.dev. to each trial
                psd_temp(psd_temp>=lowlim(ch,b)) = 0; % set unsignificant changes to zero
                psd_temp_sig = -logical(psd_temp); % set significant neg. changes to -1
                % test positive changes for significance
                psd_temp = psd_tempcopy-error;  % subtract std.dev. from each trial
                psd_temp(psd_temp<=highlim(ch,b)) = 0; % set unsignificant changes to zero
                psd_temp = logical(psd_temp); % set significant pos. changes to 1
                psd_temp_sig = psd_temp_sig + psd_temp; % comsube neg. and pos. changes in matrix
                psd_sig_sum(ch,b,p) = sum(psd_temp_sig(:)); % sum the significant scores in phase
                psd_mean(ch,b,p) = mean(psd_tempcopy(:)); % take the mean value of phase
                psd_change(ch,b,p) = (mean(psd_tempcopy(:))-psd_a1_mean(ch,b))...
                    ./psd_a1_mean(ch,b)*100; % calculate percentage change in respect to baseline
                p=p+1;
            end
        end
    end
```

```
    % zero s-values outside of interval [-2,2]
    psd_sig_sum(psd_sig_sum>-2 & psd_sig_sum<2) = 0;

    % return result
    psd_sig = psd_sig_sum;

    % return significant changes percentage change score
    psd_sig_mag = logical(psd_sig_sum).*(psd_change);
```

4.3.3 Function: test22SD(…)

```
% Grand Valley State University, Master of Science in Engineering
% ========================================================================
% author: Nadina Zweifel
% date: 3/16/2016
% advisor: Dr. Samhita Rhodes
% ------------------------------------------------------------------------
% function: [psd_sig,psd_sig_mag,psd_mean,psd_change] = test2SD(psd1,psd2,substd2)
% this function applies the two-standard deviation method to the input
% data which contains data of different phases (baseline and intervention).
% The function assumes that the phases consists of 4 data points and
% test for significance in respect of the first dataset. Significance is tested by
% counting the data points exceeding the 2*SE band (SE=standard error).
% Cases that exceed the band 2 times or more are considered as significant.
% input: psd1, psd2, substd2
% psd1 and psd2 represent the power spectral density of the EEG data under
% consideration. The input data is subned by taking the average across
% data points in a particular frequency range. Hence, psd1 and psd2 contain the
% means of the subs while substd2 contains the standard deviation of each
% sub in the second dataset psd2.
% output: psd_sig, psd_sig_mag, psd_mean, psd_change
% psd_sig is a matrix containing the significance score (s-value) for each
% sub and channel in respect of the baseline. Unsignificant changes are
% represented with a zero. The s-value is 2 or higher which is considered
% as significant. psd_sig_mag is of the same type as psd_sig but instead of
% s-values it contains the percentage change score for signifcant changes.
% psd_mean returns just the mean value for each sub and channel for each
% phase. psd_change contains the percentage change score without
% significance testing.
% ------------------------------------------------------------------------

function [psd_sig,psd_sig_mag,psd2_mean,psd_change] = test22SD(psd1,psd2,substd2)
    [nchannels,nsubs,ntrials] = size(psd1);

     % iteration over channels and subs to determine significant changes
     for chn = 1:nchannels
        for sub = 1:nsubs
            phase=1;
            for trial = 1:4:ntrials

                % calculate the baseline values in phase p of first dataset
                psd1_mean = mean(psd1(chn,sub,trial:trial+3),3); % average across 4 trials
                psd1_SE = std(psd1(chn,sub,trial:trial+3),0,3); % st.dev. across 4 trials
                psd1_SE = psd1_SE/sqrt(4); % standard error across 4 trials

                % calculate bounderies for significance testing
                lowlim = psd1_mean-2*psd1_SE;
                highlim = psd1_mean+2*psd1_SE;

                psd2_tempcopy = psd2(chn,sub,trial:trial+3); % copy phase p of second dataset
                error = substd2(chn,sub,trial:trial+3); % copy error of phase p of second dataset
                % test negative changes for significance
                psd2_temp = psd2_tempcopy+error; % add st.dev. to each trial
                psd2_temp(psd2_temp>=lowlim) = 0; % set unsignificant changes to zero
                psd2_temp_sig = -logical(psd2_temp); % set significant neg. changes to -1
```

```
                % test positive changes for significance
                psd2_temp = psd2_tempcopy-error;  % subtract std.dev. from each trial
                psd2_temp(psd2_temp<=highlim) = 0; % set unsignificant changes to zero
                psd2_temp = logical(psd2_temp); % set significant pos. changes to 1
                psd2_temp_sig = psd2_temp_sig + psd2_temp; % comsube neg. and pos. changes in
matrix
                psd_sig_sum(chn,sub,phase) = sum(psd2_temp_sig(:)); % sum the significant scores
in phase
                psd2_mean(chn,sub,phase) = mean(psd2_tempcopy(:)); % take the mean value of phase
                psd_change(chn,sub,phase) = (mean(psd2_tempcopy(:))-psd1_mean)...
                    ./psd1_mean*100; % calculate percentage change in respect to baseline
                phase=phase+1;
            end
        end
    end

    % zero s-values outside of interval [-2,2]
    psd_sig_sum(psd_sig_sum>-2 & psd_sig_sum<2) = 0;

    % return result
    psd_sig = psd_sig_sum;

    % return significant changes percentage change score
    psd_sig_mag = logical(psd_sig_sum).*(psd_change);
```

4.4 SCRIPT READANDSAVEDATA.M

```
% Grand Valley State University, Master of Science in Engineering
% ==========================================================================
% author: Nadina Zweifel
% date: 3/16/2016
% advisor: Dr. Samhita Rhodes
% --------------------------------------------------------------------------
% Master Thesis: Main script for reading the data and randomly pick an
% epoch of 35 seconds. The data is saved to a mat-file which can be loaded
% for further analysis.
% --------------------------------------------------------------------------

clear all, close all, clc

% define sample length for randomly picked epoch
nsamples = 35*128;

% define channels to analize
nchannels = 1:14;

