

Olav Finne Præsteng Larsen
William Gjerberg Tresselt

Synchronizing EEG and Eye Tracking, and Creating a Hybrid SSVEP-Based BCI Speller in an Immersive VR Environment

Bachelor's thesis in Computer Science
Supervisor: Alexander Holt
Co-supervisor: Emanuel Alexander Lorenz
May 2023

Olav Finne Præsteng Larsen
William Gjerberg Tresselt

Synchronizing EEG and Eye Tracking, and Creating a Hybrid SSVEP-Based BCI Speller in an Immersive VR Environment

Bachelor's thesis in Computer Science
Supervisor: Alexander Holt
Co-supervisor: Emanuel Alexander Lorenz
May 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science



Norwegian University of
Science and Technology

Abstract

In this project, the group investigate the synchronization of data from an eye tracker integrated within a Virtual Reality (VR) headset, and Electroencephalogram (EEG) data. The primary objective was to enable the pairing of these two technologies, thereby expanding current research possibilities at the NTNU Vizlab. The initial part of the project focused on developing a pipeline for data gathering and analyzing the results to achieve precise synchronization between the eye tracker and the EEG. The equipment was synchronized using eye blinks as a common biological marker, given their identifiable characteristics in both EEG and eye tracking data. To demonstrate the potential and capabilities of combining these two technologies, a Brain-Computer Interface (BCI) speller was developed in an immersive VR environment. Integrating eye tracking and EEG technologies in a VR setting aims to enhance the accuracy, speed, and usability of the BCI speller, which has potential applications in assistive technologies and communication for individuals with severe motor disabilities. Lastly, the accuracy of the speller was tested on a handful of participants over a period of time to assess its usability and the effect of training in using the speller.

Sammendrag

Dette prosjektet har hatt som hovedfokus å synkronisere data fra en eye tracker integrert i et Virtual Reality (VR) headset og en Electroencephalogram (EEG). Grunnen til dette er fordi det er viktig å ha synkroniserte datastrømmer fra utstyr som skal benyttes i forskning. For å oppnå dette, har gruppen lagd en prosess for datainnsamling og -analyse som baserer seg på å finne avviket mellom utstyret og deretter kompensere for det. Utstyret er synkronisert basert på blunk, ettersom datasignaturen til denne biologiske markøren er enkelt identifiserbar i begge datastrømmene. For å demonstrere potensialet til disse nylig synkroniserte teknologiene, ble et Brain-Computer Interface (BCI)-stavesystem utviklet i et VR-miljø. En slik løsning skal ha noen fordeler fremfor mer klassiske stavesystemer, som bedre nøyaktighet, hastighet og brukervennlighet. Av den grunn kan slike løsninger muligens være til nytte som kommunikasjonshjelpemiddel for personer med alvorlige motoriske funksjonshemninger. I dette prosjektet var det dog i all hovedsak lagd et slik stavesystem for å demonstrere at den synkroniserte EEGen og eye trackeren kunne brukes til noe. Dette stavesystemet ble også testet av flere deltagere ved et par anledninger for å finne ut om løsningen gruppen kom frem til var brukbar, og om trening i bruk av systemet utgjorde noen forskjell for brukbarheten.

Preface

The assignment was selected due to the unique opportunity to gain access to equipment such as an EEG and an eye tracker. We would like to express our gratitude to our main supervisor, Alexander Holt, for his assistance and guidance with eye tracking and the main report. Additionally, we would like to thank our co-supervisor, Emanuel Alexander Lorenz for his expertise in biotechnology, signal processing, training in using the EEG, and overall assistance throughout this project.

We would also like to thank all test subjects for participating in this study.

Trondheim 21st May 2023

Olav Finne Præsteng Larsen & William Gjerberg Tresselt

Olav F. P. Jansen

William G. Tresselt

Assignment

The initial project description was to combine EEG and eye tracking in a VR environment allowing for comprehensive analysis of bio-signals tracked by these sensors, with a strong focus on synchronizing the data gathered from the respective data streams. The assignment was later expanded to include the creation of a BCI speller to demonstrate a potential way to use the synchronized EEG and eye tracker, in addition to testing how accurate the implementation was on a handful of participants.

Table of Contents

List of Figures	ix
List of Tables	xi
Acronyms	xiv
Glossary	xv
1 Introduction	1
1.1 Research Questions	1
1.2 Outline	1
2 Theory	3
2.1 EEG - Electroencephalogram	3
2.1.1 Reference and Ground Electrodes	4
2.1.2 Dry and Wet EEG	4
2.1.3 Electrode Labeling	5
2.1.4 Temporal and Spatial Resolution	5
2.1.5 EMG - Electromyography	5
2.1.6 Different Types of Artifacts	6
2.2 Digital Signal Filtering	6
2.2.1 Uses of Filters	7
2.2.2 Infinite Impulse Response (IIR)	8
2.2.3 Finite Impulse Response (FIR)	10
2.3 Combining Eye Tracker and VR	10
2.3.1 Measuring the Movements of the Eye	11
2.3.2 Eye Tracking Methods	11
2.3.3 Eye Tracker Device Systems	13
2.3.4 Eye Tracker Prerequisites and Calibration	14
2.4 LSL - Lab Streaming Layer	14
2.4.1 Potential Synchronization Methods	15
2.4.2 Dejittering	15
2.5 Synchronizing EEG and Eye Tracker using a Common Marker	16
2.5.1 TTL Signaling - Transistor-Transistor Logic	16
2.5.2 Eye Blinking Signatures	16
2.5.3 Blink Signatures in Eye Tracking	17
2.6 Game Engine	18
2.7 Brain-Computer Interface (BCI) Speller	18
2.7.1 Visually Evoked Potentials	18
2.7.2 BCI Speller Paradigms	20
2.7.3 Motor Imagery (MI) Speller	20
2.7.4 Steady State Visual Evoked Potential (SSVEP) Speller	21
2.7.5 P300 Speller	22
2.7.6 Hybrid Spellers	22
2.8 Feature Extraction Techniques	23
2.8.1 Temporal Features	23
2.8.2 Frequency Domain Features	24
2.8.3 Time-Frequency Domain Features	24
2.8.4 Deep Learning Based Features	25

3	Research Methodology	26
3.1	Research methods	26
3.1.1	Qualitative Research	26
3.1.2	Quantitative Research	26
3.1.3	Design Science Research	27
3.2	Data Sampling Design	27
3.3	Research Methods Used	27
3.4	Research Participants	30
3.4.1	How Many Participants	30
3.4.2	Consequences of Multiple Tests on Participants	31
3.5	NSD/Sikt	31
4	Method	32
4.1	An Overview of Selected Sources	32
4.2	What to Download to Reproduce the Setup	32
4.2.1	NeuroPype	32
4.2.2	NeuroPype Pipelines	32
4.2.3	Unity Hub	32
4.2.4	Unity Editor Version	33
4.2.5	Unity Project	33
4.2.6	Software for Compatibility with HTC VIVE Pro Eye	33
4.2.7	Python Scripts	33
4.3	Computer Hardware Specifications	34
4.4	Software and Hardware Used	34
4.4.1	NeuroPype	34
4.4.2	Game Engine Choice	34
4.4.3	LSL4Unity	35
4.4.4	HTC VIVE Pro Eye	35
4.4.5	SRanipal	36
4.4.6	TobiiXR	36
4.4.7	ActiCAP Xpress Twist	36
4.4.8	LiveAmp	37
4.5	Combining EEG and Headset	37
4.5.1	Setting Up EEG Stream	38
4.5.2	Setting Up Eye Tracker Stream	38
4.6	Synchronizing Based on Blinking Artifact	39
4.6.1	Blink Closing	39
4.6.2	Defining Blink - Tobii	39
4.6.3	Adding EMG Channel	40
4.6.4	Unity Implementation	40
4.6.5	Data Management in NeuroPype	41
4.6.6	Recording Blinks From EMG and Eye Tracker	41
4.6.7	Analyzing the Data and Finding the Offset	42
4.7	BCI Speller	44
4.7.1	Signal Filtering	47
4.7.2	Feature Extraction	47
4.7.3	Recognition Strategy	49
4.7.4	Unity Implementation	49
4.8	BCI Speller Testing	53
5	Results	56

5.1	Scientific Results	56
5.1.1	EEG and Eye Tracker Synchronization	56
5.1.2	BCI Speller Implementation	60
5.1.3	BCI Speller Initial Testing	61
5.1.4	BCI Speller Retesting	62
5.2	Engineering Results	63
5.2.1	Results Related to Goals	63
5.2.2	BCI Difficulties	65
5.3	Administrative Results	65
5.3.1	Milestones	65
5.3.2	Meetings	65
5.3.3	Timesheets	66
6	Discussion	67
6.1	Limitations	67
6.2	EEG and Eye Tracker Synchronization	67
6.2.1	Hardware and Software Latency	68
6.2.2	Blinks as a Common Marker	69
6.2.3	Phase Distortion	69
6.2.4	LSL	69
6.2.5	Fuse Stream Node	70
6.2.6	Different Stream Length of Recordings	70
6.2.7	After Shift	72
6.2.8	Data Production Methodology	72
6.3	BCI Speller	73
6.3.1	Implementation	73
6.3.2	Performance	75
6.4	NeuroPype	78
6.5	Research Questions	78
6.5.1	RQ1 - Is it possible to synchronize the given eye tracker and EEG with less than one-millisecond offset?	78
6.5.2	RQ2 - Is it possible to create a BCI Speller combining a synchron- ized EEG and Eye Tracking in a 3D environment?	79
6.5.3	RQ3 - Does practice with the BCI Speller lead to improved accuracy?	79
6.6	Teamwork	80
6.7	Engineering Results	80
6.7.1	Discussion Related to Goals	80
6.7.2	BCI Difficulties - Part 1	86
6.7.3	BCI Difficulties - Part 2	87
6.8	Administrative Results	89
6.8.1	Milestones	89
6.8.2	Meetings	90
6.8.3	Timesheet	90
7	Conclusion and further work	91
7.1	Further Work	91
8	Societal Impact	93
8.1	Effects on Health	93
8.2	Economical Perspective	94
8.3	Societal Relevance	94

8.4 Sustainability	95
8.5 Issues Related to Professional Ethics	95

Bibliography **97**

Appendix **105**

A Letter of Consent	A-1
B Eye blink synchronization results	B-1
C BCI Speller results initial test	C-1
D BCI Speller results retest	D-1
E Preproject Plan	E-1
F Vision Document	F-1
G Project Manual	G-1
H NeuroPype Pipelines and Python Repository	H-1
I Unity Repository	I-1

List of Figures

1	Image of a person wearing an EEG, and the signal from electrodes recording the electrical signal activity in the brain being displayed.	3
2	EEG placement of electrodes where the different sections of the brain are parted according to labeling and the 10-20 system.	5
3	Noisy signal, regularly Butterworth filtered, and zero-phase Butterworth filtered. Shows how the regular filter distorts the signal from the original sinus curve. The yellow and brown circles indicate what pair of ends to compare when looking at the start/end of signal distortions to the signal. The image was created in Python by the group.	8
4	How the different IIR filters outputs the same input signal.	9
5	Image showing that vergence is the same as focal distance in the real world, but that not being the case in VR because the focal distance is always the same even though vergence can vary.	11
6	Image portraying the bright pupil effect, with the camera and light being in line with each other.	12
7	Image portraying the dark pupil system, with the camera and light off axis.	12
8	Image showing the components of the cornea glint eye tracking method.	13
9	Electrode placement in an EEG.	17
10	Shows example of blinks recorded by an EMG. Each peak corresponds to a blink.	17
11	Negative (N) and positive (P) potential created by ERP after time in ms. The number after N or P indicates the time in 100 milliseconds.	19
12	Figure a shows the EEG signal recorded of channel Oz when the user is looking at a frequency. Figure b shows the power of each frequency, using a Power Spectral Density (PSD) analysis. This suggests the user was looking at a frequency of 10 Hz. Figure c shows a scalp topography of the power spectrum summed at 10 Hz and its harmonics.	19
13	Shows the pathway for the information being detected to the visual cortex.	20
14	MI speller referred to as a Hex-O-Spell. The speller uses two stages for each letter being spelled. Firstly the user chooses what cluster the letter is in, then chooses the letter.	21
15	SSVEP speller layout. Each character flickers with a different frequency, and as the user pays attention to a single character, it is reflected in their EEG signals.	21
16	Two different P300 speller layouts. Where (a) uses a 6 x 6 layout where the entire rows are being intensified, and (b) uses a 8 x 8 layout where a single character is intensified. The user looks at the desired character, and the speller flashed different rows and columns at the user. The character being looked at is then decided through a cross-section of the rows and columns.	22
17	A Hybrid SSVEP speller layout. The layout can be divided into 6 clusters, of 2 x 3 characters. Each character in a cluster has a different frequency. Therefore there is a total of 6 different frequencies in the entire speller.	23
18	The different cycles of DSR.	27
19	How the group conducted research according to research methodology for the synchronization part.	28
20	Criteria that have to be met when conducting DSR.	29

21	How the group conducted research according to research methodology for the accuracy testing part.	30
22	Overview of electrode placement in the ActiCAP Xpress Twist.	37
23	Overview of electrode placement in the ActiCAP Xpress Twist and removed electrodes (gray) to be able put on the VR headset.	38
24	EMG electrodes connection on subject.	40
25	EMG and EEG electrodes connected to subject.	40
26	The figure shows how the equipment was put on the group member used to generate blinking data. Here you can see the member having the EEG in his lap and the EMG by the left eye, with the VR headset containing the eye tracker also equipped.	42
27	A single blink graphically shown before the shift. The blink is normalized, the y-axis is amplitude, and the x-axis is sample points. The blue graph is EMG, and the purple and orange graph is eye openness for the right and left eye respectively. The green line indicates the start of the actual blink (eye fully closed), and the red line indicates the end of the blink (eye-opening).	43
28	A single blink graphically shown after shift. The blink is normalized, the y-axis is amplitude, and the x-axis is sample points. The blue graph is EMG, and the purple and orange graph is eye openness for the right and left eye respectively. The green line indicates the start of the actual blink (eye fully closed), and the red line indicates the end of the blink (eye-opening).	44
29	Setup of a hybrid SSVEP and eye tracker speller in 2D. This shows the different clusters with characters, where each character has a different frequency. There is a total of 6 frequencies in the speller. The cluster's characters only flash if the subject's gaze is pointed at it.	45
30	Different speller paradigms labeled a-d, explored and tested by (Du et al. 2019).	46
31	Different speller paradigms' results in accuracy and STD represented by a boxplot. Speller a from figure 30 corresponds to result 1 , b to 2 and so on (Du et al. 2019).	46
32	Overview of electrode placement in the ActiCAP Xpress Twist where removed electrodes are shown in gray, and selected channels for the hybrid SSVEP speller are colored yellow. Note that the Ground (GND) electrode still is moved from the very front of the forehead to one line further back.	48
33	How the different multidimensional variables X and Y are computed as a correlation coefficient, p	49
34	Image of the speller where one of the clusters is highlighted.	50
35	A cluster is looked at by the user, causing it to fly towards the user.	51
36	After 1 second has passed, the letter-cubes in the cluster will start flickering with different frequencies.	51
37	A top-down view of the speller showing the clusters not being put on a line.	52
38	After a user has spelled a letter, it is pushed to a string visible over the speller keyboard. Here, the user has spelled 'HELLO'.	52
39	The figure shows the speller setup as seen for the participants during testing. The upper string is what the participant was trying to get and the lower string was what one might spell instead.	53

40	An image showing how a participant would wear the equipment when testing the speller.	54
41	The figure shows the frequency layout of a single cluster, from top left to bottom right, 4 Hz, 5 Hz, 5.5 Hz, 6 Hz, 7 Hz and 7.4 Hz.	54
42	The same blink as in figure 27 shown graphically with the same measurements. This signal is however filtered with a normal Butterworth filter (not zero-phase), which results in phase distortion. The peak will therefore be further away than it actually should be compared to a zero-phase filter.	57
43	Histogram with the normal distribution of offset computed when cutting half of the excess samples off at the end. X-axis is $\frac{1}{100}$ seconds. Y-axis is the number of instances with a given index. Red lines split the different indexes of 8.3 ms which is the frequency of the eye tracker (120 Hz).	59
44	How the LSL streams could have been saved compared to each other. The red stream is the stream with all samples, while the blue stream represents how the original stream could have looked compared to the red stream. Stream 2a represents the loss of samples when ending recording, Stream 2b represents sample loss when starting the stream, and Stream 2c represents a mix of both. This is important to address because indexes were used to synchronize.	71
45	A probability tree for the 2-second flickering period with the case where no symbol is correctly spelled highlighted in red. The probabilities for getting a correct and incorrect symbol are 0.4706 and 0.529 respectively.	85

List of Tables

1	Recording of the left eye blink offset between the EMG and eye tracker data before shift. Each test had the subject blink 9 times. The columns from 0-8 are the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.	56
2	STD, SE and average values for left blinks before shift. Based on values in table 1.	56
3	Recording of blink offset between the EMG and eye tracker data where all of the excess samples are cut at the start . Each test had the subject blink 10 times. The columns from 0-9 are the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.	57
4	Time offset, STD and SE for the phase distorted signal (table 3).	58

5	Recording of blink offset between the EMG and eye tracker data where half of the excess samples are cut at the start, and the rest at the end . Each test had the subject blink 10 times. The columns from 0-9 are the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.	58
6	Time offset, STD and SE for the different cut techniques used (table 5, and tables 17, 18, 19, 20, 21 from appendix B). The lowest STD and SE was obtained by the cut start 1/2.	59
7	Recording of blink offset where half of the excess samples are cut at the start after shifting found offset. This was the technique with the lowest STD (table 5).	60
8	Time offset, STD and SE from table 7 in sum using indexes, and average using both indexes and time (ms).	60
9	Different CCA values after shifting a mock EEG signal. The target frequency was 7 Hz and the sampling rate was 250 Hz. The table shows different ways the signal could have been phase distorted.	60
10	Shows an overview of how many correct symbols each test subject got for the different flickering lengths in seconds (6 s - 2 s). The lower the flickering length, the lower amount of correct symbols.	61
11	Shows an overview of how the accuracy each test subject obtained for the different flickering lengths in seconds (6 s - 2 s). The lower the flickering length, the lower the accuracy	61
12	Shows how many of the symbols the participants successfully got in total for the different flickering periods in the first round of testing found in appendix C.	62
13	Shows an overview of how many correct symbols each test subject got for the different flickering lengths in seconds (6 s-2 s).	62
14	Shows an overview of how the accuracy each test subject obtained for the different flickering lengths in seconds (6 s - 2 s)	63
15	Shows how many of the symbols the participants successfully got in total for the different flickering periods in the second round of testing found in appendix D.	63
16	Two tables containing information about the ITR of the test rounds. . . .	85

Acronyms

BCI Brain-Computer Interface. i, ii, iv, 1, 3, 18, 20–25, 27, 28, 31, 32, 34, 44, 47, 60, 63–65, 67, 68, 73, 79–81, 83, 86–95

bpm bits per minute. 85, 86

CCA Canonical Correlation Analysis. xii, 24, 47, 48, 52, 60, 73, 74, 76–81, 83, 84, 86–88, 92, 96

DSR Design Science Research. ix, 27–29

EEG Electroencephalogram. i–iv, ix, x, xii, xv, 1, 3–6, 14, 16, 17, 19–25, 32, 35–42, 44, 47–49, 52, 54, 55, 58–60, 63, 65–72, 74–81, 83, 87–89, 91, 93–96

EMG Electromyography. ix–xii, 5, 6, 16, 17, 37, 39–44, 56–58, 67, 68, 72, B-6

ERP Event-Related Potential. ix, 18–20, 22, 32, 44, 74

FIR Finite Impulse Response. 6, 7, 9, 10, 42

GND Ground. x, 4, 5, 37, 38, 48

Hz Hertz. ix, xi, xii, 6, 7, 19, 35, 38, 41, 42, 46–48, 54, 56, 58–61, 67, 72, 73, 78, 87–89, 92, B-6

IIR Infinite Impulse Response. ix, 6–10, 42, 57

ITR Information Transfer Rate. xii, 45, 84–86, 92

LSL Lab Streaming Layer. xi, 14, 15, 33–35, 37, 38, 41, 52, 68–71, 74, 78, 80, 83, 86

MI Motor Imagery. ix, 18, 20, 21, 24, 44, 93

NSD Norwegian Centre for Research Data. 31

POC Proof of concept. 65, 79, 89

PSD Power Spectral Density. ix, 19, 24, 47

REF Reference. 4, 5, 37

SDK Software Development Kit. 33–35, 39, 41

SE Standard Error. xi, xii, 43, 44, 56, 58–60, 71, 72

SSVEP Steady-State Visually Evoked Potential. ix, x, 18, 20–23, 44, 45, 48, 53, 73, 75, 76, 79, 88

STD Standard Deviation. x–xii, 43, 44, 46, 56, 58–60, 67, 68, 70–72, 78, 91

TTL Transistor-Transistor Logic. 16, 69

VEP Visually Evoked Potential. 18, 44, 73

VR Virtual Reality. i, ii, iv, ix, x, 1, 10, 11, 13, 18, 33–38, 40, 42, 49, 50, 53, 66–68, 74, 77, 80, 87, 88, 91, 92, 94, 95

XDF Extensible Data Format. 41

Glossary

API An API, Application Programming Interface, is a higher level implementation of functionality for some kind of computer system with the goal of simplifying the process of using said system (Rossen and Nätt 2022). 14, 36

artifact Artifacts in EEG represent contaminations in the quality of the recorded data, occurring when signals other than the brain's electrical activity is captured by the EEG recordings (Jiang et al. 2019). See theory 2.1.6. 6, 7, 16, 39, 40, 47, 67, 69, 75

band-pass A filter which takes some data as input and outputs only the portions of the data within the specified bounds of the filter, also called the 'band', discarding the rest (Christiano and Fitzgerald 2003). 7, 9, 24, 42, 47

dejittering Time-Stamp Post-Processing (Dejitter) is to remove the inconsistency of a signal's timestamps being streamed. Removing jitter from stream (*LabStreamingLayer submodules · GitHub* 2023). 15, 68

GameObject A GameObject is a Unity abstraction. Everything in a Unity scene is a GameObject (Technologies 2023b). 38, 40, 49, 74, 82

harmonic An oscillation with a frequency that is an integer multiple of the fundamental frequency. The fundamental frequency is the oscillation with the lowest frequency (Kristiansen, Jostein Riiser 2019). ix, 19, 21, 48, 73

jitter The variation in delays of data packets transmitted through a network. This affects the transfer of information, such as network streams (Bothner-By 2023). xv, 15, 39, 41, 43, 67, 68, 72, 78, 92

node A NeuroPype abstraction which is put on a node graph in the NeuroPype Pipeline Editor and used for all kinds of data processing (*Neuropype by Intheon* 2023) such as putting it through a Fourier Transform filter or plotting it. 41, 68, 70, 78, 87, 91

phase distortion Input signal experience varying phase shifts as they pass through the filter or system. This alters the relative timing and phase relationships between these frequency components. This leads to distortion in the output signal's waveform (Skaar 2023). xi, 7, 8, 10, 24, 42, 49, 56, 57, 69, 79

scene A scene is a Unity abstraction. Scenes work as containers for the content of a Unity application (Technologies 2023b). xv, 33, 40, 49

1 Introduction

It is important to maintain precision and accuracy in scientific research. For that reason, the NTNU Vizlab defined an assignment concerning these aspects. The goal of the assignment was to synchronize data from an EEG and an eye tracker integrated in a VR headset, allowing the combined data streams to be used for research purposes. Later, the group explored potential applications for the synchronized equipment. After brainstorming with the supervisors, they chose to develop a BCI speller, which enables users to spell words using only their eye gaze and brain activity. The implemented speller was also tested on a handful of individuals to evaluate its accuracy.

1.1 Research Questions

The research questions are as follows:

- **RQ1** Is it possible to synchronize the given eye tracker and EEG with less than 1 millisecond offset?
- **RQ2** Is it possible to create a BCI speller combining a synchronized EEG and eye tracker in a 3D environment?
- **RQ3** Does practice with the BCI speller lead to improved accuracy?

1.2 Outline

The structure of the report attempts to maintain a common thread throughout the entire project. Each chapter can generally be seen as comprising two distinct parts. Each chapter starts with information regarding the synchronization of the eye tracker and the EEG, and the second part is about the BCI speller and testing. When a chapter deviates from this structure, it is explicitly stated in its introduction.

The Theory chapters contain theory about everything needed to fully understand the work that is done in this project. Subsections 2.1-2.7 contain the information required for synchronizing the EEG and the eye tracker. The two last subsections contain additional information needed for the understanding of the BCI speller.

Following the theoretical background, the methodology is presented. The group divided the method chapter into two separate chapters, Research Methodology and Method, to create a clearer distinction between them. The first one includes theory about research methodology, as well as how the group implemented and adhered to this theory. The second one details everything needed to reproduce the research conducted by the group. The reasoning behind the method is also justified in this chapter. All information needed, from hardware to software, is provided here. This chapter is also split into two parts, where the first part is focusing on the synchronization of the eye tracker and the EEG, and the second part is addressing the implementation of the BCI speller, and how the BCI speller tests were done.

The results chapter primarily consists of tables and graphs presenting the recorded and analyzed data. As with the other chapters, this chapter starts with synchronization, then moves on to display the test results from the BCI speller. The results

also include engineering results which show the results regarding the functional and non-functional attributes set ahead of the project in the vision document (appendix F). There are also administrative results, which regard the milestones set at the beginning, meeting structure, and time sheets (appendix G).

The following chapter discusses the limitations of the project, and then the scientific results. It also contains some words about NeuroPype, answers to the research questions, and what the group thinks about their teamwork. Lastly, it discusses the engineering and administrative results from the previous chapter. The disposition of the chapter does not exactly follow that of the results chapter due to the group wanting to discuss some other factors that were relevant to the project, such as NeuroPype.

The 7th chapter concludes the report by answering the research questions from the introduction and discussing further work. After that, a chapter on the societal impact of the research conducted follows. Finally, the bibliography and the appendix are listed. The appendix provides the necessary appendices, as well as supplementary data used in the results.

2 Theory

The theory chapter provides information needed to understand the underlying principles and concepts discussed in this thesis. The section contains two different parts of the assignment. The first part presents the necessary background knowledge to understand how the group synchronized the EEG and eye tracker. The second part involves the theoretical aspects required for the development and implementation of a BCI speller. This structured approach ensures that readers gain a solid foundation in both areas, allowing for a more profound understanding of the research conducted in this project.

2.1 EEG - Electroencephalogram

The Electroencephalogram (EEG) is a technique used for capturing brain activity by recording the electrical signals generated by the brain's nerve cells (Engstrøm and Jansen 2023). These electrical signals are transmitted through neural pathways that link clusters of nerve cells together (Engstrøm and Jansen 2023). Although the electrical potentials from individual cells are weak and can have opposite directions, the coordinated activity of numerous cells and their neighbors generates measurable signals (Engstrøm and Jansen 2023). These combined signals can be recorded using electrodes attached to the exterior of the skull, amplified, and continuously displayed on a screen (Engstrøm and Jansen 2023). Figure 1 illustrates an example of such a setup.

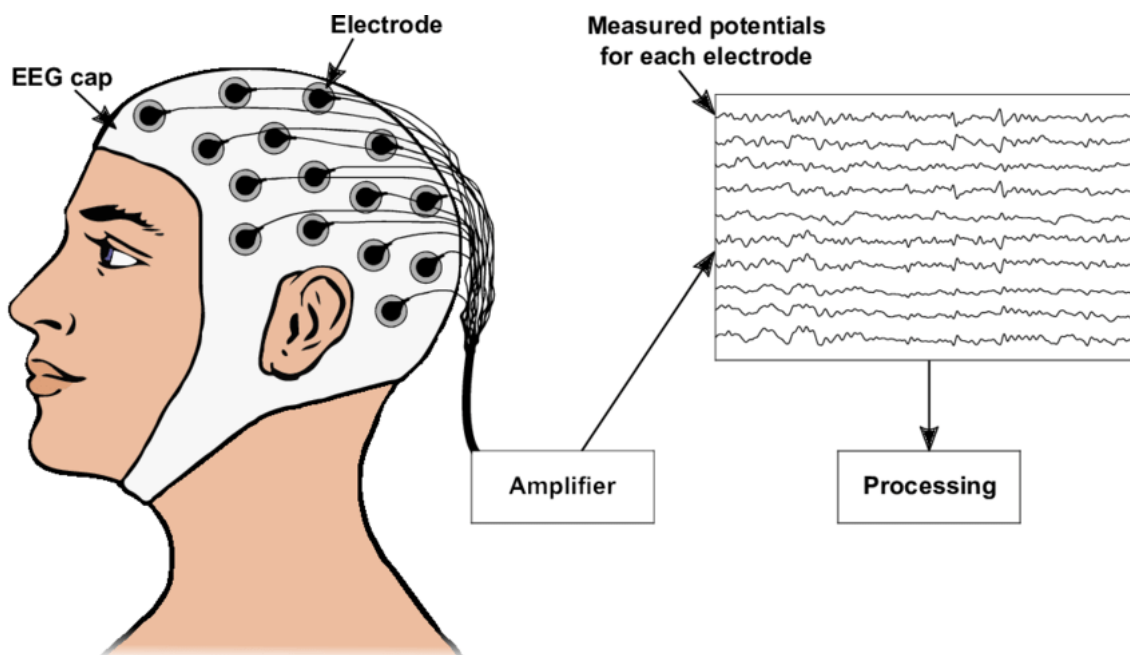


Figure 1: Image of a person wearing an EEG, and the signal from electrodes recording the electrical signal activity in the brain being displayed.

Source: (Nagel 2019)

EEGs are commonly used during the investigation of seizure disorders and as a tool to diagnose epilepsy, but also for brain function disturbances that cause confusion, cognitive impairment, reduced consciousness, and coma (Engstrøm and Jansen 2023).

2.1.1 Reference and Ground Electrodes

Electrodes are characterized as electrically conductive entities that serve the purpose of transmitting electrical current (Sandstad and Haarberg 2021). The electrodes in the EEG are measuring the difference in electrical potential primarily compared to the Ground (GND) electrode (Leuchs, Laura and Brain Products 2019). The Ground electrode is directly connected to the EEG amplifier's ground circuit, and is used to pick up electrical noise, that the other scalp electrodes don't capture (Leuchs, Laura and Brain Products 2019).

This noise does however affect the difference in voltage between the Ground and the other EEG electrodes. The noise related to the Ground is eliminated by adding a Reference (REF) electrode (Leuchs, Laura and Brain Products 2019). The Reference electrode serves as the baseline for all the other EEG channels. This means that the other electrodes are recording the difference in electrical potential compared to the Reference electrode (Leuchs, Laura and Brain Products 2019). This cancels out the noise recorded by the Ground electrode. When using an EEG, it is therefore important to remember that all the other electrodes are measuring the amplitude of the electrical signals, compared to the Reference electrode (Leuchs, Laura and Brain Products 2019).

Typically, a minimum of 20 electrodes are employed and arranged symmetrically in pairs at different locations across the head according to international standards (Engstrøm and Jansen 2023). The electrical voltage to be recorded is only a few millionths of a volt (microvolts, μV) and must be considerably amplified to be visible on a screen (Engstrøm and Jansen 2023).

2.1.2 Dry and Wet EEG

An EEG recording device can be configured in a couple of different ways, namely dry and wet (Lopez-Gordo et al. 2014). The earliest form of EEG was done utilizing the wet variant. This requires a series of steps to perform. First, to decrease the subject's skin's impedance, it is rubbed with a paste where the electrodes will be placed. The following procedure is to put electrolyte gel on the electrodes to allow for a better flow of electrical charges (Lopez-Gordo et al. 2014). This process is time-consuming and requires trained individuals to perform. The gel dries up within 5 hours, making this variant unsuitable for recordings that last more than this time period (Lopez-Gordo et al. 2014). Though inconvenient, the measurements gathered with this method are considered better, making wet EEG the gold standard (Lopez-Gordo et al. 2014). Dry recordings can be achieved with different variations of electrodes. Though they work slightly differently, and each has its strengths and weaknesses, they all share the advantage of not needing any pastes or gels to record data (Lopez-Gordo et al. 2014). Multiple studies have compared wet and dry solutions for EEG with different approaches (Lopez-Gordo et al. 2014), and they all indicate that dry solutions result in acceptable, though not as good, recordings as wet solutions.

2.1.3 Electrode Labeling

The brain is divided into distinct areas and the naming of the electrodes aligns with this (figure 2). The different letters and areas are F (frontal), C (central), T (temporal), P (posterior), and O (occipital) (Soufineyestani et al. 2020). The electrode is also numbered unless it's placed in the middle of the scalp. Then the electrode is given a Z (zero). The Z electrodes are also used for reference when placing the cap on the subject's head (Soufineyestani et al. 2020). The GND and Reference electrodes are symbolized by the GND and REF labels, respectively.

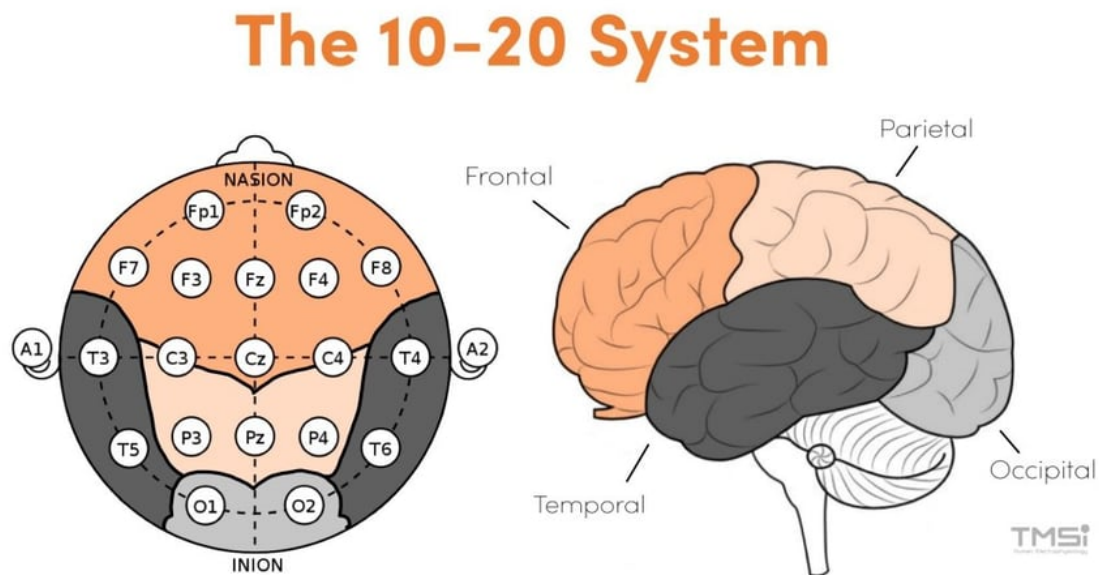


Figure 2: EEG placement of electrodes where the different sections of the brain are parted according to labeling and the 10-20 system.

Source: (TMSI 2023)

2.1.4 Temporal and Spatial Resolution

When working with electrical signals from the brain, there are two different aspects to take into consideration. Temporal resolution is the ability to tell when an activation in the brain happened, while spatial resolution is the ability to tell exactly which part of the brain is activated (Burle et al. 2015). Scalp electrodes are used to record brain electrical activity. The recorded electrical activity is events that are generated several centimeters below the recording electrodes (Burle et al. 2015). The consequence of this is that the cortical current must go through several different resistive layers. This causes a scalp-level distorted view of the brain activities, causing a blurring effect (Burle et al. 2015). This results in every spatial scalp position recorded being a mixture of underlying brain sources (Burle et al. 2015).

2.1.5 EMG - Electromyography

In contrast to EEG, which is used to measure the electric signals created by the brain, Electromyography (EMG) is a technique used to study muscle function by

measuring the electrical signals generated by muscles (Criswell 2010). Surface electromyography (SEMG) refers to the use of electrodes placed on the subject's skin to measure their muscle movements (Criswell 2010). Because EMG is a safe and non-invasive method of measuring electrical signals from muscle movements, it is frequently used in studies where muscle movement is being researched, such as eyelid movements or blinks (Blumenthal et al. 2005; Boxtel et al. 1998; VanderWerf et al. 2003).

2.1.6 Different Types of Artifacts

When collecting EEG data from recording systems, it is important to be aware of signal artifacts. These artifacts contaminate the quality of the recorded data and therefore requires comprehensive knowledge of the types of artifacts to remove the noise efficiently (Jiang et al. 2019). There are four main types of artifacts when recording EEG data.

Ocular artifacts generate significant noise in the EEG recording. These artifacts originate from eye movement and blinks, which can propagate through the scalp, and be recorded as EEG activity (Jiang et al. 2019). The eye movement artifacts are created by changes in the orientation of the retina and cornea dipole. The blink artifacts are caused by ocular conductance due to the alterations of contact to the cornea with the eyelids (Jiang et al. 2019).

Muscle artifacts can also contaminate EEG data. This is a well-known problem since it originates from different types of muscle groups. Muscle activity in close proximity to the EEG can cause these artifacts, anything from a stretch, to the subject talking, or even swallowing (Jiang et al. 2019).

Cardiac artifacts are mainly introduced when electrodes are placed near, or on a blood vessel. The movement of expansion and contraction due to the heart contaminates the data. They are hard to remove, since their frequency is around 1.2 Hz, and can occur within EEG as a similar waveform (Jiang et al. 2019).

Extrinsic artifacts are artifacts that occur from sources outside the body of the subject wearing the EEG (Jiang et al. 2019). An example of an extrinsic artifact is line noise. The source of line noise is strong power line interference, which usually creates noise around 50 or 60 Hz (Leske and Dalal 2019).

2.2 Digital Signal Filtering

To remove unwanted artifacts and noise from EEG signals, signal filtering was utilized. These digital filters are mathematical algorithms that extract information of interest and remove unwanted information (Advanced Solutions Nederland 2020). Digital filters are divided into two categories: Infinite Impulse Response (IIR) and Finite Impulse Response (FIR).

2.2.1 Uses of Filters

There are different filtering techniques when using digital filtering. A short summary of the different techniques:

Notch filtering is used to remove unwanted line noise coming from extrinsic artifacts. This filter typically removes frequencies of 50-60 Hz (Cheveigné and Nelken 2019).

Smoothing / Low-pass filtering refers to a technique, where only frequencies under a certain frequency are being allowed through, while the higher frequencies are filtered out (Cheveigné and Nelken 2019). This is useful to remove unwanted noise from extrinsic artifacts, like line noise.

High-Pass filtering is the opposite of a low-pass filter. This filter allows higher frequencies to pass through while filtering out lower frequencies (Cheveigné and Nelken 2019).

Band-Pass filtering is a combination of high-pass filtering and low-pass filtering. When using band-pass filtering, the filter allows a specific range of frequencies to pass through (Cheveigné and Nelken 2019). When deciding what frequencies to allow through in the band-pass filter, a compromise has to be made in the selection of the cutoff frequencies. The cutoff frequencies are the start and the end of the range in which frequencies are allowed through. If the cutoff frequencies are set too distant from the target band-pass, the filter will cause less signal distortion, but it will allow more noise to pass through (Winter 2009). On the other hand, if the cutoff frequencies are set too close to the target band-pass, the filter will effectively reduce the noise, but this comes at the cost of increased signal distortion (Winter 2009).

Zero-phase filtering is a technique used to counteract any phase distortion created from digital IIR filters. The filter is applied forwards, then reverse on the signal. This does, however, cause distortions to the start and end of the signal, due to the recursive nature of IIR filters (Smith 1989; Winter 2009).

Padding is not directly a filtering technique, but is used to reduce or avoid any edge artifacts or distortions created by the recursive nature of IIR filters (Smith 1989). This does not affect FIR filters, due to their non-recursiveness (Chassaing 2005a; Smith 1989). When applying a zero-phase filter, padding is needed both at the start and at the end of the signal due to the introduced distortions.

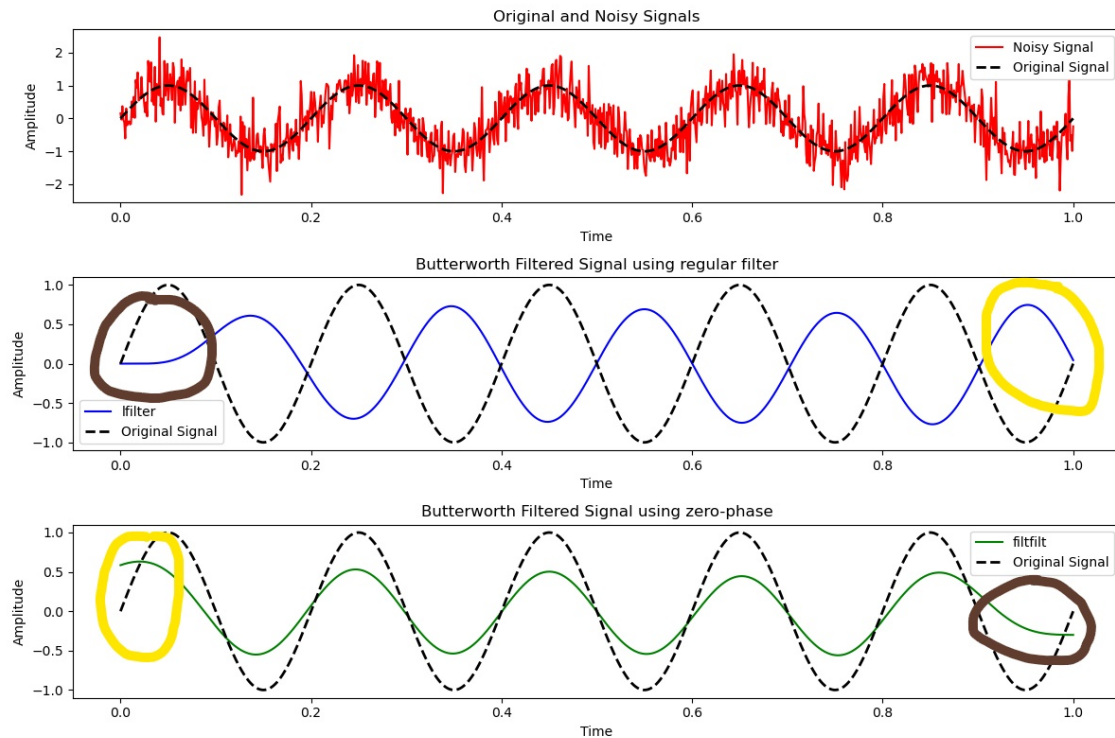


Figure 3: Noisy signal, regularly Butterworth filtered, and zero-phase Butterworth filtered. Shows how the regular filter distorts the signal from the original sinus curve. The yellow and brown circles indicate what pair of ends to compare when looking at the start/end of signal distortions to the signal. The image was created in Python by the group.

