256-CHANNEL EEG AND EYE-TRACKER INTEGRATION AND ASSOCIATED DATA ANALYSIS ON MULTIMEDIA STIMULUS

by

SIDI LIU

(Under the Direction of Tianming Liu)

ABSTRACT

Electroencephalography (EEG) provides significant guidance to multimedia analysis. Meanwhile, analyzing eye movement data is a meaningful method to understand visual perception. Essentially, EEG and Eye tracking records can serve as a bridge that reduces the gap between comprehension of multimedia content and its digital representation. Current research only focuses on EEG and Eye Tracker data in a separated manner to collect relevant information. It is necessary to combine them together to provide meaningful analysis on the mapping between multimedia features and brain's functional response.

In this thesis, I propose a platform to simultaneously record EEG and eye movement data by integrating 256-channel EEG and Eye Tracker devices together. Then a procedure is designed and tested to process the EEG and eye movement records. Finally, an experiment is performed to explore the correlation between multimedia stimulus and corresponding brain and eye response.

INDEX WORDS: Electroencephalography, Eye Tracker, Multimedia, Visualization

256-CHANNEL EEG AND EYE-TRACKER INTEGRATION AND ASSOCIATED DATA ANALYSIS ON MULTIMEDIA STIMULUS

by

SIDI LIU

B.S., Beihang University, China, 2010

A Dissertation Submitted to the Graduate Faculty

of the University of Georgia in Partial Fulfillment

of the

Requirements for the Degree

MASTER SCIENCE

ATHENS, GEORGIA

2015

256-CHANNEL EEG AND EYE-TRACKER INTEGRATION AND ASSOCIATED DATA ANALYSIS ON MULTIMEDIA STIMULUS

by

SIDI LIU

Approved:

Major Professor: Tianming Liu

Committee: Suchi Bhandarkar

Khaled Rasheed

Electronic Version Approved:

Suzanne Barbour

Dean of the Graduate School

The University of Georgia

December 2015

256-Channel EEG and Eye-Tracker Integration and Associated Data Analysis on Multimedia Stimulus

Sidi Liu

November 19, 2015

Acknowledgments

I would like to thank William Oliver in Bio-image Research Center for helping with the implementation and experiment on the EEG device. Thanks for the help of Shijie Zhao, Jinglei Lv, Qinglin Dong and other colleagues in the CAID lab as the subjects to test the platform. Meanwhile I want to thank my committee members for their valuable advice. At the same time, I appreciate the great support from Dr. Richard Neupert and his research group in the Department of Theatre and Film Studies at the University of Georgia. Without their help I could not complete this thesis. Finally, I would like to thank the UGA Willson Center of Humanities and Arts for partially funding this project including the Tobii X2-30 Eye Tracker.

Contents

Chapter 1 Introduction 1
1.1 Multimedia Analysis 1
1.2 Eye Tracking Technology 2
1.3 EEG Technology 3
1.4 Current Research Review and Motivation 7
Chapter 2 System Design and Implementation 9
2.1 System Framework 9
2.2 System Implementation 11
Chapter 3 Experiment Methods 13
3.1 Experiment Procedure 13
3.2 Data Preprocessing 15
3.3 Measurements of Synchrony 18
Chapter 4 Result Discussion 23
4.1 EEG Feature 23
4.2 EEG and Eye Feature 30
4.3 EEG and Eye feature with Shot Detection 32
Chapter 5 Conclusion 40
Chapter 6 Future Work 42
Bibliography 43
Appendix A 46

A.1 Data_acquirement.m 46
A.2 EEG_preprocessing.m 52
A.3 extract_band.m 56
A.4 EEG_and_eye_feature.m 61
A.5 EEG_Eye_correlation.m 64
A.6 EEG_coherence.m 66
A.7 EEG_synchronization_index.m 69
A.8 shot_detection.m 72

Chapter 1

Introduction

This chapter reviews the current literatures of EEG and Eye Tracking technologies applied in multimedia analysis, provides the background of devices and methods used in this thesis, and explains the motivation of simultaneously analyzing EEG and Eye Tracker data.

1.1 Multimedia Analysis

Since we entered a digital multimedia information era, it becomes significant to automatically analyze the semantic content behind a multimedia representation. The ultimate purpose of multimedia analysis is to efficiently access, digest, and retrieve information [1]. There are several strategies for multimedia analysis, including studying users' interactions and

feedbacks, designing biologically-plausible multimedia features, and a combination of low-level features and semantics description of multimedia data [2]. Recently, researchers started applying brain science knowledge and neuroimaging technologies to bridge the semantic gaps in multimedia analysis [3].

Applying neuroscience knowledge such as EEG to analyze the correspondence between low-level multimedia features and high-level semantics is becoming a hot research area in multimedia analysis. EEG-based multimedia technology measures the subject's cognitive processing while the subject is exposed to stimuli. Several researches proved that EEG-guided method can significantly enhance the accuracy of image/video analysis [4, 5].

1.2 Eye Tracking Technology

Eye Tracker is a device used in the scientific study of human perception and vision. By recording human eye movement and position, it provides accurate and quantitative evidence to study visual and attentional processes [6]. Gaze occurs when eyes are relatively stationary in one position. The process of tracking gaze point during eye movement could have a significant impact on high-level multimedia analysis [7].

Several commercial devices have been developed to record and analyze eye movements. In the present study, Tobii X2-30 Eye Tracker is used (Figure 1.1). It is a stand-alone eye tracker that can be used for detailed research of natural behavior. It could accurately record gaze coordination information at a sampling-rate of 40 Hz. Tobii also provides a software development kit that easily allows programming to operate the Eye Tracker for specific experimental purposes.

Figure 1.1: Tobii X2-30 Eye Tracker

1.3 EEG Technology

Electroencephalogram (EEG) is defined as an electrical activity recorded from the scalp head surface after being picked up by metal electrodes and conductive media [8]. In 1875, British physician Richard Caton became the first person to report the successful measurement of electrical activity of an animal brain. The first human EEG experiment was conducted by German Neurologist Hans Berger in 1924. The first international standard for the electrode

was adopted by the International Federation in Electroencephalography and Clinical Neurophysiology in 1958. It is called 10-20 electrode placement system. This system standardized physical placement and designations of electrodes on the scalp. The head is divided into proportional distances from prominent skull landmarks (nasion, preauricular points, inion) to provide adequate coverage of all regions of the brain [9]. Electrode placements are labeled according to adjacent brain areas: F (frontal), C (central), T (temporal), P (posterior), and O (occipital). Odd numbers are at the left side of the head and even numbers accompany the letters on the right side (Figure 1.2) [9].

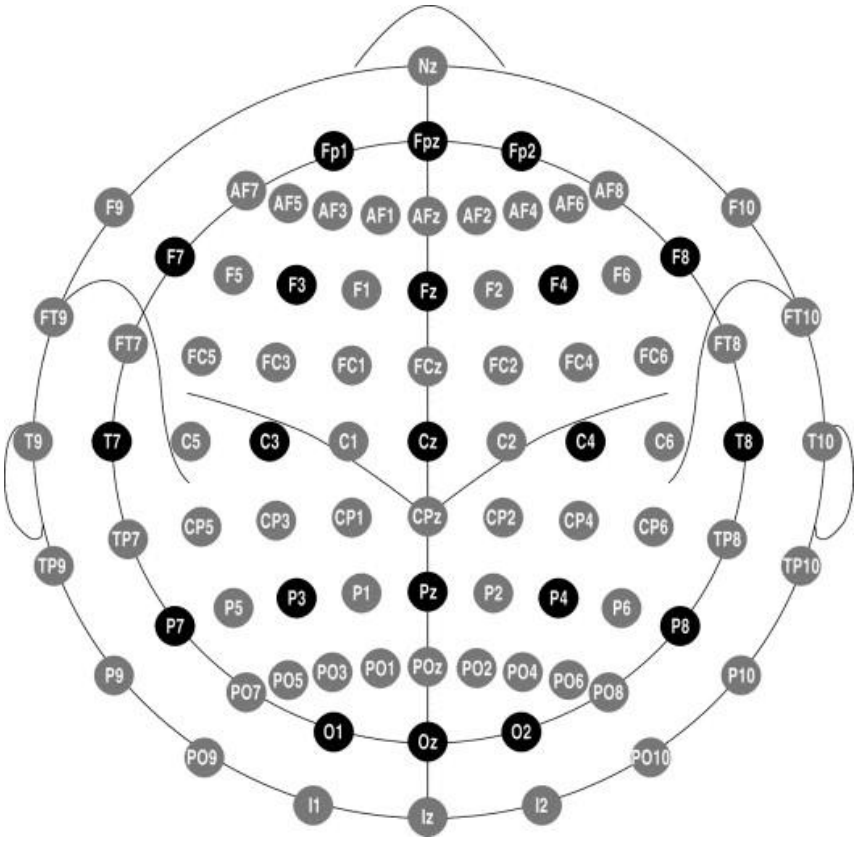


Figure 1.2: Labels for points according to 10-20 electrode placement system

EEG recorded from the surface of the scalp is mainly generated by the synchronous activity of populations of neurons on the cerebral cortex [10]. The system contains four main structures: electrodes capture the digital signal from the scalp, amplifiers enhance the microvolt signals to the proper range where they can be precisely quantified, converters transfer signals from analog to digital form, and finally endpoint devices store and display recorded data. Currently, EEG measurements are usually made with electrode caps. Commonly used electrode caps can comprise up to 128 or 256 channel active electrodes (Figure 1.3, Net Station Acquisition Technical Manual). In this thesis, a 256 channel EEG device manufactured by Electrical Geodesics Inc. is used. The overview of the system is shown in Figure 1.4 (Net Station Acquisition Technical Manual).

Figure 1.3: 256 channel EEG caps.

As the EEG procedure is non-invasive and painless, it is being widely used to study the brain organization of cognitive processes such as perception, memory, attention, language, and emotion in normal adults and children [9]. Nowadays, presenting natural visual and auditory multimedia to participants in EEG-based neurophysiology studies has been well-established [2]. Continuous collaborative efforts from the EEG fields are critically important to decipher the neural semantics of multimedia comprehension.

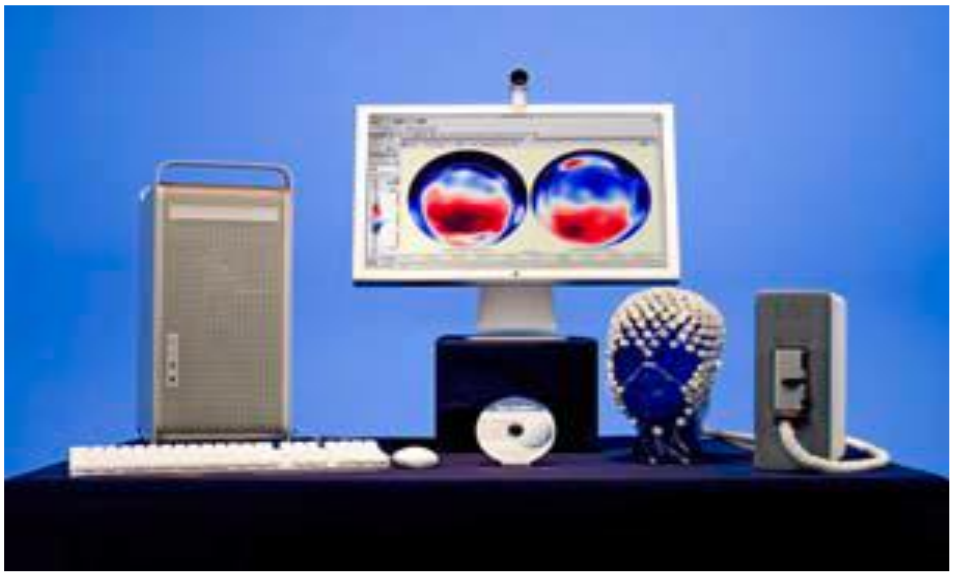


Figure 1.4: Electrical Geodesics Inc. GES 300 system.

1.4 Current Research Review and Motivation

Both EEG and Eye Tracking technology have been widely used in study of multimedia analysis. A. Kapoor developed an EEG device to measure the subconscious cognitive processing that occurs in the brain as users see images, even when they are not trying to explicitly classify them [11]. The framework integrates a discriminative visual category recognition system with information obtained from EEG measurements. The authors validated the framework with experiments using real-world data and reported significant improvement in image classification accuracy [2]. P. Sajda introduced brain-computer interfaces (BCIs) which synergistically integrate computer vision and human vision so as to construct a system for image triage. Their approach exploits machine learning for real-time decoding of brain signals which are recorded noninvasively via EEG [12]. S. Koelstra aimed to find neuro-physiological indicators to validate tags attached to video content. They presented videos and tags to participants and examined whether the shown tags were congruent with the presented videos by detecting the occurrence of an N400 event-related potential [13]. All of the above studies indicate that EEG is an effective method for multimedia analysis.

On the other hand, Eye tracking also has been widely applied to the research of analyzing visual and attentional processes. Different tools have been developed in order to implement specific approaches in eye movement analysis, such as eSeeTrack that examines patterns of sequential gaze recordings (Tsang, Tory, & Swindells, 2010), or aimed to be adapted in existing tools, such as GazeTrackerTM (Lankford, 2000) [6].

Although both EEG and Eye Tracking technologies have been used in a number of researches trying to analyze multimedia, most of them only study EEG and Eye Tracker data in a separated manner. In our view, it is important to combine them together to provide meaningful analysis of the mapping between multimedia features and brain's functional response. Even though some commercial software could analyze EEG and Eye-tracker data

together, those platforms are usually proprietary, non-extensible and non-modifiable [6]. In the present study, we design a platform to simultaneously record EEG and eye movement data by integrating the 256-channel EEG and Eye Tracker device together. Furthermore, an innovative method is provided which tries to combine EEG and Eye Tracking data together to analyze the semantic content of multimedia.

The rest of this thesis is organized as follows. In chapter 2, we explain the details about the structure and implementation of the new platform. Then in chapter 3 a procedure is designed and tested to process the EEG and eye movement records. Finally, an experiment is performed in chapter 4 to explore the connection between multimedia stimulus and corresponding brain and eye response.