% define filenames for resting 1 data
filenames = {'b1_1_tsk1_wIM125_data.mat','b1_2_tsk1_wIM125_data.mat',...
    'b1_3_tsk1_wIM125_data.mat','b1_4_tsk1_wIM125_data.mat',...
    'i1_1_tsk1_wIM125_data.mat','i1_2_tsk1_wIM125_data.mat',...
    'i1_3_tsk1_wIM125_data.mat','i1_4_tsk1_wIM125_data.mat',...
    'b2_1_tsk1_wIM125_data.mat','b2_2_tsk1_wIM125_data.mat',...
    'b2_3_tsk1_wIM125_data.mat','b2_4_tsk1_wIM125_data.mat',...
    'i2_1_tsk1_wIM125_data.mat','i2_2_tsk1_wIM125_data.mat',...
    'i2_3_tsk1_wIM125_data.mat','i2_4_tsk1_wIM125_data.mat'};

% read data files
data_rest1 = readFiles(filenames,nchannels,nsamples);

% define filenames for resting 2 data
filenames = {'b1_1_tsk1b_wIM125_data.mat','b1_2_tsk1b_wIM125_data.mat',...
    'b1_3_tsk1b_wIM125_data.mat','b1_4_tsk1b_wIM125_data.mat',...
    'i1_1_tsk1b_wIM125_data.mat','i1_2_tsk1b_wIM125_data.mat',...
    'i1_3_tsk1b_wIM125_data.mat','i1_4_tsk1b_wIM125_data.mat',...
    'b2_1_tsk1b_wIM125_data.mat','b2_2_tsk1b_wIM125_data.mat',...
    'b2_3_tsk1b_wIM125_data.mat','b2_4_tsk1b_wIM125_data.mat',...
    'i2_1_tsk1b_wIM125_data.mat','i2_2_tsk1b_wIM125_data.mat',...
    'i2_3_tsk1b_wIM125_data.mat','i2_4_tsk1b_wIM125_data.mat'};

% read data files
data_rest2 = readFiles(filenames,nchannels,nsamples);

% define filenames for interaction data
filenames = {'b1_1_tsk2_wIM125_data.mat','b1_2_tsk2_wIM125_data.mat',...
    'b1_3_tsk2_wIM125_data.mat','b1_4_tsk2_wIM125_data.mat',...
    'i1_1_tsk2_wIM125_data.mat','i1_2_tsk2_wIM125_data.mat',...
    'i1_3_tsk2_wIM125_data.mat','i1_4_tsk2_wIM125_data.mat',...
    'b2_1_tsk2_wIM125_data.mat','b2_2_tsk2_wIM125_data.mat',...
    'b2_3_tsk2_wIM125_data.mat','b2_4_tsk2_wIM125_data.mat',...
    'i2_1_tsk2_wIM125_data.mat','i2_2_tsk2_wIM125_data.mat',...
    'i2_3_tsk2_wIM125_data.mat','i2_4_tsk2_wIM125_data.mat'};

% read data files
data_interact = readFiles(filenames,nchannels,nsamples);

% save variables to mat-file
save('data.mat');
```

4.5 FUNCTION: READFILES()

```
% Grand Valley State University, Master of Science in Engineering
% ==========================================================================
% author: Nadina Zweifel
% date: 3/16/2016
% advisor: Dr. Samhita Rhodes
% --------------------------------------------------------------------------
% function: data = readFiles(filename,nchannels,nsamples)
% input: filename, nchannels, nsamples
% filename is a cell array with all the filenames to be loaded
% nchannels fines the number of channels, while nsamples defines the length
% of the epoch which is randomly picked from each file.
% output: data
% data returns a cell array with the loaded data and represents a randomly
% picked epoch of the available data in the file.
% --------------------------------------------------------------------------

function data = readFiles(filename,nchannels,nsamples)

% get number of files
nfiles = size(filename,2);

% read files and pick random epoch of length nsamples
for nf = 1:nfiles
    file = filename{nf}; % get filename
    load(file); % load file
    [pathstr,name,ext] = fileparts(file); % get variable name
    dataset = eval(name); % get values from variable
    % pick epoch randomly from dataset
    data{nf} = pickRandomEpoch(dataset,nchannels,nsamples); % save data in variable
end

% function to randomly pick an epoch of the input data
function epoch = pickRandomEpoch(data,nchannels,nsamples)

% determine length of input data
Namples = size(data,2);

% define right limit
smpmax = Namples - nsamples;

% prevent exceeding indexes
if smpmax <= 0
    tmin = 1;
else
    % get random index for left limit
    tmin = randi(smpmax,1);
end

% return epoch between left and right limit
epoch = data(nchannels,tmin:tmin+nsamples-1);
```

4.6 FUNCTION: PSDBINSWITHERR(…)

```
% Grand Valley State University, Master of Science in Engineering
% ==========================================================================
% author: Nadina Zweifel
% date: 3/16/2016
% advisor: Dr. Samhita Rhodes
% --------------------------------------------------------------------------
% function: [pxx,pxx_subs,pxx_error,freq_subs] = psdsubswitherr(data,llimit,hlimit,wsub,norm)
% input: data, llimit, hlimit, wsub, norm
% data is a matrix with the time series where the rows are the channels and
% the columns the samples. llimit and hlimit define the frequency range in
% Hz to be returned from the spectrum. wsub defines the sub with in samples
% while norm is a flag value that determines the normalization method.
% output: pxx, pxx_subs, pxx_error freq_subs
% pxx returns the unsubned spectrum normalized by the method defined in
% norm. pxx_subs returns the spectrum which is subned over the width of wsub
% samples. pxx_error returns the standard error resulting from the sub width
% wsub. freq_subs is frequency vector containing the left frequency value
% in Hz.
% --------------------------------------------------------------------------

function [pxx,freq,pxx_subs,pxx_error,freq_subs] = ...
    psdsubswitherr(data,llimit,hlimit,wsub,norm)

% calculate optimal FFT length
Fs = 128;   % sampling frequency
wlength= 320;   % window length for Welch's method in samples
NFFT = 2^nextpow2(wlength+1); % calculate FFT length