Figure 3 illustrates the phase distortion introduced by a regular filter in comparison to a zero-phase filter. Due to the phase distortion, it is more convenient to compare the opposite ends of the signals to each other. The zero-phase filtered signal exhibits signal distortions at both the beginning and the end of the signal, while the regular filter shows signal distortion only at the beginning. This can be observed by examining the similarity between the signals within the brown circles, while the signals within the yellow circles differ. The regular filtered signal appears more similar to the original signal within the yellow circle than the zero-phase filtered signal does. This shows why padding might be necessary at both the start and at the end of the zero-phase filtered signals.

2.2.2 Infinite Impulse Response (IIR)

Infinite Impulse Response (IIR) filters are recursive filters. IIR filters are categorized by their theoretical infinite impulse response,

$$y(n) = \sum_{k=0}^N a_k x(n-k) - \sum_{j=1}^M b_j y(n-j) \quad (1)$$

Equation 1 is a recursive equation, where the output depends on the input, as well as past outputs (Chassaing 2005b). The output $y(n)$ at time n , does not only depend on the current input $x(n)$ at time n , but also past inputs $x(n-1), x(n-2), \dots, x(n-N)$, but also the past outputs $y(n-1), y(n-2), \dots, y(n-M)$. This characteristic is what makes

these filters recursive, and known as infinite (Chassaing 2005b).

An advantage of IIR filters is the low implementation cost. The filter requires fewer coefficients and memory than a FIR filter to satisfy similar specifications. Because of this, IIR filters also apply low latency, which makes it suitable for real-time streaming data (Chassaing 2005a,b). IIR still has some disadvantages. The filters are generally not linear, especially near the cutoff frequencies. This means that the frequency can be phase distorted, and therefore not accurate to the original signal. IIR filters are generally more prone to numerical instability compared to FIR filters (Chassaing 2005a,b).

The order of an IIR filter is the number of previous output samples used to calculate the current output sample. Increasing the order will result in more samples used to calculate the output, which requires more computation power. This makes the filter less effective but improves the amplitude characteristics of the filter (Chassaing 2005a,b; Kleinnijenhuis 2023). There are different types of IIR filters, with different characteristics:

Butterworth filter is an IIR filter with maximally flat characteristics. The gain will decrease gradually from the center, and there are no ripples present in the stopband (Chassaing 2005b).

Chebyshev filter has constant ripples in the band-pass which oscillates between 1 and $a(\epsilon)$,

$$a(\epsilon) = (1 + \epsilon^2)^{-\frac{1}{2}} \quad (2)$$

Equation 2 represents the formula for ripple amplitude, where ϵ is referred to as the ripple factor and can be adjusted (Kleinnijenhuis 2023). The number of ripples will increase by increasing the order of the filter. The stopband and fall-off of a Chebyshev filter are better, compared to a Butterworth filter (Kleinnijenhuis 2023).

Elliptic Filter is a type of IIR filter with constant ripple height in both passband and stopband. The number of heights will increase when increasing the order of the filter. The fall-off is better for an elliptic filter, compared to a Butterworth, and Chebyshev filter (Kleinnijenhuis 2023).

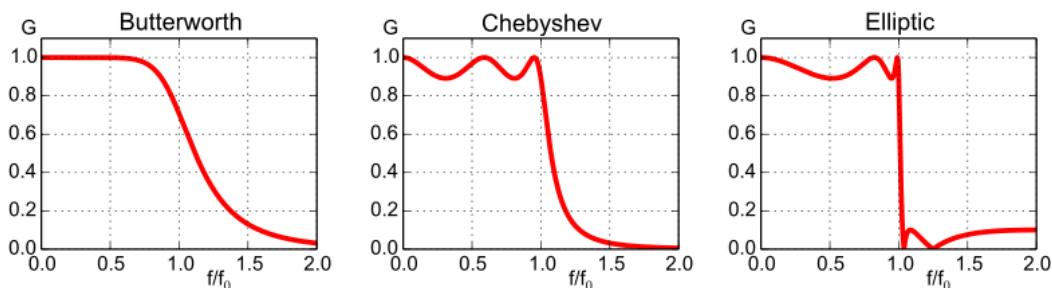


Figure 4: How the different IIR filters outputs the same input signal.

Source: (Yesmin n.d.)

Figure 4 shows the different characteristics of the mentioned IIR filters above. As described there are clear ripples in the Chebyshev and Elliptic filters, whereas the Butterworth filter has a more flat characteristic.

2.2.3 Finite Impulse Response (FIR)

A Finite Impulse Response (FIR) filter, is a digital filter whose impulse response is of finite duration, and is the opposite of earlier discussed IIR filters:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \quad (3)$$

Equation 3 is the formula for FIR filters. The equation shows how to design a FIR filter with N coefficients $h(0), h(1), \dots, h(N-1)$, and N input samples $x(n), x(n-1), \dots, x(n-(N-1))$ (Chassaing 2005a). This is how FIR filters can be implemented only with the knowledge of input $x(n)$ at time n , and knowledge about the delayed inputs $x(n-k)$ (Chassaing 2005a). Contrary to IIR filters, FIR filters are non-recursive, and no feedback or past outputs are needed (Chassaing 2005a).

Due to the large number of coefficients needed from $h(0), h(1), \dots, h(N-1)$, these filters come at a high cost of computational power and a lot of memory use. This can cause delays if the filter were to be applied to a live stream of data (Chassaing 2005a). The delays are not causing any phase distortion, due to the useful feature of FIR filters, which is the guarantee of linear phase. This means that there will not be applied any phase distortion to the filtered signal, which is important in applications where the original phase of the signal needs to be kept intact, such as in synchronization tasks (Chassaing 2005a). This is due to all input components being delayed the same amount (Chassaing 2005a).

In applications that need a fast response and are not affected by phase distortion, an IIR filter is preferred (Chassaing 2005a).

2.3 Combining Eye Tracker and VR

Virtual Reality (VR) is a virtual form of reality that is generated with information technologies of different kinds (Dvergsdal and Aabakken 2023). One common way of creating a virtual reality is with computer graphics (Dvergsdal and Aabakken 2023), which is generally experienced through devices such as headsets that have screens installed in them for each eye (Holm et al. 2023). Eye tracking is the act of tracking the eye, which can be achieved in several ways, e.g. with sensors (Krafka et al. 2016). Recent development has made it possible to put such eye-tracking devices inside VR headsets, allowing for a greater degree of control and tracking of the gaze (Clay et al. 2019). Integrating VR with eye tracking simplifies gaze observation compared to traditional eye tracking in the physical world. This is primarily due to the ease of defining regions within a virtual 3D space that can detect when the gaze hits it (Clay et al. 2019).

A challenge of VR is the disparity between vergence and focus. Our brains receive depth information from vergence and the focus of the lens of the eyes. In VR we only

receive information about vergence. Since the virtual scene in VR is only presented on a single plane at a fixed distance, as seen in figure 5, one cannot extract any depth information out of the focus of the lens (Clay et al. 2019).

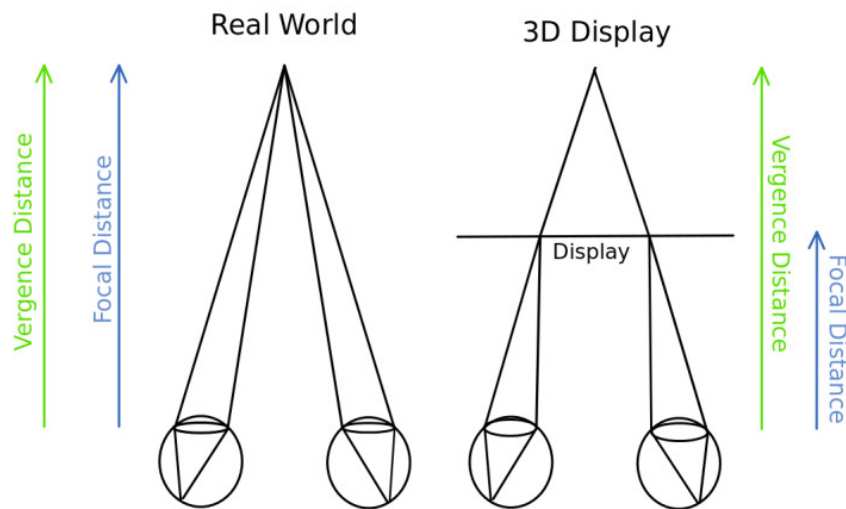


Figure 5: Image showing that vergence is the same as focal distance in the real world, but that not being the case in VR because the focal distance is always the same even though vergence can vary.

Source: (Clay et al. 2019)

To decide what direction the subject is looking in a 3D world, one needs to calculate the 3D gaze vector coming from the subject's eyes into the direction they are looking. Theoretically one can calculate the depth from the divergence of the two eyes, by calculating the crossing point of the gaze from both eyes. This is, however, considered an imprecise calculation, unless the eye-tracker is calibrated perfectly with the subject's eyes and the VR headset (Clay et al. 2019). Because of this problem, it is found a more reliable approach to depth calculation as cited by (Clay et al. 2019).

2.3.1 Measuring the Movements of the Eye

The human eye lets light through the pupil. For video-based measurements of eye movement, the pupil is important. A less known, but also important element of the eyeball is the cornea. The cornea reflects light and covers the outside of the eyeball. The reflection that can be seen from another person's eyes usually comes from the cornea. When tracking eye movement, it is typically not wanted to get more than a single reflection, which there might be many of especially with a VR headset strapped to one's face. Instead, infrared light is used such as to avoid all-natural light reflections (Holmqvist et al. 2011).

2.3.2 Eye Tracking Methods

There have been different ways of tracking eye movement throughout history, ranging from intrusive methods requiring physical contact between a subject's eye and the tracking medium to ones based on the use of cameras (Mento 2020a). The latter methods are more widespread today and can be performed in multiple ways. One such video-based eye-tracking variant tracks the pupil by sending infrared light to

the retina, and capturing the light reflected off the pupil with cameras that are in line with the light source. This form of tracking is classified as a bright pupil system (*Dark and Bright pupil tracking 2022*). Another similar one sends infrared light off the axis of the camera, lighting up everything but the pupil called a dark pupil system (*Dark and Bright pupil tracking 2022*). These methods are portrayed in figure 6 and figure 7 respectively.

Bright pupil effect



Figure 6: Image portraying the bright pupil effect, with the camera and light being in line with each other.

Source: (*Dark and Bright pupil tracking 2022*)

Dark pupil effect



Figure 7: Image portraying the dark pupil system, with the camera and light off axis.

Source: (*Dark and Bright pupil tracking 2022*)

Another way to track the gaze is by using the glint from the reflection of light bouncing off the cornea (Mento 2020a), as shown in figure 8. The calculation is done by subtracting the glint position from the pupil position (Mento 2020a) as such:

$$\text{gaze position} = \text{pupil position} - \text{cornea reflection position}$$

Moving the eye, but not the head, makes the glint stay in place while the pupil is moving, but moving the head makes the pupil and glint move together (Mento 2020a), which is the basis for this tracking method. Applying this method to eye tracking apparatus mounted on the head makes the light source move with the head, resulting in the glint being in a constant position (Mento 2020a) and is often used to deal with the impact of small relative changes between the eye tracker and the eye (Gao et al. 2012; Mento 2020a).

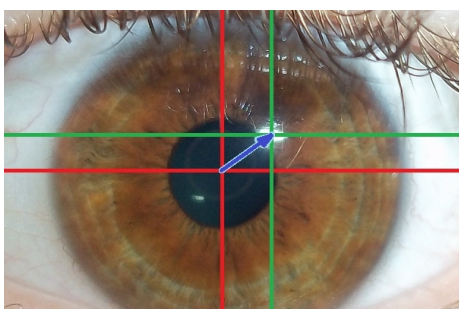


Figure 8: Image showing the components of the cornea glint eye tracking method.

Source: https://commons.wikimedia.org/wiki/File:Visible_light_eye-tracking_algorithm.jpg

One can also use web cameras as eye trackers (Mento 2020a). Although the alternative exists and web cameras probably are more widespread than other eye tracking variants, they do not result in accurate data and the data quality is dependent on the quality of the camera (Mento 2020a). Some other downsides with this tracking variant are the decreased tracking ability of the system when the one being tracked is older, has long eyelashes, wears makeup, or wears glasses to name some reasons (Mento 2020a).

2.3.3 Eye Tracker Device Systems

There are four main eye-tracking systems: head-stabilized, remote, mobile, and embedded (Mento 2020b). Head-stabilized eye tracking systems keep the subject's head still while their eyes are being tracked with a high-resolution camera, and result in accurate and precise measurements. The head is usually kept still with a chinrest or bite-bar. Due to the nature of the setup, the visual experience of the subject is highly controllable and can be used together with other technology that requires the subject to stay still like functional magnetic resonance imaging (fMRI) (Mento 2020b). These kinds of eye tracking systems usually result in high precision data, largely due to the setup's highly controlled nature and the high-level camera equipment that is utilized (Mento 2020b). This accurateness is achieved at the expense of the subject's comfort and natural movement, though, and can easily be affected by as little as 0.1 degrees of head rotation (Mento 2020b).

Remote systems do not require any contact with the subject, giving them their name (Mento 2020b). The eye tracking camera needs a view of the subject's eyes and can compensate for head movement by changing its field of view (Mento 2020b). The eye position and head orientation are tracked utilizing the pupil center and cornea reflection (Mento 2020b).

Eye trackers can be built into different kinds of apparatus, such as Virtual Reality devices (Mento 2020b). This variant allows for all visual stimuli to be predetermined or -defined, creating a highly regulated environment (Mento 2020b) which is a sought-after trait in scientific research (Wrålsen and Berntsen 2022).

2.3.4 Eye Tracker Prerequisites and Calibration

Modern eye tracking systems are robust and mainly need an unobstructed view of the eyes of the person having them tracked to work (Mento 2020a). Eye traits, like color, size, shape, and interpupillary distance, are generally not limiting factors, as these systems can adapt reasonably well to these varying traits (Mento 2020a).

To get as accurate tracking as possible, there is a need for calibration (Clay et al. 2019; Mento 2020a). Calibrating an eye tracker happens by using algorithms to link how the eye is positioned in space to where the user is looking, also called gaze (Clay et al. 2019; Mento 2020a). The calibration utilizes reference points, of which there usually are 5, 9, or 13, that the user has to look at to properly calculate their gaze (Clay et al. 2019; Mento 2020a). For each point, a translation between the eye position, minus the cornea reflex, and gaze position is calculated. The results of these calculations are used to create a matrix with interpolation between each point that covers the calibration area (Mento 2020a). A larger amount of calibration points allows for more accurate eye tracking and the accuracy of the tracking is highest inside the area defined by the points (Mento 2020a).

When using eye trackers, precision can get reduced during usage (Clay et al. 2019). For integrated systems, this can be caused by the user moving their head and therefore shifting the device the tracker is built into (Clay et al. 2019).

2.4 LSL - Lab Streaming Layer

Lab Streaming Layer (LSL) is an open-source protocol and API that makes it possible to stream, receive, synchronize, and record data from a multitude of sensors ranging from Wii remotes to the electrodes in an EEG cap (*Lab Streaming Layer 2023; Supported Devices and Tools — Labstreaminglayer 1.13 documentation 2023*). The group used this protocol to stream EEG and eye-tracking data in the project.

LSL defines several central abstractions including sample, chunk, metadata, stream, stream outlet, stream inlet, resolver, and built-in clock, as highlighted in the LSL documentation (*Introduction — Labstreaminglayer 1.13 documentation 2023*). A sample represents a single data unit from a device's channels, while a chunk consists of multiple samples sent concurrently. A chunk can be of varying size, where the larger chunks improve throughput, and smaller ones improve latency (*Introduction — Labstreaminglayer 1.13 documentation 2023*). Metadata is added through an Extensible Markup Language (XML) document that provides information about the associated stream. XML is a text-based format that signifies structured information (*XML Essentials - W3C 2023*). A stream combines sampled data and metadata, and it could have either a regular or irregular sampling rate, but the datatype must be the same throughout the stream (*Introduction — Labstreaminglayer 1.13 documentation 2023*). Streams are sent through the stream outlet, making them accessible to other network-connected devices (*Introduction — Labstreaminglayer 1.13 documentation 2023*). Devices can subscribe to these streams via a resolver and a stream inlet can be connected to it. Once a stream is subscribed to, the stream inlet gets dedicated to that particular stream, ensuring correct data order (*Introduction — Labstreaminglayer 1.13 documentation 2023*). LSL also supports type conversion and failure recovery. A resolver can query network-available streams based on a parameter list

and resolve them when discovered. Additionally, the built-in clock aids in the time-stamping of samples, enabling synchronization (*Introduction — Labstreaminglayer 1.13 documentation 2023*).

2.4.1 Potential Synchronization Methods

There are some possible ways to synchronize LSL data based on timestamps for streams sent by two computers with different clocks. A simple solution is to add the most recent clock offset value to each timestamp that is gathered remotely (*Time Synchronization — Labstreaminglayer 1.13 documentation 2023*). A more complex solution is to calculate a linear fit based on the clock offsets and remap the timestamps based on that model (*Time Synchronization — Labstreaminglayer 1.13 documentation 2023*). No mapping is done by LSL by default but is rather left up to the user to do with as they see fit (*Time Synchronization — Labstreaminglayer 1.13 documentation 2023*).

Another feasible approach is to compare two streams based on their sample points. When two streams have the same length and sampling rate, they will share the same amount of sampling points. This enables synchronization of the streams based on their samples. However, this approach requires the streams to maintain a constant sampling rate to ensure they are the same size. Alternatively, one stream can be downsampled to match the length of the corresponding stream. The central aspect is to ensure that both streams record data for the same duration.

2.4.2 Dejittering

Some jitter will most likely remain in the timestamps after they have been synchronized. This is caused by the stamping process resulting in timestamps that are not spaced with exact intervals, due to the scheduling of the stamping at the operating system and hardware level (*Time Synchronization — Labstreaminglayer 1.13 documentation 2023*). To align LSL streams at the millisecond level, some form for dejittering is necessary, e.g. by applying a trend-adjusted smoothing algorithm (*Time Synchronization — Labstreaminglayer 1.13 documentation 2023*). This process can be done either online when the data is being streamed, or offline when the data is imported from file (*Time Synchronization — Labstreaminglayer 1.13 documentation 2023*).

For offline dejittering, the clock domain is corrected and the timestamps of all streams with regular sampling rates are recalculated due to them typically being subject to some jitter (*Time Synchronization — Labstreaminglayer 1.13 documentation 2023*). A linear fit is calculated between the indices of samples and their timestamps of uninterrupted segments, which are then used to recalculate the timestamps of all samples based on their index. This is then used for time series that contain interruptions to smooth out the data (*Time Synchronization — Labstreaminglayer 1.13 documentation 2023*).

2.5 Synchronizing EEG and Eye Tracker using a Common Marker

The most common way to synchronize eye-tracking and EEG recordings are to send common marker signals into both data streams. This is to overcome the computer clock differences and can be aligned in the analysis stage (Hutton 2019). There are different strategies to send these common markers.

2.5.1 TTL Signaling - Transistor-Transistor Logic

Transistor-Transistor Logic (TTL) signaling has been used for a long time, and it remains popular to integrate into different laboratory equipment, due to its reliability (Hutton 2019). In the simplest form, TTL integration sends a stimulus that marks the onset of critical trial events. This stimulus is a number between 0 and 250, that gets translated into high or low voltages (Hutton 2019). Both data streams receive these common event markers through a parallel port, which can be used to synchronize the data at the analysis stage (Hutton 2019).

2.5.2 Eye Blinking Signatures

The most common approach to synchronizing eye-tracking and EEG recordings involves sending shared marker signals into both data streams. This method addresses the differences in computer clocks and enables alignment during the analysis stage (Hutton 2019). Various strategies can be employed to send these common markers.

Given that blinks can be recorded by both the EEG and eye tracker, it is feasible to use this event as a common marker. In a healthy and alert adult, the average blink duration is approximately 200 ms, with the reopening time being around twice as long as the closing time (Bækgaard et al. 2014). It is worth noting that blink durations can exceed 400 ms when an individual's alertness level decreases, particularly affecting closure duration and reopening time. Additionally, blink characteristics may vary depending on whether subjects are actively engaged in tasks or passively observing objects (Bækgaard et al. 2014).

Blink Signatures in the EEG / EMG To utilize the characteristics of blinks as a strong and common signature in EEG, their signatures have been extensively analyzed. The usual intent of these analyses is to extract and remove blinks since they usually are an unwanted artifact (Bækgaard et al. 2014). Dipole and eyelid movements are the main contributors to voltage generators of the EEG. A blink contains a strong signature from eyelid movement. Usually, the frontal central EEG electrodes pick up this signature, such as the FPz, FP1, and FP2 (figure 9) (Bækgaard et al. 2014). Figure 10 shows how these blinks are recorded by an EMG, where the peak of the blink is when the eye is fully closed (Sharma et al. 2020).

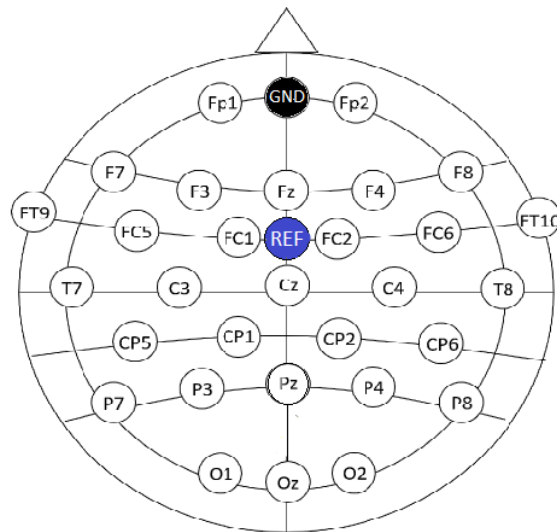


Figure 9: Electrode placement in an EEG.

Source: (Haputhanthri et al. 2019)

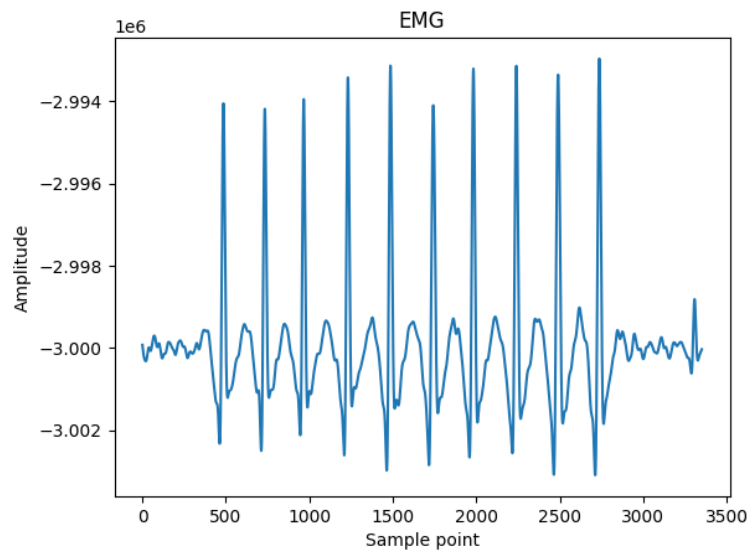


Figure 10: Shows example of blinks recorded by an EMG. Each peak corresponds to a blink.

2.5.3 Blink Signatures in Eye Tracking

Visual eye-tracking equipment relies on the movements of the pupil relative to the reflections of the cornea (Bækgaard et al. 2014). The dynamically changing size of the pupil is also measured and is registered when the gaze no longer can be traced. This can occur from either a blink or the user diverts their attention outside of the tracker's field of view. These events are explicitly recorded in the output data, which can be used directly for estimating the appearance of a blink (Bækgaard et al. 2014). A loss of pupil tracking in the interval of 50-500 ms is indicative of a spontaneous eye blink (Bækgaard et al. 2014).

2.6 Game Engine

Doom, a game created in the mid-1990s, was created in a way that clearly separated core implementations as the simulated physics and collision detection from other things in the game such as the assets and the feel of the game (Gregory 2018). This separation made it easy for developers to reuse parts that would likely not be that different from game to game, leading to the term game engine being popularized (Gregory 2018). Today, game engines are software frameworks that allow for easier development of games or simulations in both 2D and 3D. They abstract away complicated and low-level implementations like physics and rendering for ease of use (Ltd 2023). For this project, a game engine will be used to create VR interfaces.

2.7 Brain-Computer Interface (BCI) Speller

A Brain-Computer Interface, also known as a BCI, is a system that establishes direct communication between the brain and an external device (Ph.D 2020). A basic BCI system includes three components. Firstly a BCI system needs electrodes to record either electric, magnetic, or metabolic brain activity. In this project, the electrodes will record electric activity. Secondly, the system needs a processing pipeline to interpret the recordings of different signals. The pipeline should be able to extract relevant features, remove unwanted noise, etc (Ph.D 2020). Lastly, the system needs a computer or external device that operates via the generated commands from the electric signals, interpreted from the pipeline (Ph.D 2020).

2.7.1 Visually Evoked Potentials

Evoked potentials (EP) are reactions from a nerve cell or organism created by a stimulus. When this occurs in the visual cortex, it is referred to as Visually Evoked Potential (VEP) (Jansen and Dietrichs 2023). When looking at constant frequencies this potential is created in the visual cortex of the occipital lobes and can be recorded by placing electrodes at the occipital part of the scalp (Gjerstad and Bråthen 2023). This potential can e.g. be used when diagnosing illness or damage to the visual cortex (Gjerstad and Bråthen 2023), or be used for BCI applications.

Event-Related Potential A type of VEP is ERP. Event-Related Potential (ERP) are scalp-recorded voltage fluctuations that are time-locked to an event (Kropotov 2016). An Event-Related Potential (ERP) speller is when a BCI exploits the fact that the neural processing of stimulus can be modulated by attention (Treder and Blankertz 2010). Attention to an event can enhance the positive and negative peaks of the ERP time-locked to this event (figure 11). The BCI attempts to detect these modulations to infer the stimulus the user intended to choose (Treder and Blankertz 2010). ERP is utilized in P300 spellers and can also be utilized in other spellers like MI spellers (Kundu and Ari 2022).

Steady-State Visually Evoked Potential There are also different VEPs, like Steady-State Visually Evoked Potential (SSVEP), which is not a time-locked event. SSVEP is

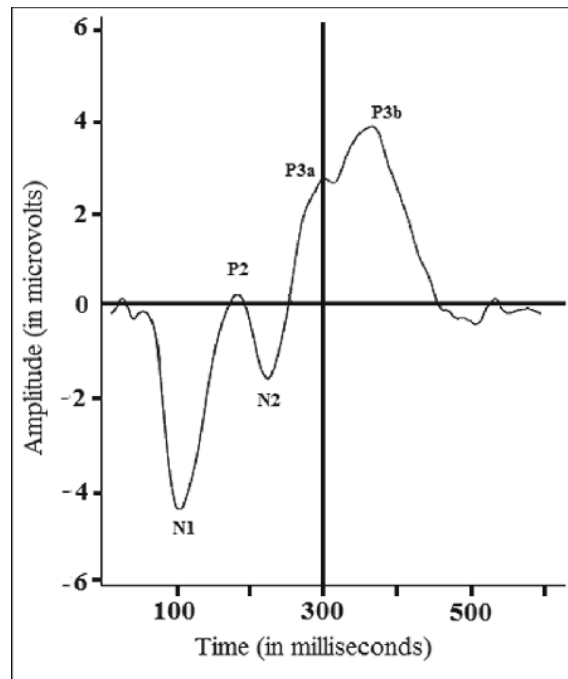


Figure 11: Negative (N) and positive (P) potential created by ERP after time in ms. The number after N or P indicates the time in 100 milliseconds.

Source: (Samima et al. 2017)

created when the user is staring at repetitive stimuli with a constant frequency. The frequency of the recorded EEG will match the stimulus frequency or harmonics of the stimulus frequency (figure 12) (Kundu and Ari 2022).

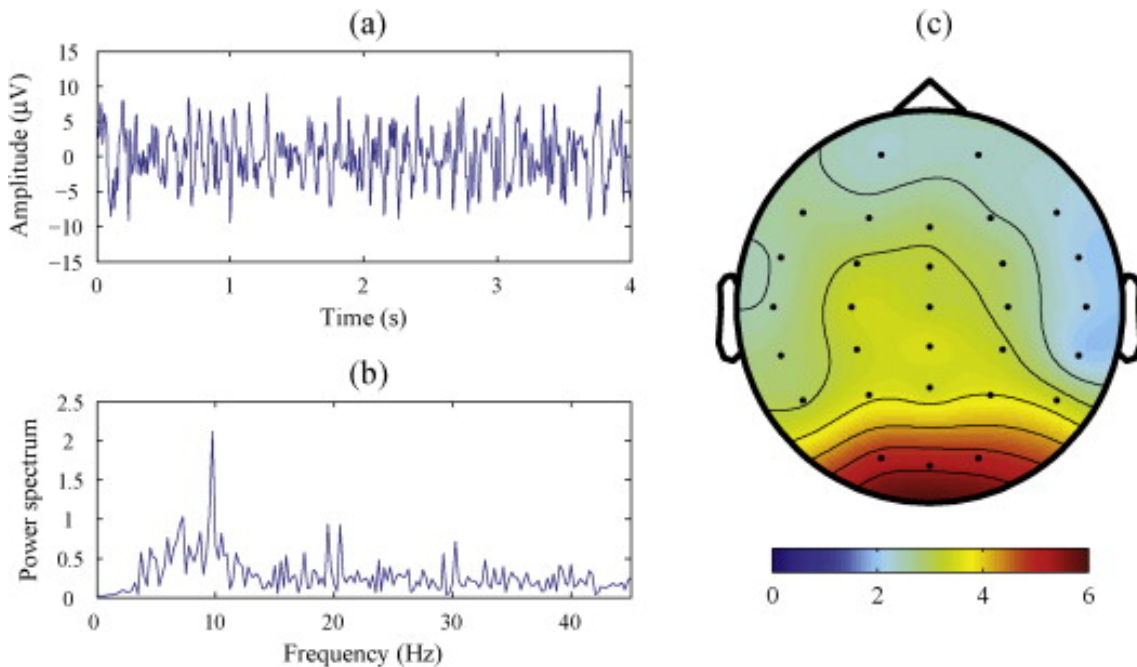


Figure 12: Figure a shows the EEG signal recorded of channel Oz when the user is looking at a frequency. Figure b shows the power of each frequency, using a Power Spectral Density (PSD) analysis. This suggests the user was looking at a frequency of 10 Hz. Figure c shows a scalp topography of the power spectrum summed at 10 Hz and its harmonics.

Source: (Zhang et al. 2015)

SSVEP also introduces a delay from the visual stimulus exposure, to the generation of corresponding oscillatory waves in the visual cortex, like ERP (Li et al. 2015). Figure 13 illustrates the pathway of the stimulus from initial detection by the eyes to it ultimately being processed in the visual cortex, which results in the generation of an oscillatory wave that aligns with the frequency of the stimulus (Li et al. 2015).

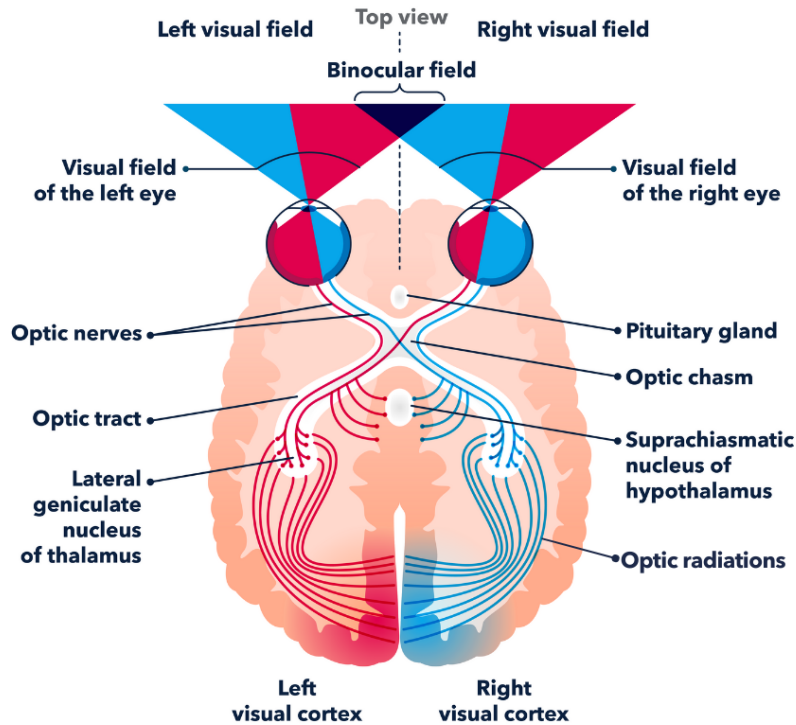


Figure 13: Shows the pathway for the information being detected to the visual cortex.

Source: <https://www.perkins.org/the-visual-pathway-from-the-eye-to-the-brain/>

2.7.2 BCI Speller Paradigms

BCI speller systems are one of the most used BCI systems by disabled patients. To enable the highest amount of information transfer per time, an efficient speller paradigm is important. A typical paradigm is an alphabet that includes numbers, other characters, and symbols (Kundu and Ari 2022). A paradigm with fewer stages will typically be more efficient. An example of a two-stage paradigm can be seen in figure 14, and a single-stage paradigm can be seen in figure 15. These paradigms can also be fully customized for the user, where commonly used words are easily accessed etc.

There are different techniques for using the EEG signals to determine what character the user is looking at. These techniques are used by different categories of spellers such as: Motor Imagery (MI) spellers, Steady-State Visually Evoked Potential (SSVEP) spellers, P300 spellers, and hybrid spellers (Kundu and Ari 2022).

2.7.3 Motor Imagery (MI) Speller

MI based BCI systems provide a communication system for disabled patients that is non-muscular (Kundu and Ari 2022). A MI speller does not require any external

stimuli. Instead, the user can imagine moving their right hand, which will create a distinct pattern in the brain and can be interpreted in the speller (Kundu and Ari 2022). A MI-based BCI speller, where the speller paradigms are hexagons that contain different letters can be seen in figure 14. This does, however, necessitate training and pattern recognition. Consequently, the user must concentrate to consistently produce the same signal patterns. This can be considered slow, and can create considerable fatigue for the user (Kundu and Ari 2022).

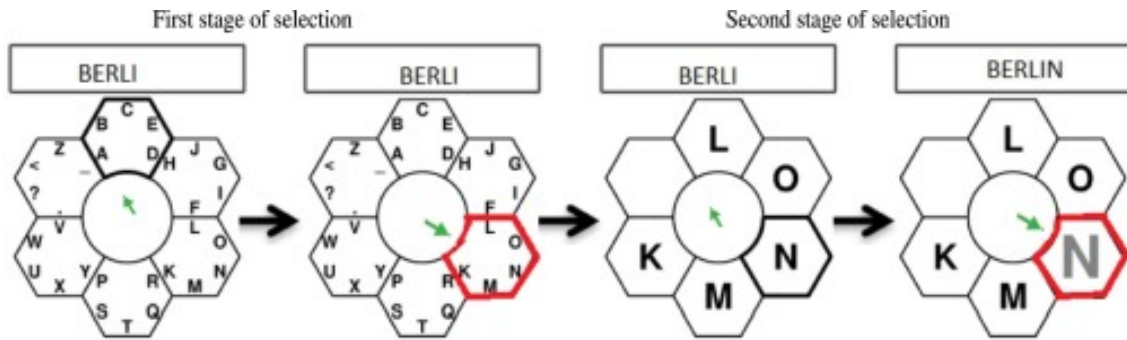


Figure 14: MI speller referred to as a Hex-O-Spell. The speller uses two stages for each letter being spelled. Firstly the user chooses what cluster the letter is in, then chooses the letter.

Source: (Kundu and Ari 2022)

2.7.4 Steady State Visual Evoked Potential (SSVEP) Speller

SSVEP speller utilized Steady-State Visually Evoked Potential. The SSVEP speller paradigm can be shown in figure 15. SSVEP has the advantage that it lets the user use the speller without any prior training, and it lets the user spell with only a single stage, while some other BCI spellers select characters through multiple stages (Kundu and Ari 2022). SSVEP can create stress and fatigue for the user, due to the multiple frequencies used for stimuli. SSVEP also performs poorly if frequencies and stimuli are too close to each other, or they are the harmonics of each other (Kundu and Ari 2022). The recognition performance of the SSVEP speller can also be affected by the brightness of the characters (Kundu and Ari 2022).

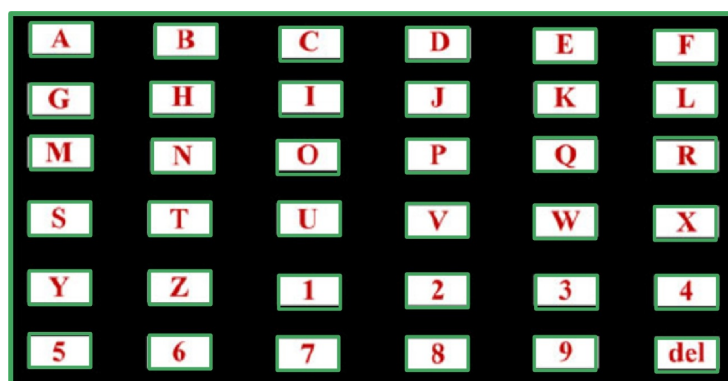


Figure 15: SSVEP speller layout. Each character flickers with a different frequency, and as the user pays attention to a single character, it is reflected in their EEG signals.

Source: (Kundu and Ari 2022)

2.7.5 P300 Speller

The P300 speller uses the same properties as the SSVEP speller, that it targets the user's EEG waves, but instead utilizes the ERP. The name P300 is due to the utilization of the positive potential created after 300 ms (figure 11). This means that instead of having many different frequencies, it focuses on a single one, and uses the time-locked event to determine what symbol the user is looking at (Kundu and Ari 2022). This creates less fatigue than the SSVEP speller. This speller also does not need any training prior to use (Kundu and Ari 2022). The speller paradigm should be a $N \times N$ table of characters. The user pays attention to the desired character. The different rows are then intensified and the BCI determines the desired character through the cross-section of rows and columns (Kundu and Ari 2022). An example of a 6×6 P300 speller (a) and a 8×8 P300 speller (b), can be seen in figure 16. The P300 speller in figure 16b intensifies a single character instead of the entire row. This improves the quality of the speller but requires more focus to complete a single sentence (Kundu and Ari 2022).



(a) A row-column paradigm used in P300

(b) A single character paradigm used in P300

Figure 16: Two different P300 speller layouts. Where (a) uses a 6×6 layout where the entire rows are being intensified, and (b) uses a 8×8 layout where a single character is intensified. The user looks at the desired character, and the speller flashed different rows and columns at the user. The character being looked at is then decided through a cross-section of the rows and columns.

Source: (Kundu and Ari 2022)

2.7.6 Hybrid Spellers

Hybrid speller systems are developed to take advantage of the different perks of the different BCI spellers (Kundu and Ari 2022). The speller in figure 17 is a combination of a SSVEP speller using eye trackers. The user looks at a cluster, then each letter in the cluster starts flickering (Kundu and Ari 2022). Hybrid spellers make it possible to create fast, accurate spellers, that cause less fatigue for the user (Kundu and Ari 2022).

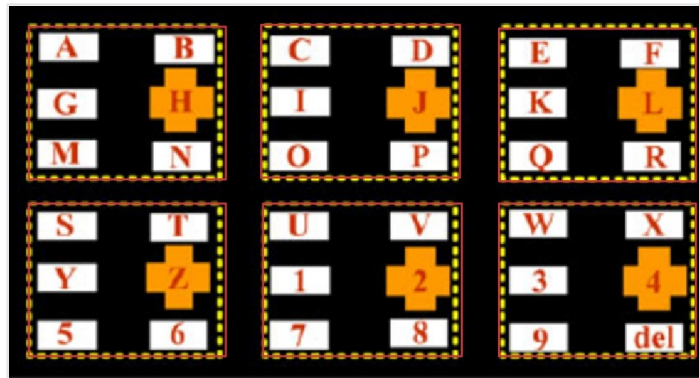


Figure 17: A Hybrid SSVEP speller layout. The layout can be divided into 6 clusters, of 2 x 3 characters. Each character in a cluster has a different frequency. Therefore there is a total of 6 different frequencies in the entire speller.

Source: (Kundu and Ari 2022)

2.8 Feature Extraction Techniques

It is crucial to identify unique features within the EEG signals in a BCI system. These features are meant to accurately describe the user's cognitive state (Kundu and Ari 2022). For the best results, it is important to focus on a limited set of features, that effectively capture the key characteristics of the user's signals. Studies emphasize the importance of feature extractions techniques in BCI systems since even the best classifiers might fail to deliver satisfactory performance and recognition accuracy if the features are inadequate (Kundu and Ari 2022). There have been done many studies on different feature extraction techniques, and they can be classified into four categories: temporal, frequency domain-based, time-frequency domain-based, and deep learning-based (Kundu and Ari 2022).