Chapter 2

System Design and Implementation

This chapter describes in details about the framework of the system, software implementation, and experiment workflow to simultaneously acquire EEG and Eye Tracking data.

2.1 System Framework

In order to use a Tobii eye tracker and an EGI EEG system in the same experiment, a two computer system is needed: one control computer running Eye-tracker SDK and stimulus, and one Mac computer running Net Station. The framework of the system is shown in Figure 2.1. The Eye Tracker is connected to the control computer via Ethernet (for example by using the

USB-Ethernet adapter provided by Tobii). Additionally, the Net Station computer and the control computer are connected via USB Serial port. In this system, the control computer will present the stimuli, collect and save the Eye Tracking data, and send digital events to the Net Station computer containing information about the onset of each stimulus that has been presented.

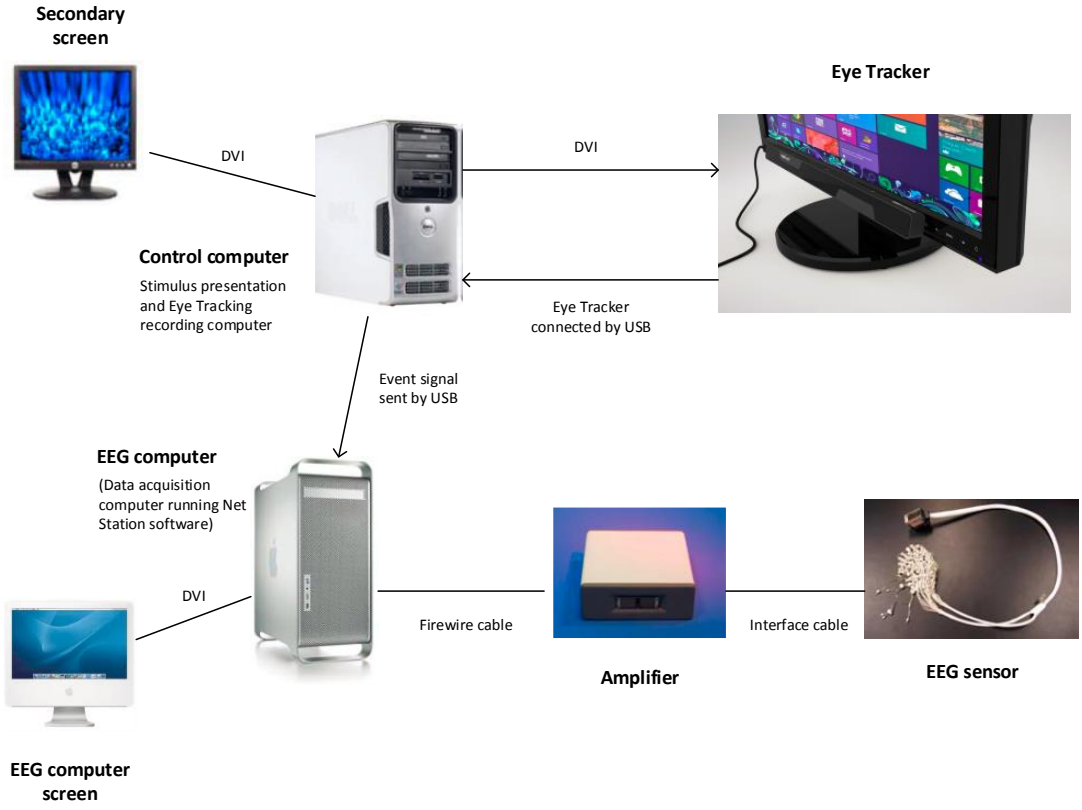


Figure 2.1: Architecture of the system.

2.2 System Implementation

The system is implemented by using the script language of MATLAB. Tobii SDK is able to communicate directly with a Tobii eye tracker and record the eye tracking data in MATLAB. The Net Station SDK from EGI also supports MATLAB format data. Our system integrates the two SDKs through API provided by each of them. The stimuli are presented by the Windows Media Player application. The workflow of the system is shown in Figure 2.2.

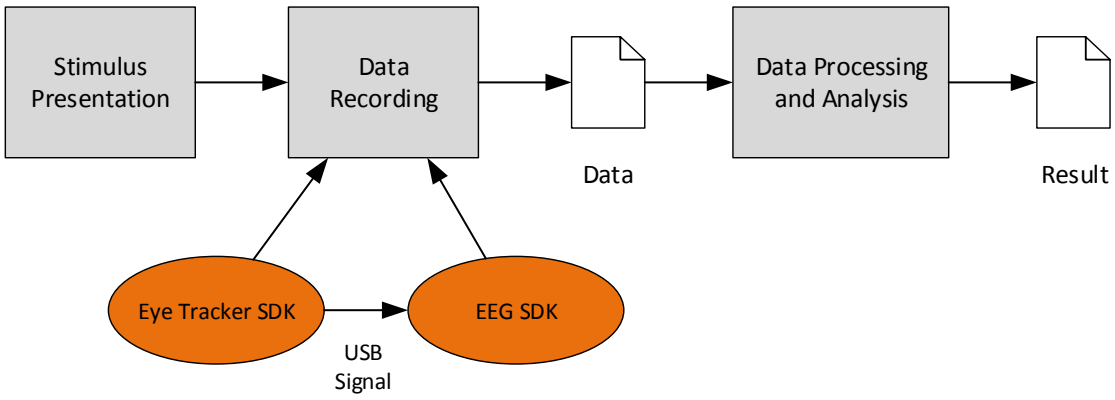


Figure 2.2: System workflow.

One key factor of the system is data alignment, which means how to precisely mark the onset of stimuli on both EEG and Eye Tracking systems. Our method is via sending a binary signal through serial port. At the beginning, Tobii SDK will calibrate the Eye Tracker. Then the Eye tracker control computer begins to record the eye movement data and send information about the onset of the stimuli to the Net Station computer through USB. Meanwhile, the Net station will start to record EEG data. This procedure makes sure that we

can get two different data files (Eye Tracking data and EEG data) with identical event information (time stamps when a certain stimulus was presented). The code of this system is listed in the appendix.

Chapter 3

Experiment Methods

This chapter explains the details about experiment procedure, introduces a method for data pre-processing, and finally introduces three measurements to evaluate the correlation between Eye Tracker and EEG data.

3.1 Experiment Procedure

The experiment is conducted on 6 volunteers, who have no history of neurological illness or damage, and have normal vision. All the participants are between 23 and 30 years of age. Before the experiment, all volunteers receive detailed instructions on the tasks they are going

to perform. Then they are asked to wear the 256 channel EEG caps and sit in a dimly lit, sound-attenuated room at 1.5 meter from a 17-inch PC monitor. The Eye Tracker is re-calibrated for each subject to provide precise measurements of the participant's gaze point during the experiment. Participants view 20 movie trailers in random order, while their Eye Tracking and EEG data is recorded.

The 20 movie trailers are selected from four different genres (Action, Drama, Comedy & Thriller), all of which are released in 2013 in the U.S [22]. They are downloaded from YouTube website. These official trailers are in English language and between 2-3 minutes in length. We select a set of movie trailers that vary considerably in commercial success, in order to avoid that most of the trailers we choose have already been seen by participants. All of the 20 trailers are pre-evaluated by a group of researchers in the Department of Theatre and Film Studies at the University of Georgia, and categorized into two groups: 10 are good movie trailers (group1) and the rest 10 are bad ones (group2). The categorization is based on the evaluation of the audiences. The audiences' evaluations could be reflected by the vote number of audiences who like the trailer and dislike the trailer [22]. The names of the movie trailers are listed in Table 3.1.

NO.	Movie Trailer Name(Group1:Bad)	Movie Trailer Name(Group2:Good)
1	Chairman of the Board	Watchman
2	The Double	Pirates of the Caribbean 3
3	Young Adult	TED
4	Zombie Nation	Seven Psychopaths
5	Sharknado	Hitchcock
6	Assault on Wall Street	The Bucket List
7	Cosmopolis	Jack the Giant Killer
8	The Wicked	Wrath Of The Titans
9	The Room	Mud
10	Bounty Killer	Die Hard 5

Table 3.1: Movie Trailers presented to subjects.

3.2 Data Preprocessing

This section describes the data structure to store Eye Tracker and EEG data. Then a method to clean EEG signal is presented. Finally, we introduce a way to down-sample the EEG signal to match the sampling rate of Eye Tracking data.

3.2.1 Data Structure

Eye Tracking data is stored in MATLAB format. The X and Y gaze positions on the screen are stored separately as two one-dimensional arrays. On the other hand, Net Station SDK uses a matrix to record EEG data. Each row represents one channel of the input signal, and each column represents the time unit.

3.2.2 Eliminate Noise from EEG Signal

EEG raw data often contains several artifacts, such as electrical noise, muscle activity, eye blinks, etc. Thus detecting and cleaning those noises becomes a significant problem in EEG signal processing and research. In the presented study, the raw data is first filtered with a 1 Hz

high-pass filter and 80 Hz low-pass filter to eliminate electronic noise. Then a 60 Hz notch filter is applied to remove alternating current (AC) noise.

The filtered data is then processed using independent component analysis (ICA) to eliminate muscle activity and eye blinks. Proposed by Bell and Sejnowski [14], ICA is suitable for blindly separating mixtures on EEG data, since it is plausible that EEG data recorded at multiple scalp sensors are linear sums of temporally independent components arising from spatially fixed, distinct brain or extra-brain networks [15]. In EEG analysis, each row of the input matrix x is the EEG signal recorded at one electrode and the columns are measurements recorded at different time points. ICA calculates a matrix W, which decomposes or linearly un-mixes the multi-channel scalp data into a sum of temporally independent and spatially fixed components. The rows of the output data matrix, u = Wx, are time courses of activation of the ICA components. The columns of the inverse matrix, W-1, give the relative projection strengths of the respective components at each of the scalp sensors. These scalp weights give the scalp topography of each component, and provide evidence for the components' physiological origins [15]. Thus, corrected EEG data could be derived as x'= (W-1) u', where u' is the matrix of activation waveforms, with rows representing artifactual components being removed. In the experiment, we apply ICA to discard all the distinct artifactual components through visual inspection. The main advantage of using ICA is that it simultaneously separates the EEG and its artifacts into independent components based on the statistics of the data; ICA does this without relying on the availability of one or more reference channels for each type of artifact. This avoids the problem of the potential mutual contamination of regressing and regressed channels [16]. Figure 3.1 shows a schematic illustration of the procedure [17].

We also use standard artifact detection and rejection procedures to remove channels containing jumps larger than 40μV/ms, segments with amplitude differences that exceeded 150μV/200ms, and segments with amplitude differences that did not exceed 0.5μV/200ms [18].

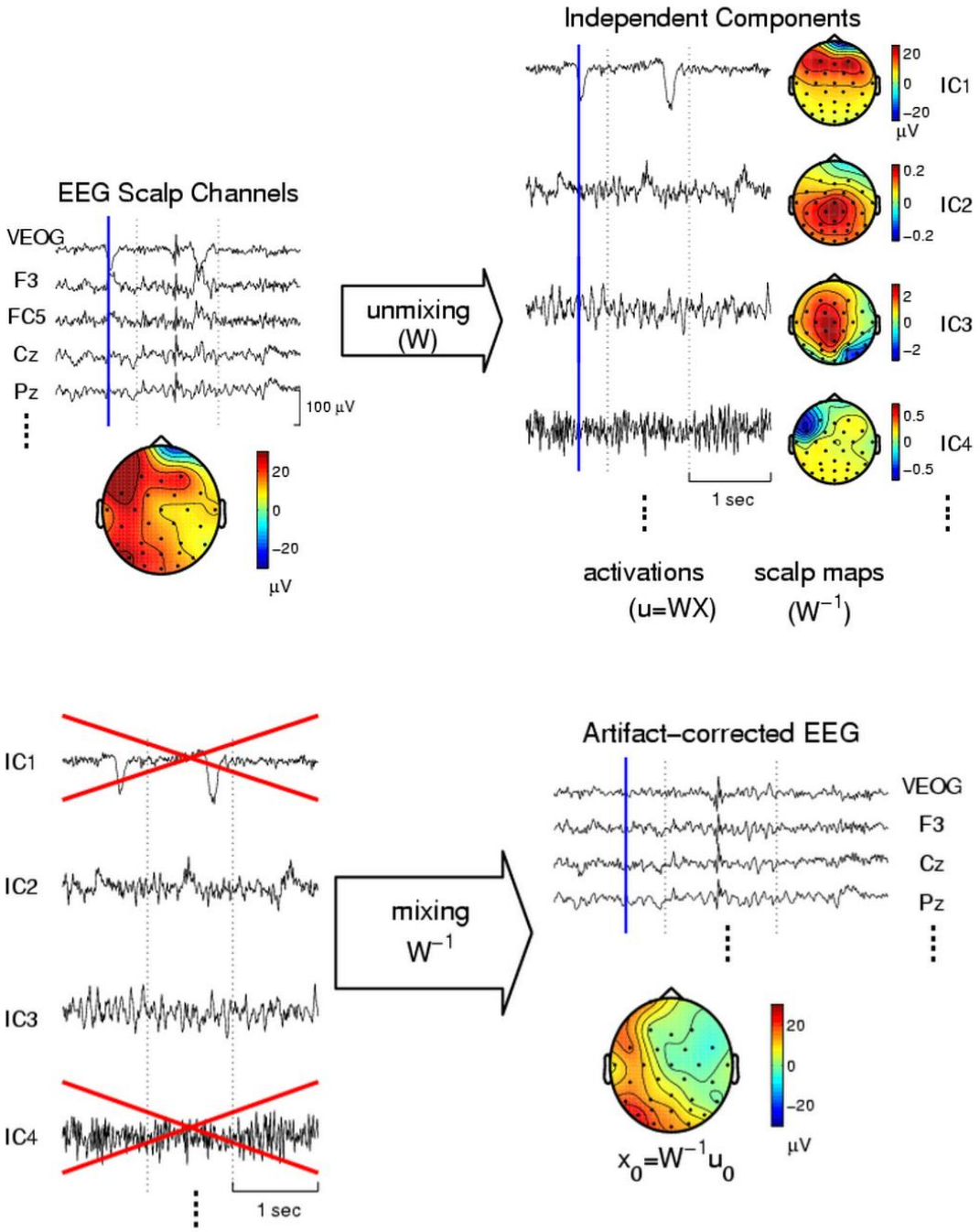


Figure 3.1: Remove artifacts from EEG signal using ICA

3.2.3 Down-sampling EEG Data

In this study, Eye Tracking data is sampled at a rate of 40 Hz, while EEG signals have the sampling rate of 250 Hz. Thus, to analyze the correlation between EEG and Eye Tracking data, down-sampling of EEG data is necessary. In practice, we use the Kaiser Window resampling method implemented in MATLAB. Further, in order to avoid starting and ending artifacts, the first and last 5s of the EEG and Eye Tracking data gathered during the experimental conditions were excluded from the analyses.