% convert frequency limits to samples
lfsmp = floor(llimit*4);
hfsmp = ceil(hlimit*4);

% determine number of channels
numch = size(data,1);

% compute spectrum iterating over channels using the Welch's method with a
% hanning window of wlength
for ch = 1:numch
    [pxxch,freq] = pwelch(data(ch,:),hanning(wlength),[],NFFT,Fs);
     pxx(ch,:) = pxxch;
end

% normalize by channel mean power if norm=0
if norm==0
    pxx = pxx./repmat(mean(pxx,2),[1 size(pxx,2)]);
%     disp('PSD normalized to channel power before subning');
end

% normalize by overall mean power if norm=1
if norm==1
    pxx = pxx./mean(pxx(:));
%     disp('PSD normalized to overall power before subning');
end

% compute the mean of each sub of width wsub between the frequency limits
% lfsmp and hfsmp
for ch = 1:numch
    b=1;
    for f = lfsmp:wsub:hfsmp-wsub
        pxx_subs(ch,b) = mean(pxx(ch,f:f+3));
        pxx_error(ch,b) = std(pxx(ch,f:f+3))/sqrt(wsub);
        b=b+1;
    end
end
```

```
% normalize to the channel mean power after subning if norm=2
if norm==2
    pxx_subs = pxx_subs./repmat(mean(pxx_subs,2),[1 size(pxx_subs,2)]);
%     disp('PSD normalized to channel power after subning');
end

% normalize to the overall mean power after subning if norm=3
if norm==3
    pxx_subs = pxx_subs/mean(pxx_subs(:));
%     disp('PSD normalized to overall power after subning');
end

% return frequency vector for computed subs
freq_subs = (lfsmp:wsub:hfsmp-wsub)/4;
```

4.7 FUNCTION: SETCOLORBAR(…)

```
% Grand Valley State University, Master of Science in Engineering
% ==========================================================================
% author: Nadina Zweifel
% date: 3/16/2016
% advisor: Dr. Samhita Rhodes
% --------------------------------------------------------------------------
% function: setColorbar(zmin,zmax,cbname,split)
% input: zmin, zmax, cbname, split, ncol
% the inputs zmin and zmax represent the lower and upper limit of the
% colorbar, while cbname is a string used for labeling the colorbar. The
% variable split is either 1 or 0 and is defines whether the zero point of
% the colorbar is set to white (1) or not (0).
% ncol defines the resolution of the colormap with ncol colors, while ncol
% = 0 is high resultion of 8192 colors using the OpenGL renderer.
% --------------------------------------------------------------------------

function setColorbar(zmin,zmax,cbname,split,ncol)

% reset colormap
colormap default

% create new colormap for change
if ncol
    newmap = jet(ncol+1);
else
    newmap = jet(8191);
    set(gcf,'renderer','OpenGL') % this allows higher resolution for the colors
end

% set colormap limits
caxis([zmin zmax]);
% determine position of the zero point
zeropos = (1-zmax/(zmax-zmin));

% if flag=1 set zero point to white
if split
    ncol = size(newmap,1);
    zpos = ceil(ncol*zeropos); % find zeropoint of colorbar
    newmap(zpos,:) = ones(1,3); %set that position to white
    % activate new colormap
    colormap(newmap);
end

% adjust settings of new colormap
h = colorbar;
set(get(h,'title'),'string',cbname,'fontsize',8);