2.8.1 Temporal Features

Temporal feature extraction methods utilize the EEG signal's temporal variation as features. A simple example of a temporal feature is the time-domain amplitude of the EEG signal (Kundu and Ari 2022). In BCI spellers, the EEG signals from multiple channels are combined to form a feature vector for classification. This technique is frequently used in BCI speller character recognition (Kundu and Ari 2022). Various temporal features can be present in EEG signals, and represent the signal's temporal dynamic, making them useful for classification. Some different temporal features to represent the EEG signal are: The maximum amplitude (A_{max}), positive area under the signal (A_p), negative area under the signal (A_n) and peak-to-peak distance (pp) (Kundu and Ari 2022). Temporal features primarily capture the signal's temporal variation, thus making their extraction easier compared to other techniques. They do, however, not convey information about the signal's frequency variation (Kundu and Ari 2022).

2.8.2 Frequency Domain Features

EEG signals consist of various frequency rhythms or components. These features will be changed based on the task performed by the user (Kundu and Ari 2022). The amplitude of these signals fluctuates during mental tasks like MI, or cognitive tasks performed in BCI applications. There are three main frequency domain extraction techniques: band power feature extraction, Power Spectral Density (PSD) feature extraction, and Canonical Correlation Analysis (CCA) (Kundu and Ari 2022).

Band power features are extracted by processing the EEG signal through a band-pass filter. After the band-pass filtering, the signal is then squared and averaged over a specific time window to calculate its power (Kundu and Ari 2022). These features are typically used in MI detection. PSD features are used to find power distribution across various frequencies (Kundu and Ari 2022). There are two main ways to compute a signal's PSD: either take the Fourier transform of the signal or compute the auto-correlation function of the signal and then perform the Fourier transform (Kundu and Ari 2022).

Canonical Correlation Analysis (CCA) is a method for the extraction of similarities between two data sets (Mannan et al. 2020). CCA is an analysis method that finds the correlation between two multidimensional variables X and Y. The linear combinations of X and Y are, respectively:

$$x = X^T w_x, y = Y^T w_y \quad (4)$$

CCA finds the vectors w_x and w_y from equation 4, that maximize the correlation between x and y by solving the problem:

$$p(x, y) = \max_{w_x, w_y} \frac{E[x^T y]}{\sqrt{E[x^T x]E[y^T y]}} = \max_{w_x, w_y} \frac{E[w_x^T X Y^T w_y]}{\sqrt{E[w_x^T X X^T w_x]E[w_y^T Y Y^T w_y]}} \quad (5)$$

The maximum canonical correlation, is the maximum of p with respect to w_x and w_y (Mannan et al. 2020) (equation 5). E is referred to as the expected value. The main benefit of using CCA compared to other feature extraction techniques is that there is no need for training data, and it is not sensitive to phase distortion (Zerafa et al. 2018).

While frequency domain features allow extraction of frequency information, they do not reveal the amplitude variation of the signal (Kundu and Ari 2022).

2.8.3 Time-Frequency Domain Features

To extract the combined information about time and frequency in an EEG signal, techniques such as Short-Time Fourier transform (STFT) and Wavelet Transform (WT) are employed (Kundu and Ari 2022). STFT involves multiplying a short, non-zero window with the input signal and then computing the Fourier transformation to the windowed signal. The Fourier transform is calculated for each window position, as the window is slid over the signal. This allows the extraction of frequency domain

information, over a brief window, which results in both time and frequency information for the signal (Kundu and Ari 2022). The main drawback of STFT compared to a Fourier transform is the use of fixed window size. This results in uniform frequency and temporal resolution across all frequency bands. It does however improve time resolution compared to Fourier transform. The limitation of the STFT can be addressed by wavelet analysis (Kundu and Ari 2022).

The WT performs a multi-resolution analysis. This technique offers better frequency localization for low-frequency signal components and improved time localization for high-frequency components (Kundu and Ari 2022). A multi-resolution analysis means, that it processes different frequencies in distinct ways. Because of these properties, WT is well-suited for most signal and image processing applications (Kundu and Ari 2022). The main perk of time-frequency domain features is that it delivers both information about time-and-frequency-domain in the EEG signal. This does however come at a cost of higher computational complexity than techniques that focus solely on the time or frequency domain (Kundu and Ari 2022).

2.8.4 Deep Learning Based Features

Unlike traditional feature extraction methods that rely on hand-crafted features that are heavily dependent on human expertise in the specific field, deep learning methods have been introduced in recent years to automatically extract important features from data sets in BCI applications (Kundu and Ari 2022). Deep learning techniques can derive high-level features from raw signals, without any manual interventions. Discriminative deep learning methods develop feature extraction models using class labels available in the training data (Kundu and Ari 2022). Discriminative architectures mainly include Deep Neural Networks (DNN), Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), and their variations (Kundu and Ari 2022). These algorithms extract deep features from the data without using the label information (Kundu and Ari 2022). This means that they can distinguish features from the data, by simply reading the signal, without any extra knowledge. While deep features offer high-level information about the signal, training the model requires a large amount of training data. Moreover, high-performance computing systems are necessary for model training, which can increase the cost of BCI systems (Kundu and Ari 2022).

3 Research Methodology

When conducting research, research methodology is used to ensure valid and reliable results. Researchers use research methodology as a baseline when designing their studies (Wrålsen and Berntsen 2022). This section will explain some different research methodologies, and what research methodologies were used in this project in addition to some words regarding research participants and their data.

3.1 Research methods

Science, according to SNL (*vitenskap – Store norske leksikon 2023*), is a systematic, methodical, and critical examination, study, or research. The reason science is needed is the knowledge it produces. Humans have a tendency to be biased and commit logical errors, which can lead to not entirely correct and/or subjective knowledge. Science tries to eliminate these human-related errors through its methods and ultimately produces knowledge that is more objective and correct than if scientific methods were not applied (Wrålsen and Berntsen 2022). Research is the act of gathering, analyzing, and interpreting gathered data to better understand phenomena (Williams 2007), and there are multiple ways to conduct research. Some general classifications are qualitative, quantitative research, and design science research.

3.1.1 Qualitative Research

Qualitative research methods try to understand or give more insight into different mechanisms and processes by analyzing data that usually is in text form (Grønmo 2023). When doing qualitative research, it can be hard to reproduce the study. This is because the context of the study can never be totally controlled, and these studies give information that might not be generalized, but rather contribute to building knowledge (*vitenskap – Store norske leksikon 2023*).

3.1.2 Quantitative Research

Quantitative research methods are concerned with collecting and analyzing data that is or can be represented numerically (Goertzen 2017). Since quantitative research focuses on measurable data, it is effective at answering the 'what' or 'how' in a given situation (Goertzen 2017).

There are a lot of different quantitative methods used for data collection, some traditional ones are questionnaires, different variants of interviews, focus groups, observations, and secondary data analysis (Axinn and Pearce 2006; Paradis et al. 2016; Tashakkori and Teddlie 2003). Questionnaires are sets of questions that can be handed out to groups of participants for them to answer said questions to get their opinions on different topics (Paradis et al. 2016). Interviews generally happen one-on-one where the interviewer in essence asks the interviewee a set of prepared questions relevant to the topic being researched (Paradis et al. 2016). Focus groups put individuals in a group situation and researchers gather information generated by the

members answering questions or having a conversation (Paradis et al. 2016). Observations are based on gathering information by utilizing one's senses (Paradis et al. 2016). Finally, secondary data analysis is analyzing already existing data to try to uncover new information from it (Paradis et al. 2016).

3.1.3 Design Science Research

Design Science Research (DSR) crosses the boundary between theory and practice, using the development of artifacts (what is being developed) as part of the learning experience for how to develop these artifacts (Dresch et al. 2015; Wrålsen and Berntsen 2022). An example could be the development of a new kind of user interface in the design discipline. One can split the regime into three cyclic, iterative processes that can also be seen in figure 18: relevance, design, and rigor. The relevance of the artifact is considered for the group it is intended, the design is assessed and rigorous use of the knowledgebase, which the artifact development also is contributing to, is used to show that the artifact is of use (Dresch et al. 2015; Wrålsen and Berntsen 2022).

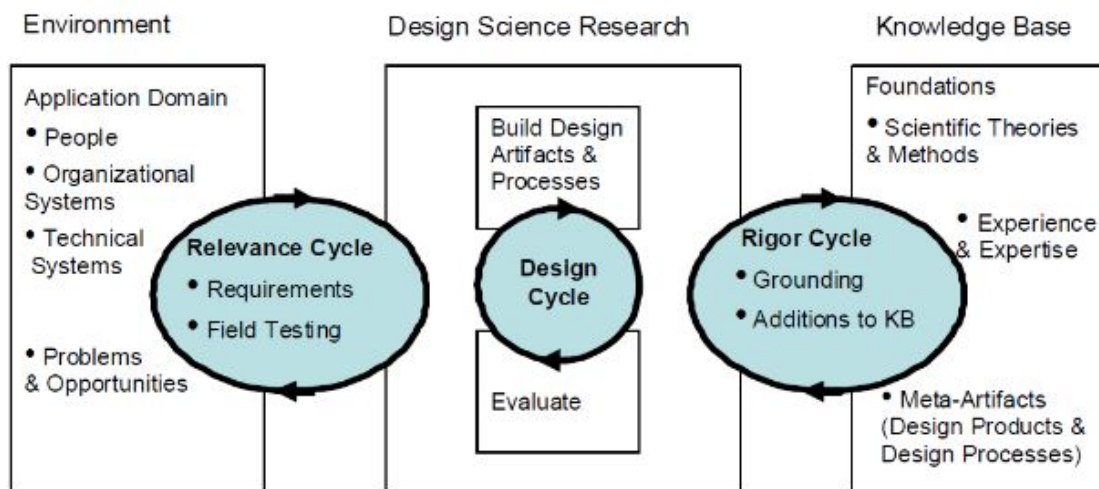


Figure 18: The different cycles of DSR.

Source: (Wrålsen and Berntsen 2022)

3.2 Data Sampling Design

When collecting data the deductive model for data sampling is used. The deductive model uses theory or abstract thought as the baseline. Then the recorded data is analyzed and compared to the hypothesis. This is used both in the synchronization of the eye trackers, and the BCI speller.

3.3 Research Methods Used

This bachelor's project has 3 research questions that each can be answered utilizing different research methods. The synchronization bit utilizes an exploratory quantit-

ative variant, the creation of the speller utilizes DSR as the main method, while the testing of the speller accuracy uses a more traditional quantitative approach. It was necessary to produce and analyze a lot of data with induced markers for the synchronization part. This data was produced in-house by systems set up by the group, and therefore not by some other more conventional quantitative data production means. The markers were necessary to have something to base the synchronization on and were chosen based on relevant theory gathered by the group. The group found no published paper on exactly how to perform the synchronization and therefore utilized a heuristical approach to figuring out the best way to synchronize the data. This approach brings with it some uncertainty and considerations to make. There is always the possibility that the method did not result in proper synchronization due to some hard-to-uncover reasons such as programmatic errors, basing the synchronization on wrong markers, or bad data. But it might be possible to get an indication of whether it was done successfully or not. If there is any way to figure out the theoretical limit and get close to it, the results could be valid.

Figure 19 shows how the group went about synchronizing the equipment. The Vizlab had a need for it, which acted as motivation, and a research question was formulated based on this. To answer the research question, an experiment was planned, but none of the proposed data generation methods fit the model the group had. The group needed to generate their own data. One could argue that this process involved observing the behavior of the data and later evaluating its accuracy, but the usage of these terms is not in line with the traditional ones (Axinn and Pearce 2006; Paradis et al. 2016; Tashakkori and Teddlie 2003). However, the fact that the synchronization is based on quantitative data analysis is clear.

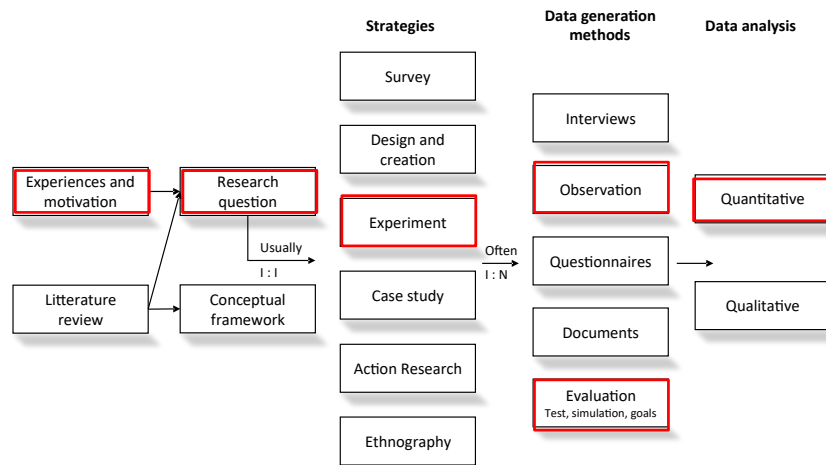


Figure 19: How the group conducted research according to research methodology for the synchronization part.

Source: (Wrålsén and Berntsen 2022)

When it came time to design and implementation of the BCI speller, DSR was utilized. Dresch et al. (Dresch et al. 2015) present some criteria that are essential to apply when following DSR and is shown in figure 20. The group constructed a speller which

is a relevant application of the synchronized equipment and used other spellers as references when creating their own, meeting criteria 1, 2, and 3. In writing about their methods, experiences, and results, and publishing these findings, criteria 4 and 7 are met. Also, later testing of the speller and evaluating its usefulness was done with methods presented by other researchers which is the essence of criteria 5 and 6.



Figure 20: Criteria that have to be met when conducting DSR.

Source: (Dresch et al. 2015)

The development of the speller was primarily guided by the rigor cycle. A visual representation of the rigor cycle can be seen on the right side of figure 18. The design of the speller was largely influenced by two specific studies, but the group also drew inspiration from a third source that had implemented a solution using a setup similar to that of the group. This had to be done because the group did not find any other implementation that used the exact setup they used, which is why they might have some knowledge to contribute to the field, completing the rigor cycle.

Lastly, the group planned to perform a test to see whether the accuracy of participants using the speller would be affected by training. For such an investigation, numbers had to be produced and compared, making a quantitative approach relevant. Here, an experiment based on observations of how the participants performed with the speller was planned. Because this last part was introduced relatively late in the project, not many participants were gathered and the number of tests conducted was few. To get better representative data and conclusions with a stronger foundation, a larger and more diverse sampling size of participants would be beneficial and is therefore a weakness of the test regime. Figure 21 illustrates the process used for testing the speller. The supervisors suggested the group test the speller accuracy, and after reading up on the subject, the group also wanted to see if learning could have some

kind of impact on potential participants' performance. A research question was then generated based on this motivation and literature review. An experiment was conducted, observing how participants performed one task multiple times at varying levels of spelling speed. This resulted in quantitative data which was used to answer the research question.

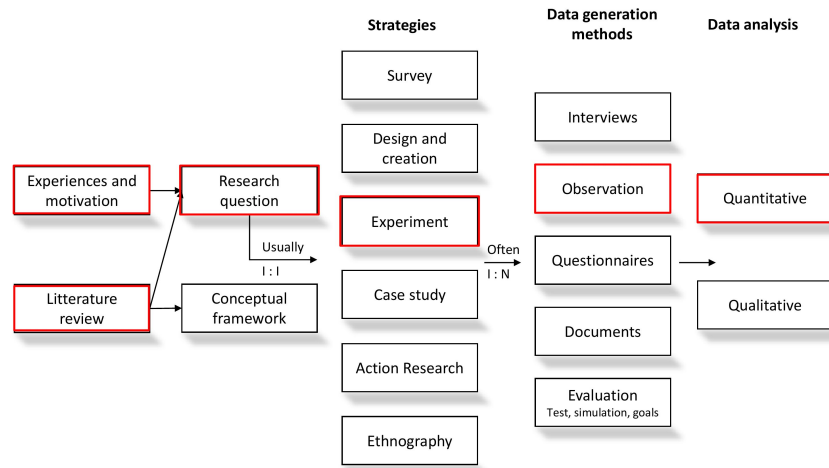


Figure 21: How the group conducted research according to research methodology for the accuracy testing part.

Source: (Wrålsen and Berntsen 2022)

3.4 Research Participants

Some research needs participants to get any findings, e.g. if a group of researchers one day want to find to which degree a group of children improve in a certain type of memory game as they grow older. These children need to be sourced from somewhere, the number of children must be decided, and factors that could influence the results must be identified. This part of the thesis looks at some theory regarding the number of participants in a study and to which degree learning through multiple attempts can have on results.

3.4.1 How Many Participants

If researchers were to observe an entire population in some kind of experiment, their discoveries would be applicable to the entire population. But there could be hundreds of millions of people in this hypothetical population, making it very time-consuming to observe. A workaround for this is for researchers to sample participants from the population in question and then generalize their findings (Cook and Campbell 2002). The participants do however have to be chosen at random to ensure that the findings are actually generalizable for the population they represent (Cook and Campbell 2002; Wrålsen and Berntsen 2022). Another thing to consider is the number of participants.

If the group size is too small, the likelihood of the results being generalizable, or at all valid, will be smaller than if the group size is large (Wrålsen and Berntsen 2022).

3.4.2 Consequences of Multiple Tests on Participants

It is known that research participants may change how they perform between tests. Some research fields commonly administer the same test to participants multiple times in order to measure the change in the results as a consequence of deteriorating cognitive abilities, age, or other factors (Calamia et al. 2012). However, results from the same test being given multiple times can also differ because of the participants learning from the last time they were tested. There is no guarantee that all participants will learn equally as much from one attempt to another due to a multitude of reasons such as applied learning strategies or the interval between tests (Calamia et al. 2012). For example, a study in which two groups of participants, one consisting of younger adults (19-29 years) and another of older adults (65 to 83 years), were to remember a set of 40 items using a mnemonic method showed a significant increase in remembered items after the training (Kliegl et al. 1989). It should be noted that all participants had to undergo 6 test sessions and a maximum of 26 training sessions, allowing for a lot of training for the participants.

3.5 NSD/Sikt

When conducting a study that involves the collection of personal data, there might be necessary to notify Norwegian Centre for Research Data (NSD) / Sikt (*Collection of personal data for research projects - Kunnskapsbasen - NTNU 2023*). NSD was a research institution that had the important role of data protection officer and competence center for data protection for Norwegian research institutions. Today Sikt has taken over this responsibility and is governed under the Norwegian government (Aspøy 2023). To ensure proper data handling and that the legalities of data collection of personal data are being fulfilled, Sikt should be notified when a new research project is being conducted (*Collection of personal data for research projects - Kunnskapsbasen - NTNU 2023*). This is to ensure the ethical and legal aspect of personal data use is being done correctly and to make sure that the rights of the participants are maintained.

Personal data was collected from the participants that tested the BCI speller created in this project. The data included their name, age, and gender. Each subject read and signed a letter of consent (appendix A), which allows the data collection and storing of this information, as well as the right of use until May 2025. The letter of consent also informs the participants of the purpose of the study, study procedures, risk of harm, study duration, confidentiality, consent, and contact information, all according to the Sikt standard. The testing of the BCI speller, with information about what was going to be collected, and how the sensitive data was going to be processed and stored, was sent in to NSD/Sikt and approved. The reference number to the approval is **975161**.

4 Method

The method chapter can be split into three parts. The first part presents the sources, software, and hardware used, along with explanations of the choices made. The following part consists of a section that describes the methods behind the synchronization of the EEG and eye tracker by analyzing locally collected quantitative data. Lastly, there are a couple of sections describing how the BCI speller was implemented utilizing the synchronization offset computed in the first part, and how the testing of the speller was completed.

4.1 An Overview of Selected Sources

The primary sources of information are gathered through Google Scholar, Oria.no, or NTNU Open and are primarily peer-reviewed. Some sources were also encouraged by the supervisors, and therefore not necessarily peer-reviewed, but instead approved by them. General information like how to install the software needed for the project and user experiences with said software was not necessarily sourced from the mentioned channels but from forums and guides or similar websites. Some sources are also blog posts, but only ones referred to or posted by manufacturers relevant to the bachelor's project. Books are also utilized to some degree. When the group members found sources that were not peer-reviewed, they sought out multiple sources stating the same to increase validity. Sources that did not fit these criteria, were not utilized.

4.2 What to Download to Reproduce the Setup

This section assumes the user has git installed and uses Windows 10.

4.2.1 NeuroPype

Download NeuroPype from <https://www.neuropype.io/> after logging in to an account. The group downloaded the NP Academic Edition v2022.0.1. Note that this program requires the user to run Windows as their operating system.

4.2.2 NeuroPype Pipelines

In the repo found in appendix H, there are a couple of pipelines in the directory of the same name. All the ones prefixed with EEG are used for the synchronization and the ones prefixed with ERP are related to the speller.

4.2.3 Unity Hub

Unity Hub is a software solution that lets a user download different versions of the Unity editor in addition to helping individuals manage all their Unity projects and create new ones (Technologies 2023a). This software can be downloaded at <https://unity.com/download>.

4.2.4 Unity Editor Version

This project uses the 2020.3.40f1 editor version for Unity. To download this, navigate to <https://unity.com/releases/editor/archive>. Then, press the 'Unity 2020.X' tab and download the correct one.

4.2.5 Unity Project

Clone the repo from the link found in appendix I. This project has all the necessary packages, but one file has to be updated in order for Unity to be able to launch the project. The downloads required are described in the next step. After the necessary files are updated, this project can be added to the 'Projects' tab in Unity Hub. The project contains two main scenes: one used for synchronization called LSLBlinking-Stream and another used for the speller called ERPSpeller.

4.2.6 Software for Compatibility with HTC VIVE Pro Eye

Navigate to <https://developer.tobii.com/xr/develop/unity/getting-started/vive-pro-eye/> and follow steps 1 and 2 in their entirety. Then follow step 4.2 to download OpenVR Unity XR Package v1.1.4 and follow step 5.1 to download the Tobii XR SDK for Unity v3.0.1. The latter should result in some files. The paths for these have to be updated in the manifest.json in the Packages directory in the cloned Unity project.

4.2.7 Python Scripts

Python 3.11 was utilized with the following packages and areas of use:

- Numpy 1.24.3 (Matrices)
- Scipy 1.9.2 (Digital filtering)
- Pandas 2.0.1 (Data managing)
- pylsl 1.16.1 (LSL in Python)
- matplotlib 3.5.3 (Plotting)

The Python script used for analyzing a single blink is the file:

- analysis_not_fuse_stream.ipynb.

The file used to find the offset by comparing many blinks is called:

- peak_analysis_not_fuse_stream.ipynb.

These files were written in a Jupiter Notebook format. These files can be found in the group's code repository (appendix H), in the folder 'jitter'.

4.3 Computer Hardware Specifications

Both NeuroPype and Unity were run on a computer provided by NTNU. It ran Windows and had an AMD Ryzen 9 5900X, an RTX 3090 and 32GB DDR4 ram.

4.4 Software and Hardware Used

The following text explains what software and hardware were utilized, what it was used for, and why it was chosen.

4.4.1 NeuroPype

The group had to find a Brain-Computer Interface (BCI) to connect and send the recordings into a pipeline to be interpreted. The pipeline can then extract relevant features, or remove unwanted noise. The group had two main BCIs to choose from, NeuroPype and TimeFlux. NeuroPype has a Graphical User Interface (GUI) and is therefore beginner-friendly. They are both new products, and it is therefore some lack of documentation and online examples on both ends. TimeFlux is open source and is very customizable. This combined with the lack of documentation and lack of GUI made it hard to use. The group decided that since they both offer the same features, NeuroPype would be easier to use and get into, therefore they chose NeuroPype. NeuroPype also had more examples and a more intuitive pipeline designer.

4.4.2 Game Engine Choice

To create an interface that allowed for easy compatibility with the VR headset the group was given, a game engine was utilized. There are many different game engines to choose from, like Godot, Unity, and Unreal Engine (Engine 2023; Technologies 2023b; *Unreal Engine | The most powerful real-time 3D creation tool 2023*). The criteria for selecting the most suitable engine for the project included: 1) compatibility with eye tracking using the HTC VIVE Pro Eye headset provided by NTNU, and 2) seamless integration with LSL.

The group found varying degrees of support for eye tracking with the different game engines for the VIVE headset. Firstly, Godot seemed to have some support through a GitHub project (*Godot driver for the HTC Vive Pro Eye equipment 2021*). However, the README mentions a lack of documentation. This is detrimental to efficiency and ease of use, making Godot less of a candidate for the game engine of choice as learning this API through reading code was decided to be outside the scope of this project by the members. Next up was Unity. There was an entire section of Tobii's developer pages dedicated to development for their HTC VIVE integrated eye tracking linking to installation, documentation, and examples (*Get started developing with the HTC VIVE Pro Eye using Tobii XR SDK 2023*). To start using Unity seemed easy, at least regarding the eye tracking part, giving that engine an edge. The last candidate, Unreal Engine, seemed to be slightly harder to utilize. On their official website, under the marketplace, there was an SDK for Tobii eye tracking, but comments on the same page suggested that it did not work with the VIVE headset used in the project (*Tobii*

Eye Tracking SDK in Code Plugins - UE Marketplace 2023). Note that the group did not try to make it work. An alternative way to communicate with the eye tracking hardware was to use SRanipal directly (*SRanipal Getting Started Steps 2019*), but the SRanipal SDK is not compatible with Unreal Engine (5) according to an HTC VIVE forum (*SRanipal Unreal Engine 5 SDK update timeline 2022*). Note that the group did not try to make this work either.

Next up was LSL integration. There was support for LSL for all the mentioned game engines, but LSL for Godot was found in another GitHub repo than LSL for Unity and Unreal Engine (*GitHub - labstreaminglayer/plugin-UE4: LabstreamingLayer plugin for Unreal Engine 4 2023*; *GitHub - UlloLabs/tutorial.GodotPythonLSL: Follow along our tutorial demoing the use of Godot engine to control a sprite with LSL 2023*). The READMEs for each of the repositories contain either examples or tutorials or links to the aforementioned.

From this, the group went with Unity as the engine of choice. LSL seemed to be supported by all the game engines, but the ease of use and documentation regarding the eye tracking part made Unity the preferred choice.

4.4.3 LSL4Unity

Since Unity was the game engine of choice for the project, the group needed to find a compatible method to stream eye-tracking data within Unity. The group considered a couple of alternatives, including implementing LSL from scratch in C# or using a predefined package called LSL4Unity available on GitHub (<https://github.com/labstreaminglayer/LSL4Unity>). They ultimately chose LSL4Unity, as the focus of this bachelor's project was not to create an LSL solution in Unity with C# but to answer the research questions as efficiently as possible, even if it meant relying on external packages.

By using the LSL4Unity package from GitHub, the group was able to stream eye-tracking data in a manner consistent with the EEG data, which also streamed through LSL. This choice was further supported by NeuroPype's full compatibility with the LSL protocol, allowing for more seamless integration of both eye-tracking and EEG data within the project.

4.4.4 HTC VIVE Pro Eye

The group was given the HTC VIVE Pro Eye as a tool for collecting eye-tracking data in and otherwise interacting with a VR environment. On top of being a VR headset, it has an integrated Tobii eye tracker. The sampling rate of the eye tracking is 120 Hz and has an accuracy between 0.5° and 1.1° (*Get started developing with the HTC VIVE Pro Eye using Tobii XR SDK 2023*).

Especially the synchronization of the EEG and the eye tracker was important to get as accurate as possible due to the strict nature of the first research question. To get 1 ms accuracy, as many sources of lag and uncertainty as possible had to be limited. Between each session of data gathering for the synchronization, the eye tracker was calibrated. The procedure occurs as follows: the user is prompted to place the headset correctly on their face, then rotate the IPD (interpupillary distance) knob to adjust the

distance between the lenses and lastly follow a dot that stops at five points with their eyes. These dots are used as reference points that the eye tracker uses to calculate where the user's gaze is pointed (Clay et al. 2019).

4.4.5 SRanipal

VIVE sense is a framework that can be used when developing virtual reality-related content (*VIVE Focus 3 Eye Tracker | VIVE Business European Union 2023*). One part of this framework is called SRanipal which is an eye-tracking SDK used to communicate with the hardware of the VIVE Pro Eye VR headset (*Downloads - Developer Resources 2023*), and exposes information such as the degree of the tracked eyes' openness individually.

4.4.6 TobiiXR

The TobiiXR SDK allows developers to communicate with the eye tracking hardware found in VR headsets such as the HTC VIVE Pro Eye (*Unity 2023*) and exposes a simple API that gets data such as whether the right or left eye is blinking as a boolean (*TobiiXR Core API Reference 2023*).

4.4.7 ActiCAP Xpress Twist

The ActiCAP Xpress Twist is a dry EEG, with 32 electrode channels (*actiCAP Xpress Twist | Brain Products GmbH > Solutions 2023*). The cap uses the common 20-10 system for electrode placement (Herwig et al. 2003). The reason the group chose the dry EEG, is because of the short preparation period. As the theory states, this comes at a cost, where the dry EEG is not as sensitive to record electrical signals, as a wet EEG (theory 2.1.2). For this project, there was no need to get perfect EEG recordings and the group would save a lot of time, by not preparing the wet EEG every time they need to test different features. It was also recommended by the supervisors to use the dry EEG, at least at the start of the project.

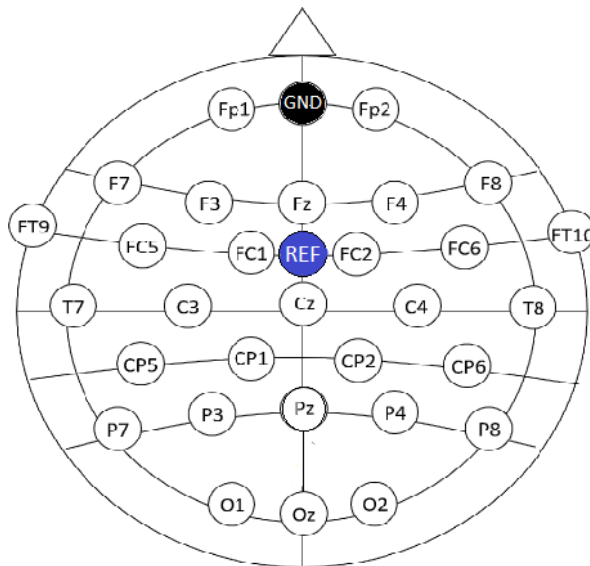


Figure 22: Overview of electrode placement in the ActiCAP Xpress Twist.

Source: (Haputhanthri et al. 2019) (Note the image is modified)

Figure 22 shows the electrode placement, as well as the Ground (GND) electrode in black and Reference (REF) electrode in blue.

4.4.8 LiveAmp

To transmit data from the EEG LiveAmp can be used. This is an EEG amplifier, that supports sending data wirelessly. This is needed to send the EEG signals from the ActiCAP Xpress Twist to a receiving computer. To connect an EMG to the data stream containing data from the LiveAmp, a Sensor & Trigger Extension box, made by the same company, can be used. Connecting EMG electrodes to one of the box's ports and the box to the LiveAmp allowed for the EMG signals to be part of the EEG stream. It was assumed that the LiveAmp and box were synchronized because they are made by the same company. Brain Products also provided a designated PC for receiving these LiveAmp streams, with all necessary software installed. This PC was used for receiving the LiveAmp stream, and then later streaming the data via LSL.

4.5 Combining EEG and Headset

When combining the usage of the EEG cap and VR headset, some of the electrodes on the cap interfered with the headset. To be able to put on the headset with the cap on, the following electrodes were removed:

- Fp1 (1)
- F3 (3)
- F7 (4)
- F4 (30)

- F8 (31)
- Fp2 (32)

This can also be seen in figure 23, where the removed electrodes are colored grey. Note that the Ground (GND) electrode was relocated from the very front of the forehead to one line further back. This electrode can't be removed, since it is essential in the core functionality of the EEG (theory 2.1.1).

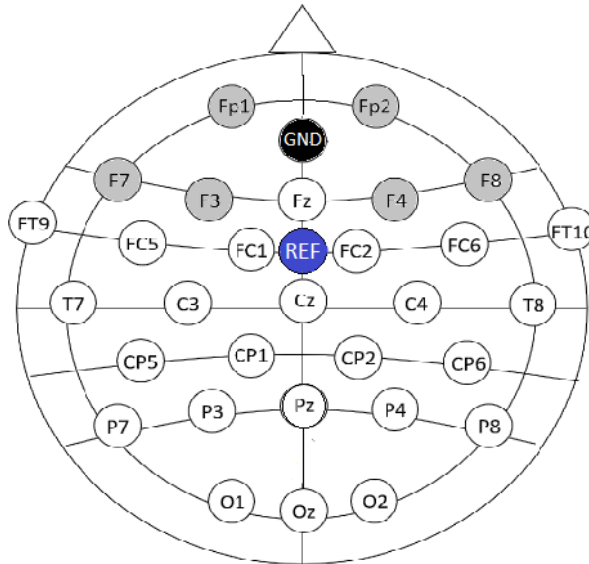


Figure 23: Overview of electrode placement in the ActiCAP Xpress Twist and removed electrodes (gray) to be able put on the VR headset.

Source: (Haputhanthri et al. 2019) (Note the image is modified)

4.5.1 Setting Up EEG Stream

When streaming the EEG data from the PC receiving the LiveAmp stream, it is firstly important to ensure good electrode connections. The Brain Products Recorder application was used to ensure these good connections when equipping the EEG. The application can be downloaded from <https://www.brainproducts.com/downloads/recorder/> and a valid license is needed to run it. To stream the EEG from the Brain Products PC through LSL an official program from Brain Product's GitHub was used. This can be cloned from <https://github.com/brain-products/LSL-LiveAmp>.

When streaming through LSL it is important for other devices that should receive this stream to be on the same local networks as the streaming source. The EEG was being streamed with a sampling rate of 500 Hz.

4.5.2 Setting Up Eye Tracker Stream

To stream the eye-tracking data accessed from the VIVE Pro Eye directly from Unity, the group used the library LSL4Unity. To stream data, a GameObject is created and given a script component that has to inherit some variant of the BaseOutlet, which is defined in LSL4Unity. Each stream is set to use the FixedUpdate hook and set to

be a regular stream, as to minimize potential jitter from irregular data transfer. The time per frame was set to 0.004 for the project, making the FixedUpdate hook run 250 times per second.

4.6 Synchronizing Based on Blinking Artifact

It was decided to try to synchronize the EEG and the eye tracker based on a common marker. As the theory state, a blink is a clear artifact that is easy to find in the EEG/EMG recording. The eye tracker has functionality that allows it to register the openness of the eye, and as an extension, if the eye is closed or not. For this reason, blinks are a marker that is easily extracted from both data streams that these kinds of equipment produce. Synchronization based on the full closure of the eye has also shown promising results in other studies (Bækgaard et al. 2014).

4.6.1 Blink Closing

Even though blinks can be considered as having three phases: 1) closing of the eyelid, 2) the closed phase, and 3) reopening of the eyelid, longer blinks primarily have a longer opening phase (Bækgaard et al. 2014). This might imply that the closing phase is the most well-defined, and it could alternatively be suggested to base the synchronization on the closing phase instead of the entire blink (Bækgaard et al. 2014). This artifact is also visible in both the EEG and eye tracker, therefore it was decided to synchronize the equipment based on when the eyes were fully closed.

4.6.2 Defining Blink - Tobii

For the eye-tracking part, a blink is defined when the tracking hardware no longer registers reflections from the pupil and cornea (*Tobii Eye Tracking SDK in Code Plugins - UE Marketplace* 2023) (theory 2.5.3). This definition is not very specific but there were difficulties formulating a proper definition, as the information found on the internet and in the documentation for the headset used for eye tracking did not specify what a blink was or when it was registered in the hardware. An email from an employee of Tobii referred to a webpage about eye openness, which was defined as “[...] the diameter in millimeters of the largest sphere that can be fitted between the upper and lower eyelids” with the upper and lower eyelids being defined “[...] by the contrast lines between the sclera and the lashes/lid structure” (*Eye openness - Tobii Pro SDK documentation* 2023). It is also mentioned that the eye openness data is available for each eye individually. With this information, it was just a matter of defining a threshold for openness where the eye goes from being open to closed.

Through the SRanipal SDK, the openness is given as a decimal number between 0 and 1, where 0 is fully closed and 1 is fully opened, as seen in the documentation (*sranipal_documentation.html* 2023) that is downloaded alongside the SDK. And the threshold for a blink is set to be 0.1 in TobiiXR, which you can find in Unity under Packages > Tobii XR SDK > Runtime > Core > Providers > Vive > ViveProvides.cs on lines 117 and 122 for the stated version. For consistencies sake, 0.1 will also be considered the threshold for a blink in this project.

4.6.3 Adding EMG Channel

Removing the frontal electrodes of the EEG reduces the visibility of the artifacts created by eye blinks in the EEG channels. This is due to no electrodes being physically close to the frontal part of the head. The group decided to incorporate EMG electrodes around the eye for more accurate measurement of eye blinks. This was achieved using a LiveAmp Sensor and Trigger Extension from BrainProducts.

Small EMG electrodes placed at the base of the upper eyelid are optimal for isolating the reflexive muscle contraction from other activity (Blumenthal et al. 2005). The most usual practice, however, is to place the recording electrodes below the lower lid. This is to avoid issues such as electrode weight, motion artifacts during lid movement, and intrusive skin preparation (Blumenthal et al. 2005). Some studies that record eye blinks with an EMG from the orbicularis oculi muscle have the recorders placed under the eyes (Blumenthal et al. 2005). Other studies have one electrode placed under the eye, while the other is placed on the temple (Anderson et al. 1999). The group only needs a good indication of eye blinks, and it is not worth the struggle to handle all the issues involving placing the electrodes at the base of the upper eyelid. Since the subject also is wearing a VR headset, the group chose the placement of one electrode under the eye and one on the temple. The ground electrode was placed under the ear, for minimum interference with muscle movements (Anderson et al. 1999). The placement can be seen in figure 24. Figure 25 shows how the subjects look after both the EEG and the EMG electrodes are connected.



Figure 24: EMG electrodes connection on sub-**Figure 25:** EMG and EEG electrodes connected to subject.

4.6.4 Unity Implementation

In the Unity project, there is a scene called LSLBlinkingStream, which is a copy of a sample scene provided by TobiiXR containing three cubes that change color when they are looked at and a plane, including a rig necessary for VR compatibility. Most of the GameObjects in the scene is deactivated except for the most relevant one named

LSLGazeRayAndBlinking, which continuously streams data provided by different eye tracking SDKs over LSL as soon as the Unity project is run. The stream had 8 channels, where the first 6 contain data from the TobiiXR SDK and the last 2 contain data from SRanipal. The x, y, and z directions of the eye are found in channels 1-3, and channels 4, 5, and 6 are either 1 or 0 based on if the left eye is blinking, the right eye is blinking or if they are both blinking, in that order. The last two channels contain a floating number between 0 and 1 based on the degree to which the left and right eye is open.

4.6.5 Data Management in NeuroPype

As mentioned are two different LSL streams being streamed, and recorded: the EEG stream and the eye-tracking stream. NeuroPype stores each stream as an XDF file. Each LSL stream was stored as raw data streams and as dejittered data streams. The streams were dejittered by using NeuroPype's dejitter-timestamps node. This was done to fix any inconsistencies/jitter in the timestamps (theory 2.4.2). The EEG stream was also downsampled from 500 Hz to 250 Hz using a designated node in NeuroPype. This was done to match the sampling rate of the eye tracking stream. The reason for this is that it would then be easier to compare the streams, due to them having the same amount of data points. E.g. after 5 seconds, both the eye tracking data and the EEG data would have $5 * 250 = 1250$ data points.

The following names were used to describe each stream with their frequency

- gaze-and-blinking.xdf (250Hz)
- dejittered-gaze-and-blinking.xdf (250Hz)
- raw-eeg.xdf (500Hz)
- dejittered-eeg.xdf (500Hz)
- decimated-dejittered-eeg.xdf (250Hz)

4.6.6 Recording Blinks From EMG and Eye Tracker

Initially, when recording the blinks from the EEG and eye tracker, the subject was set to wear both the EMG, Dry EEG, and the HTC Vive headset. Feedback was given that the EEG and HTC Vive combined were uncomfortable. This made each recording significantly short due to discomfort. Considering only the EMG channel was utilized, the EEG was removed from the head of the subject as seen in figure 26. It was still connected to the LiveAmp device, and synchronized to the EMG. This removed the dry EEG from pressing against the subject's skull, and each recording session could be for as long as necessary.



Figure 26: The figure shows how the equipment was put on the group member used to generate blinking data. Here you can see the member having the EEG in his lap and the EMG by the left eye, with the VR headset containing the eye tracker also equipped.

The recordings were conducted in sets of ten blinks each. The subject listened to a metronome set at 60 beats per minute and blinked with both eyes every two beats. This resulted in a blinking interval of approximately two seconds between each blink. This approach was taken to guarantee consistency in the recordings.

4.6.7 Analyzing the Data and Finding the Offset

When analyzing the recorded data, Python was used. The EMG signal was filtered using a zero-phase Butterworth filter with band-pass of 0.75-5 Hz and an order of 3 (Sharma et al. 2020). This was originally implemented using NeuroPype's IIR node, but the filtering was later moved to the offline analysis due to the need for a zero-phase filter (theory 2.2.1).

An IIR filter was selected due to its lower computational power requirements compared to the FIR filter (theory 2.2.1). Since a zero-phase filter was applied, making the non-phase distortion characteristics of an FIR filter unnecessary. This reinforced the choice of an IIR filter. The group opted for the Butterworth filter, with this decision being influenced by similar research projects. The Butterworth filter is the most commonly used filter in studies with comparable methodologies, in contrast to elliptic and Chebyshev filters. These filters typically introduce more ripple into the signal compared to the smoothness offered by the Butterworth filter (theory 2.2.2). As a result, the group found no reason to diverge from the approach used in other research (Sharma et al. 2020).