3.3 Measurements of Synchrony

The aim of this thesis is to detect the pattern of synchronization between each couple of EEG signals recorded at different cortical sites, as well as to find the correlation between EEG channels and eye movement data. Thus, measurements of signal synchrony act as key factors in our result analysis procedure. In order to achieve this goal, three measurements are examined to investigate the temporal synchronization of the signal.

3.3.1 Correlation coefficient

One of the most widely used measurement of synchronization is the Pearson correlation coefficient. It represents the linear dependence of two random variables, x and y. If each

variable has N scalar observations, then the Pearson correlation coefficient is defined as:

$$r_{xy} = \frac{\sum_{i=1}^{N}(x_i - \overline{x})((y_i - \overline{y})}{N(\sigma_x)(\sigma_y)}$$

Where $\overline{x}$ and σ_x are the mean and standard deviation of X, and $\overline{y}$ and σ_y are the mean and standard deviation of Y. Its absolute value is symmetric in X and Y and attains maximum value of 1 for complete synchronization, a minimum value of -1 for completely uncorrelated signals, and values close to 0 for linearly independent signals [21].

3.3.2 Coherency

Coherency is similarly defined as the standardized cross-spectrum of complex signals X and Y across trials, derived from spectral decompositions of the time series (t) for a given frequency (f), with standardization achieved by dividing the cross-spectrum by the product of the power spectrum of X and the power spectrum of Y. The cross-spectrum, analogous to the covariance in the Pearson correlation equation, is defined as the expected value of the product of the complex signal X and the complex conjugate (denoted by *) of the complex signal Y:

$$S_{xy}(f,t) = \sum_{i=1}^{N} X(f,t)Y^{*}(f,t)$$

The power spectrum of signal X at a given frequency and time across trials, analogous to the variance of x in the Pearson correlation formula, is equivalent to the cross-spectrum of X with itself and is defined as:

$$S_{xx}(f,t) = \sum_{i=1}^{N} X(f,t)X^*(f,t)$$

Y is the same:

$$S_{YY}(f,t) = \sum_{i=1}^{N} Y(f,t)Y^*(f,t)$$

Thus the magnitude squared coherence is defined as the squared absolute value of the cross-spectrum divided by the product of the power spectra of X and Y:

$$Coherence_{xy} = \frac{S_{XY}(f,t)^2}{S_{XX}S_{YY}}$$

Due to its frequency-dependence, coherence is a very useful measure for synchronization in EEG signal [21].

3.3.3 Synchronization Index

The Phase synchronization index has been found widespread use in neurophysiology since the analysis can be restricted to certain frequency bands (i.e. alpha, beta, theta, gamma, delta bands of EEG signal) reflecting specific brain rhythms, which allows relating the results to cognitive processes [21]. Two signals are said to be synchronous if their rhythms coincide. The first step in quantifying phase synchronization between two time series x and y is to determine their instantaneous phases $\emptyset_x(t)$ and $\emptyset_y(t)$. The most common technique is based on the analytic signal approach. From the continues time series x(t), the analytic signal is defined as:

$$Z(t) = \mathrm{x(t)} + \mathrm{i}\tilde{x}(t)$$

Where $\tilde{x}(t)$ is the Hilbert transform of x(t):

$$\tilde{x}(t) = \frac{1}{\pi} p.v. \int_{-\infty}^{\infty} \frac{x(t')}{t - t'} dt'$$

(here p.v. denotes the Cauchy principal value). From Z we can obtain the Hilbert phase:

$$\emptyset_x(t) = \arctan \frac{\tilde{x}(t)}{x(t)}$$

Thus, the synchronization index is defined as:

$$r_{x,y}^2 = < \cos \emptyset_{x,y}(t) >^2 + < \sin \emptyset_{x,y}(t) >^2$$

Where the brackets denote the average over time. The index is confined to the interval [0,1]. Values close to zero are attained for uncorrelated phase difference while the maximum value corresponds to perfect synchronization.

Chapter 4

Result Discussion

This chapter shows the experiment results calculated by using the data processing method mentioned in the last chapter. Then based on the experiment results, several conclusions and hypotheses are discussed.

4.1 EEG Feature

In the first part of the experiment, we selected 76 “good” channels using the noise elimination method described in chapter 3.2.2. As shown in Figure 4.1. Most of the channels are located in frontal, posterior and occipital areas. All of these channels are relabeled from 1 to 76,

according to the ascending order of their original labels. We use the coherence measure to detect phase synchronization of EEG signal and compare the difference between good and bad trailers. Each pair of the 76 electrodes from the left and the right hemispheres are analyzed together to study their relation to the preference of the trailers. By calculating the coherence, we obtain a 76×76 semi matrix for each subject of each trailer, as shown in Figure 4.2. Then for each subject, we take the average of all the 10 good trailers and find the top 50 pairs of EEG channels that have the largest coherence values. Further, we extract the common 10 out of 50 pairs of channels across the 6 participants. See Figure 4.3. The same procedure is operated on the 10 bad trailers. In order to find which pairs of EEG channels have significant variance across good and bad trailers, we also extract the top 10 pairs of EEG that have the largest differences between the two categories, shown in Figure 4.4.

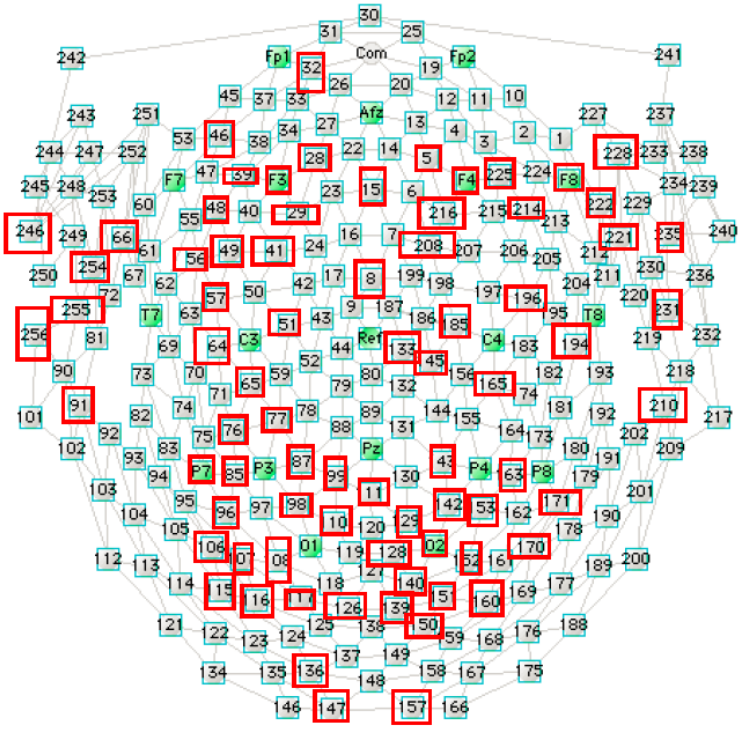
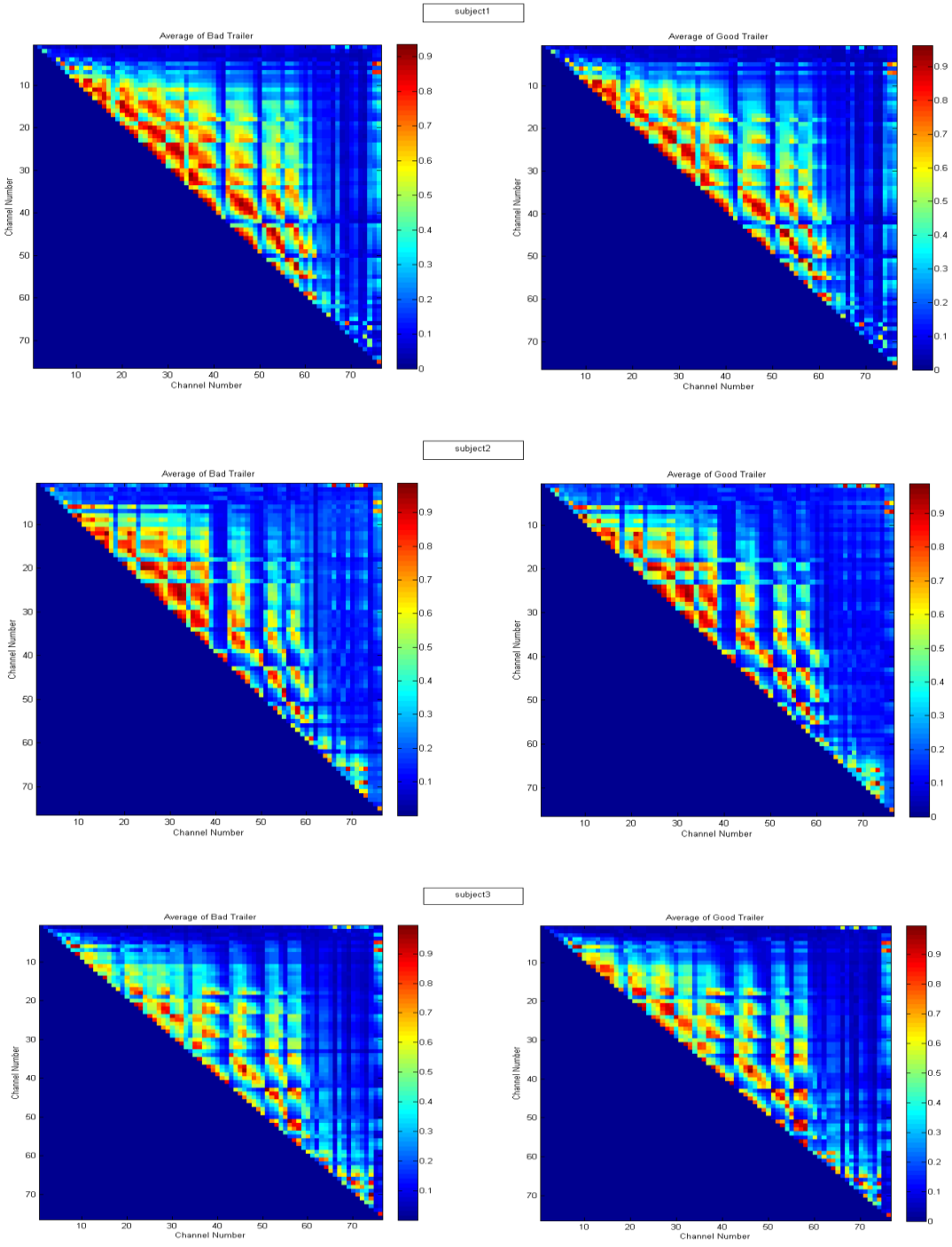


Figure 4.1: Selected 76 EEG channels in 10-20 electrode placement system


subject1
Average of Bad Trailer
Average of Good Trailer
Channel Number
Channel Number
subject2
Average of Bad Trailer
Average of Good Trailer
Channel Number
Channel Number
subject3
Average of Bad Trailer
Average of Good Trailer
Channel Number
Channel Number