end
```

5 BIBLIOGRAPHY

1. Kenyon LK, Farris JP, Gallagher C, Hammond L, Webster LM, Aldrich NJ. Power Mobility Training for Young Children with Multiple, Severe Impairments: A Case Series. *Phys Occup Ther Pediatr*. 2016;2638(January):1-16. doi:10.3109/01942638.2015.1108380.
2. Kenyon LK, Farris J, Brockway K, Hannum N, Proctor K. Promoting Self-exploration and Function Through an Individualized Power Mobility Training Program. *Pediatr Phys Ther*. 2015;27(2):200-206. doi:10.1097/PEP.0000000000000129.
3. Casey J, Paleg G, Livingstone R. Facilitating child participation through power mobility. *Br J ofOccupational Ther*. 2013;76(3):158-160.
4. Galloway JC, Ryu JC, Agrawal SK. Babies driving robots: Self-generated mobility in very young infants. *Intell Serv Robot*. 2008;1(2):123-134. doi:10.1007/s11370-007-0011-2.
5. Campos JJ, Anderson DI, Barbu-Roth M, Hubbard EM, Hertenstein MJ, Witherington D. Travel Broadens the Mind. *Infancy*. 2000;1(2):149-219. doi:10.1207/S15327078IN0102.
6. Livingstone R, Field D. The child and family experience of power mobility: A qualitative synthesis. *Dev Med Child Neurol*. 2015;57(4):317-327. doi:10.1111/dmcn.12633.
7. Livingstone R, Field D. Systematic review of power mobility outcomes for infants, children and adolescents with mobility limitations. *Clin Rehabil*. 2014;28(10):954-964. doi:10.1177/0269215514531262.
8. Vargus-Adams J. Health-related quality of life in childhood cerebral palsy. *Arch Phys Med Rehabil*. 2005;86(5):940-945. doi:10.1016/j.apmr.2004.10.036.
9. Kuntzler PM. Independent Mobility is Key to Overall Child Development. *EP Mag*. 2013:53-55. http://www.eparent.com/.

10. Dodge L, Kendall ME, Kendall E. Learning to Move. *Curr Dir Psychol Sci*. 2014;52(4):150-155.

11. Ray WJ, Cole HW. EEG activity during cognitive processing: Influence of attentional factors. *Int J Psychophysiol*. 1985;3(1):43-48. doi:10.1016/0167-8760(85)90018-2.

12. Klimesch W, Doppelmayr M, Russegger H, Pachinger T, Schwaiger J. Induced alpha band power changes in the human EEG and attention. *Neurosci Lett*. 1998;244(2):73-76. doi:10.1016/S0304-3940(98)00122-0.

13. Mak JN, Chan RHM. Spectral Modulation of Frontal EEG Activities During Motor Skill Acquisition : Task Familiarity Monitoring Using Single-Channel EEG. 2013;(1962):5638-5641.

14. Orekhova E V., Stroganova T a., Posikera IN, Elam M. EEG theta rhythm in infants and preschool children. *Clin Neurophysiol*. 2006;117(5):1047-1062. doi:10.1016/j.clinph.2005.12.027.

15. Chanel G, Kronegg J, Grandjean D, Pun T. Emotion Assessment : Arousal Evaluation Using EEG ' s and Peripheral Physiological Signals. 2005.

16. Bekkedal MY V, Rossi J, Panksepp J. Human brain EEG indices of emotions: Delineating responses to affective vocalizations by measuring frontal theta event-related synchronization. *Neurosci Biobehav Rev*. 2011;35(9):1959-1970. doi:10.1016/j.neubiorev.2011.05.001.

17. Balconi M, Pozzoli U. Arousal effect on emotional face comprehension. Frequency band changes in different time intervals. *Physiol Behav*. 2009;97(3-4):455-462. doi:10.1016/j.physbeh.2009.03.023.

18. Kuban K, Leviton A. Cerebral Palsy. *N Engl J Med*. 1994;330(3):188-195.

19. Cans C. Surveillance of cerebral palsy in Europe: a collaboration of cerebral palsy surveys and registers. Surveillance of Cerebral Palsy in Europe (SCPE). *Dev Med Child Neurol*. 2000;42(12):816-824. doi:10.1111/j.1469-8749.2000.tb00695.x.

20. Rumeau-Rouquette C, Grandjean H, Cans C, Du Mazaubrun C, Verrier A. Prevalence and time trends of disabilities in school-age children. *Int J Epidemiol*. 1997;26(1):137-145. doi:10.1093/ije/26.1.137.

21. Rumeau-Rouquette C, du Mazaubrun C, Mlika A, Dequae L. Motor disability in children in three birth cohorts . *Int J Epidemiol* . 1992;21 (2):359. http://gvsu.summon.serialssolutions.com/2.0.0/link/0/eLvHCXMwjV1LSwMxEB6UohSKj2rxCfkDW3ebZNMcRSxe6qn3kscsXdBV6l789042adWC4J4WloRl8phvMt98AeCTcZ7t7Am-VMjJXThRuBxRckOeq7T0iGrqrPo98N-qOYHC72w9bl5ex0296piW0VHcBakwofk-9Agi8zDFn0OgnhIIQkTky_M8k0oVfThIDXbwY-dHZs.

22. Beckung E, Hagberg G. Neuroimpairments, activity limitations, and participation restrictions in children with cerebral palsy. *Dev Med Child Neurol*. 2002;44(5):309-316. doi:10.1111/j.1469-8749.2002.tb00816.x.

23. Palisano R, Rosenbaum P, Walter S, Russell D, Wood E, Galuppi B. Development and reliability of a system to classify gross motor function in children with cerebral palsy. *Dev Med Child Neurol*. 1997;39(2):214-223. doi:10.1111/j.1469-8749.1997.tb07414.x.

24. Soleymani Z, Joveini G, Baghestani AR. The Communication Function Classification System: Cultural adaptation, validity, and reliability of the Farsi version for patients with cerebral palsy. *Pediatr Neurol*. 2015;52(3):333-337. doi:10.1016/j.pediatrneurol.2014.10.026.

25. Sellers D, Mandy A, Pennington L, Hankins M, Morris C. Development and reliability of a system to classify the eating and drinking ability of people with cerebral palsy. *Dev Med Child Neurol*. 2014;56(3):245-251. doi:10.1111/dmcn.12352.