The data was first visually plotted on the same graph. The EMG, openness, start, and end of the blink was plotted, for each blink. This was implemented to give an overview of how the data actually looked and to be able to debug and have a closer look if any of the data was wrong, or didn't make sense. Each blink was determined by when the eye was fully closed and then opened (Sharma et al. 2020). When the EMG signal

is filtered, the apex of the blinks will be when the eye is fully closed. To find these peaks, the scipy library was used. Each peak should align with when the eye tracker indicated a closed eye (Sharma et al. 2020). It was then counted how many indexes offset there were between these two values for each blink. This was done for every blink in the recording, and the average offset was then calculated based on this.

To find the deviation in data (jitter), the group utilized the Standard Deviation (STD) and the Standard Error (SE). The STD provides a measure of data variability, which can be viewed as equivalent to jitter in this context.

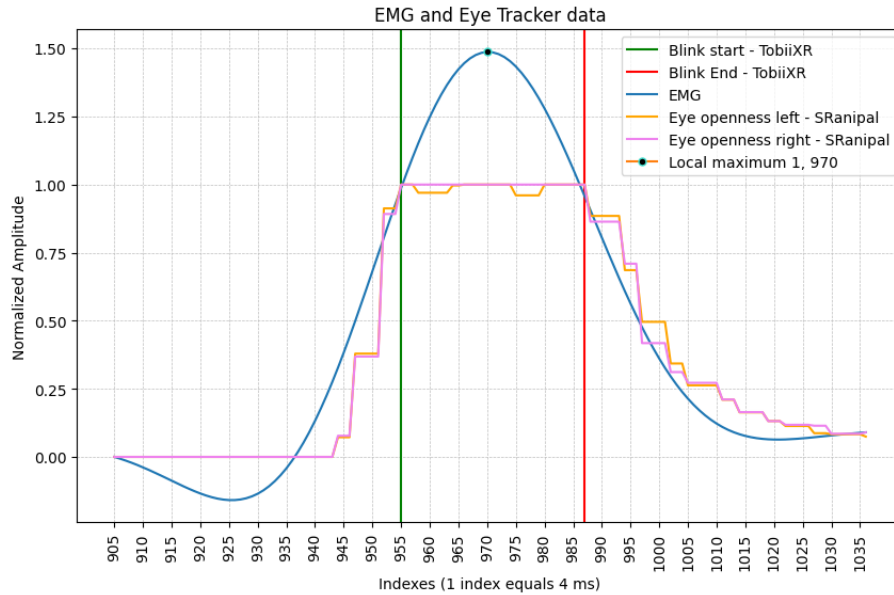


Figure 27: A single blink graphically shown **before** the shift. The blink is normalized, the y-axis is amplitude, and the x-axis is sample points. The blue graph is EMG, and the purple and orange graph is eye openness for the right and left eye respectively. The green line indicates the start of the actual blink (eye fully closed), and the red line indicates the end of the blink (eye-opening).

Figure 27 shows the normalized EMG and the eye tracking data in the same graph. As the theory states, the full closure of an eye blink will be the peak of the EMG. Therefore the blink start line (green), should be aligned with the maximum of the EMG graph.

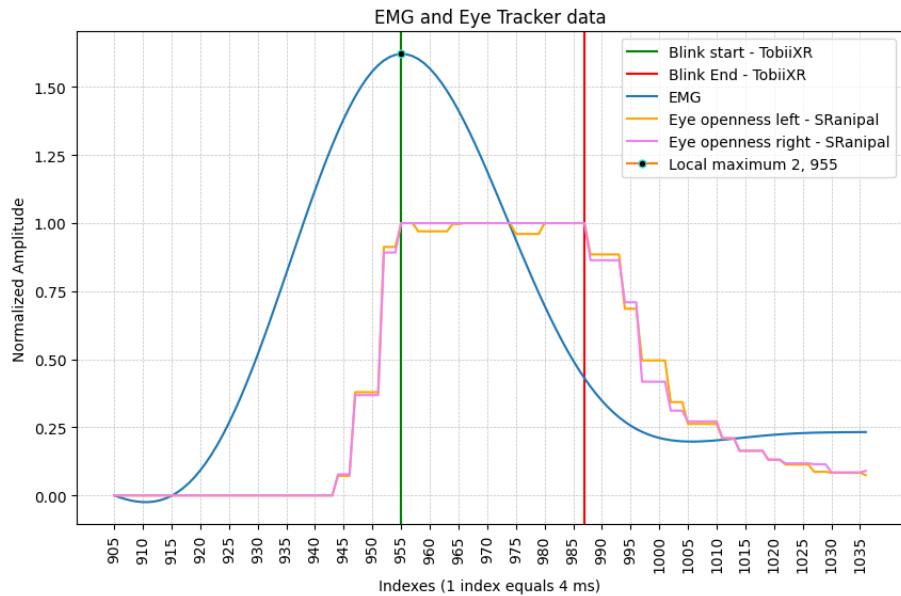


Figure 28: A single blink graphically shown **after** shift. The blink is normalized, the y-axis is amplitude, and the x-axis is sample points. The blue graph is EMG, and the purple and orange graph is eye openness for the right and left eye respectively. The green line indicates the start of the actual blink (eye fully closed), and the red line indicates the end of the blink (eye-opening).

Figure 28 shows the EMG and eye tracking data synchronized. This will then be done over many different recordings, where the mean offset is computed, as well as the Standard Deviation (STD), and Standard Error (SE). This data is the baseline of the synchronization.

4.7 BCI Speller

After synchronizing the EEG and the eye tracker, suitable applications were discussed. The decision was made to implement a BCI-Speller in a 3D environment while combining the eye tracker with the EEG. All the necessary programs to run the speller, Unity, NeuroPype and the Python scripts, were run on the computer with the specs presented at Computer Hardware Specifications.

When determining which speller to implement, different variables were taken into consideration. SSVEP offers the advantage of allowing the subject to use the speller without any training, unlike MI spellers. The SSVEP speller is also easier to analyze than P300 spellers, due to SSVEP's constant frequency characteristics compared to ERP (theory 2.7.1, 2.7.3, 2.7.4, 2.7.5). Hybrid spellers enable the creation of fast, accurate spellers that cause less fatigue for the subject (Kundu and Ari 2022) (theory 2.7.6). Consequently, the group chose a hybrid speller that utilizes eye tracking and utilizing the Visually Evoked Potential (VEP), SSVEP. This approach has also been utilized in other studies with great results (Mannan et al. 2020; Talevi et al. 2023).

The studies suggest that this combination creates a speller that is superior to other hybrid spellers in terms of comfortability, items classified, and classification accuracy (Mannan et al. 2020). Another perk is the need for only 6 different frequencies for the entire speller (Mannan et al. 2020). Figure 29 shows the hybrid setup in a 2D

environment. It uses a combination of low amount of different frequencies with the gaze of the subject, to efficiently and easily spell different words.

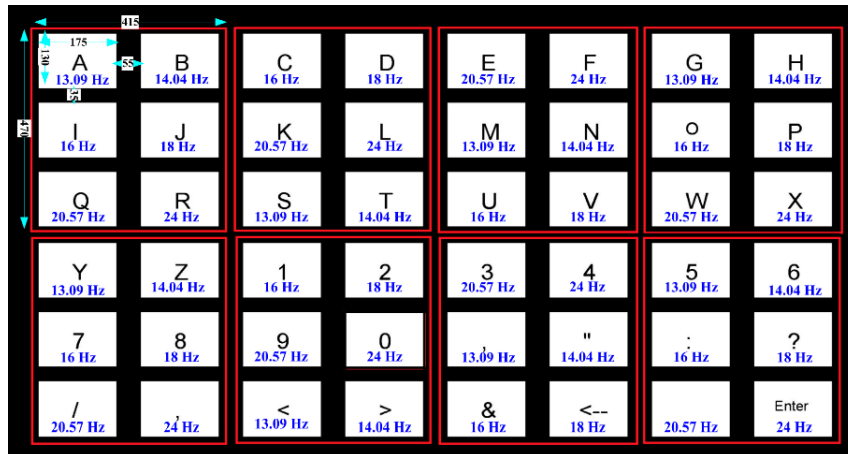


Figure 29: Setup of a hybrid SSVEP and eye tracker speller in 2D. This shows the different clusters with characters, where each character has a different frequency. There is a total of 6 frequencies in the speller. The cluster's characters only flash if the subject's gaze is pointed at it.

Source: (Mannan et al. 2020)

This hybrid speller was then implemented into a 3D space. It was decided to use a 3D paradigm because studies suggest that this is more accurate than a 2D paradigm when the number of testing rounds are few. It is also reported that the subjects have more fun and that 3D visual stimuli feel more intense and more indefatigably than that 2D visuals (Du et al. 2019).

The paradigm should also have the highest possible Information Transfer Rate (ITR), typically implemented by using fewer stages during the spelling process (theory 2.7.2). When deciding what 3D paradigm to use, a study experimented with four different paradigms and achieved high accuracy across all four. The group chose the paradigm with the lowest overall accuracy at 85%, but also the one with the lowest standard deviation, which is paradigm c and 3 in figure 30 and 31 respectively (Du et al. 2019).

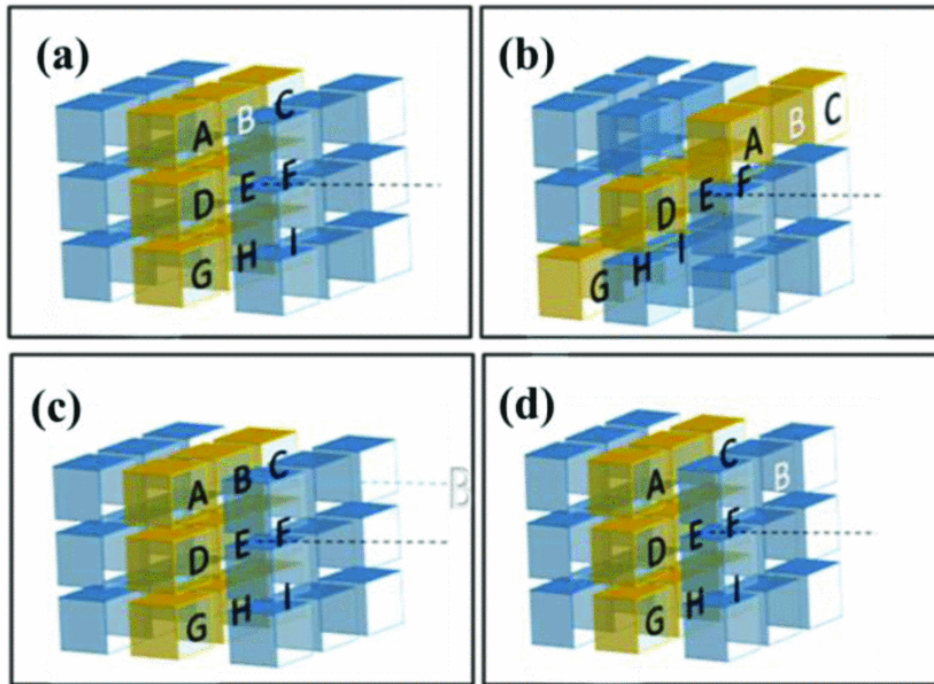


Figure 30: Different speller paradigms labeled a-d, explored and tested by (Du et al. 2019).

Source: (Du et al. 2019)

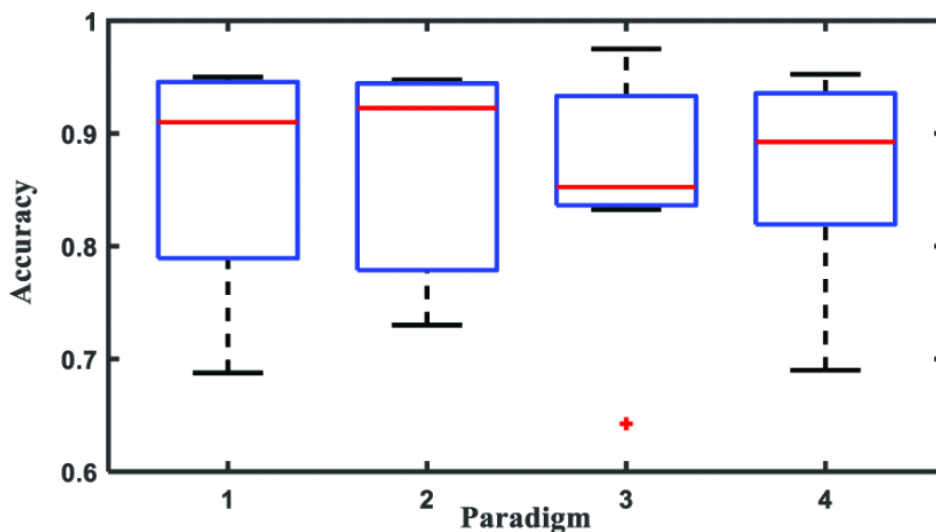


Figure 31: Different speller paradigms' results in accuracy and STD represented by a boxplot. Speller **a** from figure 30 corresponds to result **1**, **b** to **2** and so on (Du et al. 2019).

Source: (Du et al. 2019)

Because the 3D speller implementation is based on the implementation seen in figure 29, it too has a total of 6 frequencies. The chosen frequencies were 4 Hz, 5 Hz, 5.5 Hz, 6 Hz, 7 Hz and 7.4 Hz. These were found by taking a heuristical approach.

These frequencies are then combined with clusters, where each cluster consists of 6 different letters. The subject uses their eye gaze to pick the cluster with the designated letter. Then each letter is flickering at different frequencies. This can be seen, and analyzed in the subject's occipital electrode streams, to determine what letter in

the cluster the subject is looking at (theory 2.7.1) (Kundu and Ari 2022).

4.7.1 Signal Filtering

To remove unwanted noise and artifacts from the EEG signal, some signal filtering techniques were applied. A notch filter of 50 Hz was applied to remove unwanted line noise from the signal (theory 2.2.1, notch filter) (Cheveigné and Nelken 2019; Mannan et al. 2020). The signal was then band-pass filtered with a band-pass of 1 Hz to 15 Hz through a Butterworth filter of order 3. The band-pass was set to 1-15 Hz due to the compromise needed to be made to reduce signal distortions at the start and end of the signal (theory 2.2.1, band-pass) (Winter 2009). The band-pass also allows all the flickering frequencies to be included in the signals.

This was done in two different ways. The initial way was to perform the filtering in NeuroPype, and then send the filtered signal to a Python script for feature extraction. As mentioned, NeuroPype does not support zero-phase filtering when filtering live data. This meant that the streams would be phase distorted, and the computed offset of the initial task would not be correct. Therefore a second script was created, where the filtering was done in Python with zero-phase filters. Since this filtering was done with a zero-phase technique there was added padding to the signal to remove or limit any distortions created at the start and end of the signal from the digital filters (section 2.2.1, padding) (Smith 1989). After passing through the zero-phase Butterworth filter, the padding was removed from the signal for the signal to return to its original length.

These scripts were running side by side when using the BCI speller, and it was decided to use the script with the zero-phase filtering as the main script, due to the synchronization reasons mentioned above. The script using NeuroPype filtering was used as a backup analysis, to ensure that the scripts somewhat agreed on the found correlation.

4.7.2 Feature Extraction

Canonical Correlation Analysis (CCA) was used to extract similarities between the two data sets. The group chose the category of frequency domain features, due to the lack of information about the signal's frequency variation in temporal features (theory 2.8.1), the high computational complexity of time-frequency domain features (theory 2.8.3), and the training cost associated with deep learning based features (theory 2.8.4).

In selecting a frequency domain feature extraction technique, there were several options to consider. Power Spectral Density (PSD) extraction and band power extraction provide information about the signal power across various frequencies. However, techniques could indicate that the highest frequency is not one of the available options, which could complicate the analysis (theory 2.1.4). The CCA offers a more direct approach to comparing the chosen frequencies. The group, therefore, chose the CCA algorithm due to its ability to extract similarities between two data sets, and not being affected by phase distortion. (theory 2.1.4). CCA has also been implemented in previous studies with great results (Lin et al. 2006; Mannan et al. 2020).

The first data set used in the CCA, X , refers to a set of EEG channels. The channels used were the electrode channels: O1, Oz, O2, P3, P7, Pz, P8, P4. These channels can also be seen in figure 23, where they are shown in yellow. The occipital electrodes were chosen because of the fact that SSVEPs are generated in the occipital region of the brain (theory 2.7.1). The rest of the electrodes were chosen because of the spatial resolution in the EEG, where it is difficult to pinpoint the precise origin of the electrical signals (theory 2.1.4).

There was also introduced a delay to the EEG streams. This delay was a combination of the offset found in the blinking analysis (60 ms) and a delay introduced to take visual delay into consideration (theory 2.7.1). This delay was set to 100 ms, so a total of 160 ms delay was introduced between the original eye tracker signal and the original EEG signal (Mannan et al. 2020).

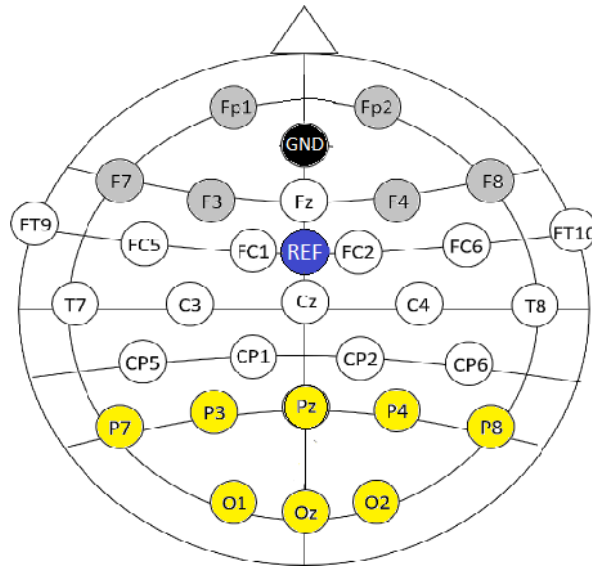


Figure 32: Overview of electrode placement in the ActiCAP Xpress Twist where removed electrodes are shown in gray, and selected channels for the hybrid SSVEP speller are colored yellow. Note that the Ground (GND) electrode still is moved from the very front of the forehead to one line further back.

Source: (Haputhanthri et al. 2019) (Note the image is modified)

The second data set, Y , refers to the set of reference signals, with the same length as X . The reference signals used in SSVEP detection are computed using equation 6 (Mannan et al. 2020).

$$Y_n = \begin{bmatrix} \sin(2\pi f_k t) \\ \cos(2\pi f_k t) \\ \vdots \\ \sin(2\pi N_h f_k t) \\ \cos(2\pi N_h f_k t) \end{bmatrix}, t = \left[\frac{1}{f_s}, \frac{2}{f_s}, \dots, \frac{N}{f_s} \right] \quad (6)$$

f_k refers to the stimulation frequency being flashed for the designated symbol. f_s is the sampling rate (Mannan et al. 2020), which in this project is set to 250 Hz. N refers to the number of samples. For a flickering of 3 seconds, this would mean $N = 250 \cdot 3 = 750$. $N_h = 3$ is the number of harmonic waves. This number of harmonics

is suggested by multiple studies, to make sure no correlation is lost due to phase distortion (Bin et al. 2009; Lin et al. 2006; Mannan et al. 2020; Talevi et al. 2023). The cosine and sine are included, to also avoid or limit the sensitivity to any phase distortion (Lin et al. 2006).

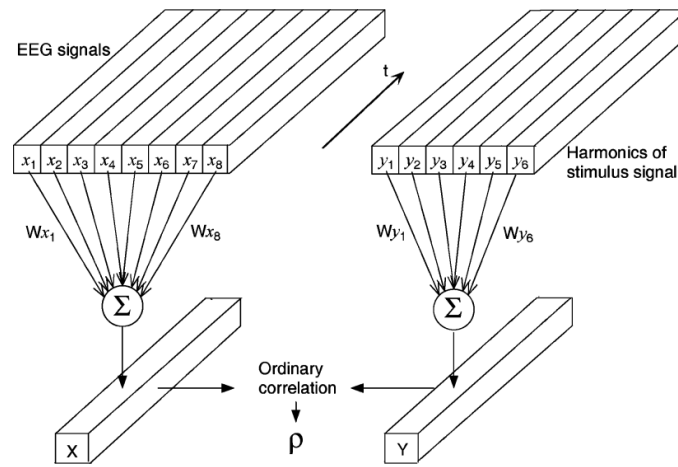


Figure 33: How the different multidimensional variables X and Y are computed as a correlation coefficient, ρ .

Source: (Lin et al. 2006)

Figure 33 shows how the different EEG signals and the stimulus signal is computed into a single component of each, and compared to find the correlation coefficient, ρ .

4.7.3 Recognition Strategy

To determine what frequency the subject is looking at, the strategy utilized is to determine the symbol by the frequency f_s , which satisfies

$$f_s = \max_f \rho(f), \quad f = f_1, f_2, \dots, f_K \quad (7)$$

Equation 7 is a greedy algorithm that predicts the frequency stared at by the subject by choosing the highest correlation coefficient (Lin et al. 2006).

4.7.4 Unity Implementation

The visual stimuli were presented in a Unity scene. After generating the scene, the default skybox was swapped out for a gray one, and a gray wall was put up after the letters to provide a neutral background. Later, the Vive VR headset support was added by copying the GameObject responsible for the rig found in an example scene that came with one of the eye tracker packages, and the default camera was deleted. As for implementation related to the stimuli for the speller, collections of cubes with letters in front of them (letter-cubes) were added. These collections will be referred to as clusters. The cubes were scaled by 1.5 and the letters were TextMeshPro instances with the same color as the background, scaled down to fit inside the perimeter of the cubes. The letter-cubes in a cluster were placed two-and-two in three rows and the

final speller setup is as seen in figure 34 with one of the clusters indicated by a red rectangle.

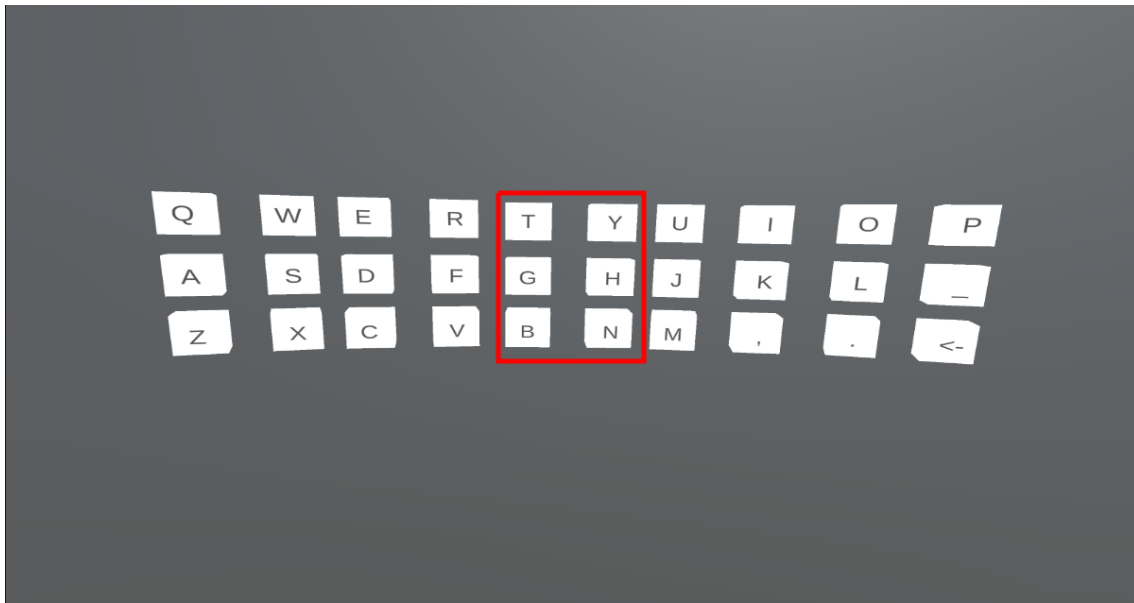


Figure 34: Image of the speller where one of the clusters is highlighted.

When looking at a cluster, its letter-cubes will immediately fly toward the user and start flickering after 1 second, giving the user of the speller some time to look at the correct letter. Figure 35 shows a cluster that has flown out and figure 36 shows that same cluster when it has started flickering. The cubes flicker with given frequencies for a given period of time, alternating color between white and black but the letter color remains gray always. After the duration, the flickering stops and the user needs to look away from the cluster to reactivate its flickering capability. Alternatively, this can be accomplished by the user closing their eyes as it is interpreted as looking away. Therefore, blinking while gazing at a cluster will reset the procedure regardless of its current state. This setup will also not be affected by the focus-accommodation-conflict introduced with eye tracking in VR, due to this 2D-like layout (theory 2.3).

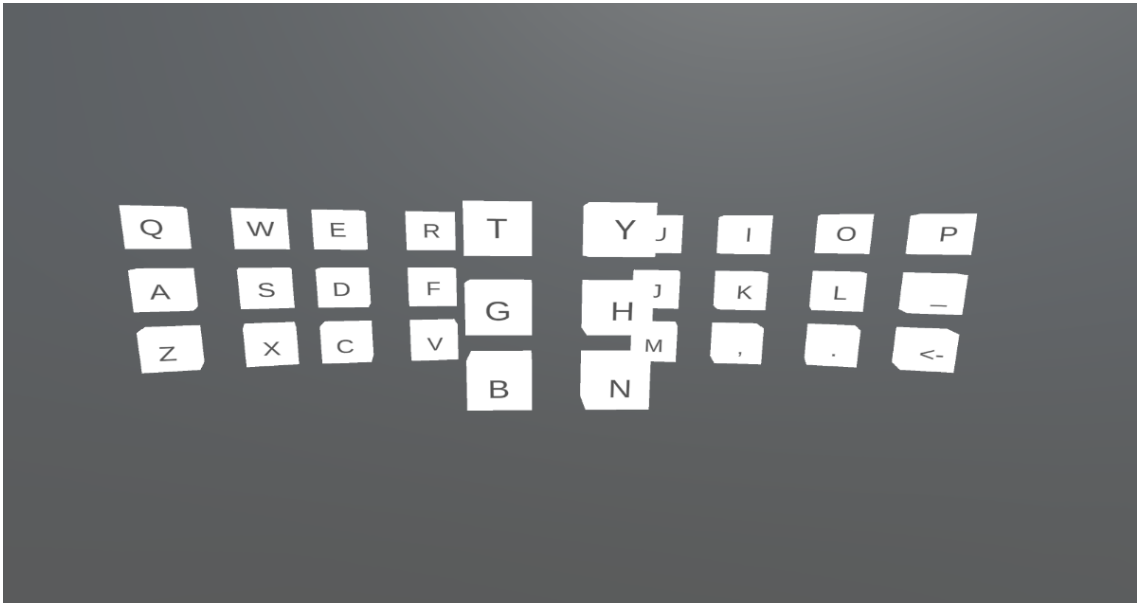


Figure 35: A cluster is looked at by the user, causing it to fly towards the user.

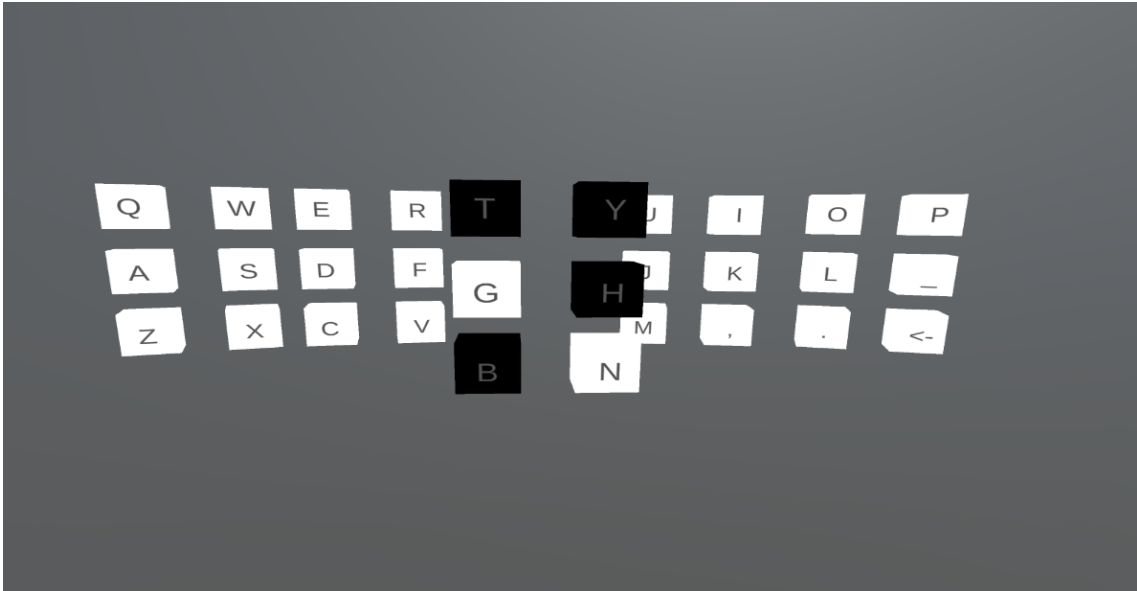


Figure 36: After 1 second has passed, the letter-cubes in the cluster will start flickering with different frequencies.

Due to the clusters flying towards the user when looked at, they were placed in a concave form as seen in figure 37. If they had not been placed that way, they would risk overlapping each other to a larger degree when flying out.

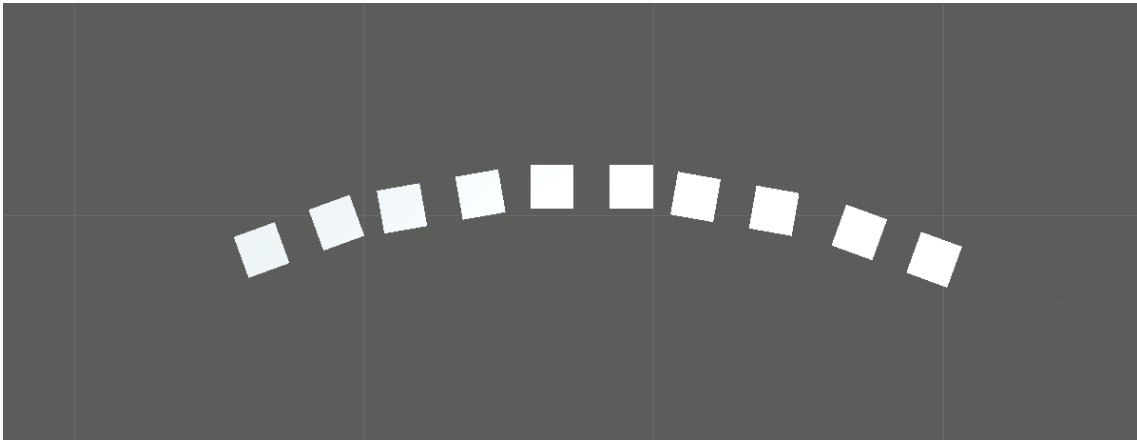


Figure 37: A top-down view of the speller showing the clusters not being put on a line.

All the while the Unity application runs, an LSL stream is sending the integer 0 at a fixed rate. When the letter-cubes in a cluster start flickering, a variable is incremented and then pushed to the outlet of the stream. This stream functions as a marker stream to indicate which when the EEG data should be analyzed by the script running the CCA algorithm. Each increment represents a sampling point and is used as the N needed to calculate t from equation 6. As soon as the letter-cubes in a cluster stop flickering, regardless of the reason, the variable is reset and the stream only sends the integer 0 again.

The Unity application also accepts an LSL stream with the name 'CCA', which is an irregular stream that contains the result of the CCA run in parallel with the application. The result comes in the form of an index that maps to a letter-cube in the cluster last looked at. When a result is registered, it is pushed to a string that the user can look at as seen in figure 38.

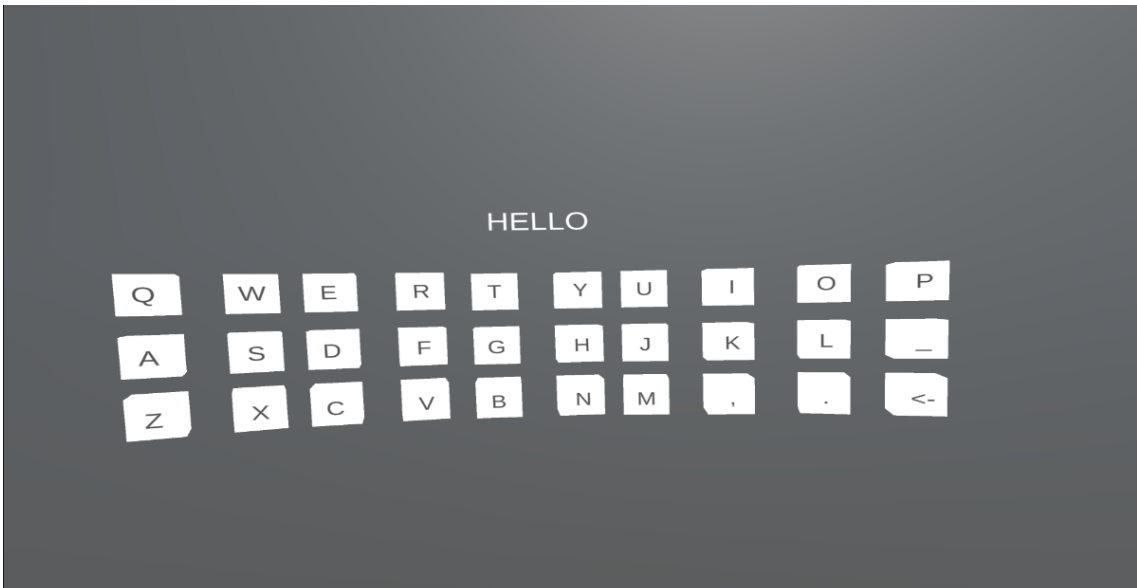


Figure 38: After a user has spelled a letter, it is pushed to a string visible over the speller keyboard. Here, the user has spelled 'HELLO'.

4.8 BCI Speller Testing

Testing the BCI speller consisted of 7 participants spelling a sentence consisting of 4 words on two different occasions, to accommodate for training. Using the speller only once could provide less accurate results due to factors such as the participants not knowing how to operate the speller. There was a period of 3 days between testing. One of the participants would be the group member mainly used for testing the speller during development, the rest would be arbitrary individuals of both sexes between the age of 20-25. Each participant had to sign a letter of consent (appendix A) before participating, as described in Research Methodology 3.5.

The speller was set up with a traditional QWERTY layout, but the bottom right was backspace and the one directly over backspace was space, resulting in 5 clusters. The final sentence the participants were to spell was also placed such that they could look at it if they needed to as seen in figure 39.

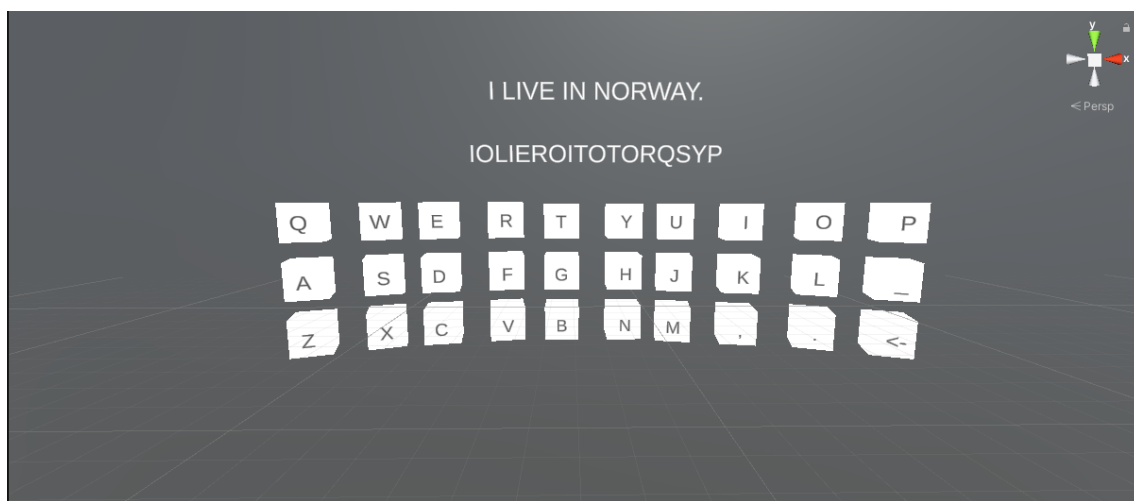


Figure 39: The figure shows the speller setup as seen for the participants during testing. The upper string is what the participant was trying to get and the lower string was what one might spell instead.

An image of how the participants wore the necessary equipment is shown in figure 40. They were all told to hold the VR headset up to their face so as not to interfere with the electrodes at the back of the head. These electrodes can not be moved, because they need to be placed at the visual cortex to accurately record the SSVEP frequencies.



Figure 40: An image showing how a participant would wear the equipment when testing the speller.

The frequencies set up for a single cluster can be seen in figure 41. This setup was the same for every cluster in the speller. The sentence the participants were asked to spell was 'I LIVE IN NORWAY.', and to continue on to the next letter even if the previously spelled letter was wrong. This test was done multiple times with varying flickering durations, first of 6 seconds, then 5 seconds, 4 seconds, 3 seconds, and lastly 2 seconds. Each subject had a break of up to 1 minute between each round, but they still had to wear the EEG during this break, due to the time consumption of equipping it correctly.

4 Hz	5 Hz
5.5 Hz	6 Hz
7 Hz	7.4 Hz

Figure 41: The figure shows the frequency layout of a single cluster, from top left to bottom right, 4 Hz, 5 Hz, 5.5 Hz, 6 Hz, 7 Hz and 7.4 Hz.

Before the participants started spelling, they were allowed to play around with the interface to get a feel for how to use it and before the testing occurred. They were also told to concentrate hard on the letter they would like to spell. While the participants were spelling, the group members counted the letters which were successfully spelled for the later accuracy analysis.

In the initial test round, the eye trackers were introduced to the delay of 60 ms found in the synchronization of the EEG and eye tracker and an ocular delay of 100 ms as mentioned. In the retesting, there was introduced a delay to the eye trackers of 60 ms, and then the eye trackers were expedited 100 ms. This created an additional

offset of 100 ms between the eye trackers and the EEG. This offset is then doubled (200 ms) when taking the ocular delay into consideration (theory 2.7.1).

5 Results

This section is split into three different parts, which display the scientific, engineering, and administrative results respectively.

5.1 Scientific Results

The results from the synchronization, speller implementation, and the two test rounds are presented here.

5.1.1 EEG and Eye Tracker Synchronization

Initially, recordings were done with a single eye blinking and both eyes blinking to determine if there was a difference in performance. Both eyes blinking gave the lowest Standard Deviation (STD) and was chosen as the main blinking technique. The data sets used contains 229 blinks, spread around 23 different recordings with 10 blinks each. Some blinks were removed because the eye tracker did not always register blinks correctly, which will be displayed as a None value. This happened because the subject tried to blink as fast as possible, which sometimes resulted in foul, and unusable blinks.

Blinking With The Left Eye Only Table 1 shows the offset of blinking with the left eye only, instead of both. Note that this was done in a small number of blinking samples (63 blinks). The blinks were streamed at a sampling rate of 120 Hz, unlike the rest of the project which utilized a rate of 250 Hz. Table 2 shows the STD and SE of the left blinks, which are 43 ms and 16 ms respectively.

Left Blinks	Blink 0	Blink 1	Blink 2	Blink 3	Blink 4	Blink 5	Blink 6	Blink 7	Blink 8	Time between index (s)	Sum (indexes)	Average (indexes)	Average time (ms)
LSLDataLeft1	-14	-13	-20	-18	-21	-15	-16	-12	-15	0.008333	-143.99	-16.00	-133.33
LSLDataLeft2	-23	-22	-23	-24	-23	-20	-21	-18	-13	0.008333	-186.99	-20.78	-173.15
LSLDataLeft3	-20	-19	-19	-20	-21	-23	-21	-20	-20	0.008333	-182.99	-20.33	-169.44
LSLDataLeft4	-14	-15	-14	-16	-18	-15	-15	-14	-14	0.008333	-134.99	-15.00	-125.00
LSLDataLeft5	-15	-14	-14	-14	-19	-19	-14	-15	-6	0.008333	-129.99	-14.44	-120.37
LSLDataLeft6	-38	-44	-44	-37	-33	-42	-22	-28	26	0.008333	-261.99	-29.11	-242.59
LSLDataLeft7	-12	-14	-20	-16	-20	-19	-15	-14	-14	0.008333	-143.99	-16.00	-133.33

Table 1: Recording of the left eye blink offset between the EMG and eye tracker data before shift. Each test had the subject blink 9 times. The columns from 0-8 are the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.

Left Blinks	Sum (indexes)	Average (indexes)	Average time (ms)
Average	-169.28	-18.81	-156.74
STD	46.72	5.19	43.26
SE	17.66	1.96	16.35

Table 2: STD, SE and average values for left blinks before shift. Based on values in table 1.

Phase Distortion The data was initially recorded without a zero-phase filter. This causes some phase distortion to the filtered data and makes the peak distorted com-

pared to the blink onset. This can be read more about in the theory about Butterworth and IIR filter's characteristics (theory 2.2.1).

Figure 42 shows the normalized, Butterworth filtered EMG and the eye tracking data in the same graph. As the theory states, the full closure of an eye blink will be the peak of the EMG. Therefore the start of the full closure of the eye (green/first vertical line), should be aligned with the maximum of the EMG graph.