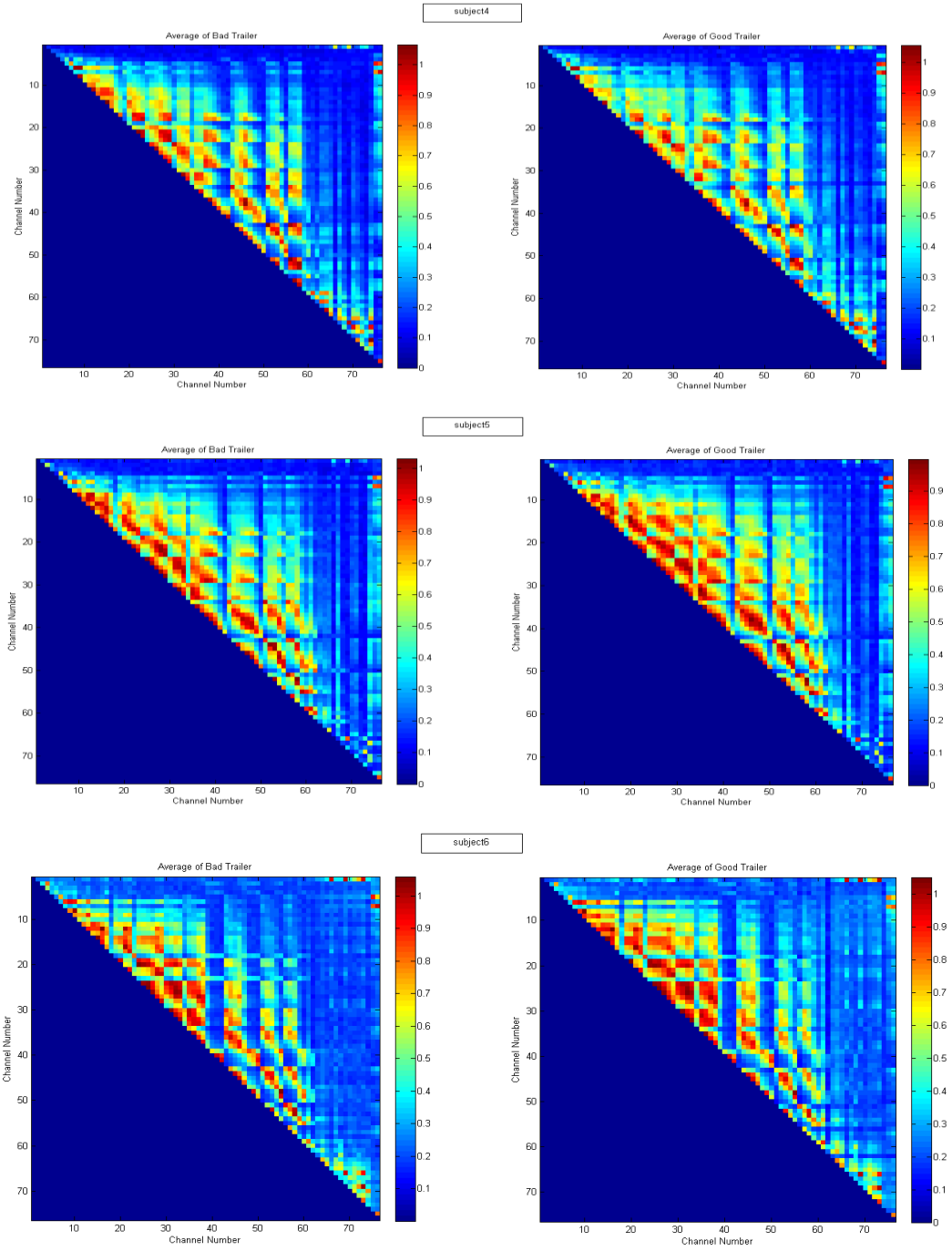


Figure 4.2: Coherence heat map for each pair of EEG channels

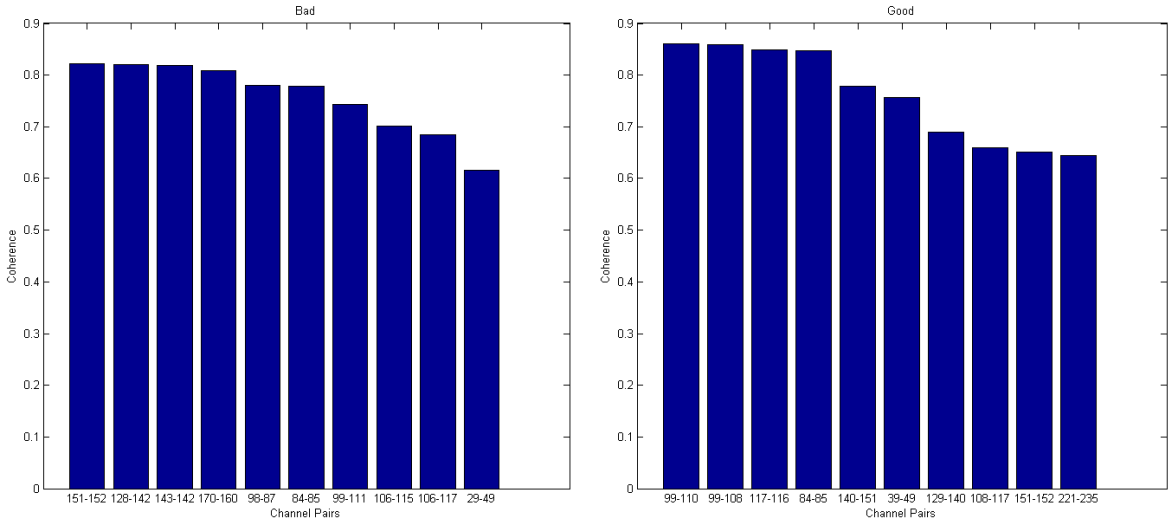


Figure 4.3: Common 10 pairs of EEG channels across 6 subjects

The aim of this study is to find the dynamic coupling between different EEG channels, regarding to changes of synchronization patterns during two kinds of preferential stimulus. The coherence result of each pair of EEG channels suggests a few important findings. First, by analyzing the common top 10 pairs of channels that have the largest value, it is noticeable that most of those channels are located at the frontal (29, 39, 49, 221, 235) and occipital (84, 85, 87, 98, 99, 106, 108, 110, 111, 115, 116, 117, 128, 129, 140, 142, 143, 151, 152, 160, 170, 171) area. This indicates the importance of cognitive processing taking place at these brain regions, which is consistent with the literature published by Costa et al. They propose that large phase synchronization values to the dynamic cooperation between cortical areas highlights the role of information exchange during emotional responses [16]. In brain science, the frontal lobe is associated with human's attention, and the occipital lobe is the visual processing center of the brain, which further supports out discovery. Furthermore, we also observe that the brain activities related to good and bad trailers shows opposite patterns. For good trailers, EEG channels are more synchronized than bad trailers, especially within the left occipital areas (29, 39, 49, 84, 85, 87, 98, 99, 106, 108, 110, 111, 115, 116, 117). On the other

hand, bad trailers are more coherent within the right hemisphere (128, 129, 140, 142, 143, 152, 151, 160, 170, 171, 221, 235).

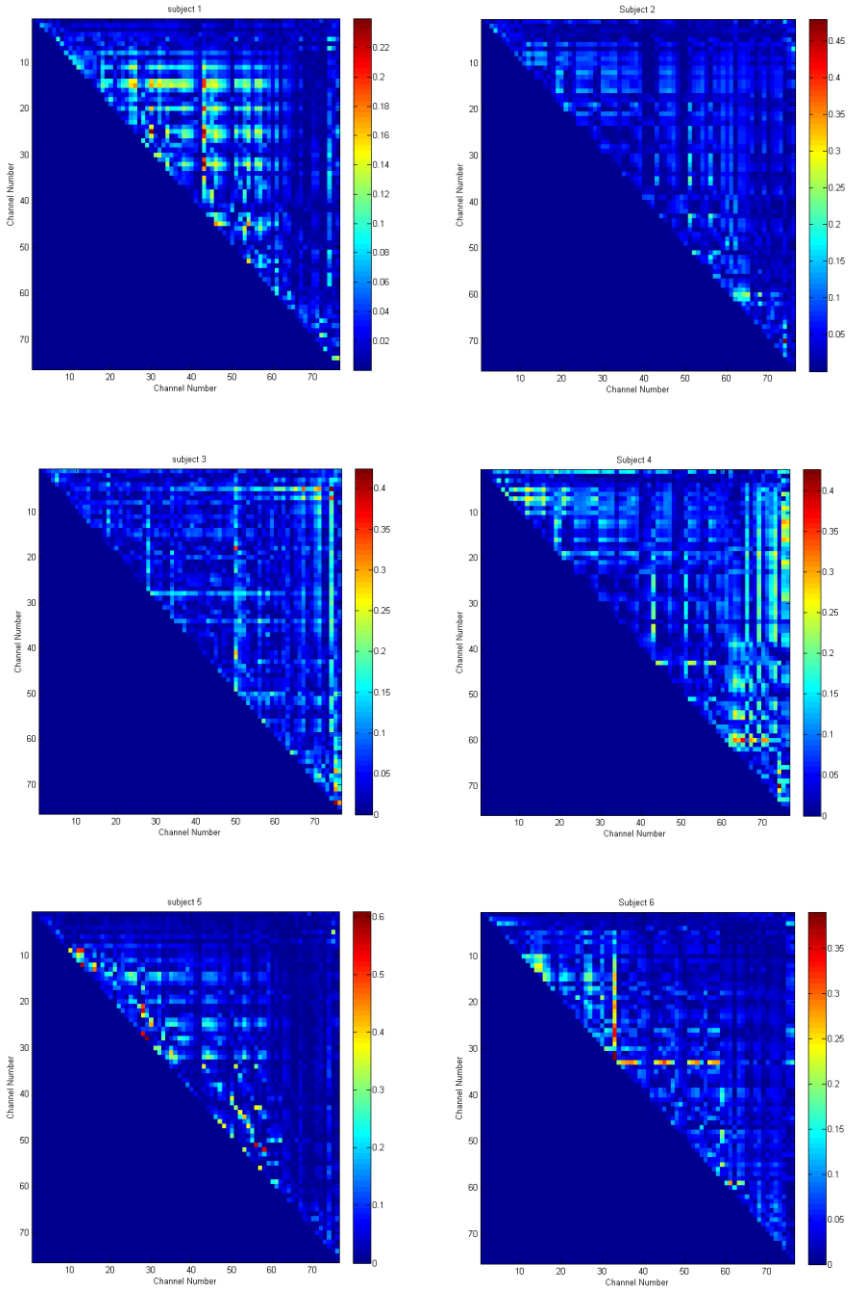


Figure 4.4: Heat map of coherence difference between good and bad trailers

This phenomenon contradicts the discussion indicated by Costa et al. that emotional sadness is more synchronized in left brain region while happiness occurs in the right hemisphere [16]. The difference may be due to the nature of the task itself: preference judgment in our case rather that emotional decision in Costa et al. Moreover, there are 2 pairs (151-152, 84-85) that have larger values in both good and bad trailers, which suggests these pairs are highly related to preference decision. At last, by comparing the top 10 pairs of channels that have the largest differences in coherence value across bad and good trailers, we find that 3 pairs (64-65, 133-8, 165-185) are located in central area, while the other 7 pairs are in frontal or occipital area. This indicates the central cortical area could also significantly reflect human preference towards multimedia stimuli.

EEG data could be separated into five spectral bands: Delta (1-3Hz), theta (4-7Hz), alpha (8-12Hz), beta (13-30Hz) and gamma (35-41Hz). In our experiment, the original data is also separated by five band-pass filters into different spectral bands. Within each band, we calculate the average Synchronization Index (SI) value of each pair of EEG channels across all trailers and subjects. From different cortical regions, we select in total 10 pairs of symmetric channels to analyze: frontal (5-28, 29-216, 35-226), central (65-165, 51-185), temporal (231-255), posterior (85-163), occipital (98-142, 116-160, 106-170), as shown in Figure 4.5. Based on the SI result, we observed several interesting discoveries, including that the frontal and occipital channels are the most synchronized channels compared to the others, which is consistent with the conclusion we mentioned above. The result also clearly indicates the importance of the alpha, beta and theta bands that reflect the highest SI at the frontal and occipital areas. These two brain regions and corresponding bands could be highly relevant to human preference towards multimedia stimuli. Besides symmetric EEG channels, we also find that for all the couples of 76 channels, the set of frontal channels showed higher SI among each other at most of the alpha, beta and theta bands.

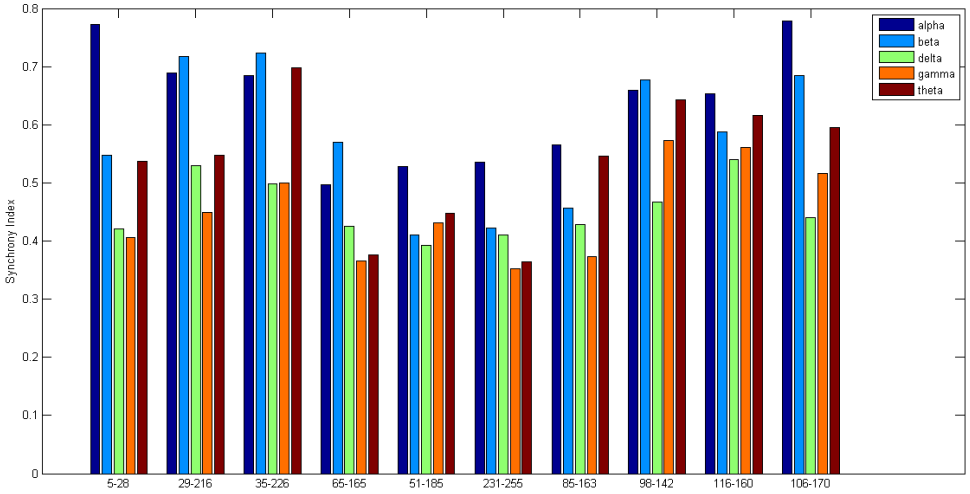


Figure 4.5: SI value in different bands of 10 pairs of channels

On the other hand, the occipital channels show their largest SI with posterior channels, instead of among each other within occipital channels. Taken as a whole, our results support the theory that synchronization provides a useful tool for analyzing and studying variation in brain activities related to subjective preference for multimedia stimuli.

4.2 EEG and Eye Feature

In order to find the correlation between EEG and eye tracking data, we down-sample the EEG data to 40 Hz, which matches the sampling rate of Eye Tracker. Then for each pair of adjacent

EEG data, we calculate the absolute difference, which represents the change of amplitude in each channel. For eye tracking data, we calculate the distance of adjacent gaze point movement. We use correlation measurement to analyze the synchronization between eye movement and each EEG channel. Thus, the analysis identifies how eye movement would affect EEG power, which is in turn represented by the amount of correlation values. The result is computed and graphed in Figure 4.6. For each subject, we take the average correlation value of 10 trailers in the "good" category. Then we extract the common top 10 channels that have the largest values. The same procedure is done to "bad" trailers as well.