26. Klimesch W. EEG alpha and theta oscillations reflect cognitive and memory performance: a review and analysis. *Brain Res Brain Res Rev*. 1999;29(2-3):169-195. doi:10.1016/S0165-0173(98)00056-3.

27. Bazanova O. Comments for Current Interpretation EEG Alpha Activity: A Review and Analysis. *J Behav Brain Sci*. 2012;02(02):239-248. doi:10.4236/jbbs.2012.22027.

28. Kulak W, Sobaniec W. Quantitative EEG analysis in children with hemiparetic cerebral palsy. *NeuroRehabilitation*. 2005;20(2):75-84.

29. Lee NG, Kang SK, Lee DR, et al. Feasibility and test-retest reliability of an electroencephalography-based brain mapping system in children with cerebral palsy: A preliminary investigation. *Arch Phys Med Rehabil*. 2012;93(5):882-888. doi:10.1016/j.apmr.2011.10.028.

30. Kułak W, Sobaniec W. Spectral analysis and EEG coherence in children with cerebral palsy: spastic diplegia . *Przegląd Lek* . 2003;60 Suppl 1:23. http://gvsu.summon.serialssolutions.com/2.0.0/link/0/eLvHCXMwjV1LS8NAEB6UXgTxWd_KnLwlxuzWNN5EUj150nPZR9YGaszBS_59Z3aT0hYK3gLLhmEzmW92-OYbAJHGSbQRE7Q1ivzZiNzlBMnKSPPoMpFlo5FOpHLrH34pu-0p_EZXcT3_ietq5pmWASgeCKgkgXG2CwPKkQX7-Ed3Uw9ReMxCKUGVUUSMXCwQ2m3aSCI9mE.

31. Rigoldi C, Molteni E, Rozbaczylo C, et al. Movement analysis and EEG recordings in children with hemiplegic cerebral palsy. *Exp Brain Res*. 2012;223(4):517-524.

doi:10.1007/s00221-012-3278-2.

32. Sajedi F, Ahmadlou M, Vameghi R, Gharib M, Hemmati S. Linear and nonlinear analysis of brain dynamics in children with cerebral palsy. *Res Dev Disabil*. 2013;34(5):1388-1396. doi:10.1016/j.ridd.2013.01.016.

33. Kulak W, Sobaniec W, Bockowski L. EEG spectral analysis and coherence in children with hemiparetic cerebral palsy. *Med Sci Monit*. 2005;11(9):CR449-CR455. doi:2440 [pii].

34. Niedermeyer E, Lopes da Silva FH. *Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*. Philadelphia: Lippincott Williams & Wilkins; 2005. http://gvsu.summon.serialssolutions.com/2.0.0/link/0/eLvHCXMwY2AwNtIz0EUrE8yNk00NU5ItzSxT0oCNWGClbJJimGhmmphqZphmmZaGGvEM-qgjGWiHOaYgj2AYABMcsMvObG4OWtDnHhEIH2EBnfQJbJ2A--oWwFrS0MjMAnrkDowP2jcPMhOpSnETZGABbTMQYmBKzRNm4PCFTnGLMIi5Qi6mAeW4goxE2JHSogyybq4hzh.

35. Davidson RJ, Jackson DC, Larson CL. Human Electroencephalography.

36. Tong S, Thankor N V. *Quantitative EEG Analysis Methods and Clinical Applications*. Norwood, MA, USA: Artech House; 2009. http://site.ebrary.com/lib/gvsu/docDetail.action?docID=10312950.

37. Scheeringa R, Bastiaansen MCM, Petersson KM, Oostenveld R, Norris DG, Hagoort P. Frontal theta EEG activity correlates negatively with the default mode network in resting state. *Int J Psychophysiol*. 2008;67(3):242-251. doi:10.1016/j.ijpsycho.2007.05.017.

38. Pfurtscheller G, Lopes Da Silva FH. Event-related EEG/MEG synchronization and desynchronization: Basic principles. *Clin Neurophysiol*. 1999;110(11):1842-1857.

doi:10.1016/S1388-2457(99)00141-8.

39. Klimesch W, Sauseng P, Hanslmayr S. EEG alpha oscillations: the inhibition-timing hypothesis. *Brain Res Rev*. 2007;53(1):63-88. doi:10.1016/j.brainresrev.2006.06.003.

40. Pfurtscheller G. Induced oscillations in the alpha band: functional meaning. *Epilepsia*. 2003;44 Suppl 1:2-8. doi:12001 [pii].

41. Bazanova OM, Vernon D. Interpreting EEG alpha activity. *Neurosci Biobehav Rev*. 2013;44:94-110. doi:10.1016/j.neubiorev.2013.05.007.

42. Barry RJ, Clarke AR, Johnstone SJ, Brown CR. EEG differences in children between eyes-closed and eyes-open resting conditions. *Clin Neurophysiol*. 2009;120(10):1806-1811. doi:10.1016/j.clinph.2009.08.006.

43. Babiloni C, Sarà M, Vecchio F, et al. Cortical sources of resting-state alpha rhythms are abnormal in persistent vegetative state patients. *Clin Neurophysiol*. 2009;120(4):719-729. doi:10.1016/j.clinph.2009.02.157.

44. Palva S, Palva JM. New vistas for alpha- frequency band oscillations. *Trends Neurosci*. 2007;30(4):150-158. doi:10.1016/j.tins.2007.02.001.

45. Hanslmayr S, Klimesch W, Sauseng P, et al. Visual discrimination performance is related to decreased alpha amplitude but increased phase locking. *Neurosci Lett*. 2005;375(1):64-68. doi:10.1016/j.neulet.2004.10.092.

46. Logan LR, Hickman RR, Harris SR, Heriza CB. Single-subject research design: Recommendations for levels of evidence and quality rating. *Dev Med Child Neurol*. 2008;50(2):99-103. doi:10.1111/j.1469-8749.2007.02005.x.

47. Klem GH, Lüders HO, Jasper HH, Elger C. The ten-twenty electrode system of the International Federation. The International Federation of Clinical Neurophysiology .

Electroencephalogr Clin Neurophysiol Suppl . 1999;52 :3. http://gvsu.summon.serialssolutions.com/2.0.0/link/0/eLvHCXMwjV1La8QgEJbCXgqlD_p-gH8gqZqH8VhKQ0895b5MjNKFbbqHhdJ_3xlNQnahpccgGqJmvs-Z8RvGMpWKZM8mGKlI-LvD84bPQHhlQSsrcQSTWRduS88WfpLdDin8tl2l_foj7VfvIdMyAsUjEQNhNB7XF8SRKZ3rbX5Sx52m5TziVlS7ETDsPINeqSVi-bQ3ZU.

48. Olaf H. Introduction to EEG and MEG. *MRC Cogn Brain Sci Unit 2009*. 2013. http://imaging.mrc-cbu.cam.ac.uk/meg/IntroEEGMEG. Accessed March 27, 2016.
49. Reis PMR, Hebenstreit F, Gabsteiger F, von Tscharner V, Lochmann M. Methodological aspects of EEG and body dynamics measurements during motion. *Front Hum Neurosci*. 2014;8(March). doi:10.3389/fnhum.2014.00156.