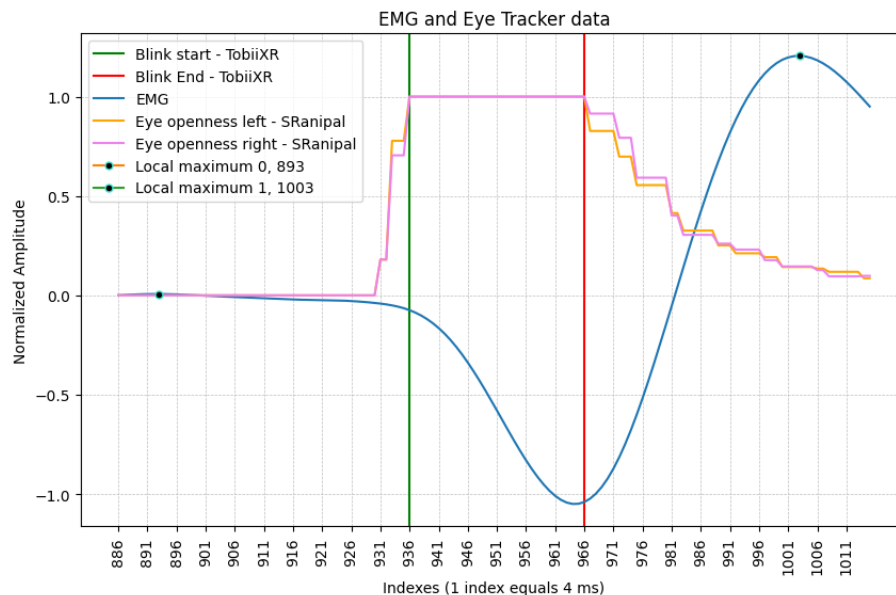


Figure 42: The same blink as in figure 27 shown graphically with the same measurements. This signal is however filtered with a normal Butterworth filter (not zero-phase), which results in phase distortion. The peak will therefore be further away than it actually should be compared to a zero-phase filter.

Table 3 shows the blinking data over all recordings filtered with a regular Butterworth filter.

Distorted Signal	Blink 1	Blink 2	Blink 3	Blink 4	Blink 5	Blink 6	Blink 7	Blink 8	Blink 9	Blink 10	Time Between Indexes (s)	Sum (indexes)	Average (indexes)	Average time (ms)
LSLData-17-04 2-1	-23	-30	-27	-28	-28	-28	-24	-29	-26	-28	0.004	-270.996	-27.1	-108.4
LSLData-17-04 2-6	-26	-26	-24	-27	-26	-26	-26	-25	-19	-23	0.004	-247.996	-24.8	-99.2
LSLData-17-04 2-14	-25	-27	-23	-25	-26	-24	-24	-25	-24	-27	0.004	-249.996	-25	-100
LSLData-17-04 8	-22	-23	-23	-25	-23	-20	-23	-24	-23	-23	0.004	-228.996	-22.9	-91.6
LSLData-17-04 2-11	-26	-27	-22	-26	-27	-23	-27	-25	-29	-26	0.004	-257.996	-25.8	-103.2
LSLData-17-04-10	-28	-28	-28	-26	-28	-22	-28	-28	-29	-29	0.004	-273.996	-27.4	-109.6
LSLData-17-04 2-3	-27	-25	-26	-32	-30	-27	-32	-29	-30	-31	0.004	-288.996	-28.9	-115.6
LSLData-17-04-11	-21	-26	-26	-22	-25	-23	-25	-23	-24	-22	0.004	-236.996	-23.7	-94.8
LSLData-17-04 2-5	-26	-27	-26	-24	-20	-24	-22	-18	-22	-23	0.004	-231.996	-23.2	-92.8
LSLData-17-04-15	-20	-19	-25	-27	-22	-19	-25	-25	-21	None	0.004	-202.996	-22.556	-90.22
LSLData-17-04 2-13	-31	-33	-30	-26	-30	-29	-30	-31	-30	-31	0.004	-300.996	-30.1	-120.39
LSLData-17-04 2-8	-27	-29	-30	-27	-24	-29	-31	-27	-30	-29	0.004	-282.996	-28.3	-113.2
LSLData-17-04 7	-21	-21	-22	-22	-22	-22	-24	-21	-21	-25	0.004	-220.996	-22.1	-88.4
LSLData-17-04-9	-23	-24	-21	-24	-23	-23	-25	-21	-23	-24	0.004	-230.996	-23.1	-92.4
LSLData-17-04 2-2	-26	-23	-24	-23	-25	-25	-27	-26	-26	-22	0.004	-246.996	-24.7	-98.8
LSLData-17-04 2-4	-25	-27	-25	-27	-29	-26	-29	-24	-25	-27	0.004	-263.996	-26.4	-105.6
LSLData-17-04-14	-25	-24	-21	-26	-20	-22	-22	-24	-25	-21	0.004	-229.996	-23	-92
LSLData-17-04 2-7	-25	-30	-29	-27	-26	-20	-27	-30	-30	-29	0.004	-272.996	-27.3	-109.2
LSLData-17-04 2-15	-34	-27	-32	-26	-31	-24	-30	-28	-30	-28	0.004	-289.996	-29	-116
LSLData-17-04 2-9	-27	-24	-23	-27	-25	-28	-23	-25	-25	-27	0.004	-253.996	-25.4	-101.6
LSLData-17-04 2-10	-26	-23	-24	-24	-23	-26	-24	-22	-24	-22	0.004	-237.996	-23.8	-95.2
LSLData-17-04 2-12	-25	-24	-26	-27	-23	-28	-33	-29	-28	-28	0.004	-270.996	-27.1	-108.4
LSLData-17-04-3	-19	-20	-19	-18	-25	-24	-22	-29	-22	-21	0.004	-218.996	-21.9	-87.6

Table 3: Recording of blink offset between the EMG and eye tracker data **where all of the excess samples are cut at the start**. Each test had the subject blink 10 times. The columns from 0-9 are the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.

The offset found from the phase-distorted signal 101 ms (table 4).

Distorted Signal	Sum (indexes)	Average (indexes)	Average time (ms)
Average	-252.74	-25.37	-101.49
STD	25.91	2.43	9.73
SE	5.40	0.51	2.03

Table 4: Time offset, STD and SE for the phase distorted signal (table 3).

Different File Lengths of Recorded Files The EEG and eye tracking recording files had different lengths. This creates uncertainty about whether the recordings were cut short at the start, end, or a mix of both. Therefore there was done analysis of the data with different cutting of the excess data at the end, start, and, 1/2, 1/3, 1/4, and 1/5 at the start. The tables of the recordings and data showing the cutting of the end, start 1/3, 1/4, and 1/5 at the start can be found in the appendix B. The histograms associated with the recordings can also be found there.

Table 5 shows an overview of the different index offsets between when the eye is fully closed in a blink recorded with the eye tracker, and the EMG peak that indicates the eye is fully closed in a blink.

Blink Cut Start %	Blink 1	Blink 2	Blink 3	Blink 4	Blink 5	Blink 6	Blink 7	Blink 8	Blink 9	Blink 10	Time Between Indexes (s)	Sum (indexes)	Average (indexes)	Average time (ms)
LSLData-17-04 2-1	-12	-19	-18	-18	-19	-20	-13	-19	-20	-19	0.004	-176.996	-17.7	-70.8
LSLData-17-04 2-6	-11	-11	-9	-15	-13	-14	-10	-14	-10	-8	0.004	-114.996	-11.5	-46
LSLData-17-04 2-14	-13	-16	-12	-13	-14	-12	-12	-13	-12	-19	0.004	-135.996	-13.6	-54.4
LSLData-17-04-8	-13	-13	-13	-13	-19	-5	-9	-12	-11	-10	0.004	-117.996	-11.8	-47.2
LSLData-17-04 2-11	-16	-15	-12	-15	-22	-13	-20	-16	-24	-18	0.004	-170.996	-17.1	-68.4
LSLData-17-04-10	-17	-19	-16	-12	-16	-9	-16	-15	-16	-18	0.004	-153.996	-15.4	-61.6
LSLData-17-04 2-3	-17	-14	-20	-23	-18	-16	-24	-18	-18	-19	0.004	-186.996	-18.7	-74.8
LSLData-17-04-11	-9	-15	-14	-15	-15	-12	-14	-12	-13	-11	0.004	-129.996	-13	-52
LSLData-17-04 2-5	-21	-18	-21	-20	-12	-18	-14	-11	-16	-15	0.004	-165.996	-16.6	-66.4
LSLData-17-04-15	-11	-15	-15	-22	-17	-8	-18	-16	-17	None	0.004	-138.996	-15.444	-61.77
LSLData-17-04 2-13	-20	-22	-19	-15	-19	-16	-27	-19	-19	-21	0.004	-196.996	-19.7	-78.8
LSLData-17-04 2-8	-16	-17	-21	-14	-11	-19	-19	-16	-18	-16	0.004	-166.996	-16.7	-66.8
LSLData-17-04-7	-10	-9	-9	-11	-13	-17	-15	-12	-15	-18	0.004	-128.996	-12.9	-51.6
LSLData-17-04-9	-13	-14	-9	-20	-13	-12	-14	-9	-19	-16	0.004	-138.996	-13.9	-55.6
LSLData-17-04 2-2	-17	-14	-12	-14	-14	-14	-20	-18	-15	-11	0.004	-148.996	-14.9	-59.6
LSLData-17-04 2-4	-15	-17	-17	-15	-20	-13	-17	-10	-17	-14	0.004	-154.996	-15.5	-62
LSLData-17-04-14	-16	-13	-11	-17	-11	-10	-15	-20	-15	-12	0.004	-139.996	-14	-56
LSLData-17-04 2-7	-12	-18	-17	-16	-15	-7	-14	-19	-18	-17	0.004	-152.996	-15.3	-61.2
LSLData-17-04 2-15	-23	-16	-23	-14	-18	-12	-18	-16	-19	-16	0.004	-174.996	-17.5	-70
LSLData-17-04 2-9	-16	-13	-9	-16	-11	-17	-8	-13	-21	-16	0.004	-139.996	-14	-56
LSLData-17-04 2-10	-16	-12	-14	-16	-12	-17	-14	-11	-14	-12	0.004	-137.996	-13.8	-55.2
LSLData-17-04 2-12	-16	-16	-16	-16	-17	-18	-23	-20	-15	-17	0.004	-173.996	-17.4	-69.6
LSLData-17-04-3	-11	-15	-14	-16	-21	-20	-18	-25	-19	-21	0.004	-179.996	-18	-72

Table 5: Recording of blink offset between the EMG and eye tracker data **where half of the excess samples are cut at the start, and the rest at the end.** Each test had the subject blink 10 times. The columns from 0-9 are the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.

The histogram (figure 43) shows the different average offsets for each recording plotted, a normal distribution is also added to the graph. The red vertical lines have a space of 8 ms which is approximately the frequency of the eye tracker (120 Hz). The purple line with points shows the normal distribution of the data, where the points are the actual average offset.

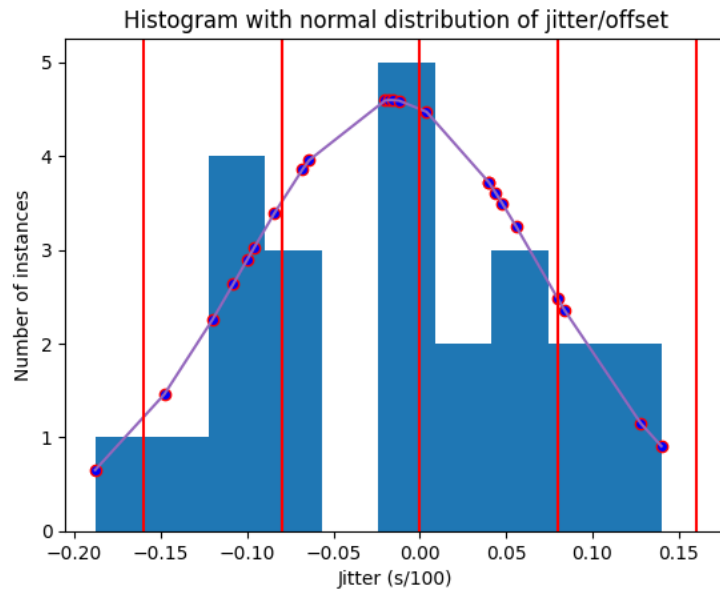


Figure 43: Histogram with the normal distribution of offset computed when cutting half of the excess samples off at the end. X-axis is $\frac{1}{100}$ seconds. Y-axis is the number of instances with a given index. Red lines split the different indexes of 8.3 ms which is the frequency of the eye tracker (120 Hz).

Table 6 shows all the different techniques to determine where the samples should start compared to each other. The blinks where half of the samples were cut from the start, and the rest from the end, gave the lowest STD and SE. When using this technique, it is determined that the offset between the EEG and the eye trackers are of 61-62 milliseconds. This means that the EEG should be moved 62 ms forward in time, or the eye tracking data should be delayed by 62 ms. The Standard Deviation (STD) in average time is 8.87-15.49 ms, and the Standard Error (SE) in average time is 1.85-2.57 ms using the different techniques.

Blink Cut End	Time offset (ms)	Time offset (indexes)	STD (ms)	STD (indexes)	SE (ms)	SE (indexes)	Single index (ms)
Cut End	-62.86	-15.71	12.26	2.57	2.56	0.64	4
Cut Start	-60.08	-15.02	15.49	4.04	3.23	0.81	4
Cut Start ½	-61.64	-15.41	8.87	2.23	1.85	0.46	4
Cut Start 1/3	-62.16	-15.54	8.93	2.03	1.86	0.47	4
Cut start ¼	-62.34	-15.58	9.56	2.11	1.99	0.50	4
Cut start 1/5	-62.34	-15.58	9.63	2.05	2.01	0.50	4

Table 6: Time offset, STD and SE for the different cut techniques used (table 5, and tables 17, 18, 19, 20, 21 from appendix B). The lowest STD and SE was obtained by the cut start 1/2.

Blink Recordings After Shift Table 7 shows the same data as in table 5, but after the data is shifted according to the found offset of 60 ms (15 indexes) (table 6). Table 8 shows the offset after shifting the data. The new offset is computed to be 1.64 ms and less than half an index.

Blink Cut Start ½ After Shift	Blink 1	Blink 2	Blink 3	Blink 4	Blink 5	Blink 6	Blink 7	Blink 8	Blink 9	Blink 10	Time Between Indexes (s)	Sum (indexes)	Average (indexes)	Average time (ms)
LSLData-17-04 2-1	3	-4	-3	-3	-4	-5	2	-4	-5	-4	0.004	-26.996	-2.7	-10.8
LSLData-17-04 2-6	4	4	6	0	2	1	5	1	5	7	0.004	35.004	3.5	14
LSLData-17-04 2-14	2	-1	3	2	1	3	3	2	3	-4	0.004	14.004	1.4	5.6
LSLData-17-04-8	2	2	2	2	-4	10	6	3	4	5	0.004	32.004	3.2	12.8
LSLData-17-04 2-11	-1	0	3	0	-7	2	-5	-1	-9	-3	0.004	-20.996	-2.1	-8.4
LSLData-17-04 2-10	-2	-4	-1	3	-1	6	-1	0	-1	-3	0.004	-3.996	-0.4	-1.6
LSLData-17-04 2-3	-2	1	-5	-8	-3	-1	-9	-3	-3	-4	0.004	-36.996	-3.7	-14.8
LSLData-17-04-11	6	0	1	0	0	3	1	3	2	4	0.004	20.004	2	8
LSLData-17-04 2-5	-6	-3	-6	-5	3	-3	1	4	-1	0	0.004	-15.996	-1.6	-6.4
LSLData-17-04-15	4	0	0	-7	-2	7	-3	-1	-2	None	0.004	-3.996	-0.444	-1.78
LSLData-17-04 2-13	-5	-7	-4	0	-4	-1	-12	-4	-4	-6	0.004	-46.996	-4.7	-18.8
LSLData-17-04 2-8	-1	-2	-6	1	4	-4	-4	-1	-3	-1	0.004	-16.996	-1.7	-6.8
LSLData-17-04-7	5	6	6	4	2	-2	0	3	0	-3	0.004	21.004	2.1	8.4
LSLData-17-04-9	2	1	6	-5	2	3	1	6	-4	-1	0.004	11.004	1.1	4.4
LSLData-17-04 2-2	-2	1	3	1	1	1	-5	-3	0	4	0.004	1.004	0.1	0.4
LSLData-17-04 2-4	0	-2	-2	0	-5	2	-2	5	-2	1	0.004	-4.996	-0.5	-2
LSLData-17-04-14	-1	2	4	-2	4	5	0	5	0	3	0.004	10.004	1	4
LSLData-17-04 2-7	3	-3	-2	-1	0	8	1	-4	-3	-2	0.004	-2.996	-0.3	-1.2
LSLData-17-04 2-15	-8	-1	-8	1	-3	3	-3	-1	-4	-1	0.004	-24.996	-2.5	-10
LSLData-17-04 2-9	-1	2	6	-1	4	-2	7	2	-6	-1	0.004	10.004	1	4
LSLData-17-04 2-10	-1	3	1	-1	3	-2	1	4	1	3	0.004	12.004	1.2	4.8
LSLData-17-04 2-12	-1	-1	-1	-1	-2	-3	-8	-5	0	-2	0.004	-23.996	-2.4	-9.6
LSLData-17-04-3	4	0	1	-1	-6	-5	-3	-10	-4	-6	0.004	-29.996	-3	-12

Table 7: Recording of blink offset where half of the excess samples are cut at the start after shifting found offset. This was the technique with the lowest STD (table 5).

Blink Cut Start ½ After Shift	Sum (indexes)	Average (indexes)	Average time (ms)
Average	-4.08	-0.41	-1.64
STD	22.18	2.23	8.87
SE	4.63	0.46	1.85

Table 8: Time offset, STD and SE from table 7 in sum using indexes, and average using both indexes and time (ms).

5.1.2 BCI Speller Implementation

Results regarding the implementation of the BCI speller.

Table 9 shows an overview of the correlation coefficients computed by a CCA on a mock EEG signal. This shows that when shifting the signal, there was no significant difference. This indicated that the CCA is not phase sensitive.

Phase sensitivity check	4 Hz	5 Hz	5.5 Hz	6 Hz	7 Hz	7.4 Hz
Shifted 0 Samples	0.029	0.051	0.081	0.041	0.815	0.063
Shifted 25 Samples	0.046	0.041	0.079	0.053	0.821	0.038
Shifted 50 Samples	0.034	0.030	0.046	0.033	0.819	0.049
Shifted 75 Samples	0.032	0.034	0.061	0.053	0.824	0.036
Shifted 100 Samples	0.050	0.019	0.037	0.032	0.813	0.024
Shifted 125 Samples	0.042	0.017	0.061	0.028	0.823	0.040
Shifted 150 Samples	0.045	0.043	0.063	0.030	0.818	0.034
Shifted 175 Samples	0.069	0.038	0.045	0.055	0.818	0.021
Shifted 200 Samples	0.033	0.022	0.051	0.033	0.821	0.038
Shifted 225 Samples	0.026	0.039	0.042	0.033	0.811	0.027
Shifted 250 Samples	0.033	0.047	0.069	0.047	0.820	0.036

Table 9: Different CCA values after shifting a mock EEG signal. The target frequency was 7 Hz and the sampling rate was 250 Hz. The table shows different ways the signal could have been phase distorted.

5.1.3 BCI Speller Initial Testing

After finalizing the speller, the group tested it on 7 participants on two separate occasions. The following section shows the result of the accuracy of the speller for these individuals when they initially tested it.

Table 10 is a collection of how many correct symbols the subjects got for each of the flickering periods. The rightmost column contains the theoretical maximum amount of symbols a subject could get, and the bottommost row contains the average amount of symbols the subjects got for each flickering period.

Subject	6 s	5 s	4 s	3 s	2 s	Total symbols
Subject 1	17	17	17	17	16	17
Subject 2	15	10	5	6	9	17
Subject 3	14	11	9	9	8	17
Subject 4	7	6	6	4	5	17
Subject 5	14	15	11	3	6	17
Subject 6	13	10	9	10	5	17
Subject 7	15	17	12	12	7	17
Average	13.57	12.29	9.86	8.71	8	17

Table 10: Shows an overview of how many correct symbols each test subject got for the different flickering lengths in seconds (6 s - 2 s). The lower the flickering length, the lower amount of correct symbols.

Table 11 shows the accuracy for each subject for each flickering period and a couple of average accuracies. The row at the bottom shows the accuracy for each flickering period across all subjects.

Subject	6 s	5 s	4 s	3 s	2 s
Subject 1	100.00%	100.00%	100.00%	100.00%	94.12%
Subject 2	88.24%	58.82%	29.41%	35.29%	52.94%
Subject 3	82.35%	64.71%	52.94%	52.94%	47.06%
Subject 4	41.18%	35.29%	35.29%	23.53%	29.41%
Subject 5	82.35%	88.24%	64.71%	17.65%	35.29%
Subject 6	76.47%	58.82%	52.94%	58.82%	29.41%
Subject 7	88.24%	100.00%	70.59%	70.59%	41.18%
Average	79.83%	72.27%	57.98%	51.26%	47.06%

Table 11: Shows an overview of how the accuracy each test subject obtained for the different flickering lengths in seconds (6 s - 2 s). The lower the flickering length, the lower the accuracy

Table 12, shows how many of each frequency the participants successfully got for the different flickering periods in total. The rightmost column shows how many symbols it would be possible to get, e.g. for 4 Hz a total of 14 would be the theoretical max. The row next to last at the bottom is labeled 'Total' and tallies all the values of the preceding cells in their respective columns. In the bottom row, the percentage of the total symbols which was successfully spelled for each flickering period is presented.

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	13	14	13	12	13	14
5 Hz	38	30	24	22	22	42
5.5 Hz	11	11	10	7	9	14
6 Hz	15	15	9	10	5	21
7 Hz	4	4	5	2	2	7
7.4 Hz	14	12	8	8	5	21
Total	95	86	69	61	56	119
Accuracy	79.83%	72.27%	57.98%	51.26%	47.06%	

Table 12: Shows how many of the symbols the participants successfully got in total for the different flickering periods in the first round of testing found in appendix C.

The results from the initial speller testing are based on the recordings of each subject. A more detailed overview of each subject result and accuracy can be found in appendix C.

5.1.4 BCI Speller Retesting

After 3 days had passed, the 7 participants came to test the speller one more time. The following section shows the result of the accuracy of the speller for these individuals the second time.

Table 13 is a collection of how many correct symbols the subjects got for each of the flickering periods. The rightmost column contains the theoretical maximum amount of symbols a subject could get, and the bottommost row contains the average amount of symbols the subjects got for each flickering period.

Subject	6 s	5 s	4 s	3 s	2 s	Total symbols
Subject 1	17	17	17	17	14	17
Subject 2	8	5	6	4	5	17
Subject 3	7	7	2	5	9	17
Subject 4	4	6	7	3	2	17
Subject 5	3	3	14	12	9	17
Subject 6	2	2	3	2	2	17
Subject 7	13	7	11	9	6	17
Average	7.71	6.71	8.57	7.43	6.71	17

Table 13: Shows an overview of how many correct symbols each test subject got for the different flickering lengths in seconds (6 s-2 s).

Table 14 shows the accuracy for each subject for each flickering period. The row at the bottom shows the accuracy for each flickering period across all subjects.

Subject	6 s	5 s	4 s	3 s	2 s
Subject 1	100.00%	100.00%	100.00%	100.00%	82.35%
Subject 2	47.06%	29.41%	35.29%	23.53%	29.41%
Subject 3	41.18%	41.18%	11.76%	29.41%	52.94%
Subject 4	23.53%	35.29%	41.18%	17.65%	11.76%
Subject 5	17.65%	17.65%	82.35%	70.59%	52.94%
Subject 6	11.76%	11.76%	17.65%	11.76%	29.41%
Subject 7	76.47%	41.18%	64.71%	52.94%	35.29%
Average	45.38%	39.50%	50.42%	43.70%	42.02%

Table 14: Shows an overview of how the accuracy each test subject obtained for the different flickering lengths in seconds (6 s - 2 s)

The results from the retesting of the speller are based on the recordings of each subject. A more detailed overview of each subject result and accuracy can be found in appendix D and a summary of these observations is found in table 15. The table shows how all participants did in total for each of the flickering periods and each of the frequencies. Each row and column has the same explanation as table 12 but with different data.

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	10	11	11	10	10	14
5 Hz	17	19	24	22	21	42
5.5 Hz	6	4	6	4	7	14
6 Hz	8	5	10	7	4	21
7 Hz	3	3	3	2	2	7
7.4 Hz	10	5	6	7	3	21
Total	54	47	60	52	47	119
Accuracy	45.38%	39.50%	50.42%	43.70%	39.50%	

Table 15: Shows how many of the symbols the participants successfully got in total for the different flickering periods in the second round of testing found in appendix D.

5.2 Engineering Results

The engineering results include goals defined earlier in the project, and other bits of knowledge and discoveries made during the project with a main focus on the practical aspect. After presenting the goals for the project, which later will be discussed, a section regarding difficulties in developing the BCI speller will follow.

5.2.1 Results Related to Goals

The functional and non-functional attributes defined in the vision document, as seen in appendix F, are considered engineering goals. Firstly the functional attributes will be in focus followed by the non-functional ones.

Synchronized equipment The first functional attribute defined in the vision document was that the EEG and eye tracker should be synchronized within a 1 ms margin

of error. This lines up with research question 1 and was the main focus of the first part of the project. Ultimately, this was not possible to do.

Working speller The second functional attribute was to use the synchronized equipment to create a working BCI speller. This aligns with the second research question and was the main focus of the second part of the project. The group considers this goal met.

Little to no discrimination The following functional attribute was for the speller to work with multiple individuals, which it did to some degree.

Accuracy of acceptable degree The last functional attribute was for the speller to be at least 80% accurate. The testing of the speller showed that it was more accurate for some individuals than others.

Follow standard scientific conventions Over to the non-functional attributes, the first one was for the group to follow standard scientific conventions such as controlling sources of error during the entire project, both when synchronizing the equipment and testing the BCI speller. This was done to a considerable degree.

Maintainable and readable code The second non-functional attribute was for the group to produce maintainable and readable code with comments explaining what happens in the code, utilizing proper and descriptive names for variables, methods, and other places where that might be relevant. The reason for this attribute being defined was to make the project as easy as possible for any future individuals to continue this project and was considered fulfilled by the group.

Usability When creating a user interface, usability should be considered. The speller is no exception and should per the third non-functional attribute follow general usability guidelines according to the WCAG 2.1 standard, Perceivable. Here, the group thinks they followed the most relevant ones.

The speller should be robust The next attribute was for the underlying implementation of the speller to be robust and work as expected leading to few or no crashes in addition to the controls being intuitive. This was taken into consideration during development.

The speller should be relatively efficient The last non-functional attribute was that using the speller should not be slow to increase the amount of information one can convey over a short period of time. After the testing of the speller, the data indicated that it was not relatively efficient.

5.2.2 BCI Difficulties

The group had some trouble developing the BCI speller, but identified potential sources of error and systematically worked through these to find the most probable reason. While trying to eliminate the potential sources of error, the group was not sure whether the speller did not work due to it malfunctioning or the test person was not compatible with BCIs. Therefore, the search for the root cause is split into two: one where it was unknown whether the test person was compatible with BCIs and another where the person was tested for BCI compatibility. After checking out and fixing several of the potential error sources, one of them resulted in the speller working to some degree and was related to how the flickering was generated.

5.3 Administrative Results

This subsection contains results regarding milestones, meetings, and timesheets in the project.

5.3.1 Milestones

The planned milestones set by the group (appendix G, 2.1 Planned Milestones), at the start of the semester included important deadlines for when delivering the pre-project plan, the poster presentation of the research questions, self-imposed deadline of the final report, absolute deadline and when the digital presentation of the project was due. These milestones are more about the administrative aspects of the project. The milestones that were defined were among others: when the data gathering pipelines should be complete, when the synchronization of the eye tracking and EEG data should be complete, and when to gather test subjects for data collection.

The actual milestones (appendix G, 2.2 Actual Milestones) shared the same administrative goals as the planned milestones. As for the others, they underwent significant changes. The synchronization and setup of the pipeline was completed faster than expected and was officially accomplished on the 7th of March, instead of the predicted 6th of April. After these milestones, additional tasks regarding the BCI speller were added to the assignment. The implementation of the speller was started on the 7th of March, and a POC was presented on the 30th of March. A completely working BCI speller was accomplished on the 14th of April. The initial testing of the speller was done on the 24th of April, and the retesting was done on the 28th of April. Between these two test days, the group participated in the Sensor Tech Day and presented their BCI speller on the 26th of April. After the testing, the rest of the milestones were the self-imposed, and actual deadlines of the thesis.

5.3.2 Meetings

The group had a meeting every other week. However, there were sometimes more frequent meetings if there were upcoming deadlines, or fewer meetings if there were holidays or if the group felt no meeting was needed. There was a weekly meeting the

last three weeks before the deadline of the thesis (appendix G, 3. Meeting notices and Meeting reports).

The meetings were initialized with a meeting invitation following a template provided by NTNU. The meeting invitation included the date, time, and location of the meeting, including a Mazemap link. The meeting participants were then listed, and an agenda was provided. If the participants couldn't show up, they were told to notify the group by email or phone. These invitations were sent out by Tresselt.

The meeting reports are a summary of what was discussed during the meetings in the different cases on the agenda. It included the date, time, and location. It also included who was present, and who was absent. It also included the chair of the meeting, most of the time Tresselt, and the secretary, which was Larsen. Then then the discussion around the different cases on the agenda was transcribed by the secretary.

5.3.3 Timesheets

The group tracked the time spent by each member, and what each member had done during that time. A summary of what work was done each week was also added (appendix G, Timesheets and Status reports). This was done using a template provided by NTNU.

Weeks 1-11 had about 15 hours each, then the amounts gradually increased. Week 9-11 had the least amount of work done. Week 14 was the Easter holiday, and therefore 0 hours of work were done. During the last five weeks of the project, each member worked 9 hours every day. The group reached the target total of 500 plus minus 10 % hours and worked a total of 1004 hours. There was only a single hour separating the two group members (501.5 hours, and 502.5 hours).

The first few weeks of the project were devoted to research of relevant topics, preparing the pre-project plan and vision document, and EEG equipment training. Additionally, the group had to assist the supervisor in setting up the new PC that would provide the group with more computational power. This PC had a number of issues, which the group spent a lot of time resolving.

It was necessary to share equipment because another bachelor group was also using the VR headset with the eye tracker. Each group used the equipment for 2 - 3 days a week. Being compelled to work on the primary report and other administrative components of the project, was actually a benefit for the group.

6 Discussion

The discussion is divided into seven primary sections. The initial section contains some thoughts regarding the limitations of the project. The two subsequent sections explore the results obtained from the synchronization of the EEG and eye tracker, as well as the outcomes from the initial testing and retesting of the BCI speller. The following section discusses some aspects regarding the use of NeuroPype. Subsequently, the teamwork, engineering results, and administrative results will be discussed, providing a complete understanding of the various aspects involved in this project.

6.1 Limitations

Before addressing the results of the synchronization, it is important to acknowledge the limitations of the project. The main limitation for finding the offset and jitter between the EEG and the eye tracker is the sampling rate of the eye tracker. The eye tracker has a sampling rate of 120 Hz. This gives an accuracy of:

$$\frac{1}{120} s = 0.00833 s = 8.33 ms$$

This makes it impossible to get the specified accuracy of less than 1 ms. To confidently achieve synchronization of less than a millisecond, both the eye tracker and the EEG would have to have a minimum sampling rate of 1000 Hz (1 sample every millisecond).

The rest of the equipment also introduced some limitations, like the VR headset only having a refresh rate of 90 Hz, limiting the potential flickering rate of the speller. Mannan et al. (Mannan et al. 2020) used a 144 Hz screen which might have made it possible for them to display higher frequencies, for example. Limitations can also be related to artifacts or noise in the eye tracking and, or EEG/EMG data. This could delay things like the sample of when the subject is closing their eyes in the eye tracker, or distort the blink signal recorded by the EMG.

Another limitation of this project is the time aspect. The group only had 500 hours each to complete the assignment. This included the synchronization and the demonstration of the synchronized equipment. To ensure the group had time to demonstrate the synchronization of the equipment, the number of blinks in the offset analysis was limited.

6.2 EEG and Eye Tracker Synchronization

When synchronizing the EEG and the eye tracker the main goal was to find the offset and jitter between the two data streams. When referring to offset, it means the time between the registration of the eye blink in the EEG and in the eye tracker. When referring to jitter, it means the variation in the offset, which is the same as the Standard Deviation (STD) in the results.

This offset is assumed to be a combination of hardware and potentially software latency between the equipment. The latency introduced is not necessarily constant, which could explain the jitter in the results.

6.2.1 Hardware and Software Latency

The group has treated the offset as a static value, whereas the jitter and latency created by hardware, network transfer, and software might not be constant and, therefore could vary based on the load on the hardware, software, or other factors. As mentioned the jitter is the STD, which can be found in table 6. The group does however not know the amount of jitter created by each hardware component. This is a tedious job to map and compute, due to a large amount of potential jitter sources. To find out how much jitter each component creates is also not the main part of the assignment, but it would have been nice to know what hardware creates the most jitter. Then the group could have explored ways to limit its impact on the synchronization.

The research conducted by the group uses plenty of different hardware like the EEG, the LiveAmps, and the VR headset. It is also important to consider other sources of latency like network latencies when streaming the data, and even biological latencies. The network latency could have been measured similarly to the hardware and software latencies. The potential biological latencies could have been explored by using different individuals for gathering blinking data.

There are some things the group didn't take into consideration in the synchronization, like when equipping the LiveAmp Extender for the EMG. This could have introduced extra hardware latency, compared to when only using the LiveAmp. The impact of latency is unknown, and should maybe have been measured. The group could also always have had the LiveAmp Extender connected, to remove this uncertainty. The consequence of this is that if this did create a delay, the entire pipeline would be delayed accordingly, which is unwanted in live applications that depend on fast responsiveness.

Software latency is also something to take into consideration, especially when streaming the data live. The data was streamed through LSL, which introduces some network jitter, even though this can be counteracted through dejittering (theory 2.4.2). This sort of software latency also is introduced when using NeuroPype, since the data is continuously streamed live through many different nodes and filters. This was however counteracted by filtering in the Python scripts after the streams that were going to be analyzed were stored. This ensured there was no latency to the signals since they weren't streamed when filtered. This was the case for both the synchronization and the BCI speller.

The software latency could also be higher during high processing load. This was not explored by the group but would have been nice to test in hindsight. The group did use a powerful computer, and during recordings, all other applications other than those necessary for the blink recordings were closed. It should therefore not have been too different processing load during the different recordings. Note that things like CPU load, RAM usage, and other things were not controlled, and it is therefore not possible to guarantee the same load during the different recordings. The other components, like the LiveAmp, EEG and VR headset, should have been under the same constant load, due to being handled and set up the same way for each recording. If the load affects the synchronization, it might explain the varying jitter in the blinking dataset (figure 43). The group can however not guarantee this.

6.2.2 Blinks as a Common Marker

The group decided to base the synchronization on the blink artifact as explained in Synchronizing Based on Blinking Artifact.

The equipment could also have been synchronized utilizing another common marker, such as TTL (theory 2.5). This could have been done by manually adding an integer value to both streams at the same time. The group could then have used this marker to find the offset in the analysis stage. However, it would not be applicable for this situation, because the group would have needed to access the hardware of the eye tracker directly. This would have been a time-consuming task, and the group had no idea if it was actually possible. The actual hardware delay between the EEG and eye tracker would be interesting, but not applicable in a real-world scenario. Extra latency could be created from additional hardware, additional software, or by utilizing different information transfer techniques. There is no guarantee that these latencies are the same for the EEG and eye tracker, due to using different solutions before arriving in the same NeuroPype pipeline. Therefore the group opted to try to synchronize the equipment using eye blinks as a common marker.

6.2.3 Phase Distortion

When taking phase distortion into consideration, it is clear that figure 42 and 27 are not agreeing on the offset between the EEG and the eye tracker. Figure 42 which is not filtered using zero-phase Butterworth, shows the peak more delayed than in figure 27. This can also be seen in table 4 and 6, where the computed offset is 101 ms and 61-62 ms respectively. The zero-phase filter filters the signal twice: left to right, then backward, which compensates for any initial phase distortion (theory 2.2.1). When streaming data live and applying zero-phase filters, it is necessary to split the stream into chunks. This is because the filter needs to know the end of the data stream to perform the second filtering stage (theory 2.2.1). Therefore zero-phase filtering is not supported while streaming data in NeuroPype. This among other things resulted in the logic originally implemented in NeuroPype being slowly phased out in favor of a custom-made solution using Python.

6.2.4 LSL

The protocol Lab Streaming Layer (LSL) was used to stream the data between the computer recording the EEG and the computer running the rest of the applications (NeuroPype, Unity, etc). The group utilized one of the features of LSL which allows for streams to be consistent with predetermined data rate. This made it possible to assume that the two data streams had about the same offset relative to each other over different recordings. Utilizing LSL is a lot easier when conducting this type of research. Instead of having physical cables connected to transfer the different data streams, the group is able to simply stream them via the local network. This made it easier to set up and modify the pipelines and streams. Another benefit of utilizing LSL is their extensive and descriptive documentation, and popularity, resulting in many forums discussing relevant matters.

When synchronizing the two streams using LSL, the group chose to compare them

by index. This was done because LSL does not guarantee synchronization between different data streams, but rather leaves that up to the one implementing LSL in their software solution, as stated in their docs (*LabStreamingLayer submodules · GitHub* 2023). They also inform that a stream is marked with a timestamp based on the clock of the computer which generates the stream. Due to the fact that two computer clocks are not guaranteed to be exactly in sync (Levine 2020), two LSL streams generated on two different computers are not guaranteed to have exactly the same timestamps even if they were generated at the same point in time. But say the computer clocks are synchronized. There is still the issue of process scheduling. The processor has many tasks to complete, and it is not impossible that setting a timestamp on one computer is scheduled differently than setting a timestamp on the other computer, potentially resulting in different timestamps. Therefore the streams are compared by samples, instead of by the time stamp. Note that this approach requires the streams to be constant, and can not guarantee the same offset from session to session. This however seemed to be fine when doing the blink recordings as can be seen by the low STD in the results (table 6).

6.2.5 Fuse Stream Node

Early implementation of the NeuroPype pipeline utilized for the synchronization part of the project used a node called 'FuseStreams'. The idea was that fusing the streams from the EEG and eye tracker with this node would help with the synchronization by NeuroPype fusing the streams in a smart way and also to have all the data gathered in one file. Later, when reading the description of the 'FuseStreams' node, the group noticed that it was in beta at the time of doing the project, which made them question the usage of 'FuseStreams'.

Inspecting the data showed that the data from each stream had been put in the same file, as expected, but the time stamps did not align well with either of the original data when they were separate. This suggests the usage of some sort of algorithm that manipulates the timestamps of both data streams instead of aligning one of the streams to the other with some constant time shift. For applications where the timestamp does not matter, this is fine, but if someone else would like to utilize this setup in the future and needs to retain as much information about the timestamps as possible, FuseStreams should not be utilized. For that reason, the group omitted the node from the pipeline. When trying to reproduce this data to include it in the report, the fuse stream node was not working (NeuroPype 2022.0.1). This is another reason to not use this experimental beta node.

6.2.6 Different Stream Length of Recordings

After removing the fuse stream node from the pipeline, the different data streams were separate, and needed to be aligned before synchronization based on eye blinks could be done. NeuroPype did however save the different data streams with different lengths. The streams had different lengths from 0 to 11 samples at most. This meant that the stream could have an offset from 0 ms to $11 \cdot 4ms = 44ms$, already before synchronizing. Because NeuroPype saves these streams with different lengths, it is hard to know whether the streams begin at different times, stop at different times, or a combination of both. The reason this is happening is that the streams are

updated every 4th millisecond, which might add samples to the second stream, while NeuroPype has stopped and is writing the first stream to a file. This might happen if NeuroPype runs different threads for different streams, or if it follows a sequential algorithm, where the streams stop recording samples at different times. This is crucial to know for the project because the data sets are compared based on indexes.

To test if the streams were cut short at the start of recording, or at the end of recording, the difference of length between the EEG stream and eye tracking stream was found. Then the Standard Deviation (STD) and Standard Error (SE) of the data set were found. This was done both when the excess samples of the stream were cut at the end and at the start. This resulted in a high STD for both streams (table 6). This suggests that the stream was not only cut short at the start of recording, nor only at the end. Note that there wasn't consistency in which of the streams had missing samples, it could either be the EEG or the eye tracking stream that had fewer samples. This varied from recording to recording.

There is however a possibility that one of the streams was written to file later than the other, and finished writing before. This might cause a loss of samples at the start of the stream and at the end of the stream. This can be seen in figure 44, where the blue streams represent different variations of samples missing (start, end, or both). To test for this, the shift was set to $\frac{1}{2}$, $\frac{1}{3}$, and $\frac{1}{4}$. As table 6 shows, the lowest STD was achieved when half of the offset was designated to initial write delay, and the rest of the excessive samples were cut off at the end of the recording.

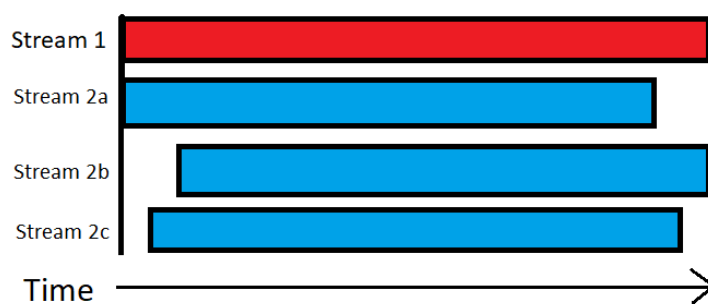


Figure 44: How the LSL streams could have been saved compared to each other. The red stream is the stream with all samples, while the blue stream represents how the original stream could have looked compared to the red stream. Stream 2a represents the loss of samples when ending recording, Stream 2b represents sample loss when starting the stream, and Stream 2c represents a mix of both. This is important to address because indexes were used to synchronize.

The group is aware that the different lengths of streams are a flaw in the research work done, due to introducing unnecessary uncertainties to the project. If it was able to synchronize the equipment with less than a millisecond offset, it could have been disastrous. However, since there already are some limitations introduced due to the equipment's sampling rate, there wouldn't be too accurate synchronization, not that it doesn't mean this should have been avoided, but rather it wouldn't have that much of an impact on the actual result. This can be seen when calculating offset between the EEG and eye tracker when using the cutting techniques of not only cutting the start/end (table 6). This could have been because not all the streams used in the analysis had different lengths, and since there were many recordings, this might have evened out.