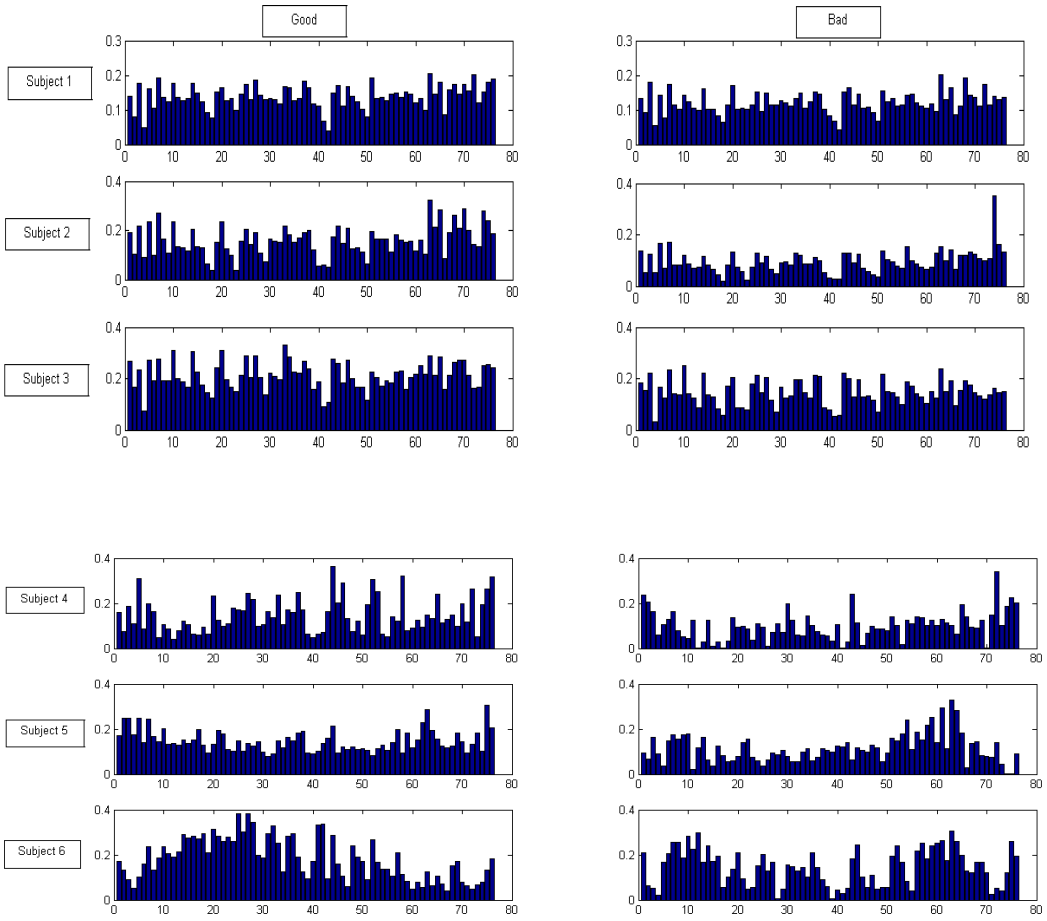


Figure 4.6: Correlation value between eye movement and each EEG channel

First of all, we find that on average, the good trailers have higher correlation value than bad trailers. That could be due to the subject's attention while watching the trailers. There is high probability that participants are more likely to stare at the screen while they are watching the good trailers, so the coherence of their eye movements and brain's functional responses are relatively high. On the other hand, they tend to lose concentration when viewing bad trailers. As a result, their eye movement will be larger and disordered, which leads to less consistency with their EEG responses. Moreover, channels in frontal (15, 32, 214, 208, 254), temporal (91, 256, 231, 246), and occipital (98, 106, 108, 110, 141, 151, 152, 170) regions tend to be highly correlated to eye movements. The occipital response has been related to the encoding of visual stimuli in the literature [19]. In the present study, we suggest that frontal and temporal regions could also have significant impact on human visual response to multimedia stimuli. Finally, we extract the common top 10 EEG channels that have the highest correlation value with eye movement across all categories and subjects: 39, 91, 106, 128, 141, 32, 165, 96, 160, 270. Most of them are located in frontal and occipital areas, which further supports our hypothesis.

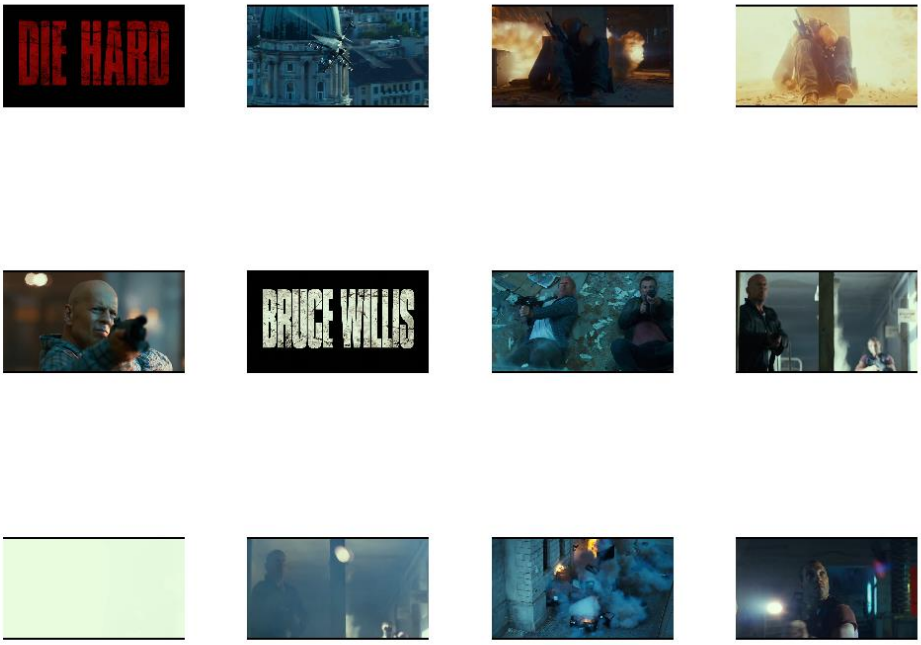
4.3 EEG and Eye feature with Shot Detection

In film studies, a shot is a group of correlated sequential images taken contiguously by a single camera and representing a continuous action in time and space. Shot detection is used to split up a film into basic temporal units [20]. One commonly used method of shot detection is called abrupt video transition detection, which could detect the difference between two transition frames. This difference could be measured based on some global characteristics of frames such as intensity histogram [20]. In a continuous video frame sequence, the histogram difference is not obvious, whereas the difference increases when the frame transition occurs.

Thus the difference of intensity histogram is an effective method in detecting abrupt transitions.

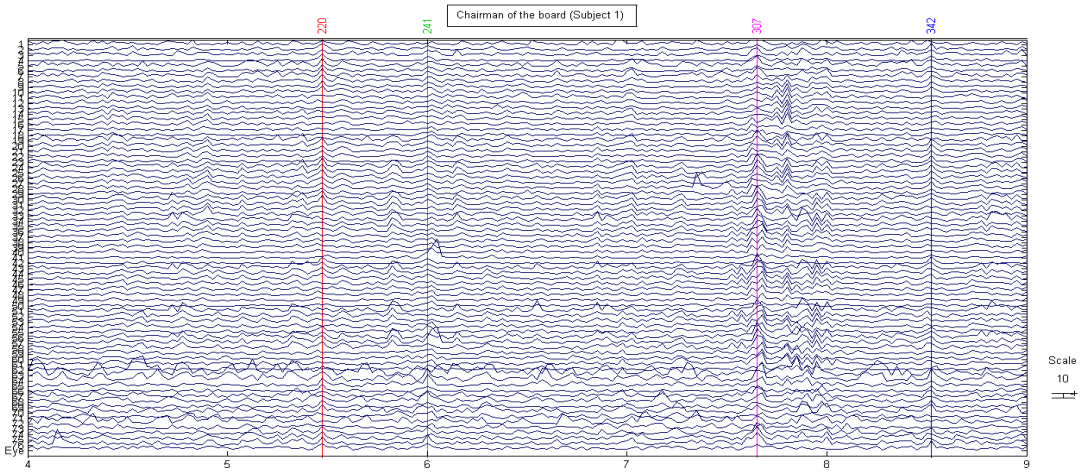
In the presented study, our hypothesis is that EEG and eye movement data would follow the same trend when shot changing occurs. In another word, shot changing could have significant impact on gazing points as well as the human brain's functional response. In order to find this trend, we first apply the shot detection algorithm on all the 20 trailers, and obtain a set of key frames of each trailer. Here we take 2 trailers as an example. From each of them, we select 12 sequential key frames, as shown in Figure 4.7. Then, we visualize all the 76 EEG channels in time series, as well as the eye movement data. We mark all the key frames using colored lines to show the time point where shot change happens. Again we select the same 2 trailers as an illustration, as shown in Figure 4.8. From the figure, we can observe that near the shot changing point, there are peaks in both eye movement data and most of the EEG channels data. The result supports our hypothesis that when shot changing occurs, EEG magnitude variation and eye movement would follow the same trend, especially in frontal, temporal and occipital regions.

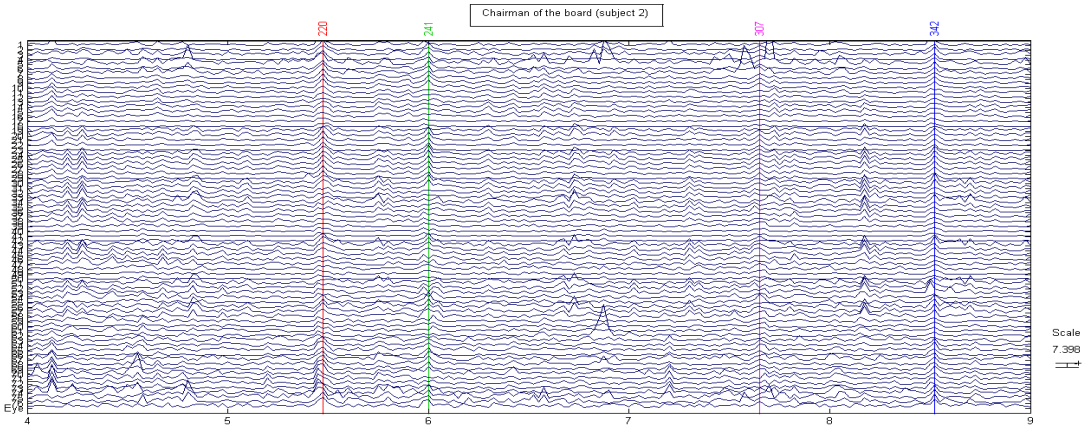
(a) Chairman of the board (Bad trailer)

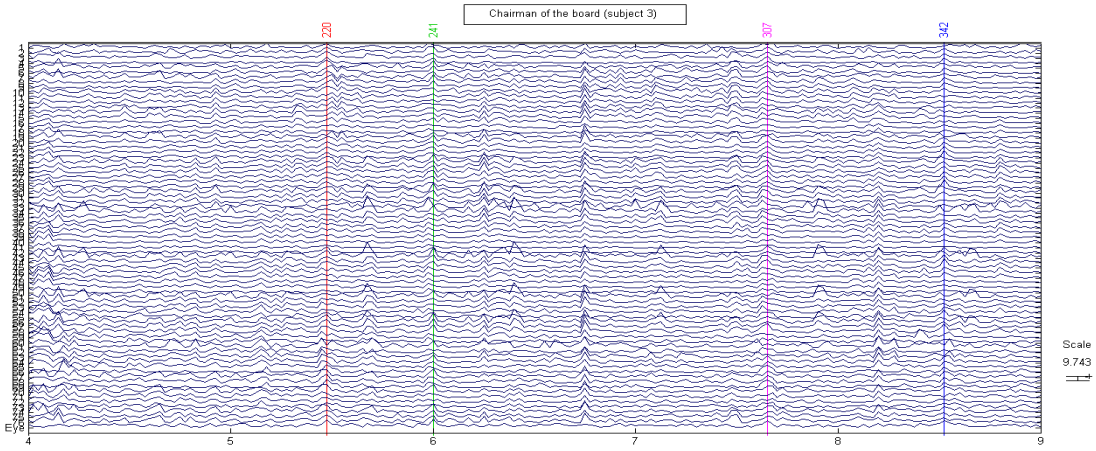


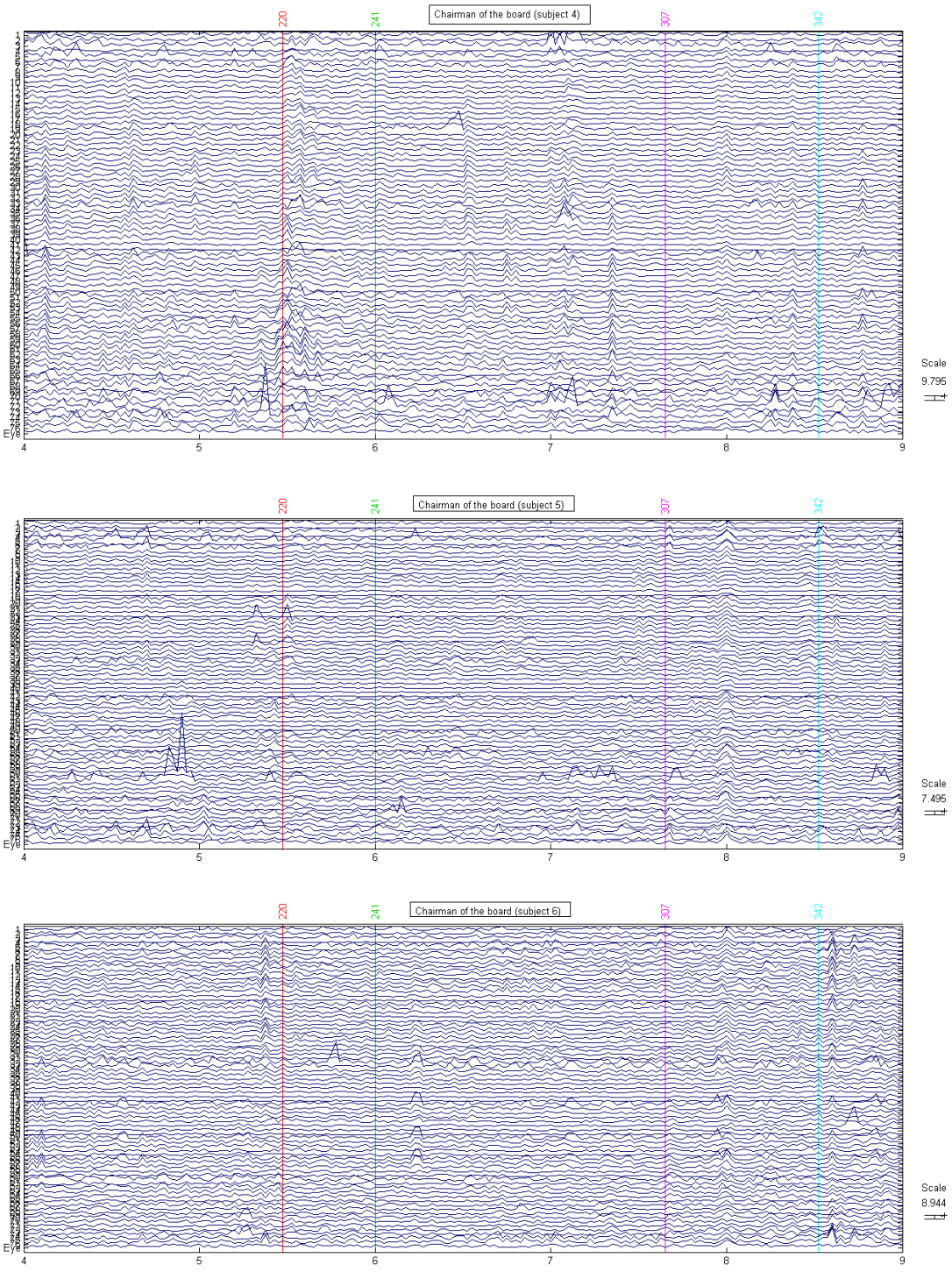
(b) Die Hard 5 (Good trailer)