50. Schneider S, Strüder HK. Functional Neuroimaging in Exercise and Sport Sciences. 2012:197-212. doi:10.1007/978-1-4614-3293-7.
51. Castellanos NP, Makarov V a. Recovering EEG brain signals: Artifact suppression with wavelet enhanced independent component analysis. *J Neurosci Methods*. 2006;158(2):300-312. doi:10.1016/j.jneumeth.2006.05.033.
52. Klemm M, Haueisen J, Ivanova G. Independent component analysis: comparison of algorithms for the investigation of surface electrical brain activity. *Med Biol Eng Comput*. 2009;47(4):413-423. doi:10.1007/s11517-009-0452-1.
53. Cochran WT, Cooley JW, Favin DL, et al. What is the fast Fourier transform? *Proc IEEE*. 1967;55(10):1664-1674. doi:10.1109/PROC.1967.5957.
54. Welch P. The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Trans Audio*

Electroacoust. 1967;15(2):70-73. doi:10.1109/TAU.1967.1161901.

55. Goncharova II, McFarland DJ, Vaughan TM, Wolpaw JR. EMG contamination of EEG: Spectral and topographical characteristics. *Clin Neurophysiol*. 2003;114(9):1580-1593. doi:10.1016/S1388-2457(03)00093-2.

56. Wolery M, Harris SR. Interpreting results of single-subject research designs. *Phys Ther*. 1982;62(4):445-452.

57. Nourbakhsh MR, Ottenbacher KJ. The statistical analysis of single-subject data: a comparative examination. *Phys Ther*. 1994;74(8):768-776.

58. Klimesch W, Doppelmayr M, Wimmer H, et al. Theta band power changes in normal and dyslexic children. *Clin Neurophysiol*. 2001;112(7):1186-1195. doi:10.1016/S1388-2457(01)00543-0.

59. Hagemann D, Naumann E. The effects of ocular artifacts on (lateralized) broadband power in the EEG. *Clin Neurophysiol*. 2001;112(2):215-231. doi:10.1016/S1388-2457(00)00541-1.

60. Sauseng P, Klimesch W, Freunberger R, Pecherstorfer T, Hanslmayr S, Doppelmayr M. Relevance of EEG alpha and theta oscillations during task switching. *Exp Brain Res*. 2006;170(3):295-301. doi:10.1007/s00221-005-0211-y.

61. Filley C. *Neurobehavioral Anatomy (3rd Edition)*. Boulder, CO, USA: University Press of Colorado; 2011. http://site.ebrary.com/lib/gvsu/docDetail.action?docID=10457067.

62. Badcock N a, Mousikou P, Mahajan Y, de Lissa P, Thie J, McArthur G. Validation of the Emotiv EPOC(®) EEG gaming system for measuring research quality auditory ERPs. *PeerJ*. 2013;1:e38. doi:10.7717/peerj.38.

63. David Hairston W, Whitaker KW, Ries AJ, et al. Usability of four commercially-oriented

EEG systems. *J Neural Eng*. 2014;11:046018. doi:10.1088/1741-2560/11/4/046018.

64. Vourvopoulos A, Liarokapis F. Evaluation of commercial brain–computer interfaces in real and virtual world environment: A pilot study. *Comput Electr Eng*. 2014;40(2):714-729. doi:10.1016/j.compeleceng.2013.10.009.

65. Debener S, Minow F, Emkes R, Gandras K, de Vos M. How about taking a low-cost, small, and wireless EEG for a walk? *Psychophysiology*. 2012;49(11):1617-1621. doi:10.1111/j.1469-8986.2012.01471.x.

66. Schlögl a., Keinrath C, Zimmermann D, Scherer R, Leeb R, Pfurtscheller G. A fully automated correction method of EOG artifacts in EEG recordings. *Clin Neurophysiol*. 2007;118(1):98-104. doi:10.1016/j.clinph.2006.09.003.

67. Gasser T, Sroka L, Moecks J. The Correction of EOG Artifacts by Frequency Dependent and Frequency Independent Methods. *Psychophysiology*. 1986;23(6):704-712.

68. Pham TTH, Croft RJ, Cadusch PJ, Barry RJ. A test of four EOG correction methods using an improved validation technique. *Int J Psychophysiol*. 2011;79(2):203-210. doi:10.1016/j.ijpsycho.2010.10.008.

69. Croft RJ, Barry RJ. Removal of ocular artifact from the EEG: A review. *Neurophysiol Clin*. 2000;30(1):5-19. doi:10.1016/S0987-7053(00)00055-1.

70. Freeman WJ, Quiroga RQ, Bash E. *Imaging Brain Function With EEG: Advanced Temporal and Spatial Analysis of Electroencephalographic Signals* . New York, NY : Springer New York ; 2013. doi:10.1017/CBO9781107415324.004.

71. Carvalhaes C, de Barros JA. The Surface Laplacian Technique in EEG: Theory and Methods. 2014. http://arxiv.org/abs/1406.0458.

72. Kayser J, Tenke CE. Principal components analysis of Laplacian waveforms as a generic

method for identifying ERP generator patterns: II. Adequacy of low-density estimates. *Clin Neurophysiol*. 2006;117(2):369-380. doi:10.1016/j.clinph.2005.08.033.

73. Fitzgibbon SP, DeLosAngeles D, Lewis TW, et al. Surface Laplacian of scalp electrical signals and independent component analysis resolve EMG contamination of electroencephalogram. *Int J Psychophysiol*. 2014;97(3):277-284. doi:10.1016/j.ijpsycho.2014.10.006.

74. Becker H, Albera L, Comon P, et al. EEG extended source localization: Tensor-based vs. conventional methods . *NEUROIMAGE* . 2014;96 :143-157. doi:10.1016/j.neuroimage.2014.03.043.

75. Cohen MX. Rigor and replication in time-frequency analyses of cognitive electrophysiology data. *Int J Psychophysiol*. 2016:1-8. doi:10.1016/j.ijpsycho.2016.02.001.

76. Mayfield Clinic & Spine Institute. *Anatomy of the Brain*. Cincinnati, Ohio; 2013. http://www.mayfieldclinic.com/PDF/PE-AnatBrain.pdf.