Figure 43 shows the blinking data plotted in a histogram, when cutting the excess samples are half cut at the start, and half at the end. The line with the dots shows the normal distribution of the blinking offsets computed. The majority of the offsets that were found were between 15 milliseconds in front of and behind the predicted target. This implies that a blink is most likely to happen between -2 and +2 sample points away from the actual target.

The final offset found from the preferred cutting method of the streams was an offset of around 62 ms (table 6). The STD/jitter is however more interesting since it shows how much the data varies. A jitter of 8.87 ms is close to a jitter of 1 sample, due to the sampling rate being 8.33 ms. That makes for a difference of approximately 6%, making the group satisfied with the result, especially taking the limitations and other problems, like the different stream lengths into consideration.

6.2.7 After Shift

When referring to the data after shift, it means that the data points had been moved to align based on the computed offset found in table 6. The found offset was 62 ms, which is the same as 15.5 indexes when using a sampling rate of 250 Hz. This is however not possible to shift, therefore the EEG data was shifted 15 indexes (60 ms). The data in table 7, is the same data as in table 5, except the data is shifted 15 indexes. The data shown in table 7 shows how many indexes each blink is off compared to the computed offset. When looking at the average indexes column, the blinks are in the range of 5 indexes too fast, and at most 4 indexes delayed. This is a great result, considering the eye tracker only sampled about every other index. This means that if the average index off is 2 indexes, the actual offset is a single sample, due to the recorded sampling rate of 250 Hz, and the eye tracking sampling rate of 120 Hz. This was done as explained earlier to easier compare the EEG and eye tracker data streams.

When looking at table 8 the STD and SE stay the same compared to the results before the shift, which indicated the shifting of the data is implemented correctly. The average time offset is still not 0 milliseconds, due to the sampling rate limitations described prior.

6.2.8 Data Production Methodology

The method for creating markers in the data has not been consistent throughout the project. Considering the EMG was only put on the left eye of the member used for producing data, early production was based on blinking only with the left eye. The subject noted that more muscles around the left eye were recruited, and they were contracted at a higher intensity when blinking only with the left eye compared to when blinking with both. A paper (Mercer et al. 2006) shows that higher muscle contraction results in increased EMG signals, and recruiting more muscles would certainly also increase the signals. The subject also noted that they found blinking unilaterally to be a harder cognitive task, than just blinking bilaterally.

This lead the group to wonder whether these factors could have an impact on the synchronization. This was also supported by the difference in offset found when comparing the two methods (table 2 and 6). The final work on synchronization was therefore

based on unilateral blinking data.

It might also be a synchronization flaw to only use a single subject. It would have been interesting to compare the offsets of various subjects and discover any variations. It might not matter because biological variations may not have an impact on manual blinking, but it would still be great to have ruled out any variation.

As mentioned in Eye Blinking Signatures, natural blinking speeds become slower when fatigued. Even though the member producing the blinking data did so by blinking manually, fatigue could affect these kinds of blinks too. Especially the later blink recordings, when the member might've been more tired than at the start, could be subject to this phenomenon. This could partly be counteracted by the usage of the metronome. Having a reference for when to blink could help keep blink speed and consistency up, because of the urgency of having to blink at a set interval. The group did not find any trend in the data indicating that fatigue had a large impact.

6.3 BCI Speller

The following section discusses the implementation and performance of the BCI speller.

6.3.1 Implementation

This subsection discusses the different aspects of the implementation of the speller.

Deviations from sources of inspiration This speller implementation took heavy inspiration from Mannan et al. (Mannan et al. 2020) and Du et al. (Du et al. 2019) but differs in some fundamental ways. Mannan et al. chose which frequencies to use for their speller based on the refresh rate of the monitor they used for testing. The frequencies used for the speller created by the group were originally calculated with the same approach as Mannan et al. but were deemed not very usable during development. While testing the speller with the frequencies calculated similarly to Mannan et al., its accuracy was very low compared to the frequencies which were used in the testing state. Note that these 'working' frequencies were based on Mannan et al. mentioning that Visually Evoked Potential (VEP) were very prominent in stimuli with low frequency (between 4-12 Hz) and some heuristics. They ended up being 4 Hz, 5 Hz, 5.5 Hz, 6 Hz, 7 Hz and 7.4 Hz, which is on the lower end of the spectrum mentioned by Mannan et al. but were found to work the best out of the frequencies that were tested. Some of the frequencies that were chosen try to take the CCA into consideration, avoiding being multiples of one another. This is the reason why the highest frequency is 7.4 Hz and not 7.5 Hz, because $7.5 \times 2 = 15$ and $5 \times 3 = 15$. Though not all frequencies are that carefully selected, such as 4 Hz and 6 Hz because $4 \times 3 = 12$ and $6 \times 2 = 12$, meaning they could increase each other's covariance factor. To combat this, the harmonic frequencies are added to the Y data set in the CCA as described in equation 6.

Since the pipeline doing the CCA is working with lower frequencies, there might be something making Unity not being able to display the higher frequencies. This is a bit unfortunate, due to higher frequencies being easier to distinguish in the visual cortex when utilizing Steady-State Visually Evoked Potential (SSVEP). This is due to

fewer similar frequencies being signaled. Higher frequencies also require a smaller flickering time window, due to the wave being more frequent than lower frequencies (theory 2.7.4).

The string used in testing, 'I LIVE IN NORWAY.', is similar to what Mannan et al. made their participants spell ('I LIVE IN KOREA'), but tweaked based on regional differences for where the testing was conducted.

The other inspiration source was Du et al., who created a regime in 3D where the letters of their speller were arranged in different ways to see if such a stimulus variant could be better for ERP-spellers. The group originally planned to implement the SACFO (Same depth information Array Character Fly Out) paradigm, where only the letter flew out toward the user, due to its high precision as mentioned in 4.7. But due to limitations with the implementation of how the eye tracker interacts with GameObjects in Unity, making the exact character a user looks at fly out proved difficult. The middle ground found by the group was to make entire clusters fly out when a letter from it was looked at, hopefully preserving the precision of the SACFO paradigm.

CCA The initial speller was intended to be implemented using NeuroPype solely. This was however hard because of the lack of documentation which caused a lot of frustration. When the speller wasn't working due to an unknown source of error, the group had to investigate and identify the cause of this error. This error and lack of documentation made the group question if the NeuroPype pipeline was correctly implemented. Therefore much of the functionality was moved away from NeuroPype into a self-written Python script. The group was more familiar with Python, and therefore it was easier to ensure correct implementation, due to the large amount of documentation and support available online.

The reason Python was used, instead of C# which was already used in Unity, was that Python supports data analysis with a lot of easy-to-use frameworks created for analyzing, plotting, and altering data. NeuroPype is also created in Python, therefore all functionality present in NeuroPype should be recreatable in Python (*Neuropype by Intheon 2023*).

The script received the data streams from the EEG and the eye tracker. When the eye tracker notices the subject looking at a letter-cube, a signal was sent to start analyzing the data. This script was then split into two different scripts. The first script used the filtered signals from NeuroPype to do the CCA, while the other script did this filtering on its own.

Two scripts for the CCA were created. One used the filtered data from NeuroPype and another where the filter process was done in the script itself, utilizing zero-phase filtering. The result of the CCA was then streamed through LSL, so other applications could receive the predicted frequency. Both streams were running at the same time and showed similar results. In hindsight, the group recognizes that all the CCA results should also have been saved, so they could have been used for comparison and later analysis.

Equipping strategy As described in BCI Speller Testing, the participants held the VR headset up to their faces. This might influence how the participants interacted with the speller, as how the participants held it may have varied. Some might have

pressed it more loosely against their face, making the eye tracking less accurate since the actual trackers would be further from the eyes than if it had been pressed harder against their face. This approach could lead to shifting of the eye tracker in relation to the eyes because of small movements caused by the subjects not holding the headset at the exact same position at all times, introducing a possibility for worsened eye tracking as mentioned in Eye Tracker Prerequisites and Calibration. The potentially worsened eye tracking would luckily not impact the usage of the speller in any considerable way, but it could lead to the interaction with the clusters being more frustrating. During development, such a scenario was experienced where the cluster was jumping sporadically because of some unintended behavior of the gaze ray. This was produced by the eye tracker randomly colliding with the cluster, especially at its edge. The headset might also hit the EEG when the subjects were looking around, and this might mess with the signal. The subjects were also not allowed to speak during the testing to ensure as little artifacts from muscle movements in the face affecting the EEG as possible.

6.3.2 Performance

Each subject was tested twice, initially to test if the speller was working, and subsequently to test if training would improve the accuracy. The initial round had some promising results, while the second round of testing included some errors which will be discussed.

Initial testing The accuracies measured for each participant, as seen in table 11, varied both for each flickering period and in between the participants. Comparing subjects 1 and 4 might be of interest, as they scored the highest and lowest overall. There are many potential reasons for this. For example, subject 4 had very thick hair growth which may have led to the dry EEG not getting as good contact as it may have needed to gather and send accurate, clean data. It may also have been an error while putting the EEG cap on, leading to the data not being as good as it should be. As for subject 1, this individual was used for testing during the development of the speller. For that reason, subject 1 has a lot of training in using the speller, which may have led to the high accuracy. Another reason is that since subject 1 was the one testing the speller during development, there is a possibility that the setup is biased towards that individual since it essentially was made to fit them. That was one of the dangers of only testing for one person while creating the speller. But the results of subject 7 suggests that this might not be the case entirely, considering that individual got relatively high accuracies for most of the tests. It could also imply that the speller works better for some individuals than others. Though the exact reason why is unknown, it could for example be that SSVEP can take longer time to generate in some individuals than others (theory 2.7.1).

As a whole, the speller tends to be most accurate the more data it has to calculate the covariance between the flickering happening in Unity and the registered potentials in the brain, which is also seen in table 11. The bottom row, which contains averages for all subjects' accuracies for the different flickering periods, shows that 6 seconds of flickering has the highest accuracy with a steady decline down to 2 seconds. This finding is consistent with results produced by similar studies (Lin et al. 2006; Mannan et al. 2020).

Something to note about the speller setup is the symbols, and their associated frequencies, that the chosen string 'I LIVE IN NORWAY.' results in. As seen in table 22, there are more symbols associated with lower flickering frequency, almost 1/2 of the total symbols, than the ones with higher frequency. The speller setup used for testing always had lower flickering frequencies at the top left and higher frequencies in the following letter-cubes, which differs from how (Mannan et al. 2020) created theirs. They had spread their chosen frequencies differently through all their clusters. If the created speller variant had done something similar, it would probably have spread the frequencies more evenly through the keyboard which would then avoid 1/2 of the total symbols in the test string being of lower frequencies.

Retesting The second round of testing was initially planned to see if the subjects would get higher accuracy with training in using the speller. There was however a little change in the code before doing the retesting. In the initial testing, the time shift was set to delay the eye tracker by 160 ms as mentioned. Before the second round of testing, one of the group members thought this was incorrectly implemented, and implemented a shift where the eye tracker was delayed 60 ms, and then the eye tracker was expedited 100 ms. This is incorrect, and instead of synchronizing the eye tracker, it introduces an offset of 100 ms from the EEG and the eye tracker, then an additional 100 ms, if the ocular delay is taken into consideration.

To ensure that the code change didn't completely ruin the speller, the script using the NeuroPype to filter the data was running. The two scripts gave similar results, and the group members were not alerted by a big difference between the two scripts, therefore it is hard to believe this was the reason for the different results. The group members acknowledge however that this was a big mistake and an introduction to an unnecessary source of error. Even if the initial implementation was incorrect, the changes should not have been introduced. Changing multiple variables between each test round makes it uncertain which change(s) affected the result.

The retesting introduced strange results in the speller accuracy (table 11 and 14). The overall accuracy of flickering lengths of 4, 3, and 2 seconds dropped by some percentages (7.5%, 7.5%, and 5.0%). This drop in performance could be explained by the introduction of incorrect synchronization. This resulted in some misleading data being added to the SSVEP signals, and therefore the accuracy naturally fell a bit. This can however also be seen as something that suggests the synchronization offset computed was reasonable, due to the loss in accuracy when changing this.

The percentage drops in the 6-, and 5-second flickering (34.4%, and 32.7%) can't be explained by the incorrect synchronization. This amount of accuracy drop should not have been possible if the only difference between the initial test, and the retesting were the added offset between the EEG and eye tracker. This is also supported by the fact that table 9 shows that CCA is not phase sensitive, therefore the introduced offset shouldn't influence the results with longer flickering, due to less percentage-wise bad data in the entire data set.

Another anomaly with the results is that subject 1 kept a high accuracy, while the other subjects' accuracy drastically fell. Subject 1 was the subject with the most experience, and shows that the speller is working, even with the code change. The result in 2 seconds was however worse than in the initial testing, but this can be explained by the introduced offset from the code change, or simply be random.

This was however not the case when looking at the rest of the results. There are some inconsistencies in the results, like subject 5 having a 10 percent accuracy when the flickering was at 6 and 5 seconds, then jumping up to 80 percent, when the flickering is at 4 seconds. Subject 6 went from an accuracy of around 50-60% from 6 to 3 seconds, for then in the second test, never achieving anything higher than 18% in the same intervals. Then, when presented with the two seconds flickering, the accuracy jumped to around 30%. Due to this inconsistency, the group believes there was noise or other sources of disturbances that affected the EEG signals negatively.

When equipping subject 6 with the EEG, the group also noticed a lot of noise in the EEG signal. When unplugging the charger of a neighboring group's laptop, this noise disappeared. This is just a single example of noise that could have been created by other groups using the lab. The group doesn't know whether the charger was plugged in during any of the tests, but it suggests that there might have been some power noise during testing. Some of the subjects also reported finding it hard to concentrate this time, due to disturbances from other users of the lab or them generally being fatigued for various reasons. The testing station was also exposed to people entering and leaving the lab, which might cause some distractions for the subjects when testing the speller.

The group also had some issues with the LiveAmp equipment, where the LiveAmps sometimes needed to be replaced because of showing noise, then when restarting it, the noise was gone. Since the group is confident in the scripts analyzing the signal and doing the CCA is working, this might be the source of the bad results. If the EEG signal is noisy when analyzing, it will result in low accuracy. Even though the EEG signals were shown to be good when equipping the EEG, the LiveAmps acted weirdly as described. The group also spent much less time getting a good signal from each subject than in the initial testing. This could be because the group was getting better at equipping the EEG, or it might be because the LiveAmps did not behave as intended and the software used to visually control if the signal was ok, indicated good signals even though they were not.

It would have been interesting to test if the use of the wet EEG could have limited the hypothetical noise, where the EEG signals would have been stronger compared to the dry EEG (theory 2.1.2). A better signal from the EEG could make the CCA give more accurate results, ultimately making the spelling solution more effective.

Between the two testing sessions, the equipment used was moved from the lab, for the group to participate in a stand, where the group displayed their work and held a live demo. This required the entire setup, everything from the computers being used, VR headset, and EEG to be moved. The equipment was returned later the same day. It is unlikely that this should affect the speller, but it is worth mentioning, as the hardware setup used in the first testing session, might not be completely the same as in the retesting session.

The group would also like to address that using the CCA algorithm to show how well the equipment is synchronized, is not the most suited technique. This is due to the algorithm not being phase sensitive. This can result in a false sense of security that the equipment is synchronized, because even poorly synchronized equipment, such as during the retest, could give somewhat good results. One can not conclude that the found offset is correct based on the results from the initial test round, but it could indicate that it is more correct than the offset in the retest.

6.4 NeuroPype

At the start of the project, the group was encouraged to use a pipeline designer to handle data streams and processing, to focus more on the research part of the synchronization, than the implementation of software, like support for LSL streams, etc. The group then chose NeuroPype as explained in the Method chapter. Even though NeuroPype has a lot of nodes and functionality, it was hard to fully utilize this because of the lack of documentation and forums answering questions. This caused a lot of frustration, and without support for zero-phase filtering online, the group slowly moved functionality out of NeuroPype. It started with implementing a simple Butterworth filter in Python, then continued to implement the CCA, and the correlating frequencies. NeuroPype has support for all of this functionality, but it was impossible to implement without support from them directly due to a mismatch of axes like frequency/time and space/time couldn't be compared in a CCA node.

NeuroPype is very good at handling the LSL data stream and was mostly used to dejitter the signals, re-sample, and store them to file.

6.5 Research Questions

Answers to the different research questions including a short summary of each are presented here.

6.5.1 RQ1 - Is it possible to synchronize the given eye tracker and EEG with less than one-millisecond offset?

The initial task of the bachelor project was to synchronize the EEG with the eye tracker. These were intended to be synchronized with an offset of less than a single millisecond. The equipment that was available was an EEG with a sampling rate of up to 1000 Hz. This means that there is

$$\frac{1}{1000}s = 1 \text{ ms}$$

between each data point. The eye tracker, however, had only a sampling rate of 120Hz. This is equivalent to

$$\frac{1}{120}s = 8.33 \text{ ms}$$

between each sampling point.

Due to these limitations, the best achievable synchronization offset between the EEG and eye tracker is approximately 8.33 ms. Figures 6 and 8 illustrate that the lowest Standard Deviation (STD) observed was 8.87 ms. The reason this is not quite 8.33 ms, could be factors like network lag during LSL streams, or other jitter caused by hardware, software or both.

It is not possible to achieve synchronization of less than 1 ms between the eye tracker and the EEG, mainly because of the limitations imposed by the eye tracker's sampling rate.

6.5.2 RQ2 - Is it possible to create a BCI Speller combining a synchronized EEG and Eye Tracking in a 3D environment?

In order to demonstrate the successful synchronization of the eye tracker and the EEG, the group chose to develop a BCI speller as a Proof of concept (POC). This speller was designed as a hybrid SSVEP-based speller, utilizing a 3D environment and Canonical Correlation Analysis (CCA) for feature extraction. Due to the limitations mentioned about the synchronization, it would be impossible to implement a BCI system, or feature extraction method that needs a more precise synchronization to work properly. CCA was chosen as the feature extraction method, due to the lack of sensitivity to phase distortion.

Despite the small sample size of 7 participants, initial testing (as seen in table 10 and 11) yielded promising results. The tests indicate that the speller gave satisfactory results and that the synchronization was implemented correctly.

It should be noted that the speller is not the fastest BCI speller available. The speller gave results of almost 80 % at six seconds of flickering, and this accuracy gradually declined with lowering the length of the flickering.

In summary, the initial testing of the BCI speller demonstrates that the synchronization between the eye tracker and the EEG was sufficient to create a BCI speller in a 3D environment.

6.5.3 RQ3 - Does practice with the BCI Speller lead to improved accuracy?

Subject 1 had the highest accuracy, with almost 100% accuracy across all different flickering lengths (table 22). This subject was also the subject used for testing when implementing the speller. This meant that the subject had gotten a lot of training in using the speller, and it might suggest that the results from the initial speller test would improve if the subjects were to use the speller more. To test this hypothesis, each subject was scheduled for a retest three days after the first test.

Unfortunately, due to an error made by one of the group members, an offset of at least 200 ms was introduced between the eye tracker and the EEG, between the test rounds. This would most likely impact the lower flickering lengths, rather than the longer flickering. The retest results were inconsistent, with a subject having 10% accuracy for flickering lengths of 5 and 6 seconds, but then suddenly achieving 80% accuracy when the flickering length was reduced to 4 seconds.

Another aspect to consider is the amount of training each participant got. As seen in (Kliegl et al. 1989), the participants had the opportunity to practice a lot. For this experiment, the participants only tried to spell one sentence a total of 10 times on two different days. To potentially increase the effect of training, the participants should have been called in more frequently and have had the opportunity to get more spelling practice. Another benefit of this would be a larger data pool with more representative results, which could counteract the effect of exterior variables.

Given the offset mistake and the inconsistency in the data due to unknown error sources, it is impossible to definitively conclude that subjects achieve higher accuracy with more speller usage. Likewise, it is also impossible to determine if their performance worsens, based on the results (table 14 and 13).

6.6 Teamwork

During the course of the project, the group consisting of Larsen and Tresselt demonstrated great teamwork and collaboration. The division of tasks was well-balanced, with each team member contributing their unique skill set to the project. The good distribution of tasks is also reflected in the timesheet, appendix G, where both members have worked almost the exact same amount of hours.

Tresselt had prior knowledge about Unity and VR and therefore the eye tracker and Unity were made into his primary responsibility. This made it possible for him to use his knowledge to develop the core functionality of both the eye-tracking application, and the BCI speller. Tresselt also got a basic understanding of NeuroPype and LSL quickly, and he was therefore able to set up the different pipelines. He ensured that the Unity programs were efficient and worked as intended.

Larsen focused on the EEG and the principles of signal handling and filtering. This included reading a lot of similar research and theory, especially for signal filtering considering he had no prior knowledge on the subject. He also wrote the Python scripts used for analyzing blinks, as well as the scripts used to run the CCA algorithm during the BCI speller. Larsen also created most of the graphs used in the report, utilizing his Python knowledge.

In addition to their individual responsibilities, both team members actively read up on relevant theory. They shared their findings to enhance the project's scope and depth. Their collaborative efforts also extended to the preparation of the main report, where they combined their insights. Despite the division of tasks, Larsen and Tresselt devoted equal amounts of time and maintained a consistent work ethic throughout the project. There were no problems spending the weekend at the lab or spending some time in the afternoon doing work related to the project.

This resulted in the team accomplishing most of the outcome and process goals set at the start of the project in the collaboration agreement, which can be found in the appendix to the pre-project plan (appendix E). The group managed to collaborate and communicate efficiently and increased their knowledge of EEG, eye tracking, VR, and scientific methods. They also managed to deliver the obligatory tasks on time, but won't know their grade until after the thesis is delivered. Both members put in at least 500 hours each, and the project was completed more than one week before the deadline. The final report is approved by the group to get published, and the group hopes the findings of the project can lead be used as a baseline for further research. The group did utilize scientific methodology.

6.7 Engineering Results

The goals and difficulties regarding the development of the BCI speller presented in section 5.2 are discussed in greater length in this section.

6.7.1 Discussion Related to Goals

Here, the status of each functional and non-functional attribute defined in appendix F will be discussed. This section contains goals that might already have been discussed,

but they are added anyways to ensure consistency, but not discussed in great detail.

Synchronized equipment The first functional attribute defined in the vision document was that the EEG and eye tracker should be synchronized within a 1 ms margin of error. This lines up with the first research question and due to reasons discussed earlier in section 6.5.1, the theoretically best possible synchronization would be within 8.33 ms, which means that this goal could not be met.

Working speller The second functional attribute was to use the synchronized equipment to create a working BCI speller. This aligns with the second research question and was discussed earlier in section 6.5.2, where the group argues for a satisfactory implementation of the speller.

Little to no discrimination The following functional attribute was for the speller to work with multiple individuals. Based on the observations from the speller testing, the results could indicate it works with multiple individuals. As seen in e.g. appendix C and D, especially subject 1 and subject 7 showed a relatively high accuracy across the board, indicating that they are able to use it relatively efficiently and the goal is met. But the results of the other subjects are not as consistent, suggesting that the speller is not compatible with the majority of individuals and does in fact discriminate to some degree.

Accuracy of acceptable degree The last functional attribute was for the speller to be at least 80% accurate, which table 11 and 14 shows was not the average case. Even though some of the test participants did manage to get accuracies of over 80% for some flickering periods which can be seen in appendix C and D, others did not. The attribute is therefore partly fulfilled when looking at individual cases, but not when looking at averages.

Follow standard scientific conventions While writing the report, some conventions were also followed. These include presenting accumulated data in tables and graphs and basing conclusions on this instead of what the members thought and felt, all the while writing in a non-personal manner. When working on all the other parts of the project, the group was careful to base their decisions on prior reputable research and thoroughly thought out reflections to a large degree. This is both for the theory presented and methods used in development and testing. But, some slip-ups are inevitable, and during testing of the speller, some parts of the program running the CCA were changed. This brought with it some consequences which have been discussed earlier in Retesting. There is also the matter of the creation of the speller. It was discovered that the speller did not work as intended after it was initially created, and the group then systematically discovered potential reasons as to why. But implementing the fixes was not as systematic. Up to a couple of them were deployed at once, meaning isolating the actual cause was less likely to happen. But when finally discovering the root cause, it was clear that it was largely due to that reason, and the subpar fix implementation methodology did not have large repercussions.

Maintainable and readable code The second non-functional attribute was for the group to produce maintainable and readable code, which the group has been conscious of throughout the project. Each member of the group regularly commented their code when writing it. This was intensified at the final part of development, the members went through all the code they had produced and wrote comments for the parts they considered most relevant. They also refactored the names of variables, functions, file names and Unity GameObjects to make each part of the project as clear and self-explanatory as possible. But it could have been done better. None of the two GitLab repos created for the project have a README, but adding well-formulated and descriptive ones could have made the repositories easier to understand.

Usability Because the speller has a user interface, it should take usability into consideration. Though it is not web content, it will be based on WCAG 2.1 Perceivable considering it is important to follow some usability guidelines when creating interfaces. Because it in essence is a keyboard, the letter-cubes should be visible on the background and the letter associated with each letter-cube should be visible. The RGB value of the background behind the cubes and of the letters are the same gray color, RGB(82,82,82), and the cubes are as white as Unity allows, RGB(255,255,255), resulting in a contrast ratio of 7.81 between the background and the cubes and the same ratio between the cubes and the letters. Also, the string that shows what the user has spelled has the same RGB value as the cubes and is presented on the gray background. This results in a ratio of 7.81 and is sufficient for Level AAA of criteria 1.4.6 Contrast (Enhanced) and Level AA of criteria 1.4.11 Non-text Contrast (*Web Content Accessibility Guidelines (WCAG) 2.1 2023*), which were considered essential to meet by the group. But when the cubes are flickering, they change color to black, RGB(0,0,0), resulting in all mentioned contrast ratios, except for the result string on the background, being 2.69. This value does not follow WCAG standards but is not the default state and the change from black to white and back is rapid, leading the low ratio to be less relevant. And there is also the matter of the cubes changing color from white to black, which in itself has a contrast ratio of 21. Another criterium met under Distinguishable was 1.4.1 Use of Color, since the speller does not use color as the only visual means of conveying information, indicating an action, prompting a response, or distinguishing a visual element.

Other criteria were not met, though. Due to the speller not being customizable when the application runs, it is not very adaptable and the criteria under Adaptable that apply to the speller is therefore not achieved. The speller relies on presentation and instructions from the group to be used and is not customizable or explained in text and a user has to move around to fully utilize its capabilities, which does not allow for it to follow 1.3.1 Info and Relationships, 1.3.3 Sensory Characteristics and 1.3.4 Orientation (*Web Content Accessibility Guidelines (WCAG) 2.1 2023*). The other criteria under Adaptable were not as relevant. Due to the rigid nature of the speller, it did not fulfill the Distinguishable 1.4.4 Resize text and 1.4.8 Visual Presentation which both are related to deciding characteristics for the text presented in the application (*Web Content Accessibility Guidelines (WCAG) 2.1 2023*).

The group was unsure regarding criteria 1.4.12 Text Spacing (*Web Content Accessibility Guidelines (WCAG) 2.1 2023*), because they used the default values for the Text-mesh Pro text in Unity except for the color and did not pay mind to the letter or word spacing, but assumed it had considered usability in its default mode. They did not

find any information about whether the creators of the text package had considered usability, which makes it uncertain whether the mentioned criterium is fulfilled.

Many of the WCAG Perceivable criteria are created for individuals who are visually impaired, but the speller requires the user to have functioning eyesight. For this reason, some criteria are not as relevant as others. The criteria that are not mentioned were deemed irrelevant, considering they were not applicable to the created interface (*Web Content Accessibility Guidelines (WCAG) 2.1 2023*).

The aspect of usability when designing an interface is important and should in general not be down-prioritized, but for this project, the focus was on figuring out whether or not the synchronized EEG and eye tracker could have applications. Had this project only been to create a BCI speller, the group would have prioritized this goal to a larger degree. Ultimately, quite a lot of what the group considered the most important usability criteria were followed.

The speller should be robust A program should be intuitive and not be prone to bugs or crashes to give a good user experience, which is why the group wanted to focus on the speller being robust. The speller interface is in essence a keyboard, and there were some alternatives for the layout. The two main ones considered were to simply put each letter in alphabetical order, such as Mannan et al. (Mannan et al. 2020), or a QWERTY layout with some modifications. But due to QWERTY being an ISO standard (ISO 2009; Noyes 1983), it was chosen. The modifications done to the layout were trying to use symbols that would be associated with the function, that being '<' with a single dash for backspace and '_' for space as seen in the bottom right of figure 38, even though these are not standard symbols or had standard placements. These were inspired by Mannan et al., who used '<' with two dashes for backspace, which the group almost used but would be too large to fit inside the letter-cube. Mannan et al. used a blank square to indicate space, but the group feared that an empty letter-cube might cause confusion and decided to put '_' instead.

The flickering of the letter-cubes was also important to get right. Throughout development, these have had different behavior such as continuing to flicker after being looked away from or clusters not being able to be switched between. The final version does not have such bugs and works consistently as expected. E.g. when a cluster is looked at, it is flown out and then shortly after starts flickering and when it is looked away from, it stops flickering (if it is not already finished flickering) and another cluster is able to be looked at.

The application had to work well behind the scenes, too. For this reason, the Python script running the CCA takes into consideration if a flickering period is valid. When a cluster starts flickering, it sends non-zero data via LSL which the mentioned python script picks up. Only if the non-zero data is registered over a period that matches the flickering period, is it considered valid. This means that if a user looks at a cluster and it flickers for a total of 2 seconds while the flickering period is set to 5 seconds, this is not valid and no result from the CCA is pushed to the Unity application. But if the user looks for the full 5 seconds, the Python script returns a result that the Unity application then handles appropriately. The only downside of the current implementation regarding this validity check is that the flickering period has to be set in both Unity and Python, making the system less dynamic. Ideally, the period is set in Unity and sent via LSL or set in some kind of environment variable where both the

programs could read what period to use.

The speller should be relatively efficient The last non-functional attribute was meant to target the effectiveness of the speller and was left pretty vague giving it some flexibility.

It would be inefficient for the speller to take too long even though it might be accurate and it would be inefficient for it to be fast but not be accurate, after all. To get an idea of how efficient it turned out, one may take a look at the probability of different cases for spelling symbols with different flickering periods and compare them. Note, that this is not a standard way to assess a speller's efficiency. A more common approach is to use Information Transfer Rate (ITR), which will be discussed later. During testing, the flickering periods used ranged from 6 to 2 seconds. Some of the participants reported that the longer flickering periods, especially 6 and 5, were slow. Though this was the case, the speller was most accurate at these periods, especially for the first test round as seen in table 11.

The following thought experiment uses the accuracies found in the table 11 for the 6-second flickering period. Consider the probability of spelling at least one correct symbol within the longest time possible, which is 7.5 seconds. First, a user has to look at a cluster and wait 1 second for it to start flickering. The longest flickering time is 6 seconds. After the current total of 7 seconds has passed, assume that it takes another 0.5 seconds for the CCA to finish and for the symbol to appear in Unity, adding up to 7.5 seconds. Therefore the probability of the speller giving at least one correct symbol for the longest period is

$$P(\text{At least one correct})_{6s \text{ flicker}} = 0.7983$$

But with the shortest flickering period, one might fit 2 spelled symbols. Looking at a cluster will make it flicker after 1 second, it will flicker for 2 seconds and the result from the CCA uses 0.5 seconds to Unity as before, which adds up to 3.5 seconds. Assume a user looks at the next cluster containing their next symbol immediately. The entire process can happen again, and 2 symbols are spelled in 7 seconds, which is within the longest period of 7.5 seconds. Because there are 2 tries, and there is an ordered sampling with replacement, the probability of getting at least one correct symbol is

$$P(\text{At least one correct})_{2s \text{ flicker}} = 1 - (0.5294 * 0.5294) \approx 1 - 0.2803 = 0.7197$$

where 0.5294 is the probability of getting a wrong symbol for the 2-second period. Looking at the probability tree for the given case represented in figure 45, the only path where no correct symbols are spelled is highlighted. Because that is the only path with no correct symbols, all the other paths must contain at least one correct symbol. Instead of summing all those other paths to get the probability, it is possible to take the max value of P, which is 1, and subtract the path that is not of interest.

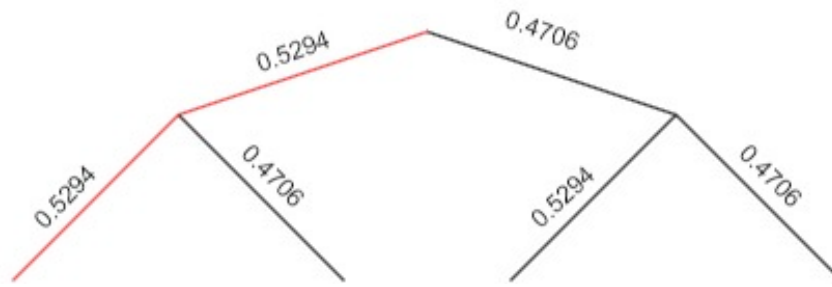


Figure 45: A probability tree for the 2-second flickering period with the case where no symbol is correctly spelled highlighted in red. The probabilities for getting a correct and incorrect symbol are 0.4706 and 0.529 respectively.

The probabilities presented for utilizing the two flickering periods are pretty similar, but with the one for the 6-second flickering coming out on top by some percentage points. Though the case presented was limited and very constructed, it shows an indication of how efficient the speller is for the different flickering periods over the shortest period of time possible and that the two most extreme flickering periods tested are not very different in terms of efficiency.

The more proper way of checking how efficient a speller is is by finding its Information Transfer Rate (ITR). This tells how much information can be transferred over a period of time and is given in bits per minute (bpm). Mannan et al. calculate the ITR as shown in equation 8:

$$ITR = \frac{1}{T} [\log_2 M + P \log_2 P + (1 - P) \log_2 \frac{1 - P}{M - 1}] \cdot 60 \quad (8)$$

where M is the total targets, P is the classification accuracy, and T is the average selection time (Mannan et al. 2020). For this project, there are 4 clusters with 6 targets each, resulting in an M=24. The group will assume for simplicities sake that selection time is flickering period + 1 second because the flickering needs 1 second to start and excludes the time it takes to look for a letter. Finally, P is the average accuracy for each flickering period from the test rounds. Based on this, the ITR for each test round is as given in table 16a and 16b.

Flicker period	Accuracy	ITR (bpm)
6 s	79.83%	25.26
5 s	72.27%	24.79
4 s	57.98%	20.43
3 s	51.26%	20.71
2 s	47.06%	23.85

(a) The ITR in bits per minute (bpm) and accuracies for the **first** test round for all tested flickering periods.

Flicker period	Accuracy	ITR (bpm)
6 s	45.38%	9.60
5 s	39.50%	8.80
4 s	50.42%	16.11
3 s	43.70%	15.74
2 s	39.50%	19.61

(b) The ITR in bits per minute (bpm) and accuracies for the **second** test round for all tested flickering periods.

Table 16: Two tables containing information about the ITR of the test rounds.

As indicated by the discussion comparing the probability of spelling at least one correct symbol for the 6-second and 2-second flickering periods, these periods have similar transfer rates for the first test round, with the longer period being slightly more ef-

ficient. The second test round shows a lower ITR, especially for the longer flickering periods of the second test round, not even breaking 10 bpm and being only 45% accurate or lower. Though the values are low, they are expected because of the large T and low P utilized in the calculation. But as discussed earlier in the Retesting, the results from the second test round are not very valid due to lots of potential sources of error.

Compared to the ITRs mentioned by Mannan et al., both for the values calculated by themselves and the values of other speller solutions, the values for this speller are low. In their paper, one of the slowest systems mentioned has an ITR of 50.41 bpm at 95.18% accuracy. Using this and the other discussed systems as a baseline, the speller created by the group is not very efficient, even when cherry-picking the best result observed at an ITR of 25.26 bpm and accuracy of 79.83%.

6.7.2 BCI Difficulties - Part 1

Shortly after the initial implementation of the BCI speller in Unity and NeuroPype, it was not working correctly. With many different software and hardware solutions interacting, it was hard to know why it did not work properly. Therefore, different sources of error were identified and solved to make the speller work properly. The following text presents and discusses the identified potential sources of error.

CCA phase sensitive One of the potential problems could've been that CCA was phase sensitive. The group suspected that if the signals compared in the CCA did not covary exactly, it would lead to the output of the computation showing little to no correlation. To check whether this was the case or not, a Python script was created that generated a sinus wave with some noise. This signal was then phase-shifted 0.1 seconds until a full period was achieved. Each phase shift iteration was then compared with another sinus wave with the same frequency as the original one using the CCA implementation written for the speller setup. This series of comparisons showed that the correlation was no less than 0.8 for each iteration, suggesting that CCA is not phase-sensitive (table 9).

Visual system shift When the subject looks at the designated frequency, there is a delay in the visual system. Researchers take this delay into consideration by applying a delay of 100-200 milliseconds (Mannan et al. 2020). However, due to the CCA not being phase-sensitive, this potential source of error was ruled out.

Unity inconsistency How the speller was implemented in Unity may be part of the problem, but Unity in itself could also influence the performance of the speller. The LSL stream from Unity was implemented such that it should be as computationally simple as possible, sending only an incrementing floating number, to avoid introducing lag or other inconsistencies from Unity. LSL4Unity allows for a stream to be regular or irregular, of which the group chose regular and sent out the data in the FixedUpdate hook. Though the setup suggests a consistent stream, this might not be the case due to underlying factors the group members are not aware of.

NeuroPype - Fuse Streams The node Fuse Streams that NeuroPype offers was originally used for merging the EEG recordings with the eye tracker. This node is however in a beta state, which means it is experimental and might not work as intended. When inspecting the data that the node it was somewhat strange, and since the node was experimental, it was decided to remove the node from the pipeline.

CCA implemented in Python To ensure that the CCA method offered by the python library scipy was correctly implemented, a synthetic EEG signal was sent into the pipeline. The CCA was able to find a large correlation between the synthetic signal and the stimulus, suggesting that was not causing the setup to not work as intended.

Flickering duration The flickering duration used when analyzing the EEG signal was suggested by related studies (Mannan et al. 2020). Research suggested that the longer the flickering duration is, the higher the accuracy should be (Mannan et al. 2020). To test this the flickering duration was raised up to ten seconds, but this still gave the same results.

Letter distance in Unity Something else to consider was the distance between the letter-cubes in the speller. If the letter-cubes were too close, it might be hard to focus on a single frequency, when all the other letter-cubes are flickering in close proximity. If too far apart, the number of symbols in the speller could be needed to be lowered due to suboptimal use of space. When changing the distance between the letter-cubes, there was little to no change in accuracy.

Contrast of flickering Research suggests that the contrast of the flickering letters is an important factor in creating well-defined frequencies in the EEG signals (Guo et al. 2019). When changing from the original red and white flickering to black and white flickering, the features should be easier to extract (Guo et al. 2019).

Hardware limitations The possibility that the VR headset might not be able to provide consistent flickering of the letter-cubes was also considered. Assuming that the Unity implementation works, the VR headset might be the limiting factor. Though the refresh frequency of the headset is rated at 90 Hz, it might not be able to output those frames consistently resulting in the signals observed in the EEG not being compatible with the compared values in the CCA. If that were the case, the speller would not be implementable, and therefore it was assumed that the VR headset is able to represent these frequencies.

6.7.3 BCI Difficulties - Part 2

The speller did not yet seem to work, and it was unsure whether the main test subject could even use BCIs. Due to excessive use, the EEG cap had been stretched to fit that person's head. This resulted in the electrodes responsible for gathering data related to the visual system not having proper contact with the head of the other potential test subjects. This potential source of error would be ruled out by keeping

the same subject for testing. Earlier tests indicated that the NeuroPype pipeline and CCA calculation worked as expected. To figure out whether the main subject could use BCIs or not, the group tried eliminating the parts of the system that introduced the most uncertainty: Unity and the VR headset.

This was done by asking ChatGPT to write a program in C which created a 2D window that flickered from black to white with four different frequencies in one quadrant each. The reason the group used ChatGPT for this part was that the main focus of this bachelor's project was not to create such a flickering program from the ground up, but it was rather used as a tool to eliminate uncertainties that ultimately could lead to a working speller. The group initially tried to find such a program on GitHub but did not find anything of use. It was important that the program was written in C, to ensure it was fast enough to keep the frequencies consistent.

The program was run on a Huawei laptop with a 1080p 60Hz screen with flickering frequencies of 4 Hz, 5 Hz, 6 Hz, and 7 Hz, but also up to 15 Hz. While looking at each of these frequencies, the group found that the main test subject could in fact use BCIs, because the CCA analysis then showed correlations between 0.7-0.9 for the actual frequency being looked at. Though it had a higher correlation for lower frequencies of 4-10 Hz than higher ones, of 11-15 Hz. This data was unfortunately not recorded. Later work would then consist of figuring out whether something related to Unity or the headset itself was the root of the problem. Some potential sources of error were identified: 1) noisy or unusable EEG data, 2) the contrast of the blinking cubes, 3) the implementation of the blinking, 4) Unity not being able to display the refresh rate that is being indicated, 5) hardware limitations in the form of the VR headset, 6) something being wrong with how all the parts of the entire system was being connected, 7) not being synchronized correctly and 8) there being too much visual noise because too many frequencies were visible at once.

Noisy or unusable EEG data A BCI speller heavily relies on usable EEG data. If there is a lot of noise, it might be hard, or impossible to extract the features needed. The EEG data was however proven to be usable when the group was able to recreate a SSVEP speller in a 2D environment with low frequencies.

Contrast The group changed the way the boxes were rendered to an unlit variant so as to make the white more white and the black more black than in the original implementation, increasing the contrast even more. But it had no notable effect on the CCA result.