Figure 4.7: Shot detection of 12 key frames from 2 trailers

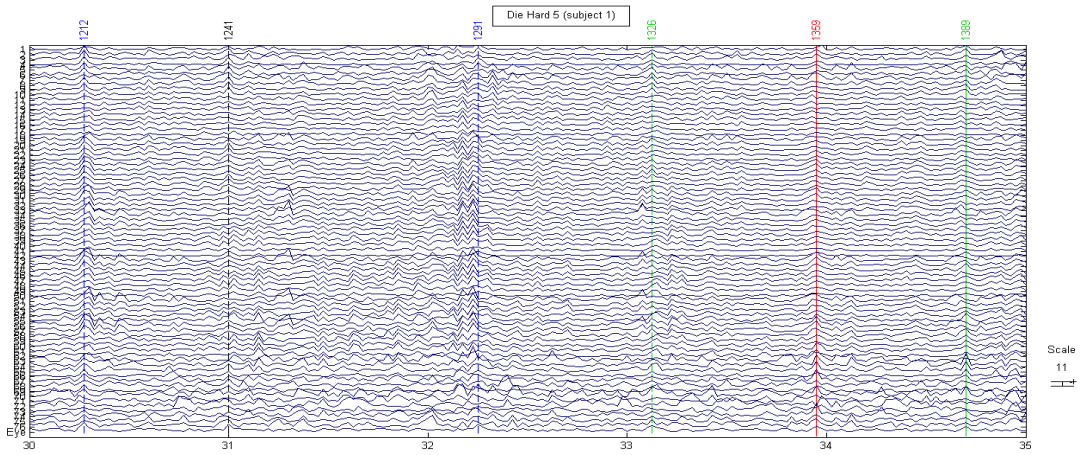

Chairman of the board (Subject 1)
220
241
307
342
Scale
10
4
5
6
7
8
9

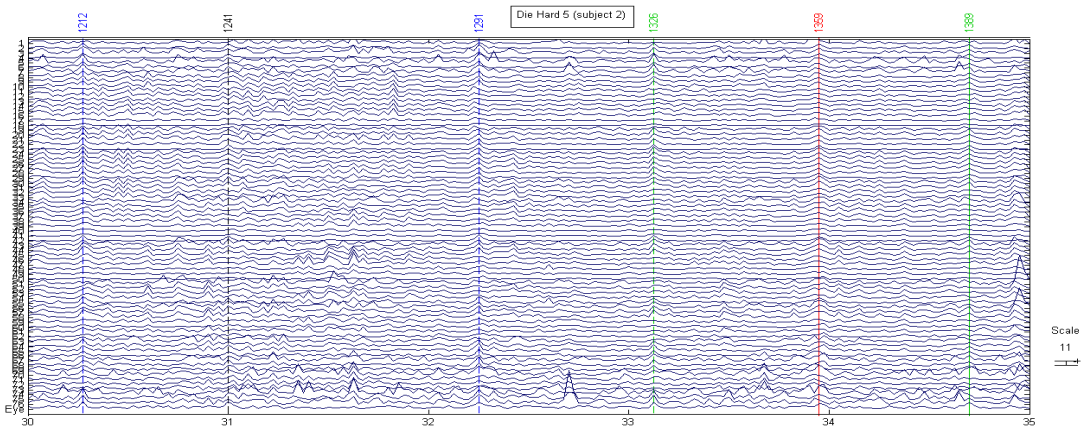

Chairman of the board (subject 2)
220
241
307
342
Scale
7.398
4
5
6
7
8
9

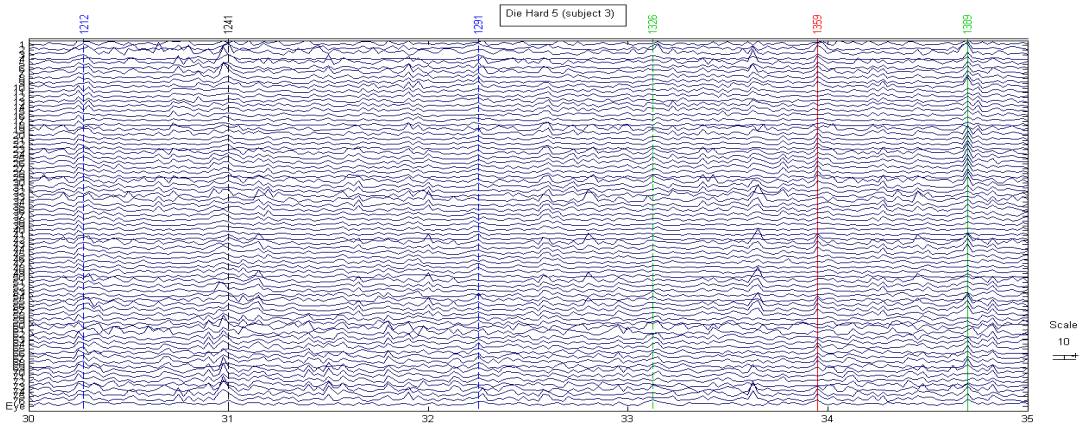

Chairman of the board (subject 3)
220
241
307
342
Scale
9.743
4
5
6
7
8
9

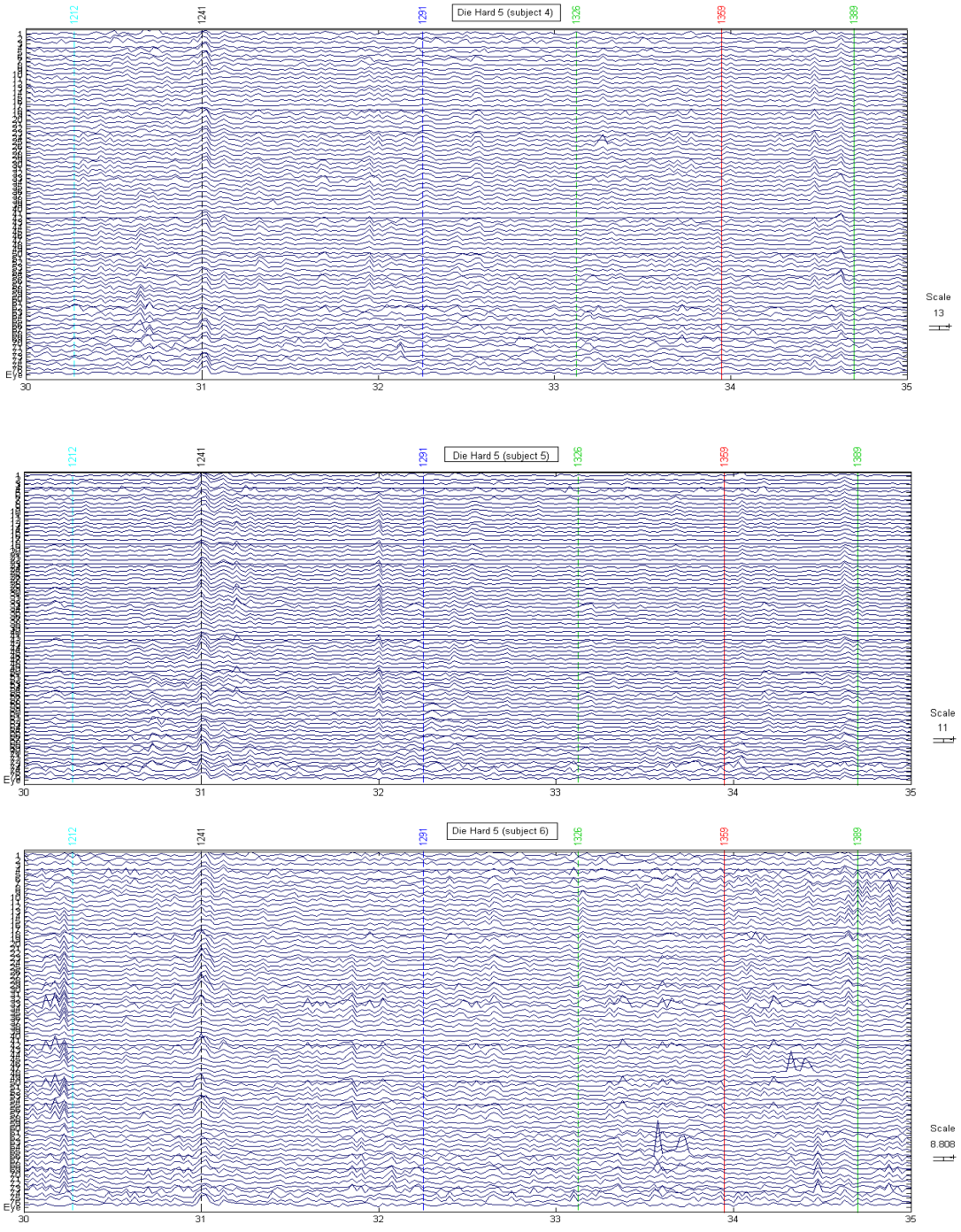


(a) Chairman of the board (Bad trailer)


Die Hard 5 (subject 1)
1212
1241
1291
1326
1369
1389
Scale
11
30
31
32
33
34
35


Die Hard 5 (subject 2)
1212
1241
1291
1326
1369
1389
Scale
11
30
31
32
33
34
35


Die Hard 5 (subject 3)
1212
1241
1291
1326
1369
1389
Scale
10
30
31
32
33
34
35



(b) Die Hard 5 (Good trailer)

Figure 4.7: Shot detection of 12 key frames from 2 trailers

Chapter 5

Conclusion

In this study, we use commercially available Tobii X2-30 Eye Tracker and 256-channel GES 300 EEG system to investigate the eye movement and corresponding brain's functional responses taking place during multimedia stimuli. The main contributions of our work are as follows:

1. We provide a platform to simultaneously record 256-channel EEG and Eye Tracker data. In this thesis, both hardware structure and software implementation are explained in detail.

2. A procedure is designed and tested to process the EEG and eye movement records, including eliminating noise from EEG data, calculating eye gazing point movements, and defining proper standards to measure the synchronization between EEG and Eye Tracking data.

3. A set of 20 movie trailers, categorized by two groups, are presented to 6 subjects. When studying their EEG and eye movement data, several important findings emerges. First, our results support the theory that synchronization between EEG channels provides a significant and useful tool for deeply understanding the variation in different brain regions when multimedia stimuli are presented. Specifically, we find that frontal and occipital regions have larger coherence value than the rest areas, which indicates the importance of cognitive processing taking place at these brain regions. Moreover, we observe that the brain activities related to good and bad trailers shows opposite patterns: EEG channels turn out to be more synchronized than bad trailers within the left occipital areas, whereas bad trailers tend to be more coherent within the right hemisphere. Second, we discover the importance of the alpha, beta and theta bands that reflect the highest SI at the frontal and occipital areas. These two brain regions and corresponding bands could be highly relevant to human preference towards multimedia stimuli. Third, by analyzing EEG and eye movement data together, we find that on average, the good trailers have higher correlation value than bad trailers. Finally, we visualize all the EEG channels and corresponding eye movement in time series, combining with shot detection. The result further supports our hypothesis that when shot changing occurs, EEG and eye movement would follow the same trend, especially in frontal, temporal and occipital regions.

Chapter 6

Future Work

In the final part of this thesis, it should be noted that the present work has a number of limits. First of all, there are a small number of subjects. Thus the statistical analysis is not strong enough to support our conclusion. Here, we report the evidence that EEG and Eye Tracking data could provide indicative information about "good" and "bad" movie trailers. Importantly, more experimental results are needed to establish the robustness of these findings, and whether they can be extended to other types of movies. In addition, in the present study, we only use 3 measurements to represent the synchronization of the data. However, there are other methods to measure the neural signal synchrony, such as Mutual Information and Phase Locking Value. It is necessary to test them to see if we can find other results. Lastly, future work should relate certain EEG channels or brain regions to specific brain functional responses, i.e. adding fMRI data to further analyze the correlation between brain activity and eye movement in response to multimedia stimuli.

Bibliography

[1] Y Wang, Z Liu, JC Huang. "*Multimedia content analysis-using both audio and visual clues".* Signal Processing Magazine, IEEE, Volume: 17, Issue: 6. 2000.

[2] Tianming Liu, Xintao Hu, Xiaojin Li, Mo Chen, Junwei Han, Lei Guo. "*Merging Neuroimaging and Multimedia: Methods, Opportunities and Challenges"*. IEEE Transactions on Human-Machine Systems, 2014.

[3] A. Kapoor, P. Shenoy, and D. Tan, "*Combining brain computer interfaces with vision for object categorization,*" in Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, 2008, pp. 1-8.

[4] S. Scholler, S. Bosse, M. S. Treder et al., "*Toward a Direct Measure of Video Quality Perception Using EEG,*" Image Processing, IEEE Transactions on, vol. 21, no. 5, pp. 2619-2629, 2012.

[5] J. Antons, R. Schleicher, S. Arndt et al., "*Analyzing speech quality perception using electro-encephalography,*" IEEE Journal of Selected Topics in Signal Processing, vol. 6, no.

6, pp. 721-731, 2012.

[6] Vassiliki Filippakopoulou and Byron Nakos. "*Eyemmv toolbox: An eye movement post-analysis tool based on a two-step spatial dispersion threshold for fixation identification*". Journal of Eye Movement Research, 7(1):1–10, 2014.