77. Marcus EM. *Neuroanatomy for the Neuroscientist*. Boston, MA: Springer US; 2011.

78. Widmaier E, aff, ershel, Strang K. *Vander's Human Physiology: 13th Edition*. McGraw-Hill Higher Education; 2013. http://books.google.com/books?id=aXwgAAAAQBAJ&pgis=1. Accessed November 23, 2014.

79. Lopes da Silva F. EEG and MEG: Relevance to neuroscience. *Neuron*. 2013;80(5):1112-1128. doi:10.1016/j.neuron.2013.10.017.

80. Ryynanen ORM, Hyttinen JAK, Laarne PH, Malmivuo JA. Effect of electrode density and measurement noise on the spatial resolution of cortical potential distribution. *IEEE Trans*

Biomed Eng. 2004;51(9):1547-1554.

81. Akalin Acar Z, Makeig S. Effects of Forward Model Errors on EEG Source Localization . *Brain Topogr* . 2013;26 (3):378-396. doi:10.1007/s10548-012-0274-6.

82. Canuet L, Ishii R, Pascual-Marqui RD, et al. Resting-State EEG Source Localization and Functional Connectivity in Schizophrenia-Like Psychosis of Epilepsy . *PLOS ONE* . 2011;6 (11):e27863. doi:10.1371/journal.pone.0027863.

83. Babiloni C, Del Percio C, Lizio R, et al. Cortical sources of resting state electroencephalographic alpha rhythms deteriorate across time in subjects with amnesic mild cognitive impairment. *Neurobiol Aging*. 2014;35(1):130-142. doi:10.1016/j.neurobiolaging.2013.06.019.

84. Rodriguez-Rivera A, Baryshnikov B V, Van Veen BD, Wakai RT. MEG and EEG source localization in beamspace . *IEEE Trans Biomed Eng* . 2006;53 (3):430-441. doi:10.1109/TBME.2005.869764.

85. Xu P, Tian Y, Chen H, Yao D. Lp Norm Iterative Sparse Solution for EEG Source Localization . *IEEE Trans Biomed Eng* . 2007;54 (3):400-409. doi:10.1109/TBME.2006.886640.

86. Gross J. Analytical methods and experimental approaches for electrophysiological studies of brain oscillations. *J Neurosci Methods*. 2014;228:57-66. doi:10.1016/j.jneumeth.2014.03.007.

87. Luck SJ. *Introduction to the Event-Related Potential Technique (2nd Edition)*. Cambridge, MA, USA: A Bradford Book; 2014. http://site.ebrary.com/lib/gvsu/docDetail.action?docID=10883348.

88. Makeig S, Debener S, Onton J, Delorme A. Mining event-related brain dynamics. *Trends*

Cogn Sci. 2004;8(5):204-210. doi:10.1016/j.tics.2004.03.008.

89. Michel CM, Murray MM. Towards the utilization of EEG as a brain imaging tool. *Neuroimage*. 2012;61(2):371-385. doi:10.1016/j.neuroimage.2011.12.039.

90. Wan F, Nan W, Vai MI, Rosa A. Resting alpha activity predicts learning ability in alpha neurofeedback. *Front Hum Neurosci*. 2014;8(July):500. doi:10.3389/fnhum.2014.00500.

91. Wu J, Srinivasan R, Kaur A, Cramer SC. Resting-state cortical connectivity predicts motor skill acquisition. *Neuroimage*. 2014;91:84-90. doi:10.1016/j.neuroimage.2014.01.026.

92. Kang J, Wang L, Yan C, Wang J, Liang X, He Y. Characterizing dynamic functional connectivity in the resting brain using variable parameter regression and Kalman filtering approaches. *Neuroimage*. 2011;56(3):1222-1234. doi:10.1016/j.neuroimage.2011.03.033.

93. Sanei S, Chambers JA. *EEG Signal Processing*. 1st ed. Wiley; 2008.

94. Jung T-P, Makeig S, Stensmo M, Sejnowski TJ. Estimating Alertness from EEG Power Spectrum. *IEEE Trans Biomed Eng*. 1997;44(1):60-69.

95. Wiedemann HR. Hans Berger (1873-1941). *Eur J Pediatr*. 1994;153(10):705. doi:10.1007/s00415-002-0872-4.

96. Klimesch W, Schimke H, Pfurtscheller G. Alpha frequency, cognitive load and memory performance. *Brain Topogr*. 1993;5(3):241-251. doi:10.1007/BF01128991.

97. Mathewson KE, Basak C, Maclin EL, et al. Different slopes for different folks: Alpha and delta EEG power predict subsequent video game learning rate and improvements in cognitive control tasks. *Psychophysiology*. 2012;49(12):1558-1570. doi:10.1111/j.1469-8986.2012.01474.x.

98. Mihajlovi V. The Impact of Head Movements on EEG and Contact Impedance : An

Adaptive Filtering Solution for Motion Artifact Reduction. *36th Annu Int Conf IEEE Eng Med Biol Soc*. 2014:5064-5067. doi:10.1109/EMBC.2014.6944763.

99. Ma J, Tao P, Bayram S, Svetnik V. Muscle artifacts in multichannel EEG: Characteristics and reduction. *Clin Neurophysiol*. 2012;123(8):1676-1686. doi:10.1016/j.clinph.2011.11.083.

100. Junghöfer M, Elbert T, Tucker DM, Rockstroh B. Statistical control of artifacts in dense array EEG/MEG studies. *Psychophysiology*. 2000;37(4):523-532. doi:10.1111/1469-8986.3740523.

101. Jung T-P, Humphries C, Lee T-W, et al. Removing electroencephalographic artifacts: comparison between ICAand PCA. *Neural Networks Signal Process VIII Proc 1998 IEEE Signal Process Soc Work (Cat No98TH8378)*. 1998. doi:10.1109/NNSP.1998.710633.

102. Brunner DP, Vasko RC, Detka CS, Monahan JP, Reynolds CF, Kupfer DJ. Muscle artifacts in the sleep EEG: Automated detection and effect on all-night EEG power spectra. *J Sleep Res*. 1996;5(3):155-164. http://gvsu.summon.serialssolutions.com/2.0.0/link/0/eLvHCXMw3V1bS8MwFA6CL4KIV5wXyIP4Mir2kq0TfJCtKqIwcPqqSZPgcNZhO8F_7zlJetkE8dnHde0YydeT79y-Q0gYnJx6CzZBMyEj7geyKxjjUTcUXUCL4EoDYVDSn9_4pTJkVl_7Dxt_N8vhCooXjbFrIS8rGfOJUtN2klyZWOCseAeuCmxTqkK5ceGZdOUdmEHgk4.