To make the flickering more pronounced, the letter in front of the cube was also shrunk. No notable changes in the result. The background color was also controlled to be more gray and uniform. No notable changes in the result.

Incorrect Unity implementation Using another SSVEP speller implemented in Unity as inspiration showed promising results. The repository for that project is found at https://github.com/ryanlintott/SSVEP_keyboard. Early flickering in the project was based on Unity's Time library and changed color when a certain threshold was met. The other speller is based on a sine wave and changes between two colors using a Unity function called 'lerp', which was also implemented in the speller created for this project. This

change made the speller more accurate for some of the tested frequencies, which were 4 Hz, 5 Hz, 6 Hz, 7 Hz, and 9 Hz but did not consistently get 11 Hz right, just as when testing with higher frequencies in the C program. Something else to note is that the changing of color was also changed from the Update hook to the FixedUpdate hook, though it is not clear if that had anything to do with the more accurate results.

Due to the main problem seemingly being fixed, the group did not check the remaining potential sources of error but rather continued making small adjustments to the layout and usability of the speller and prepared for the testing.

6.8 Administrative Results

The administrative results were mostly done according to the different templates. Nevertheless, the group deviated slightly from the template in some aspects to better accommodate their project.

6.8.1 Milestones

The group's supervisor advised against creating a Gantt chart, given the open and vague nature of the assignment. This made it difficult to plan precisely what would be implemented and when. The group's lack of experience with this type of project further complicated time estimation. Instead, the group opted to develop a list of milestones. Gantt charts are useful for obtaining a clear overview and plan of the time allocation for each task component. Milestones, on the other hand, offer more flexibility and are less time-consuming to create than Gantt charts, particularly when considering the likelihood of discarding the chart due to the assignment's nature.

As the results indicate, the planned milestones and the actual milestones did not align perfectly. The synchronization of the EEG and eye tracker was completed by the beginning of March, significantly ahead of schedule. It is unclear whether this was due to an overestimation of the task's size or the group's working speed. Consequently, an additional component was introduced to the assignment: using the synchronized eye tracker and EEG to develop a BCI speller. This addition was also introduced at the beginning of March.

Feeling confident, the group scheduled the POC presentation of the BCI speller for the 30th of March, the last day before the Easter holiday. This was not a presentation of a functional BCI speller, due to the group encountering some challenges with the BCI speller, ultimately presenting a fully functional speller after the Easter holidays on the 14th of April. The group required only a few days to get the speller working; the extended timeframe was mainly due to the Easter break.

Once the functional speller was implemented, the group conducted an initial test to verify the BCI speller's effectiveness and a retest to assess whether participants' accuracy improved with training. Between the two test days, the group relocated their equipment to a booth at the Sensor Tech Day at NTNU Øya and showcased their BCI speller. The testing of the speller was carried out as planned, with all subjects attending on both days and completing the sessions.

Overall, utilizing milestones to plan tasks and estimate the time required for each

task was appropriate for this assignment. As the actual milestones demonstrated, accurately predicting the time allocation for different components proved challenging, especially for the BCI speller, which was introduced midway through the project. Other milestones like the deadline of the main report were easier to set.

6.8.2 Meetings

To maintain regular communication with the supervisors, the group scheduled meetings every other week. During crucial parts of the project, the group was also allowed to contact the supervisor outside of meeting hours. Since the assignment involved novel technologies and concepts like biotechnology, biology, and signal processing, the group appreciated this opportunity. This was particularly helpful when working on the biological aspects of the task, where Emanuel Lorenz had extensive expertise.

As the project neared its completion, the group held more frequent meetings to obtain regular feedback on the main report before the deadline. The group managed to maintain structure around meetings and formal aspects, such as meeting invitations and reports. Tresselt had the responsibility to send the meeting invitations, as well as being the chair of each meeting. Larsen was set to be the secretary. The group believes that the structured and consistent approach to meetings played a significant role in the successful outcome of the assignment. This effective meeting management can be applied to the rest of the project, ensuring continued progress and positive results

6.8.3 Timesheet

The bachelor thesis accounts for only 20 study points, and the group had to study another subject concurrently with the project. This subject required 10 study credits to be completed and was scheduled over 10 weeks. Consequently, the group had to deprioritize the bachelor's project in favor of this subject. This can be observed in the reduced number of hours worked by each member during these weeks. Especially during weeks 9-11, few hours were worked due to the preparation for the exam in the other subject.

Following the exam in week 11, the bachelor project became the group's primary focus, leading to a significant increase in hours worked on the project. The group decided to take the Easter holiday in week 14 and worked 0 hours. The group also had to share equipment with another bachelor group, which meant that a maximum of 3 days a week were dedicated to working on the coding/research part of the project while the remaining time was spent on documentation and writing the main report. As deadlines approached, the group also had to work on weekends to ensure they met the deadline. Due to some hours lost at the start, the group had to work more towards the end of the project and committed to working 9 hours every weekday to ensure they reached the 500 hours mark. They worked a total of 1004 hours, where only a single hour separated their time use. When taking the 1004 hours into consideration, this single hour is less than 0.05% of the entire workload of the project.

The group also tracked what area of work was done and maintained a weekly summary of the work accomplished during that week. This was done in accordance with the NTNU template.

7 Conclusion and further work

This project tried to utilize the common marker of eye blinks to synchronize an EEG and an eye tracker integrated into a VR headset. It was not possible to synchronize these with less than a millisecond offset, due to limitations in the equipment, making it impossible to fulfill the first research question, and functional attribute defined in the vision document (appendix F). The group was, however, able to synchronize with a STD of 8.87 ms, which is a little more than the time it takes to get one sample from the slowest equipment (8.33 ms). The group found an offset of around 62 ms.

To prove the synchronization of the eye tracker and the EEG a BCI speller was implemented. This speller was initially tested with an accuracy of 79.83 % with 6-second flickering, and down to 47.06% with 2-second flickering. The test subject with the most training did however have 100% accuracy for every flickering length except 2 seconds, where the subject had 94.12% accuracy, which is the same as missing a single symbol. Because of the subject's amount of training, the hypothesis of better results with more training was raised. The group can only conclude with the speller working to some extent, mainly based on the results from the initial test round, fulfilling the second functional attribute defined in the vision document.

The retesting of the speller resulted in overall worse performance, where some of it is suspected to come from a logical error implemented between the two testing sessions. This error does however not explain everything, like 4 seconds of flickering giving the highest results. It is suspected that noise in the EEG was introduced during some of the tests, therefore resulting in extremely varying results. This is supported by findings, like it being discovered that the neighboring group used PC chargers that affected the EEG signal. Even though the group tried to follow standard scientific conventions and control each and every variable one at a time, they sometimes got a bit ahead of themselves by changing multiple things at once or they were affected by other sources of error outside of their control. Other non-functional goals were met to a larger degree, though. The code produced by the group had comments throughout explaining what was happening and general naming conventions for e.g. variables and files were clear. What the group considered the most important usability criteria were also implemented, even though the speller was just meant as an informal test to see if the synchronized equipment could be used for something. They also managed to make the speller robust, taking unexpected behavior such as quitting spelling a letter halfway into consideration, but did not manage to make the speller relatively efficient.

The group also regrets not saving all the data when testing the BCI speller. This resulted in the group not being able to fully support their findings, such as the weird behavior of the fuse streams node, and the low accuracy of the speller using higher frequencies. This would have made the claims regarding these results more credible.

7.1 Further Work

Future research based on the synchronization of the equipment is encouraged. To ensure a more accurate synchronization, the blinks should be recorded using an eye tracker with a higher sampling rate. It would be interesting to see if the results show a similar, higher, or lower STD. It would also be interesting to explore the effects of

high system load on synchronization. Would an excess of CPU-intensive programs running on the machine streaming and recording the data introduce more jitter? And if so, to which degree does it affect the jitter?

When it comes to the speller, the group suggests investigating the reasons why the use of higher frequencies did not work. Many other BCI speller solutions use higher frequencies than implemented here (Bin et al. 2009; Mannan et al. 2020; Talevi et al. 2023). Could the reason be a flaw originating from Unity, could it be something to do with the VR headset or other equipment, could it be an implementation error on the part of the group, or something else entirely? These questions warrant further exploration.

Regarding the testing of the speller, the group suggests redoing the second test round and allow for the participants to practice more with the speller. It could also be interesting to have more test rounds, such as to be able to have a stronger foundation for concluding whether training affects the usage of the speller. As seen in the study mentioned in section 3.4.2, the participants had the opportunity to practice way more than the ones in this project, which might be a better approach to seeing the effect of training.

Another matter that could be subject to further work is if the distribution of flickering frequencies throughout the cluster has any impact on the speller performance. Mannan et al. did not have all the frequencies in their cluster in order (Mannan et al. 2020), and so changing it up might have made the results different. Results from the two test rounds, which can be seen in table 12 and 15, indicated a bias towards the lower frequencies. More of the frequencies from 4-5.5 Hz had successful hits than from 6-7.4 Hz, especially for the 2-second flickering period. However, the results might be biased since there are more symbols associated with lower than higher frequencies, which might drive up the probability of the lower frequency symbols being spelled than high ones.

Additionally, some noteworthy observation from the testing was that the CCA seemed to give relatively high output for 4 Hz for almost all of the spelled symbols, regardless of the participant. This led to many of the misspelled symbols being symbols associated with lower frequencies, particularly 4 Hz. As per this thesis, there are no concrete data on which symbols were spelled instead of the desired symbol, but it could be interesting to look at in further work to get an idea of to which degree the lower frequencies are biased.

The group also encourages others to explore the possibilities regarding creating a more efficient speller, with a higher Information Transfer Rate (ITR). The speller implemented in this project did not focus on speed, and there might be more possibilities regarding utilizing the 3D environment of the VR glasses.

8 Societal Impact

The following section discusses some societal impacts this project could have. Both the synchronized eye tracker and EEG, and the BCI speller can have a great impact on society. Some issues related to professional ethics are also presented.

8.1 Effects on Health

Utilizing an EEG and an eye tracker combined can have beneficial effects on health. This combination of equipment could be used both in medicine and in psychology to expand already known diagnostic methods and techniques regarding either an EEG or an eye tracker. The benefits of a single BCI system can be demonstrated by using the BCI speller as an example:

If individuals who struggle with communicating get access to a BCI speller solution, they could express themselves and their needs to a larger degree. This could have many benefits for the workforce in the health sector. One example is improved decision-making for patients who usually have difficulties communicating. If such individuals can inform healthcare personnel about their current state, e.g. if they are hurt somewhere or are fatigued, this can help the workers make decisions to better manage their patients' health. Focusing on the patient's needs is what one could call patient-centered, and has become more relevant in recent years (Breen et al. 2009). This could also be beneficial for individuals with communication difficulties that also have caregivers. Being able to express their needs and other relevant information makes the caregiver's job more concrete, which could help reduce stress and frustration for both the ones with difficulties communicating and their caregivers. A study looking at individuals with Amyotrophic lateral sclerosis (ALS) and their caregivers using BCI spellers to communicate were generally enthusiastic about it (Blain-Moraes et al. 2012).

There is also the matter of innovation and accessibility. If more research is done on BCI spellers, innovation might follow which in turn might lead to new applications of such technology. Then, if innovation takes place, BCI related technology will be more available and accessible, making it easier for the health sector to invest in due to the lower cost associated with such development. And with more establishments offering more ways to communicate for struggling individuals, this will in turn lead to all the benefits related to increased communication presented.

Another possibility is the usage of BCI systems as a tool in rehabilitation and therapy in stroke patients (Tonin et al. 2017). This has been studied over the last several decades as pointed out by this paper (Lazarou et al. 2018). Though such BCI systems might rather be based on Motor Imagery (MI) (theory 2.7.3) and not spelling, they can help paralyzed people or individuals with other motor impairments regain functionality to some degree. This can also probably be applied to other areas. This combination of equipment could also for example extend the Apple Test, which is a test used on patients to determine if they suffer from visual neglect (Bickerton et al. 2011).

8.2 Economical Perspective

As some kinds of technology mature, the cost usually goes down as seen with e.g. solar power (Hirth 2015) and ultimately be more accessible to the public. Both eye trackers and EEGs are expensive equipment. Therefore it is needed for continuous research and maturing of the technology, to make it more affordable over time.

Such a scenario could be beneficial for individuals who have the need for BCI system, such as a BCI speller, but with few economic resources. It is shown that disabled individuals often generally are poor (BERESFORD 1996). It would especially be beneficial for those in that group who live in developing countries, bearing in mind that these countries generally are worse off than developed countries.

The equipment used during the project is more on the expensive side, but brands such as NeuroSky and Emotiv offer consumer-level EEG solutions (*EEG Headsets | NeuroSky Store 2023; INSIGHT - 5 Channel EEG Brainwear® 2023*) and with HTC offering eye tracking capabilities as an add-on for already bought VR headsets (*VIVE Focus 3 Eye Tracker | VIVE Business European Union 2023*), a future with such equipment easily accessible is plausible. With this technology available to the public for a reasonable amount of money, it could speed up the development of new BCI systems, due to more developers being able to work with this equipment. This could result in more people reaping the benefit of the potential use cases the systems provide. Some are already mentioned in this section, and others might yet be discovered.

8.3 Societal Relevance

The synchronized eye tracker and EEG could open new possibilities in diagnostic techniques. Combining this with the 3D environment, researchers are able to create a fully controlled environment, exceeding current possibilities. This could greatly impact society by giving a better diagnostic tool in areas such as medicine, and in psychology, where the use of EEG and eye tracking is becoming more popular (Svartdal 2023).

The BCI speller is also just a single BCI system that can positively impact society. With approximately 1.3 billion people worldwide living with a significant disability, this technology could potentially improve the lives of those who struggle with movement and vocalization but have functional eyes and average brain activity (*Disability 2023*). By enabling easier communication, these individuals can more effectively express their thoughts and opinions.

The importance of being heard is particularly crucial for some people with disabilities, as demonstrated in a study on individuals with developmental disabilities, where the majority expressed a desire to be seen and heard like anyone else (Wilkenfeld 2015). While some are less concerned, the majority are affected by this issue. By facilitating communication and other solutions for those with disabilities, this technology may improve well-being and reduce inequalities, aligning with the United Nations' sustainability goals 3 (good health and well-being) and 10 (reducing inequalities) (*FNs baerekraftsmaal 2023*).

This combination of equipment could also create new ways people interact online. The popularity of VR is relatively high, as seen in the Steam charts for VR games played (*Most played VR Games Steam Charts 2023*), and might have an impact on

society. VRChat is a game that bases itself around the concept of interacting with others and exploring different experiences created by others (VRChat 2023). The all-time peak of VRChat is over 48 000 concurrent users, which is a large number of individuals. All these people most likely got some cultural input from the other people they met online, learning more about how individuals from other parts of the world behave and what values they hold. And if EEG became a more accessible gadget, its integration could be used for new and innovative VR experiences such as extracting one's emotional state based on certain brain waves and displaying it in an avatar. When expressing more emotion, an increase in empathy might follow which again could lead to a deeper understanding of one's fellow human beings. This is but an example of what could happen, should this equipment be used online.

8.4 Sustainability

The overall description of goals 3 and 10 of the UN's sustainability goals fit into the cases presented so far. Goal 3 is all about indiscriminate increased well-being and ensuring people live healthy lives (*God helse og livskvalitet* 2023). This fits well with the introduction of new, better diagnostic tools, as discussed in Effects on Health and Societal Relevance. This has the possibility of increasing people's quality of life in both a health and societal sense.

As for goal 10, its main focus is to decrease the inequality between and within countries (*Mindre ulikhet* 2023). The focus here will be on decreasing inequality within countries. A BCI system has the power to give people with disabilities an easier life, like the ability to communicate more freely when using a BCI speller. This might make them involve themselves more in e.g. education, the job market, and politics, due to them having a higher quality of life, and being able to do things they couldn't be doing otherwise. This could be things like controlling robotic limbs or speaking only using their mind and eyes.

8.5 Issues Related to Professional Ethics

The group declares no conflict of interest associated with this project.

The group followed ethical guidelines when handling sensitive data and the project was approved by NSD/Sikt (research methodology 3.5). This is to ensure the processing of sensitive data was done correctly to avoid any breaches of confidentiality. This is especially important when working with sensitive data like, EEG signals. A person's EEG signals are unique. There are studies that have used the EEG signals of a person as a personal unique key, like a fingerprint (Demuru and Fraschini 2020). This shows the vulnerability of the EEG data, and is why the group never stored this data from the testing of the subjects, including it not being legal, since the letter of consent did not include the use or sampling of biological data.

The group has previously throughout the report discussed the scientific rigor of this project. Instead of referring to previous arguments, that are somewhat scattered across the report, the following text is a short summary of the things that went well, and the things that could have been done differently.

The group was happy with the synchronization of the EEG and eye tracker. How to

synchronize the equipment was well defined due to diligent investigation of how to synchronize generally and clear definitions of what to synchronize on, leading to the group efficiently completing the initial part of the project. The group also had some difficulties implementing the speller as previously discussed in BCI Difficulties - Part 1 and BCI Difficulties - Part 2. The group did however find a solution through guidance and following other people's implementations. Another strength of the project was during development and testing, the group considered and took some potential sources of error into account. For example, early in development, it was discovered that the power supplies for some PCs' chargers were highly prone to leaking. This caused lots of noise in the EEG signals and was taken into consideration, but mainly for the group members. Other groups in close proximity were not as easy to control, though. Something else that was avoided was physical contact with any kinds of conductive materials for the person using the EEG cap because this had a tendency to also cause noisy signals. During the speller testing, participants with smartwatches were also asked to take these off, in case they interfered with the EEG signals.

Other parts of the project were not as optimal, though. During the development of the speller, up to multiple fixes were implemented at once, both for the CCA part and the stimuli presentation part, which is not very scientific. Ideally, one variable should be changed at a time to ensure each one is isolated and controlled. Another example of suboptimal scientificity is from the testing phase. To increase the likelihood of the results being valid and not affected as much by randomness, more participants should have been called in for testing. The lab was also not completely controlled for potential sources of error such as power supply leakage introducing noise in the EEG signal, which likely happened especially during the second test phase. Though there was a conscious attempt by the group to eliminate as many sources of error as possible, other groups in the same room were not controlled as extensively. There was also the fact that the code used for testing the speller was changed from the initial test round to the second one. Ideally, the test circumstances should have been identical so as to eliminate the effect of as many variables as possible, but that was not the case for this project. Luckily, it was tested and found that CCA is not phase sensitive, as discussed earlier in Engineering results, which ultimately means that the additional shift that was introduced should not have had a large influence on the final accuracy result. But whether the shift affected the results in any meaningful way in the second test round can't be said for certain, given the results that were recorded and the other sources of error present. Though some sources are more likely to have a more prominent effect, one can not eliminate all the other sources from consideration entirely.

Bibliography

- actiCAP Xpress Twist | Brain Products GmbH > Solutions* (2023). en-US. URL: <https://www.brainproducts.com/solutions/acticap-xpress-twist/> (visited on 7th Feb. 2023).
- Advanced Solutions Nederland (Apr. 2020). *Difference between IIR and FIR filters: a practical design guide*. en-US. Section: Algorithms. URL: <https://www.advsolned.com/difference-between-iir-and-fir-filters-a-practical-design-guide/> (visited on 11th Apr. 2023).
- Anderson, Vitas et al. (Jan. 1999). 'Nerve Conduction Velocity and Mobile Phones'. In: pp. 749–752. ISBN: 978-1-4613-7208-0. DOI: 10.1007/978-1-4615-4867-6_178.
- Aspøy, Arild (Feb. 2023). *Sikt – offentlig virksomhet*. no. URL: https://snl.no/Sikt_-_offentlig_virksomhet (visited on 27th Apr. 2023).
- Axinn, William G. and Lisa D. Pearce (Sept. 2006). *Mixed Method Data Collection Strategies*. en. Google-Books-ID: uZbjHzUaMhAC. Cambridge University Press. ISBN: 978-1-139-45838-2.
- BERESFORD, PETER (Dec. 1996). 'Poverty and Disabled People: Challenging dominant debates and policies'. In: *Disability & Society* 11.4. Publisher: Routledge _eprint: <https://doi.org/10.1080/09687599627598>, pp. 553–568. ISSN: 0968-7599. DOI: 10.1080/09687599627598. URL: <https://doi.org/10.1080/09687599627598> (visited on 10th May 2023).
- Bickerton, Wai Ling et al. (Sept. 2011). 'Separating forms of neglect using the Apples Test: validation and functional prediction in chronic and acute stroke'. eng. In: *Neuropsychology* 25.5, pp. 567–580. ISSN: 1931-1559. DOI: 10.1037/a0023501.
- Bin, Guangyu et al. (Aug. 2009). 'An online multi-channel SSVEP-based brain-computer interface using a canonical correlation analysis method'. en. In: *Journal of Neural Engineering* 6.4, p. 046002. ISSN: 1741-2560, 1741-2552. DOI: 10.1088/1741-2560/6/4/046002. URL: <https://iopscience.iop.org/article/10.1088/1741-2560/6/4/046002> (visited on 29th Mar. 2023).
- Blain-Moraes, Stefanie et al. (May 2012). 'Barriers to and mediators of brain-computer interface user acceptance: focus group findings'. In: *Ergonomics* 55.5. Publisher: Taylor & Francis _eprint: <https://doi.org/10.1080/00140139.2012.661082>, pp. 516–525. ISSN: 0014-0139. DOI: 10.1080/00140139.2012.661082. URL: <https://doi.org/10.1080/00140139.2012.661082> (visited on 18th May 2023).
- Blumenthal, Terry D. et al. (Jan. 2005). 'Committee report: Guidelines for human startle eyeblink electromyographic studies'. en. In: *Psychophysiology* 42.1, pp. 1–15. ISSN: 0048-5772, 1469-8986. DOI: 10.1111/j.1469-8986.2005.00271.x. URL: <https://onlinelibrary.wiley.com/doi/10.1111/j.1469-8986.2005.00271.x> (visited on 16th Feb. 2023).
- Bøekgaard, Per, Michael Kai Petersen and Jakob Eg Larsen (May 2014). 'In the twinkling of an eye: Synchronization of EEG and eye tracking based on blink signatures'. In: *2014 4th International Workshop on Cognitive Information Processing (CIP)*. ISSN: 2327-1698, pp. 1–6. DOI: 10.1109/CIP.2014.6844504.
- Bothner-By, Halvor (Jan. 2023). *pakkejitter*. no. URL: <https://snl.no/pakkejitter> (visited on 8th May 2023).
- Boxtel, A. van, A.j.w. Boelhouwer and A.r. Bos (1998). 'Optimal EMG signal bandwidth and interelectrode distance for the recording of acoustic, electrocutaneous, and photic blink reflexes'. en. In: *Psychophysiology* 35.6. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/1469-8986.3560690>, pp. 690–697. ISSN: 1469-8986. DOI: 10.1111/1469-8986.3560690. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/1469-8986.3560690> (visited on 25th Feb. 2023).

-
- Breen, Gerald-Mark et al. (Apr. 2009). 'Improving Doctor–Patient Communication: Examining Innovative Modalities Vis-à-vis Effective Patient-Centric Care Management Technology'. en. In: *Journal of Medical Systems* 33.2, pp. 155–162. ISSN: 1573-689X. DOI: 10.1007/s10916-008-9175-3. URL: <https://doi.org/10.1007/s10916-008-9175-3> (visited on 9th May 2023).
- Burle, Borís et al. (Sept. 2015). 'Spatial and temporal resolutions of EEG: Is it really black and white? A scalp current density view'. en. In: *International Journal of Psychophysiology*. On the benefits of using surface Laplacian (current source density) methodology in electrophysiology 97.3, pp. 210–220. ISSN: 0167-8760. DOI: 10.1016/j.ijpsycho.2015.05.004. URL: <https://www.sciencedirect.com/science/article/pii/S0167876015001865> (visited on 9th Feb. 2023).
- Calamia, Matthew, Kristian Markon and Daniel Tranel (2012). 'Scoring higher the second time around: meta-analyses of practice effects in neuropsychological assessment'. eng. In: *The Clinical Neuropsychologist* 26.4, pp. 543–570. ISSN: 1744-4144. DOI: 10.1080/13854046.2012.680913.
- Chassaing, Rulph (2005a). 'Finite Impulse Response Filters'. en. In: *Digital signal processing and applications with the C6713 and C6416 DSK*. Topics in digital signal processing. OCLC: ocm55682340. Hoboken, N.J: Wiley-Interscience, pp. 119–148. ISBN: 978-0-471-69007-8.
- (2005b). 'Infinite Impulse Response Filters'. en. In: *Digital signal processing and applications with the C6713 and C6416 DSK*. Topics in digital signal processing. OCLC: ocm55682340. Hoboken, N.J: Wiley-Interscience, pp. 177–204. ISBN: 978-0-471-69007-8.
- Cheveigné, Alain de and Israel Nelken (Apr. 2019). 'Filters: When, Why, and How (Not) to Use Them'. en. In: *Neuron* 102.2, pp. 280–293. ISSN: 0896-6273. DOI: 10.1016/j.neuron.2019.02.039. URL: <https://www.sciencedirect.com/science/article/pii/S0896627319301746> (visited on 11th Apr. 2023).
- Christiano, Lawrence J. and Terry J. Fitzgerald (2003). 'The Band Pass Filter*'. en. In: *International Economic Review* 44.2, pp. 435–465. ISSN: 1468-2354. DOI: 10.1111/1468-2354.t01-1-00076. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/1468-2354.t01-1-00076> (visited on 16th Apr. 2023).
- Clay, Viviane, Peter König and Sabine König (Apr. 2019). 'Eye Tracking in Virtual Reality'. In: *Journal of Eye Movement Research* 12.1, 10.16910/jemr.12.1.3. ISSN: 1995-8692. DOI: 10.16910/jemr.12.1.3. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7903250/> (visited on 13th Mar. 2023).
- Collection of personal data for research projects - Kunnskapsbasen - NTNU* (2023). URL: <https://i.ntnu.no/wiki/-/wiki/English/Collection+of+personal+data+for+research+projects> (visited on 27th Apr. 2023).
- Cook, ThomasD and Donald T Campbell (2002). 'Experimental and Quasi-Experimental Designs for Generalized Causal Inference'. en. In.
- Criswell, Eleanor (Mar. 2010). *Cram's Introduction to Surface Electromyography*. en. Google-Books-ID: ADYm0TqiDo8C. Jones & Bartlett Publishers. ISBN: 978-1-4496-6362-9.
- Dark and Bright pupil tracking* (Nov. 2022). en-US. URL: https://connect.tobii.com/s/article/What-is-dark-and-bright-pupil-tracking?language=en_US (visited on 21st May 2023).
- Demuru, Matteo and Matteo Fraschini (May 2020). 'EEG fingerprinting: Subject-specific signature based on the aperiodic component of power spectrum'. eng. In: *Computers in Biology and Medicine* 120, p. 103748. ISSN: 1879-0534. DOI: 10.1016/j.compbimed.2020.103748.
-

-
- Disability* (2023). en. URL: <https://www.who.int/news-room/fact-sheets/detail/disability-and-health> (visited on 4th May 2023).
- Downloads - Developer Resources* (2023). URL: <https://developer.vive.com/resources/downloads/> (visited on 3rd Mar. 2023).
- Dresch, Aline, Daniel Pacheco Lacerda and José Antônio Valle Antunes (2015). 'Design Science Research'. en. In: *Design Science Research: A Method for Science and Technology Advancement*. Ed. by Aline Dresch, Daniel Pacheco Lacerda and José Antônio Valle Antunes Jr. Cham: Springer International Publishing, pp. 67–102. ISBN: 978-3-319-07374-3. DOI: 10.1007/978-3-319-07374-3_4. URL: https://doi.org/10.1007/978-3-319-07374-3_4 (visited on 8th May 2023).
- Du, Jiale et al. (Mar. 2019). '3D Stimulus Presentation of ERP-Speller in Virtual Reality'. In: *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*. ISSN: 1948-3554, pp. 167–170. DOI: 10.1109/NER.2019.8717164.
- Dvergsdal, Henrik and Lars Aabakken (Jan. 2023). *virtuell virkelighet*. no. URL: https://snl.no/virtuell_virkelighet (visited on 3rd Mar. 2023).
- EEG Headsets | NeuroSky Store* (2023). URL: <https://store.neurosky.com/> (visited on 19th May 2023).
- Engine, Godot (2023). *Godot Engine - Free and open source 2D and 3D game engine*. en. URL: <https://godotengine.org/> (visited on 7th Feb. 2023).
- Engstrøm, Morten and Jan K. S. Jansen (Jan. 2023). *elektroencefalografi*. no. URL: <http://sml.snl.no/elektroencefalografi> (visited on 27th Jan. 2023).
- Eye openness - Tobii Pro SDK documentation* (2023). URL: <https://developer.tobii.com/commonconcepts/eyeopenness.html> (visited on 2nd Mar. 2023).
- FNs baerekraftsmaal* (Apr. 2023). no. URL: <https://www.fn.no/om-fn/fns-baerekraftsmaal> (visited on 10th May 2023).
- Gao, Di et al. (Jan. 2012). 'Non-Invasive Eye Tracking Technology Based on Corneal Reflex'. en. In: *Procedia Engineering*. 2012 International Workshop on Information and Electronics Engineering 29, pp. 3608–3612. ISSN: 1877-7058. DOI: 10.1016/j.proeng.2012.01.539. URL: <https://www.sciencedirect.com/science/article/pii/S1877705812005498> (visited on 31st Mar. 2023).
- Get started developing with the HTC VIVE Pro Eye using Tobii XR SDK* (2023). URL: <https://developer.tobii.com/xr/develop/unity/getting-started/vive-pro-eye/> (visited on 25th Jan. 2023).
- GitHub - labstreaminglayer/plugin-UE4: LabstreamingLayer plugin for Unreal Engine 4* (2023). URL: <https://github.com/labstreaminglayer/plugin-UE4> (visited on 9th Feb. 2023).
- GitHub - Ullolabs/tutorial.GodotPythonLSL: Follow along our tutorial demoing the use of Godot engine to control a sprite with LSL* (2023). URL: <https://github.com/Ullolabs/tutorial.GodotPythonLSL> (visited on 9th Feb. 2023).
- Gjerstad, Leif and Geir Bråthen (Jan. 2023). *visuelt evoked potensial/response*. no. URL: https://sml.snl.no/visuelt_evoked_potensial/response (visited on 10th May 2023).
- God helse og livskvalitet* (Feb. 2023). no. URL: <https://www.fn.no/om-fn/fns-baerekraftsmaal/god-helse-og-livskvalitet> (visited on 10th May 2023).
- Godot driver for the HTC Vive Pro Eye equipment* (Aug. 2021). original-date: 2019-07-17T13:59:57Z. URL: <https://github.com/FAU-Inf2/godot-vive-pro-eye> (visited on 7th Feb. 2023).
- Goertzen, Melissa J. (May 2017). 'Chapter 3. Introduction to Quantitative Research and Data'. en. In: *Library Technology Reports* 53.4. Number: 4, pp. 12–18. ISSN: 0024-2586. URL: <https://journals.ala.org/index.php/ltr/article/view/6325> (visited on 8th Mar. 2023).
-

-
- Gregory, Jason (July 2018). *Game Engine Architecture, Third Edition*. en. Google-Books-ID: EwlpDwAAQBAJ. CRC Press. ISBN: 978-1-351-97427-1.
- Grønmo, Sigmund (Jan. 2023). *kvalitativ metode*. no. URL: https://snl.no/kvalitativ_metode (visited on 15th May 2023).
- Guo, Miaoji et al. (2019). 'Investigation of Visual Stimulus With Various Colors and the Layout for the Oddball Paradigm in Evoked Related Potential-Based Brain-Computer Interface'. In: *Frontiers in Computational Neuroscience* 13. ISSN: 1662-5188. URL: <https://www.frontiersin.org/articles/10.3389/fncom.2019.00024> (visited on 12th Apr. 2023).
- Haputhanthri, Dilantha et al. (July 2019). 'An EEG based Channel Optimized Classification Approach for Autism Spectrum Disorder'. In: pp. 123–128. DOI: 10.1109/MERCon.2019.8818814.
- Herwig, Uwe, Peyman Satrapi and Carlos Schönfeldt-Lecuona (Dec. 2003). 'Using the International 10-20 EEG System for Positioning of Transcranial Magnetic Stimulation'. en. In: *Brain Topography* 16.2, pp. 95–99. ISSN: 1573-6792. DOI: 10.1023/B:BRAT.0000006333.93597.9d. URL: <https://doi.org/10.1023/B:BRAT.0000006333.93597.9d> (visited on 13th Mar. 2023).
- Hirth, Lion (2015). 'Market value of solar power: Is photovoltaics cost-competitive?' en. In: *IET Renewable Power Generation* 9.1, pp. 37–45. ISSN: 1752-1424. DOI: 10.1049/iet-rpg.2014.0101. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1049/iet-rpg.2014.0101> (visited on 10th May 2023).
- Holm, Adrian, Anne Eilertsen and Tor-Ivar Krogsæter (Jan. 2023). *dataspill*. no. URL: <http://snl.no/dataspill> (visited on 3rd Mar. 2023).
- Holmqvist, Kenneth et al. (Sept. 2011). *Eye Tracking: A comprehensive guide to methods and measures*. en. Google-Books-ID: 5rIDPV1EoLUC. OUP Oxford. ISBN: 978-0-19-162542-8.
- Hutton, Sam (Apr. 2019). *Eye Tracking and EEG*. en-US. URL: <https://www.sr-research.com/eye-tracking-blog/background/eye-tracking-and-eeeg/> (visited on 20th Jan. 2023).
- INSIGHT - 5 Channel EEG Brainwear®* (2023). en-US. URL: <https://www.emotiv.com/insight/> (visited on 19th May 2023).
- Introduction – Labstreaminglayer 1.13 documentation* (2023). URL: <https://labstreaminglayer.readthedocs.io/info/intro.html> (visited on 3rd Feb. 2023).
- ISO (Dec. 2009). *ISO/IEC 9995-2:2009*. en. URL: <https://www.iso.org/standard/51644.html> (visited on 6th May 2023).
- Jansen, Jan K. S. and Espen Dietrichs (Jan. 2023). *evoked response*. no. URL: https://sml.snl.no/evoked_response (visited on 10th May 2023).
- Jiang, Xiao, Gui-Bin Bian and Zean Tian (Jan. 2019). 'Removal of Artifacts from EEG Signals: A Review'. en. In: *Sensors* 19.5. Number: 5 Publisher: Multidisciplinary Digital Publishing Institute, p. 987. ISSN: 1424-8220. DOI: 10.3390/s19050987. URL: <https://www.mdpi.com/1424-8220/19/5/987> (visited on 20th Jan. 2023).
- Kleinnijenhuis, Michiel (2023). *Real-Time Filtering in BioExplorer*. en. URL: https://brainclinics.com/wp-content/uploads/Filtering_in_BioExplorer.pdf (visited on 9th May 2023).
- Kliegl, Reinhold, Jacqui Smith and Paul Baltes (1989). *Testing-the-limits and the study of adult age differences in cognitive plasticity of a mnemonic skill*. URL: <https://psycnet.apa.org/fulltext/1989-21959-001.html> (visited on 19th Apr. 2023).
- Krafka, Kyle et al. (June 2016). 'Eye Tracking for Everyone'. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. ISSN: 1063-6919, pp. 2176–2184. DOI: 10.1109/CVPR.2016.239.
-

-
- Kristiansen, Jostein Riiser (Dec. 2019). *harmonisk svingning*. no. URL: https://snl.no/harmonisk_svingning (visited on 16th May 2023).
- Kropotov, Juri D. (Jan. 2016). 'Chapter 1.6 - Event-Related Potentials'. en. In: *Functional Neuromarkers for Psychiatry*. Ed. by Juri D. Kropotov. San Diego: Academic Press, pp. 59–78. ISBN: 978-0-12-410513-3. DOI: 10.1016/B978-0-12-410513-3.00006-1. URL: <https://www.sciencedirect.com/science/article/pii/B9780124105133000061> (visited on 8th Mar. 2023).
- Kundu, S. and S. Ari (Aug. 2022). 'Brain-Computer Interface Speller System for Alternative Communication: A Review'. en. In: *IRBM* 43.4, pp. 317–324. ISSN: 1959-0318. DOI: 10.1016/j.irbm.2021.07.001. URL: <https://www.sciencedirect.com/science/article/pii/S1959031821000944> (visited on 13th Mar. 2023).
- Lab Streaming Layer* (2023). URL: <https://labstreaminglayer.org/#/> (visited on 27th Jan. 2023).
- LabStreamingLayer submodules · GitHub* (2023). URL: <https://github.com/labstreaminglayer> (visited on 9th Feb. 2023).
- Lazarou, Ioulietta et al. (2018). 'EEG-Based Brain-Computer Interfaces for Communication and Rehabilitation of People with Motor Impairment: A Novel Approach of the 21st Century'. In: *Frontiers in Human Neuroscience* 12. ISSN: 1662-5161. URL: <https://www.frontiersin.org/articles/10.3389/fnhum.2018.00014> (visited on 9th May 2023).
- Leske, Sabine and Sarang S. Dalal (Apr. 2019). 'Reducing power line noise in EEG and MEG data via spectrum interpolation'. en. In: *NeuroImage* 189, pp. 763–776. ISSN: 1053-8119. DOI: 10.1016/j.neuroimage.2019.01.026. URL: <https://www.sciencedirect.com/science/article/pii/S1053811919300266> (visited on 11th Apr. 2023).
- Leuchs, Laura and Brain Products (May 2019). *Choosing your reference & why it matters - a BrainVision Analyzer Tip*. en-US. URL: <https://pressrelease.brainproducts.com/referencing/> (visited on 15th May 2023).
- Levine, Judah (Feb. 2020). 'The Statistics of Computer Clocks and the Design of Synchronization Algorithms'. en. In: *Journal of Research of the National Institute of Standards and Technology* 125, p. 125008. ISSN: 2165-7254. DOI: 10.6028/jres.125.008. URL: <https://nvlpubs.nist.gov/nistpubs/jres/125/jres.125.008.pdf> (visited on 20th Apr. 2023).
- Li, Fali et al. (Oct. 2015). 'The enhanced information flow from visual cortex to frontal area facilitates SSVEP response: evidence from model-driven and data-driven causality analysis'. en. In: *Scientific Reports* 5.1. Number: 1 Publisher: Nature Publishing Group, p. 14765. ISSN: 2045-2322. DOI: 10.1038/srep14765. URL: <https://www.nature.com/articles/srep14765> (visited on 11th May 2023).
- Lin, Zhonglin et al. (Dec. 2006). 'Frequency Recognition Based on Canonical Correlation Analysis for SSVEP-Based BCIs'. In: *IEEE Transactions on Biomedical Engineering* 53.12. Conference Name: IEEE Transactions on Biomedical Engineering, pp. 2610–2614. ISSN: 1558-2531. DOI: 10.1109/TBME.2006.886577.
- Lopez-Gordo, M. A., D. Sanchez-Morillo and F. Pelayo Valle (July 2014). 'Dry EEG Electrodes'. en. In: *Sensors* 14.7. Number: 7 Publisher: Multidisciplinary Digital Publishing Institute, pp. 12847–12870. ISSN: 1424-8220. DOI: 10.3390/s140712847. URL: <https://www.mdpi.com/1424-8220/14/7/12847> (visited on 27th Jan. 2023).
- Ltd, Arm (2023). *What is a Gaming Engine?* en. URL: <https://www.arm.com/glossary/gaming-engines> (visited on 7th Feb. 2023).
- Mannan, Malik M. Naeem et al. (Jan. 2020). 'A Hybrid Speller Design Using Eye Tracking and SSVEP Brain-Computer Interface'. en. In: *Sensors* 20.3. Number: 3 Publisher: Multidisciplinary Digital Publishing Institute, p. 891. ISSN: 1424-8220. DOI:
-

-
- 10.3390/s20030891. URL: <https://www.mdpi.com/1424-8220/20/3/891> (visited on 8th Mar. 2023).
- Mento, Mark A. (May 2020a). *Learn here what are the different eye-tracking techniques and methods to record eye movements accurately, how to calibrate, and what are the limitations*. en. Section: Neurotechnology, Business & Marketing, Science & Research. URL: <https://www.bitbrain.com/blog/eye-tracking-technology> (visited on 14th Mar. 2023).
- (June 2020b). *The Different Kinds of Eye Tracking Devices*. en. Section: Neurotechnology. URL: <https://www.bitbrain.com/blog/eye-tracking-devices> (visited on 14th Mar. 2023).
- Mercer, J. A. et al. (Apr. 2006). 'EMG sensor location: Does it influence the ability to detect differences in muscle contraction conditions?' en. In: *Journal of Electromyography and Kinesiology* 16.2, pp. 198–204. ISSN: 1050-6411. DOI: 10.1016/j.jelekin.2005.07.002. URL: <https://www.sciencedirect.com/science/article/pii/S1050641105000878> (visited on 20th Apr. 2023).
- Mindre ulikhet* (Feb. 2023). URL: <https://www.fn.no/om-fn/fns-baerekraftsmaal/mindre-ulikhet> (visited on 10th May 2023).
- Most played VR Games Steam Charts* (2023). en. URL: <https://steamdb.info/charts/?tagid=21978> (visited on 19th May 2023).
- Nagel, Sebastian (Dec. 2019). 'Towards a home-use BCI: fast asynchronous control and robust non-control state detection'. PhD thesis. DOI: 10.15496/publikation-37739.
- Neuropype by Intheon* (2023). URL: <https://www.neuropype.io/docs/nodes/utilities.html#dejittertimestamps> (visited on 27th Apr. 2023).
- Noyes, Jan (Mar. 1983). 'The QWERTY keyboard: a review'. en. In: *International Journal of Man-Machine Studies* 18.3, pp. 265–281. ISSN: 0020-7373. DOI: 10.1016/S0020-7373(83)80010-8. URL: <https://www.sciencedirect.com/science/article/pii/S0020737383800108> (visited on 6th May 2023).
- Paradis, Elise et al. (May 2016). 'Design: Selection of Data Collection Methods'. In: *Journal of Graduate Medical Education* 8.2, pp. 263–264. ISSN: 1949-8349. DOI: 10.4300/JGME-D-16-00098.1. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4857496/> (visited on 27th Apr. 2023).
- Ph.D, Eduardo López-Larraz (Mar. 2020). *What is BCI? An introduction to brain-computer interface using EEG*. en. Section: Neurotechnology, Science & Research. URL: <https://www.bitbrain.com/blog/brain-computer-interface-using-eeg-signals> (visited on 7th Feb. 2023).
- Rossen, Eirik and Tom Heine Nätt (Dec. 2022). *API*. no. URL: <https://snl.no/API> (visited on 16th Apr. 2023).
- Samima, Shabnam, Monalisa Sarma and Debasis Samanta (Dec. 2017). 'Detecting Vigilance in People Performing Continual Monitoring Task'. In: pp. 202–214. ISBN: 978-3-319-72037-1. DOI: 10.1007/978-3-319-72038-8_16.
- Sandstad, Jakob and Geir Martin Haarberg (Nov. 2021). *elektrode*. no. URL: <https://snl.no/elektrode> (visited on 15th May 2023).
- Sharma, Kamal, Neeraj Jain and Prabir K. Pal (Jan. 2020). 'Detection of eye closing/opening from EOG and its application in robotic arm control'. en. In: *Biocybernetics and Biomedical Engineering* 40.1, pp. 173–186. ISSN: 0208-5216. DOI: 10.1016/j.bbe.2019.10.004. URL: <https://www.sciencedirect.com/science/article/pii/S0208521619304723> (visited on 23rd Feb. 2023).
- Skaar, Johannes (Jan. 2023). *faseforskyvning*. no. URL: <https://snl.no/faseforskyvning> (visited on 8th May 2023).
-