[7] Salvucci, D. D. "*An interactive model-based environment for eye-movement protocol analysis and visualization*", In Proceedings of the Symposium on Eye Tracking Research and Applications, 57-63. 2000

[8] E. Niedermeyer, F. H. Lopes da Silva. 1993. *Electroencephalography: Basic principles, clinical applications and related fields*, 3rd edition, Lippincott, Williams & Wilkins, Philadelphia.

[9] M.Teplan. *Foundamentals of eeg measurement*. Measurement Science Review, 2, 2002.

[10] Väisänen, O. Multichannel *EEG methods to improve the spatial resolution of cortical potential distribution and the signal quality of deep brain sources*. Tampere, Finland, Tampere University of Technology. Doctoral Dissertation. 2008

[11] A. Kapoor, P. Shenoy, and D. Tan, "*Combining brain computer interfaces with vision for object categorization*" in Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, 2008, pp. 1-8.

[12] P. Sajda, E. Pohlmeyer, J. Wang et al., "*In a blink of an eye and a switch of a transistor: cortically coupled computer vision*," Proceedings of the IEEE, vol. 98, no. 3, pp. 462-478, 2010.

[13] S. Koelstra, C. Muhl, and I. Patras, "*EEG analysis for implicit tagging of video data*" in Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference on, 2009, pp. 1-6.

[14] Bell AJ, Sejnowski TJ. "*An information-maximisation approach to blind separation and*

blind deconvolution". Neural Computation, 7(6):1129-59. 1995

[15] Jung T-P, Humphries C, Lee TW, Makeig S, McKeown MJ, Iragui V, and Sejnowski TJ, "*Removing Electroencephalographic Artifacts:Comparison between ICA and PCA*", Neural Networks for Signal Processing VIII, 63-72, 1998.

[16] Costa, T., Rognoni, E., & Galati, D. "*EEG phase synchronization during emotional response to positive and negative film stimuli*". Neuroscience Letters, 406, 159–164, 2006.

[17] Jung T-P, Makeig S, Humphries C , Lee TW, McKeown MJ, Iragui V, and Sejnowski TJ, "*Removing Electroencephalographic Artifacts by Blind Source Separation,*" Psychophysiology, 37:163-78, 2000

[18] Maarten A.S. Boksem and Ale Smidts (2015). "*Brain Responses to Movie Trailers Predict Individual Preferences for Movies and Their Population-Wide Commercial Success*". Journal of Marketing Research: August 2015, Vol. 52, No. 4, pp. 482-492

[19] Hald, L. A., Bastiaansen, M. C. M., & Hagoort, P. (2006). *EEG theta and gamma responses to semantic violations in online sentence processing*. Brain and Language, 96, 90–105.

[20] Patel, Nilesh V., and Ishwar K. Sethi. "*Video shot detection and characterization for video databases*." Pattern Recognition 30.4 (1997): 583-592.

[21] Roach, Brian J., and Daniel H. Mathalon. "*Event-Related EEG Time-Frequency Analysis: An Overview of Measures and An Analysis of Early Gamma Band Phase Locking in Schizophrenia.*" Schizophrenia Bulletin 34.5 (2008): 907–926. PMC. Web. 14 Oct. 2015.

[22] Yimin Hou, Ting Xiao. el. "*Predicting Movie Trailer Viewer's "Like/Dislike" via Learned Shot Editing Patterns*". IEEE Transactions on Affective Computing, 2015. in press

Appendix A

Main Function Source Code

A.1 Data_acquirement.m

```
% Eye Tracker and EEG data acquirement
% Sidi Liu
% 10/12/2015

 clc
 clear all
 close all

subName = input('Please enter your name :');
movieNum = input('Which movie you want to play :');
```

```
% ************************************************************************
% Open a com port for connecting to EEG
% ************************************************************************
s = serial('COM1');
fopen(s);

% ************************************************************************
% Load Tobii SDK
% ************************************************************************

addpath('functions');
addpath('tetio');

% ************************************************************************
% Initialization and connection to the Tobii Eye-tracker
% ************************************************************************
disp('Initializing tetio...');
tetio_init();

% Set to tracker ID to the product ID of the tracker you want to connect to.
trackerId = '%%% Tracker ID provided by Tobii %%%';

% %   FUNCTION "SEARCH FOR TRACKERS" IF NOTSET
if (strcmp(trackerId, 'NotSet'))
```

```
    warning('tetio_matlab:EyeTracking', 'Variable trackerId has not been set.');
    disp('Browsing for trackers...');

    trackerinfo = tetio_getTrackers();
    for i = 1:size(trackerinfo,2)
        disp(trackerinfo(i).ProductId);
    end

    tetio_cleanUp();
    error('Error: the variable trackerId has not been set. Edit the EyeTrackingSample.m script and replace "NOTSET" with your tracker id (should be in the list above) before running this script again.');
end

fprintf('Connecting to tracker "%s"...\n', trackerId);
tetio_connectTracker(trackerId)

currentFrameRate = tetio_getFrameRate;
fprintf('Frame rate: %d Hz.\n', currentFrameRate);

% ********************************************************************************
% Prepare stimulus
% ********************************************************************************
close all;
movie_name = sprintf('%%%% movieFile %%%%', movieNum);
```

```
movie_path = fullfile('%%% moviePath %%%',movie_name);
mov = VideoReader( movie_name );
movDuration = mov.Duration;
% call for MediaPlayer to show the trailer
h=actxserver('WMPlayer.OCX.7');

hold on;

% ********************************************************************************
%
% Eye Tracker calibration
%
% ********************************************************************************

SetCalibParams;
% Display the track status window showing the participant's eyes (to position the participant).
TrackStatus; % Track status window will stay open until user key press.

% Perform calibration
HandleCalibWorkflow(Calib);
close all

% ********************************************************************************
%
```

```
% Start tracking and gathering the gaze data.
%
% ************************************************************************

tetio_startTracking;

leftEyeAll = [];
rightEyeAll = [];
timeStampAll = [];

%set the sampling rate
pauseTimeInSeconds = 1/currentFrameRate;
durationInSeconds = movDuration;
steps = floor(durationInSeconds/pauseTimeInSeconds);

fprintf('show the movie and start to collect the data\n');
h.openPlayer(movie_path);

% Send signal to EEG to start acquiring EEG data
fwrite(s,bin2dec('1'),'uint8');

for i = 1:steps
    pause(pauseTimeInSeconds);
   [lefteye, righteye, timestamp, trigSignal] = tetio_readGazeData;
```

```
    if isempty(lefteye)
        continue;
    end
    leftEyeAll = vertcat(leftEyeAll, lefteye(:,[7,8,12,13]));
    rightEyeAll = vertcat(rightEyeAll, righteye(:,[7,8,12,13]));
    timeStampAll = vertcat(timeStampAll, timestamp(:,1));
end

tetio_stopTracking;

% send another signal to EEG to mark the end of the trailer
fwrite(s,bin2dec('1'),'uint8');

% close Eye Tracker
tetio_disconnectTracker;
tetio_cleanUp;
h.close;

% close com port
fclose(s);
delete(s);
clear s;

fprintf('writing the data, hold on!\n');
```

```
disp('only collect the gaze data ');
[gazex,gazey]=DisplayData(leftEyeAll, rightEyeAll );
resultEYEfile = sprintf('result-EYE%s%d.mat',subName, movieNum);
save(resultEYEfile, 'leftEyeAll', 'rightEyeAll', 'timeStampAll', 'gazex', 'gazey');

scatter (gazex,gazey,50,'filled');
axis([0 1 0 1]);
resultEYEPosition = sprintf('result-EYE%s%d',subName, movieNum);
saveas(gcf, resultEYEPosition, 'fig')
```

A.2 EEG_preprocessing.m

```
% EEG data pre-processing
% Sidi Liu
% 10/14/2015

clc;
close all;
clear all;

%load EEGLAB
addpath(\eeglab13_4_4b');
```

```
eeglab;

DataSet_catagory = {...
    'bad',...
    'good'};

DataSet_subject = {...
    'subject 1',...
    'subject 2',...
    'subject 3'};

for cataFile = 1:numel(DataSet_catagory)
    cataName=cellstr(DataSet_catagory(cataFile));
    for subFile = 1:numel(DataSet_subject)
        subName=cellstr(DataSet_subject(subFile));
        for iNum=1:10
            eegdatafile = [%%% eegPath %%%];
            movfile = [%%% movPath %%%];
            mov = VideoReader(movfile);
            movDuration = mov.Duration;

            clear EEG;

            % ********************************************************************
```

```
% Load EEG data to EEGLAB.
% ********************************************************************
EEG = pop_readegi(eegdatafile, [],[],'auto');
EEG = eeg_checkset( EEG );
% ********************************************************************
% Load EEG channel location file.
% ********************************************************************
EEG=pop_chanedit(EEG, 'load',{'G:\\matlab\\MS\\code\\eeglab13_4_4b\\chanloc\\256.elp' 'filetype' 'autodetect'});
EEG = eeg_checkset( EEG );
% ********************************************************************
% Eliminate bad channels.
% ********************************************************************
EEG = pop_select( EEG,'nochannel',{'E72' 'E81' 'E90' 'E91' 'E101' 'E102' 'E112' 'E121' 'E134' 'E146' ...
   'E166' 'E175' 'E188' 'E200' 'E209' 'E210' 'E217' 'E218' 'E219' 'E220' ...
   'E229' 'E230' 'E231' 'E232' 'E233' 'E234' 'E235' 'E236' 'E237' 'E238' ...
   'E239' 'E240' 'E241' 'E242' 'E243' 'E244' 'E245' 'E246' 'E247' 'E248' ...
   'E249' 'E250' 'E253' 'E254' 'E255' 'E256'});
EEG = eeg_checkset( EEG );
EEG = pop_rejchan(EEG, 'elec',[1:210] ,'threshold',5,'norm','on','measure','kurt');
EEG = eeg_checkset( EEG );
% ********************************************************************
% Exclude the first and last 5 second from the original data.
% ********************************************************************
```

```
startTime = round(EEG.event(1).latency/EEG.srate)+5;
endTime = round(movDuration)-5;
EEG = pop_select( EEG,'time',[startTime endTime]);
EEG = eeg_checkset( EEG );
% ********************************************************************
% Bandpass filter and notch filter.
% ********************************************************************
EEG = pop_eegfiltnew(EEG, [], 1, 826, true, [], 0);
EEG = eeg_checkset( EEG );
EEG = pop_eegfiltnew(EEG, [], 80, 42, 0, [], 0);
EEG = eeg_checkset( EEG );
EEG = pop_eegfiltnew(EEG, 55, 65, 414, 1, [], 0);
EEG = eeg_checkset( EEG );
% ********************************************************************
% Run ICA.
% ********************************************************************
EEG = pop_runica(EEG, 'extended',1,'interupt','on');
EEG = eeg_checkset( EEG );
% ********************************************************************
% Exclude bad components using ICA.
% ********************************************************************
EEG = pop_subcomp( EEG, [1  2  3  5], 0);
EEG = eeg_checkset( EEG );
```

```
            resultPath = [%%% resultFolder %%%];
            resultName = [%%% fileName %%%];
            EEG = pop_saveset( EEG, 'filename',resultName,'filepath',resultPath);
            EEG = eeg_checkset( EEG );
        end
    end
end
```

A.3 extract_band.m

```
% Extract five bands from original EEG data
% Sidi Liu
% 10/14/2015

clc;
close all;
clear all;

addpath(\eeglab13_4_4b');
eeglab;

DataSet_catagory = {...
```

```
    'bad',...
    'good'};

DataSet_subject = {...
    'subject 1',...
    'subject 2',...
    'subject 3'};

for cataFile = 1:numel(DataSet_catagory)
    cataName=cellstr(DataSet_catagory(cataFile));
    for subFile = 1:numel(DataSet_subject)
        subName=cellstr(DataSet_subject(subFile));
        for iNum=1:10
            sourcePath = [%%% sourceFolder %%%];
            sourceName = [%%% fileName %%%];
            clear originEEG;
            originEEG = pop_loadset('filename',sourceName,'filepath',sourcePath);
            originEEG = eeg_checkset( originEEG );

            % ********************************************************************
            % using band pass filter to filter the data into 5 bands.
            % ********************************************************************
            %delta
            clear tempEEG;
```

```
clear tempData;
tempEEG = originEEG;
tempEEG = pop_eegfiltnew(tempEEG, [], 0.5, 1650, true, [], 0);
tempEEG = eeg_checkset( tempEEG );
tempEEG = pop_eegfiltnew(tempEEG, [], 3, 414, 0, [], 0);
tempEEG = eeg_checkset( tempEEG );
EEGresultPath = [%%%  resulteFolder %%%];
EEGresultName = [%%% fileName %%%];
tempEEG = pop_saveset( tempEEG, 'filename',EEGresultName,'filepath',EEGresultPath);
tempEEG = eeg_checkset( tempEEG );

%theta
clear tempEEG;
clear tempData;
tempEEG = originEEG;
tempEEG = pop_eegfiltnew(tempEEG, [], 4, 414, true, [], 0);
tempEEG = eeg_checkset( tempEEG );
tempEEG = pop_eegfiltnew(tempEEG, [], 7, 414, 0, [], 0);
tempEEG = eeg_checkset( tempEEG );
EEGresultPath = [%%%  resulteFolder %%%];
EEGresultName = [%%% fileName %%%];
tempEEG = pop_saveset( tempEEG, 'filename',EEGresultName,'filepath',EEGresultPath);
tempEEG = eeg_checkset( tempEEG );
tempEEG = pop_saveset( tempEEG, 'filename',EEGresultName,'filepath',EEGresultPath);
```

```
tempEEG = eeg_checkset( tempEEG );

%alpha
clear tempEEG;
clear tempData;
tempEEG = originEEG;
tempEEG = pop_eegfiltnew(tempEEG, [], 8, 414, true, [], 0);
tempEEG = eeg_checkset( tempEEG );
tempEEG = pop_eegfiltnew(tempEEG, [], 12, 276, 0, [], 0);
tempEEG = eeg_checkset( tempEEG );
EEGresultPath = [%%%  resulteFolder %%%];
EEGresultName = [%%% fileName %%%];
tempEEG = pop_saveset( tempEEG, 'filename',EEGresultName,'filepath',EEGresultPath);
tempEEG = eeg_checkset( tempEEG );

%beta
clear tempEEG;
clear tempData;
tempEEG = originEEG;
tempEEG = pop_eegfiltnew(tempEEG, [], 13, 254, true, [], 0);
tempEEG = eeg_checkset( tempEEG );
tempEEG = pop_eegfiltnew(tempEEG, [], 30, 110, 0, [], 0);
tempEEG = eeg_checkset( tempEEG );
EEGresultPath = [%%%  resulteFolder %%%];
```

```
        EEGresultName = [%%% fileName %%%];
        tempEEG = pop_saveset( tempEEG, 'filename',EEGresultName,'filepath',EEGresultPath);
        tempEEG = eeg_checkset( tempEEG );

        %gamma
        clear tempEEG;
        clear tempData;
        tempEEG = originEEG;
        tempEEG = pop_eegfiltnew(tempEEG, [], 35, 96, true, [], 0);
        tempEEG = eeg_checkset( tempEEG );
        tempEEG = pop_eegfiltnew(tempEEG, [], 41, 82, 0, [], 0);
        tempEEG = eeg_checkset( tempEEG );
        EEGresultPath = [%%%  resulteFolder %%%];
        EEGresultName = [%%% fileName %%%];
        tempEEG = pop_saveset( tempEEG, 'filename',EEGresultName,'filepath',EEGresultPath);
        tempEEG = eeg_checkset( tempEEG );


      end
   end
end
```