103. Anderer P, Roberts SJ, Schlögl A, et al. Artifact Processing in Computerized Analysis of Sleep EEG - a Review. *Neuropsychobiology*. 1999;40:150-157.

104. Jung TP, Makeig S, Humphries C, et al. Removing electroencephalographic artifacts by blind source separation. *Psychophysiology*. 2000;37(2):163-178. doi:10.1111/1469-8986.3720163.

105. Boudet S, Peyrodie L, Gallois P, Vasseur C. Filtering by optimal projection and application to automatic artifact removal from EEG. *Signal Processing*. 2007;87(8):1978-1992. doi:10.1016/j.sigpro.2007.01.026.

106. Nolan H, Whelan R, Reilly RB. FASTER: Fully Automated Statistical Thresholding for EEG artifact Rejection. *J Neurosci Methods*. 2010;192(1):152-162. doi:10.1016/j.jneumeth.2010.07.015.

107. Gao JF, Yang Y, Lin P, Wang P, Zheng CX. Automatic removal of eye-movement and blink artifacts from EEG signals. *Brain Topogr*. 2010;23(1):105-114. doi:10.1007/s10548-009-0131-4.

108. Keil A, Debener S, Gratton G, et al. Committee report: Publication guidelines and recommendations for studies using electroencephalography and magnetoencephalography. *Psychophysiology*. 2014;51(1):1-21. doi:10.1111/psyp.12147.

109. McFarland DJ, McCane LM, David S V., Wolpaw JR. Spatial filter selection for EEG-based communication. *Electroencephalogr Clin Neurophysiol*. 1997;103(3):386-394. doi:10.1016/S0013-4694(97)00022-2.

110. Hyvärinen a., Oja E. Independent component analysis: Algorithms and applications. *Neural Networks*. 2000;13(4-5):411-430. doi:10.1016/S0893-6080(00)00026-5.

111. Gwin JT, Gramann K, Makeig S, Ferris DP. Removal of movement artifact from high-density EEG recorded during walking and running. *J Neurophysiol*. 2010;103(6):3526-3534. doi:10.1152/jn.00105.2010.

112. Delorme A, Sejnowski T, Makeig S. Enhanced detection of artifacts in EEG data using higher-order statistics and independent component analysis. *Neuroimage*. 2007;34(4):1443-1449. doi:10.1016/j.neuroimage.2006.11.004.

113. Wacker M, Witte H. Time-frequency techniques in biomedical signal analysis: A tutorial review of similarities and differences. *Methods Inf Med*. 2013;52(4):279-296. doi:10.3414/ME12-01-0083.

114. Weidong ZWZ, Yingyuan LYL. EEG multiresolution analysis using wavelet transform. *2001 Conf Proc 23rd Annu Int Conf IEEE Eng Med Biol Soc*. 2001;2(x):1854-1856. doi:10.1109/IEMBS.2001.1020584.

115. Freeman WJ. Hilbert transform for brain waves. *Scholarpedia*. 2007;2:1338. http://www.scholarpedia.org/article/Hilbert_transform_for_brain_waves.

116. Emotiv I. Emotiv Media. 2014. https://emotiv.com/media/. Accessed April 4, 2016.

117. Urigüen JA, Garcia-Zapirain B. EEG artifact removal-state-of-the-art and guidelines. *J Neural Eng*. 2015;12(3):31001. doi:10.1088/1741-2560/12/3/031001.

118. Belouchrani a., Abed-Meraim K, Cardoso JF, Moulines E. A blind source separation technique using second-order statistics. *IEEE Trans Signal Process*. 1997;45(2):434-444. doi:10.1109/78.554307.

119. Bell AJ, Sejnowski TJ. An information-maximization approach to blind separation and blind deconvolution . *Neural Comput* . 1995;7 (6):1129-1159. http://gvsu.summon.serialssolutions.com/2.0.0/link/0/eLvHCXMw3V1LT4QwEG5MvJgY38Zn0oPxQjBdWqAcPOCKcZUFsgtqvBigbeLBjYmP3--Ux8LuwXj2SNsQ0q988820nUGIWhfEXOIEbtFcuoUrKS9F4ZaCUcJzIqmtLFtVuXV6wM_rXnZt_wF4PzJGEfh247qgzth_An56rh_8JJnE_vDWSGPjKhxF18Y0SPxJ0wmPdeO1Lm.

120. Hyvärinen A, Oja E, Giroux HA. A Fast Fixed-Point Algorithm for Independent Component Analysis . *Neural Comput* . 1997;9 (7):1483-1492.

doi:10.1162/neco.1997.9.7.1483.

121. Palmer J a, Kreutz-Delgado K, Makeig S. AMICA: An Adaptive Mixture of Independent Component Analyzers with Shared Components. 2011:1-15. http://sccn.ucsd.edu/~jason/amica_a.pdf\npapers2://publication/uuid/E6296FC1-7F6B-400C-85D0-3A292A27F710.

122. Mammone N, La Foresta F, Morabito FC. Automatic artifact rejection from multichannel scalp EEG by wavelet ICA. *IEEE Sens J*. 2012;12(3):533-542. doi:10.1109/JSEN.2011.2115236.

123. Zhou W, Gotman J. Automatic removal of eye movement artifacts from the EEG using ICA and the dipole model. *Prog Nat Sci*. 2009;19(9):1165-1170. doi:10.1016/j.pnsc.2008.11.013.

124. Winkler I, Haufe S, Tangermann M. Automatic Classification of Artifactual ICA-Components for Artifact Removal in EEG Signals. *Behav Brain Funct*. 2011;7(1):30. doi:10.1186/1744-9081-7-30.

125. Wallstrom GL, Kass RE, Miller A, Cohn JF, Fox N a. Automatic correction of ocular artifacts in the EEG: A comparison of regression-based and component-based methods. *Int J Psychophysiol*. 2004;53(2):105-119. doi:10.1016/j.ijpsycho.2004.03.007.

126. Jolliffe IT. *Principal Component Analysis*. Secaucus, NJ, USA: Springer; 2002. http://site.ebrary.com/lib/gvsu/docDetail.action?docID=10047693.

127. Zhang J, Sanderson AC. JADE: Adaptive Differential Evolution With Optional External Archive . *IEEE Trans Evol Comput* . 2009;13 (5):945-958. doi:10.1109/TEVC.2009.2014613.

128. Sweeney KT, Ayaz H, Ward TE, Izzetoglu M, McLoone SF, Onaral B. A methodology

for validating artifact removal techniques for physiological signals. *IEEE Trans Inf Technol Biomed*. 2012;16(5):918-926. doi:10.1109/TITB.2012.2207400.