A.4 EEG_and_eye_feature.m

```
% Calculate gaze movment and EEG magnitude change
% Sidi Liu
% 10/15/2015

clc;
close all;
clear all;

addpath(\eeglab13_4_4b');
eeglab;

DataSet_catagory = {...
    'bad',...
    'good'};

DataSet_subject = {...
    'subject 1',...
    'subject 2',...
    'subject 3'};

for cataFile = 1:numel(DataSet_catagory)
```

```
cataName=cellstr(DataSet_catagory(cataFile));
for subFile = 1:numel(DataSet_subject)
   subName=cellstr(DataSet_subject(subFile));
   for iNum=1:10
      EEGsourcePath = [%%% ¡¥ourceFolder %%%];
         EEGsourceName = [%%% fileName %%%];
         clear EEG;
         clear dataEEG;
         EEG = pop_loadset('filename',EEGsourceName,'filepath',EEGsourcePath);
         EEG = eeg_checkset( EEG );
         % resample to 40 Hz
         EEG = pop_resample( EEG, 40);
         EEG = eeg_checkset( EEG );
         dataEEG = (EEG.data)';

         EyesourcePath = [%%%  sourceFolder %%%];
         EyesourceName = [%%% fileName %%%];
         clear Eye;
         clear gazex;
         clear gazey;
         Eye = load([EyesourcePath,EyesourceName],'gazex','gazey');
         gazex=Eye.('gazex');
         gazey=Eye.('gazey');
         %%% eliminate 5s from start and end. Sampling rate is 40
```

```
gazex = gazex(200:end-200);
gazey = gazey(200:end-200);

EyeX = data.gazex;
EyeY = data.gazey;
signallength = length(EEG(:,1));
chanNum = length(EEG(1,:));
result = zeros(chanNum,3);

for p = 1:signallength-1
x1 = EyeX(p);
y1 = EyeY(p);
x2 = EyeX(p+1);
y2 = EyeY(p+1);
if x1==-1
   x1=0;
   y1=0;
end
if x2==-1
   x2=0;
   y2=0;
end
d=sqrt((x1-x2)^2+(y1-y2)^2);
diffEye(p,1)=single(d);
```

```
            for q=1:chanNum
               diffEEG(p,q)=abs(EEG(p+1,q)-EEG(p,q));
            end
            end

            EEG = pop_saveset( EEG, 'filename',EEGLABresultName,'filepath',EEGLABresultPath);
            EEG = eeg_checkset( EEG );
            save([resultPath,resultName],'dataEEG','gazex','gazey');

        end
    end
end
```

A.5 EEG_Eye_correlation.m

```
% calculate the correlation
% Sidi Liu
% 10/14/2015

clc;
close all;
clear all;
```

```
DataSet_catagory = {...
    'bad',...
    'good'};

DataSet_subject = {...
    'subject 1',...
    'subject 2',...
    'subject 3'};

for cataFile = 1:numel(DataSet_catagory)
    cataName=cellstr(DataSet_catagory(cataFile));
    for subFile = 1:numel(DataSet_subject)
        subName=cellstr(DataSet_subject(subFile));
        for iNum=1:10
                sourcePath = [%%% sourceFolder %%%];
                sourceName = [%%% fileName %%%];
                clear EEG;
                clear result;
                EEG = load([sourcePath,sourceName]);
                EEG = EEG.data;
    Eye = load([sourcePath,sourceName]);
                Eye = Eye.data;
                chanNum = length(EEG(:,1));
```

```
            result = zeros(chanNum);

            for i = 1:chanNum-1
                correlation=corrcoef(EEG(i,:),Eye(1,:));
                result(i) = correlation(1,2);
            end

            resultPath = [%%% reslutFolder %%%];
            resultName = [%%% fileName %%%];
            save([resultPath,resultName],'result');

        end
    end
end
```

A.6 EEG_coherence.m

```
% calculate the coherence
% Sidi Liu
% 10/17/2015

clc;
```

```
close all;
clear all;

DataSet_catagory = {...
    'bad',...
    'good'};

DataSet_subject = {...
    'subject 1',...
    'subject 2',...
    'subject 3'};

for cataFile = 1:numel(DataSet_catagory)
    cataName=cellstr(DataSet_catagory(cataFile));
    for subFile = 1:numel(DataSet_subject)
        subName=cellstr(DataSet_subject(subFile));
        for iNum=1:10
            sourcePath = [%%% sourceFolder %%%];
            sourceName = [%%% fileName %%%];

            clear EEG;
            clear result;
            clear resultI;
            EEG = load([sourcePath,sourceName]);
```

```
EEG = (EEG.data)';


Fs = 250;  %sampling rate
W_length = min(size(EEG,1)/5,2^10);
W = hamming(W_length);
chanNum = length(EEG(1,:));
result = zeros(chanNum,chanNum);
resultI = zeros(chanNum,chanNum);
[t,f]=mscohere(EEG(:,1),EEG(:,1),W,[],W_length,Fs);
Index = find(floor(f)>= 1 &  floor(f)<= 60);
FrequencyIndex= [Index(1) Index(end)];
   for i = 1:chanNum-1
      [Pxx] = cpsd(EEG(:,i), EEG(:,i), W, [], W_length, Fs);
      for j = i+1:chanNum
         [Pyy] = cpsd(EEG(:,j), EEG(:,j), W, [], W_length, Fs);
         [Pxy] = cpsd(EEG(:,i), EEG(:,j), W, [], W_length, Fs);
         Cxy=Pxy./sqrt(Pxx.*Pyy);
         Coh  = (abs(Cxy)).^2;
         ICoh = (imag(Cxy)).^2;
         Coh = 0.5*log((1+Coh)./(1-Coh)); %first do Fisher's Z
         result(i,j) = nanmean(Coh(FrequencyIndex(1):FrequencyIndex(2)));
         result(i,j) = (exp(2*result(i,j))-1)./(exp(2*result(i,j))+1); %now do an inverse-Fisher's
Z to transform back to coherence
         ICoh = 0.5*log((1+ICoh)./(1-ICoh)); %first do Fisher's Z
```

```
                resultI(i,j) = nanmean(ICoh(FrequencyIndex(1):FrequencyIndex(2)));
                resultI(i,j) = (exp(2*resultI(i,j))-1)./(exp(2*resultI(i,j))+1); %now do an inverse-Fisher's Z to transform back to coherence
            end
        end

    resultPath = [%%% reslutFolder %%%];
    resultName = [%%% fileName %%%];
    save([resultPath,resultName],'result','resultI');

  end
 end
end
```

A.7 EEG_synchronization_index.m

```
% calculate the synchronization index
% Sidi Liu
% 10/12/2015

clc;
close all;
clear all;
```

```
DataSet_catagory = {...
    'bad',...
    'good'};

DataSet_subject = {...
    'subject 1',...
    'subject 2',...
    'subject 3'};

DataSet_band = {...
    'alpha',...
    'beta',...
    'delta',...
    'gamma',...
    'theta'};

for cataFile = 1:numel(DataSet_catagory)
    cataName=cellstr(DataSet_catagory(cataFile));
    for subFile = 1:numel(DataSet_subject)
        subName=cellstr(DataSet_subject(subFile));
        for bandFile = 1:numel(DataSet_band)
            bandName=cellstr(DataSet_band(bandFile));
            for iNum=1:10
```

```
sourcePath = [%%% sourceFolder %%%];
sourceName = [%%% fileName %%%];
clear EEG;
clear result;
EEG = load([sourcePath,sourceName]);
EEG = EEG.tempData;
signallength = length(EEG(1,:));
chanNum = length(EEG(:,1));
result = zeros(chanNum,chanNum,3);

for i = 1:chanNum-1
   for j = i+1:chanNum
      SignalHilb1 = hilbert(EEG(i,:)');
      SignalHilb2 = hilbert(EEG(j,:)');
      phase1 = unwrap(angle(SignalHilb1));
      phase2 = unwrap(angle(SignalHilb2));
      % exclude 10% of the signal before and after because of distorsion
      % introduced by hilbert transform
      perc10w =  floor(signallength*0.1);
      phase1 = phase1(perc10w:end-perc10w);
      phase2 = phase2(perc10w:end-perc10w);
      [index1nm,index2nm, index3nm] = nbt_n_m_detection(phase1,phase2,1,1);
      result(i,j,1)=index1nm;
      result(i,j,2)=index2nm;
```

```
                result(i,j,3)=index3nm;
            end
        end

        resultPath = [%%% reslutFolder %%%];
        resultName = [%%% fileName %%%];
        save([resultPath,resultName],'result');
    end
  end
end
```

A.8 shot_detection.m

```
% shot detection
% Sidi Liu
% 10/18/2015

clc;
close all;
clear all;
```

```
DataSet = {...
    'Trailer 1',...
    'Trailer 2'};

sampleratio = 1/40;

for iFile = 1:numel(DataSet)
    movName=cellstr(DataSet(iFile));
    movePath = [%%% sourceFolder %%%];
    video = VideoReader(movePath);
    rateofFrame = video.FrameRate;
    keyFrameFile = [%%% sourceFolder %%%];
    tempSegData = load(keyFrameFile,'keyFrame');
    segData = tempSegData.('keyFrame');
    clear tempSegData;
    numKeyFrames = size(segData,2);

   % ************************************************
   % Shot detection using Patel, Nilesh V's algorithm
   % ************************************************
    numOfFrames = video.NumberOfFrames;
    frameHeight = video.Height;
    frameWidth = video.Width;
    %compute the color histogram
```

```
B = 5;
numOfBins = 2^B;
colorInt = 256/numOfBins;
HGray = zeros(numOfFrames, numOfBins);
stdGray = zeros(1, numOfFrames);
for i=1:1:numOfFrames
   try
      lFrame = read(video, i);
   catch
      break;
   end
   lRFrame = lFrame(:,:,1);
   lGFrame = lFrame(:,:,2);
   lBFrame = lFrame(:,:,3);
   %get the intensity
   lGray = 0.299*lRFrame + 0.587*lGFrame + 0.114*lBFrame;
   lGrayReshaped = reshape(lGray, 1, frameHeight*frameWidth);
   stdGray(i) = std(double(lGrayReshaped), 0, 2);
   lindexGray = uint8(floor(double(lGray)./colorInt + 1));
   for j=1:1:frameHeight
      for k=1:1:frameWidth
         HGray(i, lindexGray(j, k)) = HGray(i, lindexGray(j, k)) + 1;
      end
   end
```

```
end
%calculate the histogram difference
HDGray = [zeros(1, numOfFrames-1)];
for i=1:1:numOfFrames-1
   HDGray(i) = sum(sum(abs(HGray(i, :) - HGray(i+1, :))));
end
%calculate the mean and variance of the frame-to-frame difference, and
%compute the threshold of Tb = mean + alpha*variance
alpha = 3;
mu = mean(HDGray);
sigma = std(HDGray);
Tb = mu + alpha*sigma;
%calculate the low threshold Ts as the bigger value of the two: 1. the
%mean value of HDGray. 2. the value of HDGray at the midde of right slope of the
%peak in histogram of HDGray
DHNumOfBins = 100;
DHInt = max(HDGray)/DHNumOfBins + 1;
DHist = zeros(1, DHNumOfBins);
for i=1:1:numOfFrames-1
   index = uint8(floor(double(HDGray(i))/DHInt+1));
   DHist(index) = DHist(index) + 1;
end
mxDHist = max(DHist);
mxIndex = find(DHist==mxDHist);
```

```
Ts = max((mxIndex+2)*DHInt, mu);
scaleF = Tb/max(stdGray);
stdGray = stdGray.*scaleF;
figure, plot(1:numOfFrames-1, HDGray, 1:numOfFrames-1, Tb, 1:numOfFrames-1, Ts);
%get the cut transition frame number and output the frame
%check the neighboring difference, see if there're multiple spikes near
%each other, if so, we treat it as false positive
for i=1:1:numOfFrames-1
   if (HDGray(i) > Tb)
      highCnt = 1;
      for j=2:1:10
         if ((i-j >=1) & (HDGray(i-j) > Tb/3) & HDGray(i-j) > 5000)
            highCnt = highCnt + 1;
         end
         if ((i+j < numOfFrames-1) & (HDGray(i+j) > Tb/3) & HDGray(i+j) > 5000)
            highCnt = highCnt + 1;
         end
      end
      if (highCnt < 2)
         lFrame1 = read(video, i);
         lFrame2 = read(video, i+1);
      end
   end
end
```

```
    clear eyeSegData;
    count =1;
    for indexKF=1:numKeyFrames
        seg_start = segData(indexKF);
        tempEye = round(seg_start/rateofFrame/sampleratio);
        if tempEye < 200
            continue;
        end
        eyeSegData(count,1) = tempEye - 199;
        count=count+1;
    end
    resultname = [%%% reslutName %%%];
    save([resultFolder,resultname],'eyeSegData');
end
```