-
- Smith, Gerald (Jan. 1989). 'Padding point extrapolation techniques for the butterworth digital filter'. en. In: *Journal of Biomechanics* 22.8, pp. 967–971. ISSN: 0021-9290. DOI: 10.1016/0021-9290(89)90082-1. URL: <https://www.sciencedirect.com/science/article/pii/0021929089900821> (visited on 4th May 2023).
- Soufineyestani, Mahsa, Dale Dowling and Arshia Khan (Jan. 2020). 'Electroencephalography (EEG) Technology Applications and Available Devices'. en. In: *Applied Sciences* 10.21. Number: 21 Publisher: Multidisciplinary Digital Publishing Institute, p. 7453. ISSN: 2076-3417. DOI: 10.3390/app10217453. URL: <https://www.mdpi.com/2076-3417/10/21/7453> (visited on 13th Feb. 2023).
- SRanipal Getting Started Steps* (May 2019). en-US. URL: <https://forum.htc.com/topic/5642-sranipal-getting-started-steps/> (visited on 9th Feb. 2023).
- SRanipal Unreal Engine 5 SDK update timeline* (Apr. 2022). en-US. URL: <https://forum.htc.com/topic/12881-sranipal-unreal-engine-5-sdk-update-timeline/> (visited on 9th Feb. 2023).
- sranipal_documentation.html* (2023). URL: <https://developer-express.vive.com/resources/vive-sense/eye-and-facial-tracking-sdk/documentation/> (visited on 23rd Feb. 2023).
- Supported Devices and Tools – Labstreaminglayer 1.13 documentation* (2023). URL: https://labstreaminglayer.readthedocs.io/info/supported_devices.html (visited on 3rd Feb. 2023).
- Svartdal, Frode (Jan. 2023). *forskningsmetoder i psykologien*. no. URL: https://snl.no/forskningsmetoder_i_psykologien (visited on 16th May 2023).
- Talevi, Luca et al. (2023). 'A Ssvpep Bci Based On Canonical Correlation Analysis'. en. In: (). ISBN: 9783851255331 Publisher: Verlag der Technischen Universität Graz. DOI: 10.3217/978-3-85125-533-1-88. URL: <https://openlib.tugraz.at/download.php?id=5e6a0b27e3434&location=medra> (visited on 28th Mar. 2023).
- Tashakkori, Abbas and Charles Teddlie (2003). *Handbook of Mixed Methods in Social & Behavioral Research*. en. Google-Books-ID: F8BFOM8DCKoC. SAGE. ISBN: 978-0-7619-2073-1.
- Technologies, Unity (2023a). *Start Your Creative Projects and Download the Unity Hub | Unity*. en. URL: <https://unity.com/download> (visited on 12th Apr. 2023).
- (2023b). *Unity - Scripting API: GameObject*. en. URL: <https://docs.unity3d.com/2023.2/Documentation/ScriptReference/GameObject.html> (visited on 26th Apr. 2023).
- Time Synchronization – Labstreaminglayer 1.13 documentation* (2023). URL: https://labstreaminglayer.readthedocs.io/info/time_synchronization.html (visited on 7th Feb. 2023).
- TMSI (2023). *The 10-20 System for EEG*. en. URL: <https://info.tmsi.com/blog/the-10-20-system-for-eeeg> (visited on 21st May 2023).
- Tobii Eye Tracking SDK in Code Plugins - UE Marketplace* (2023). en-US. URL: <https://www.unrealengine.com/marketplace/en-US/product/tobii-eye-tracking-sdk/questions> (visited on 9th Feb. 2023).
- TobiiXR Core API Reference* (2023). URL: <https://developer.tobii.com/xr/develop/unity/documentation/api-reference/core/> (visited on 2nd Mar. 2023).
- Tonin, Luca et al. (2017). 'Behavioral and Cortical Effects during Attention Driven Brain-Computer Interface Operations in Spatial Neglect: A Feasibility Case Study'. In: *Frontiers in Human Neuroscience* 11. ISSN: 1662-5161. URL: <https://www.frontiersin.org/articles/10.3389/fnhum.2017.00336> (visited on 19th May 2023).
- Treder, Matthias S. and Benjamin Blankertz (May 2010). '(C)overt attention and visual speller design in an ERP-based brain-computer interface'. en. In: *Behavioral and Brain Functions* 6.1, p. 28. ISSN: 1744-9081. DOI: 10.1186/1744-9081-6-28. URL: <https://doi.org/10.1186/1744-9081-6-28> (visited on 8th Mar. 2023).
-

-
- Unity* (2023). URL: <https://developer.tobii.com/xr/develop/unity/> (visited on 12th Apr. 2023).
- Unreal Engine | The most powerful real-time 3D creation tool* (2023). en-US. URL: <https://www.unrealengine.com/en-US> (visited on 7th Feb. 2023).
- VanderWerf, Frans et al. (May 2003). 'Eyelid Movements: Behavioral Studies of Blinking in Humans Under Different Stimulus Conditions'. In: *Journal of Neurophysiology* 89.5. Publisher: American Physiological Society, pp. 2784–2796. ISSN: 0022-3077. DOI: 10.1152/jn.00557.2002. URL: <https://journals.physiology.org/doi/full/10.1152/jn.00557.2002> (visited on 23rd Feb. 2023).
- vitenskap – Store norske leksikon* (2023). URL: <https://snl.no/vitenskap> (visited on 9th Feb. 2023).
- VIVE Focus 3 Eye Tracker | VIVE Business European Union* (2023). en. URL: <https://business.vive.com/eu/product/vive-focus-3-eye-tracker/> (visited on 19th May 2023).
- VRChat* (2023). en-US. URL: <https://hello.vrchat.com> (visited on 19th May 2023).
- Web Content Accessibility Guidelines (WCAG) 2.1* (2023). URL: <https://www.w3.org/TR/WCAG21/> (visited on 5th May 2023).
- Wilkenfeld, Bonnie Fader (Oct. 2015). "Being Heard": Qualitative Research Conundrums With Individuals With Developmental Disabilities'. en. In: *Research on Social Work Practice* 25.6. Publisher: SAGE Publications Inc, pp. 702–710. ISSN: 1049-7315. DOI: 10.1177/1049731514547768. URL: <https://doi.org/10.1177/1049731514547768> (visited on 4th May 2023).
- Williams, Carrie (Mar. 2007). 'Research Methods'. en. In: *Journal of Business & Economics Research (JBER)* 5.3. Number: 3. ISSN: 2157-8893. DOI: 10.19030/jber.v5i3.2532. URL: <https://clutejournals.com> (visited on 27th Apr. 2023).
- Winter, David A. (2009). 'Signal Processing'. en. In: *Biomechanics and motor control of human movement*. 4th ed. OCLC: ocn318408191. Hoboken, N.J: Wiley, pp. 14–43. ISBN: 978-0-470-39818-0.
- Wrålsen, Anette and Kristi E. Berntsen (2022). *Kompendie Vitenskapelighet i BA-oppgaven - Wraalesen VT1 og Berntsen VT2 - IDI AIT.pdf*.
- XML Essentials - W3C* (2023). URL: <https://www.w3.org/standards/xml/core> (visited on 15th May 2023).
- Yesmin, Farzana (n.d.). 'IDENTIFICATION OF PHARMACEUTICAL SUBSTANCES WITH RAMAN SPECTROSCOPY'. en. In: ().
- Zerafa, R. et al. (July 2018). 'To train or not to train? A survey on training of feature extraction methods for SSVEP-based BCIs'. en. In: *Journal of Neural Engineering* 15.5. Publisher: IOP Publishing, p. 051001. ISSN: 1741-2552. DOI: 10.1088/1741-2552/aaca6e. URL: <https://dx.doi.org/10.1088/1741-2552/aaca6e> (visited on 31st Mar. 2023).
- Zhang, Yu et al. (Apr. 2015). 'SSVEP recognition using common feature analysis in brain-computer interface'. en. In: *Journal of Neuroscience Methods. Brain Computer Interfaces; Tribute to Greg A. Gerhardt* 244, pp. 8–15. ISSN: 0165-0270. DOI: 10.1016/j.jneumeth.2014.03.012. URL: <https://www.sciencedirect.com/science/article/pii/S016502701400106X> (visited on 10th May 2023).

Appendix

A Letter of Consent

PURPOSE OF THE STUDY

You are invited to participate in a research study at the Department of computer science (IDI) at the Norwegian University of Science and Technology (NTNU). Before you decide to participate in this study, it is important that you understand why the research is being done and what your participation will involve. Please read the following information carefully and feel free to ask the researcher if anything is unclear or if you need more information. The purpose of the study is to determine the accuracy of a hybrid SSVEP-based BCI speller.

STUDY PROCEDURES

The user is going to spell "I LIVE IN NORWAY." using the hybrid SSVEP-based speller. The user will do this five times, where each time the flickering length of each letter will be shortened by 1 second. The flickering length will start at 6 seconds, and eventually end at 2 seconds per letter. The user will be wearing a dry EEG and a VR headset.

DURATION

Each user is assigned 45 minutes for the test. The researchers however suggest that the user will use 10 minutes to equip the equipment (EEG) and around 20 minutes to do the test.

VOLUNTARY PARTICIPATION

Your participation in this study is voluntary. It is up to you whether or not you decide to participate. You will be asked to sign this consent form if you choose to participate. After signing this consent form, you are free to withdraw at any time and without giving a reason. If you withdraw from the study before data collection is completed, your data will be destroyed. No data will be recorded, besides name, signature, age, and gender.

RISK OF HARM

This research involves using a virtual reality headset; some users may experience cybersickness. Cybersickness may cause the user to experience discomfort, such as motion sickness, nausea, disorientation, blurred vision, eye strain, and eye fatigue. The EEG cap can also cause some pain due to electrodes putting pressure on the scalp. The user will also be exposed to flickering lights which can cause an epileptic seizure. The user is therefore asked to not participate if the user has any previous history of epilepsy.

CONFIDENTIALITY

The information recorded about you will only be used as described under the purpose of the project until June 2023 and is planned to be stored for 2 years (August 2023) in a restricted-access, secure file storage area at NTNU. During the project, your personal information will be treated confidentially and in accordance with data protection legislation (General Data Protection Regulation and the Personal Data Act). We reserve the right to reuse your anonymized information for a related project within the two years. After this period, all your information will be irreversibly deleted. Any extensions in use and storage time can only occur after approval from the NSD (Norwegian Centre for Research Data) and other relevant authorities. You have the right to access the information registered about you and request the deletion of the information or correct any errors in the registered information. You also have the right to access the security measures in place

for processing the information. You can file a complaint about the processing of your information with the data protection authority and the institution's data protection officer. All information will be processed without name, or other directly identifiable information. A number links you to your information through a name list, and this list is stored in a secure, access-restricted location. Only the bachelor group has access to this list. The final publication of our results is a necessary part of the research process. All publications should be made so that individual participants cannot be recognized, but we are obliged to inform you that we cannot exclude the possibility of this happening.

CONSENT

The Norwegian Centre for Research Data (NSD) has evaluated the project and has approved data storage and processing.

Following the new personal data protection law, the data controller NTNU, the bachelor students and main supervisor Alexander Holt have independent responsibility for ensuring that the processing of your information has a legal basis. This project has a legal basis in the EU's General Data Protection Regulation (GDPR) Article 6(1)(a) and Article 9(2)(a), and your consent. You have the right to file a complaint about the processing of your information with the data protection authority.

CONTACT INFORMATION

If you have any questions about this study, you may contact the researchers at oflarsen@stud.ntnu.no at any time.

If you would like to file a complaint regarding the processing of your personal information, please contact:

- NSD - Norwegian Centre for Research Data AS (Email: personverntjenester@nsd.no, Phone: +47 55 58 21 17)
- Main supervisor - Alexander Holt (alexander.holt@ntnu.no)

I have read the provided information, or it has been read to me. I agree to participate in the project and to my data being used as described.

Name

Date

Participant's signature

Researcher's signature

B Eye blink synchronization results

Blink Cut End	Blink 1	Blink 2	Blink 3	Blink 4	Blink 5	Blink 6	Blink 7	Blink 8	Blink 9	Blink 10	Time Between Indexes (s)	Sum (indexes)	Average (indexes)	Average time (ms)
LSLData-17-04 2-1	-10	-17	-16	-16	-17	-18	-11	-17	-18	-17	0.004	-156.996	-15.7	-62.8
LSLData-17-04 2-6	-17	-17	-15	-21	-19	-20	-16	-20	-16	-14	0.004	-174.996	-17.5	-70
LSLData-17-04 2-14	-12	-15	-11	-12	-13	-11	-11	-12	-11	-18	0.004	-125.996	-12.6	-50.4
LSLData-17-04 8	-16	-16	-16	-16	-22	-8	-12	-15	-14	-13	0.004	-147.996	-14.8	-59.2
LSLData-17-04 2-11	-14	-13	-10	-13	-20	-11	-18	-14	-22	-16	0.004	-150.996	-15.1	-60.4
LSLData-17-04-10	-19	-21	-18	-14	-18	-11	-18	-17	-18	-20	0.004	-173.996	-17.4	-69.6
LSLData-17-04 2-3	-17	-14	-20	-23	-18	-16	-24	-18	-18	-19	0.004	-186.996	-18.7	-74.8
LSLData-17-04-11	-9	-15	-14	-15	-15	-12	-14	-12	-13	-11	0.004	-129.996	-13	-52
LSLData-17-04 2-5	-18	-15	-18	-17	-9	-15	-11	-8	-13	-12	0.004	-135.996	-13.6	-54.4
LSLData-17-04-15	-7	-11	-11	-18	-13	-4	-14	-12	-13	None	0.004	-102.996	-11.444	-45.78
LSLData-17-04 2-13	-20	-22	-19	-15	-19	-16	-27	-19	-19	-21	0.004	-196.996	-19.7	-78.8
LSLData-17-04 2-8	-20	-21	-25	-18	-15	-23	-23	-20	-22	-20	0.004	-206.996	-20.7	-82.8
LSLData-17-04-7	-12	-11	-11	-13	-15	-19	-17	-14	-17	-20	0.004	-148.996	-14.9	-59.6
LSLData-17-04-9	-13	-14	-9	-20	-13	-12	-14	-9	-19	-16	0.004	-138.996	-13.9	-55.6
LSLData-17-04 2-2	-15	-12	-10	-12	-12	-12	-18	-16	-13	-9	0.004	-128.996	-12.9	-51.6
LSLData-17-04 2-4	-17	-19	-19	-17	-22	-15	-19	-12	-19	-16	0.004	-174.996	-17.5	-70
LSLData-17-04-14	-14	-11	-9	-15	-9	-8	-13	-18	-13	-10	0.004	-119.996	-12	-48
LSLData-17-04 2-7	-13	-19	-18	-17	-16	-8	-15	-20	-19	-18	0.004	-162.996	-16.3	-65.2
LSLData-17-04 2-15	-22	-15	-22	-13	-17	-11	-17	-15	-18	-15	0.004	-164.996	-16.5	-66
LSLData-17-04 2-9	-18	-15	-11	-18	-13	-19	-10	-15	-23	-18	0.004	-159.996	-16	-64
LSLData-17-04 2-10	-13	-9	-11	-13	-9	-14	-11	-8	-11	-9	0.004	-107.996	-10.8	-43.2
LSLData-17-04 2-12	-16	-16	-16	-16	-17	-18	-23	-20	-15	-17	0.004	-173.996	-17.4	-69.6
LSLData-17-04-3	-16	-20	-19	-21	-26	-25	-23	-30	-24	-26	0.004	-229.996	-23	-92

Table 17: Recording of blink offset between the EMG and Eye Tracker data where excess samples are cut off at the end. Each test had the subject blink 10 times. The columns from 0-9 is the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.

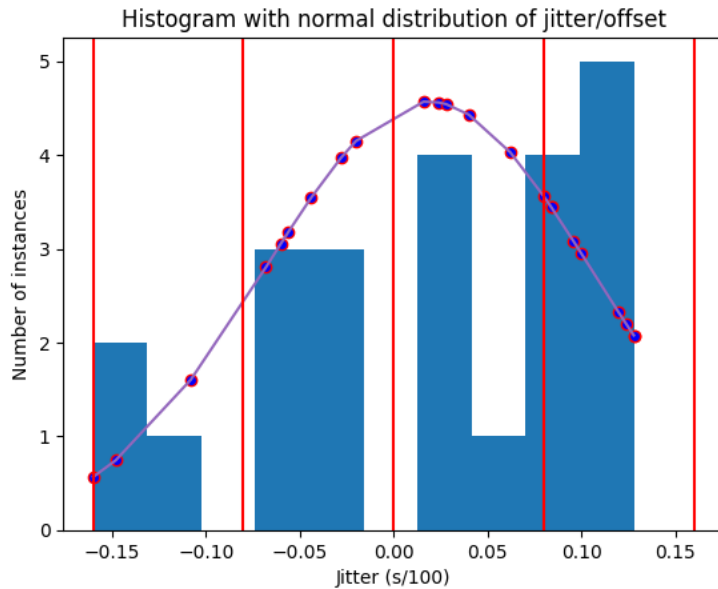


Figure 46: Histogram with normal distribution of offset computed when cutting excess samples off at the end. X-axis is $\frac{1}{100}$ seconds. Y-axis is number of instances with given index. Red lines splits the different indexes of 8.3 ms which is the frequency of the eye tracker (120 Hz).

Blink Cut Start	Blink 1	Blink 2	Blink 3	Blink 4	Blink 5	Blink 6	Blink 7	Blink 8	Blink 9	Blink 10	Time Between Indexes (s)	Sum (indexes)	Average (indexes)	Average time (ms)
LSLData-17-04 2-1	-13	-20	-19	-19	-20	-21	-14	-20	-21	-20	0.004	-186.996	-18.7	-74.8
LSLData-17-04 2-6	-6	-6	-4	-10	-8	-9	-5	-9	-5	-3	0.004	-64.996	-6.5	-26
LSLData-17-04 2-14	-14	-17	-13	-14	-15	-13	-13	-14	-13	-20	0.004	-145.996	-14.6	-58.4
LSLData-17-04 8	-10	-10	-10	-10	-16	-2	-6	-9	-8	-7	0.004	-87.996	-8.8	-35.2
LSLData-17-04 2-11	-19	-18	-15	-18	-25	-16	-23	-19	-27	-21	0.004	-200.996	-20.1	-80.4
LSLData-17-04 10	-14	-16	-13	-9	-13	-6	-13	-12	-13	-15	0.004	-123.996	-12.4	-49.6
LSLData-17-04 2-3	-18	-15	-21	-24	-19	-17	-25	-19	-19	-20	0.004	-196.996	-19.7	-78.8
LSLData-17-04 11	-8	-14	-13	-14	-14	-11	-13	-11	-12	-10	0.004	-119.996	-12	-48
LSLData-17-04 2-5	-24	-21	-24	-23	-15	-21	-17	-14	-19	-18	0.004	-195.996	-19.6	-78.4
LSLData-17-04 15	-15	-19	-19	-26	-21	-12	-22	-20	-21	None	0.004	-174.996	-19.444	-77.77
LSLData-17-04 2-13	-19	-21	-18	-14	-18	-15	-26	-18	-18	-20	0.004	-186.996	-18.7	-74.8
LSLData-17-04 2-8	-11	-12	-16	-9	-6	-14	-14	-11	-13	-11	0.004	-116.996	-11.7	-46.8
LSLData-17-04 7	-7	-6	-6	-8	-10	-14	-12	-9	-12	-15	0.004	-98.996	-9.9	-39.6
LSLData-17-04 9	-12	-13	-8	-19	-12	-11	-13	-8	-18	-15	0.004	-128.996	-12.9	-51.6
LSLData-17-04 2-2	-18	-15	-13	-15	-15	-15	-21	-19	-16	-12	0.004	-158.996	-15.9	-63.6
LSLData-17-04 2-4	-14	-16	-16	-14	-19	-12	-16	-9	-16	-13	0.004	-144.996	-14.5	-58
LSLData-17-04 14	-19	-16	-14	-20	-14	-13	-18	-23	-18	-15	0.004	-169.996	-17	-68
LSLData-17-04 2-7	-11	-17	-16	-15	-14	-6	-13	-18	-17	-16	0.004	-142.996	-14.3	-57.2
LSLData-17-04 2-15	-24	-17	-24	-15	-19	-13	-19	-17	-20	-17	0.004	-184.996	-18.5	-74
LSLData-17-04 2-9	-14	-11	-7	-14	-9	-15	-6	-11	-19	-14	0.004	-119.996	-12	-48
LSLData-17-04 2-10	-19	-15	-17	-19	-15	-20	-17	-14	-17	-15	0.004	-167.996	-16.8	-67.2
LSLData-17-04 2-12	-17	-17	-17	-17	-18	-19	-24	-21	-16	-18	0.004	-183.996	-18.4	-73.6
LSLData-17-04 3	-6	-10	-9	-11	-16	-15	-13	-20	-14	-16	0.004	-129.996	-13	-52

Table 18: Recording of blink offset between the EMG and Eye Tracker data where all of the excess samples are cut at the start. Each test had the subject blink 10 times. The columns from 0-9 is the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.

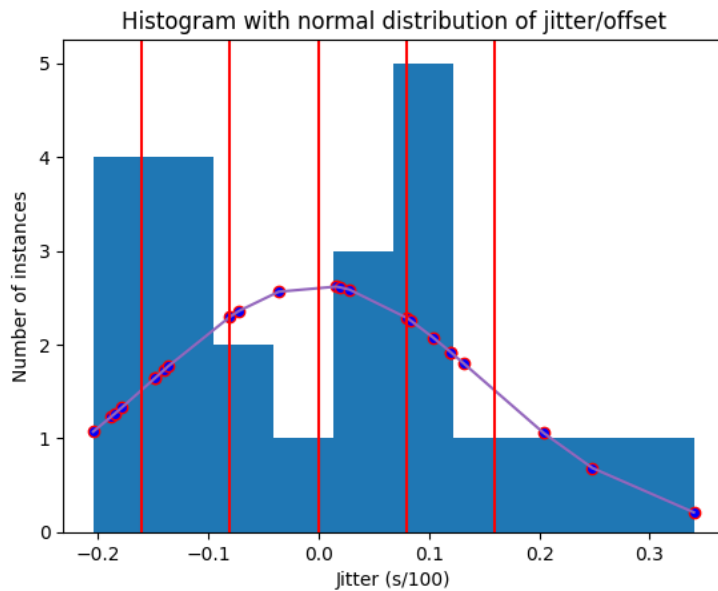


Figure 47: Histogram with normal distribution of offset computed when cutting excess samples off at the start. X-axis is $\frac{1}{100}$ seconds. Y-axis is number of instances with given index. Red lines splits the different indexes of 8.3 ms which is the frequency of the eye tracker (120 Hz).

Blink Cut Start 1/3	Blink 1	Blink 2	Blink 3	Blink 4	Blink 5	Blink 6	Blink 7	Blink 8	Blink 9	Blink 10	Time Between Indexes (s)	Sum (indexes)	Average (indexes)	Average time (ms)
LSLData-17-04 2-1	-11	-18	-17	-17	-18	-19	-12	-18	-19	-18	0.004	-166.996	-16.7	-66.8
LSLData-17-04 2-6	-13	-13	-11	-17	-15	-16	-12	-16	-12	-10	0.004	-134.996	-13.5	-54
LSLData-17-04 2-14	-13	-16	-12	-13	-14	-12	-12	-13	-12	-19	0.004	-135.996	-13.6	-54.4
LSLData-17-04 8	-14	-14	-14	-14	-20	-6	-10	-13	-12	-11	0.004	-127.996	-12.8	-51.2
LSLData-17-04 2-11	-16	-15	-12	-15	-22	-13	-20	-16	-24	-18	0.004	-170.996	-17.1	-68.4
LSLData-17-04-10	-17	-19	-16	-12	-16	-9	-16	-15	-16	-18	0.004	-153.996	-15.4	-61.6
LSLData-17-04 2-3	-17	-14	-20	-23	-18	-16	-24	-18	-18	-19	0.004	-186.996	-18.7	-74.8
LSLData-17-04-11	-9	-15	-14	-15	-15	-12	-14	-12	-13	-11	0.004	-129.996	-13	-52
LSLData-17-04 2-5	-20	-17	-20	-19	-11	-17	-13	-10	-15	-14	0.004	-155.996	-15.6	-62.4
LSLData-17-04-15	-10	-14	-14	-21	-16	-7	-17	-15	-16	None	0.004	-129.996	-14.444	-57.77
LSLData-17-04 2-13	-20	-22	-19	-15	-19	-16	-27	-19	-19	-21	0.004	-196.996	-19.7	-78.8
LSLData-17-04 2-8	-17	-18	-22	-15	-12	-20	-20	-17	-19	-17	0.004	-176.996	-17.7	-70.8
LSLData-17-04-7	-10	-9	-9	-11	-13	-17	-15	-12	-15	-18	0.004	-128.996	-12.9	-51.6
LSLData-17-04-9	-13	-14	-9	-20	-13	-12	-14	-9	-19	-16	0.004	-138.996	-13.9	-55.6
LSLData-17-04 2-2	-16	-13	-11	-13	-13	-13	-19	-17	-14	-10	0.004	-138.996	-13.9	-55.6
LSLData-17-04 2-4	-16	-18	-18	-16	-21	-14	-18	-11	-18	-15	0.004	-164.996	-16.5	-66
LSLData-17-04-14	-16	-13	-11	-17	-11	-10	-15	-20	-15	-12	0.004	-139.996	-14	-56
LSLData-17-04 2-7	-12	-18	-17	-16	-15	-7	-14	-19	-18	-17	0.004	-152.996	-15.3	-61.2
LSLData-17-04 2-15	-23	-16	-23	-14	-18	-12	-18	-16	-19	-16	0.004	-174.996	-17.5	-70
LSLData-17-04 2-9	-17	-14	-10	-17	-12	-18	-9	-14	-22	-17	0.004	-149.996	-15	-60
LSLData-17-04 2-10	-15	-11	-13	-15	-11	-16	-13	-10	-13	-11	0.004	-127.996	-12.8	-51.2
LSLData-17-04 2-12	-16	-16	-16	-16	-17	-18	-23	-20	-15	-17	0.004	-173.996	-17.4	-69.6
LSLData-17-04-3	-13	-17	-16	-18	-23	-22	-20	-27	-21	-23	0.004	-199.996	-20	-80

Table 19: Recording of blink offset between the EMG and Eye Tracker data where a third of the excess samples are cut at the start, and the rest at the end. Each test had the subject blink 10 times. The columns from 0-9 is the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.

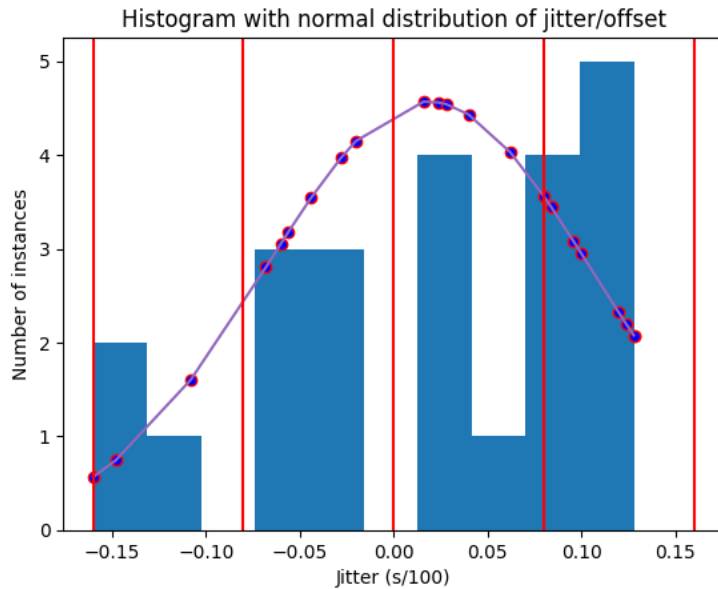


Figure 48: Histogram with normal distribution of offset computed when cutting a third of excess samples off at the end. X-axis is $\frac{1}{100}$ seconds. Y-axis is number of instances with given index. Red lines splits the different indexes of 8.3 ms which is the frequency of the eye tracker (120 Hz).

Blink Cut Start %	Blink 1	Blink 2	Blink 3	Blink 4	Blink 5	Blink 6	Blink 7	Blink 8	Blink 9	Blink 10	Time Between Indexes (s)	Sum (indexes)	Average (indexes)	Average time (ms)
LSLData-17-04 2-1	-11	-18	-17	-17	-18	-19	-12	-18	-19	-18	0.004	-166.996	-16.7	-66.8
LSLData-17-04 2-6	-14	-14	-12	-18	-16	-17	-13	-17	-13	-11	0.004	-144.996	-14.5	-58
LSLData-17-04 2-14	-12	-15	-11	-12	-13	-11	-11	-12	-11	-18	0.004	-125.996	-12.6	-50.4
LSLData-17-04 8	-14	-14	-14	-14	-20	-6	-10	-13	-12	-11	0.004	-127.996	-12.8	-51.2
LSLData-17-04 2-11	-15	-14	-11	-14	-21	-12	-19	-15	-23	-17	0.004	-160.996	-16.1	-64.4
LSLData-17-04 10	-18	-20	-17	-13	-17	-10	-17	-16	-17	-19	0.004	-163.996	-16.4	-65.6
LSLData-17-04 2-3	-17	-14	-20	-23	-18	-16	-24	-18	-18	-19	0.004	-186.996	-18.7	-74.8
LSLData-17-04 11	-9	-15	-14	-15	-15	-12	-14	-12	-13	-11	0.004	-129.996	-13	-52
LSLData-17-04 2-5	-20	-17	-20	-19	-11	-17	-13	-10	-15	-14	0.004	-155.996	-15.6	-62.4
LSLData-17-04 15	-9	-13	-13	-20	-15	-6	-16	-14	None		0.004	-120.996	-13.444	-53.77
LSLData-17-04 2-13	-20	-22	-19	-15	-19	-16	-27	-19	-19	-21	0.004	-196.996	-19.7	-78.8
LSLData-17-04 2-8	-18	-19	-23	-16	-13	-21	-21	-18	-20	-18	0.004	-186.996	-18.7	-74.8
LSLData-17-04 7	-11	-10	-10	-12	-14	-18	-16	-13	-16	-19	0.004	-138.996	-13.9	-55.6
LSLData-17-04 9	-13	-14	-9	-20	-13	-12	-14	-9	-19	-16	0.004	-138.996	-13.9	-55.6
LSLData-17-04 2-2	-16	-13	-11	-13	-13	-13	-19	-17	-14	-10	0.004	-138.996	-13.9	-55.6
LSLData-17-04 2-4	-16	-18	-18	-16	-21	-14	-18	-11	-18	-15	0.004	-164.996	-16.5	-66
LSLData-17-04 14	-15	-12	-10	-16	-10	-9	-14	-19	-14	-11	0.004	-129.996	-13	-52
LSLData-17-04 2-7	-13	-19	-18	-17	-16	-8	-15	-20	-19	-18	0.004	-162.996	-16.3	-65.2
LSLData-17-04 2-15	-22	-15	-22	-13	-17	-11	-17	-15	-18	-15	0.004	-164.996	-16.5	-66
LSLData-17-04 2-9	-17	-14	-10	-17	-12	-18	-9	-14	-22	-17	0.004	-149.996	-15	-60
LSLData-17-04 2-10	-15	-11	-13	-15	-11	-16	-13	-10	-13	-11	0.004	-127.996	-12.8	-51.2
LSLData-17-04 2-12	-16	-16	-16	-16	-17	-18	-23	-20	-15	-17	0.004	-173.996	-17.4	-69.6
LSLData-17-04 3	-14	-18	-17	-19	-24	-23	-21	-28	-22	-24	0.004	-209.996	-21	-84

Table 20: Recording of blink offset between the EMG and Eye Tracker data where a fourth of the excess samples are cut at the start, and the rest at the end. Each test had the subject blink 10 times. The columns from 0-9 is the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.

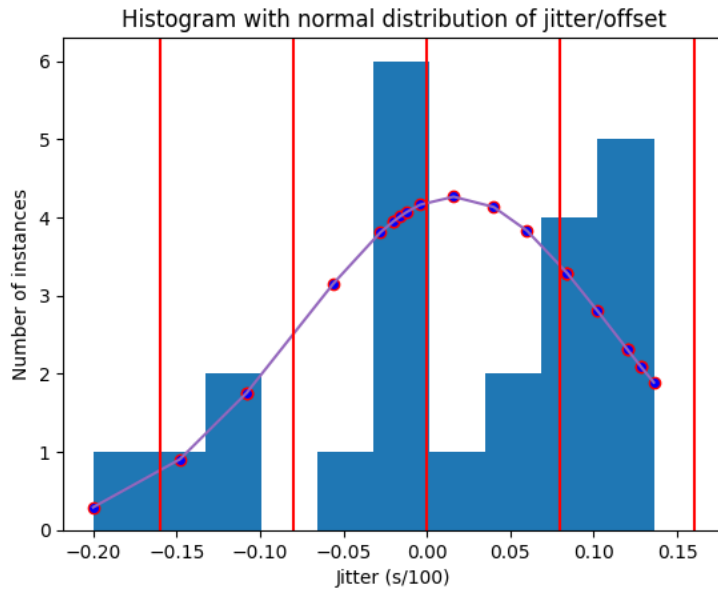


Figure 49: Histogram with normal distribution of offset computed when cutting a fourth of excess samples off at the end. X-axis is $\frac{1}{100}$ seconds. Y-axis is number of instances with given index. Red lines splits the different indexes of 8.3 ms which is the frequency of the eye tracker (120 Hz).

Blink Cut Start 1/5	Blink 1	Blink 2	Blink 3	Blink 4	Blink 5	Blink 6	Blink 7	Blink 8	Blink 9	Blink 10	Time Between Indexes (s)	Sum (indexes)	Average (indexes)	Average time (ms)
LSLData-17-04 2-1	-11	-18	-17	-17	-18	-19	-12	-18	-19	-18	0.004	-166.996	-16.7	-66.8
LSLData-17-04 2-6	-15	-15	-13	-19	-17	-18	-14	-18	-14	-12	0.004	-154.996	-15.5	-62
LSLData-17-04 2-14	-12	-15	-11	-12	-13	-11	-11	-12	-11	-18	0.004	-125.996	-12.6	-50.4
LSLData-17-04-8	-15	-15	-15	-15	-21	-7	-11	-14	-13	-12	0.004	-137.996	-13.8	-55.2
LSLData-17-04 2-11	-15	-14	-11	-14	-21	-12	-19	-15	-23	-17	0.004	-160.996	-16.1	-64.4
LSLData-17-04-10	-18	-20	-17	-13	-17	-10	-17	-16	-17	-19	0.004	-163.996	-16.4	-65.6
LSLData-17-04 2-3	-17	-14	-20	-23	-18	-16	-24	-18	-18	-19	0.004	-186.996	-18.7	-74.8
LSLData-17-04-11	-9	-15	-14	-15	-15	-12	-14	-12	-13	-11	0.004	-129.996	-13	-52
LSLData-17-04 2-5	-19	-16	-19	-18	-10	-16	-12	-9	-14	-13	0.004	-145.996	-14.6	-58.4
LSLData-17-04-15	-9	-13	-13	-20	-15	-6	-16	-14	-15	None	0.004	-120.996	-13.444	-53.77
LSLData-17-04 2-13	-20	-22	-19	-15	-19	-16	-27	-19	-19	-21	0.004	-196.996	-19.7	-78.8
LSLData-17-04 2-8	-18	-19	-23	-16	-13	-21	-21	-18	-20	-18	0.004	-186.996	-18.7	-74.8
LSLData-17-04-7	-11	-10	-10	-12	-14	-18	-16	-13	-16	-19	0.004	-138.996	-13.9	-55.6
LSLData-17-04-9	-13	-14	-9	-20	-13	-12	-14	-9	-19	-16	0.004	-138.996	-13.9	-55.6
LSLData-17-04 2-2	-16	-13	-11	-13	-13	-13	-19	-17	-14	-10	0.004	-138.996	-13.9	-55.6
LSLData-17-04 2-4	-16	-18	-18	-16	-21	-14	-18	-11	-18	-15	0.004	-164.996	-16.5	-66
LSLData-17-04-14	-15	-12	-10	-16	-10	-9	-14	-19	-14	-11	0.004	-129.996	-13	-52
LSLData-17-04 2-7	-13	-19	-18	-17	-16	-8	-15	-20	-19	-18	0.004	-162.996	-16.3	-65.2
LSLData-17-04 2-15	-22	-15	-22	-13	-17	-11	-17	-15	-18	-15	0.004	-164.996	-16.5	-66
LSLData-17-04 2-9	-17	-14	-10	-17	-12	-18	-9	-14	-22	-17	0.004	-149.996	-15	-60
LSLData-17-04 2-10	-14	-10	-12	-14	-10	-15	-12	-9	-12	-10	0.004	-117.996	-11.8	-47.2
LSLData-17-04 2-12	-16	-16	-16	-16	-17	-18	-23	-20	-15	-17	0.004	-173.996	-17.4	-69.6
LSLData-17-04-3	-14	-18	-17	-19	-24	-23	-21	-28	-22	-24	0.004	-209.996	-21	-84

Table 21: Recording of blink offset between the EMG and Eye Tracker data where a fifth of the excess samples are cut at the start, and the rest at the end. Each test had the subject blink 10 times. The columns from 0-9 is the index of the blink. The offset of each blink is in indexes, which means each blink is offset = index times time between indexes. These indexes are then summarized and the average offset is found for each test, both in seconds and indexes. Also includes mean offset in seconds and indexes from all tests together.

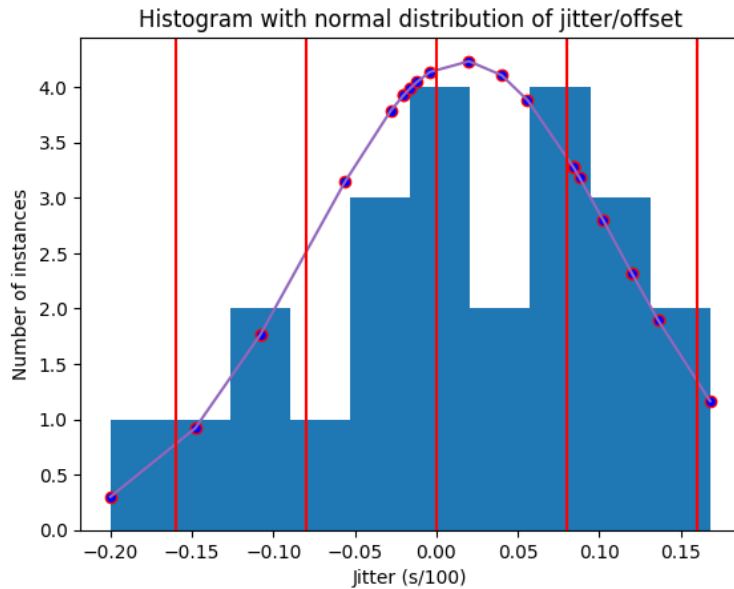


Figure 50: Histogram with normal distribution of offset computed when cutting a fifth of excess samples off at the end. X-axis is $\frac{1}{100}$ seconds. Y-axis is number of instances with given index. Red lines splits the different indexes of 8.3 ms which is the frequency of the eye tracker (120 Hz).

C BCI Speller results initial test

Subject 1

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	2	2	2	2	2	2
5 Hz	6	6	6	6	6	6
5.5 Hz	2	2	2	2	2	2
6 Hz	3	3	3	3	3	3
7 Hz	1	1	1	1	0	1
7.4 Hz	3	3	3	3	3	3
Total	17	17	17	17	16	17
Accuracy	100.00%	100.00%	100.00%	100.00%	94.12%	

Table 22: Shows the accuracy of Subject 1 ranging from 6 seconds to 2 seconds in the initial testing of the speller.

Subject 2

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	2	2	3	1	2	2
5 Hz	5	3	1	2	4	6
5.5 Hz	1	1	0	0	2	2
6 Hz	3	1	0	1	0	3
7 Hz	1	1	1	1	1	1
7.4 Hz	3	2	0	1	0	3
Total	15	10	5	6	9	17
Accuracy	88.24%	58.82%	29.41%	35.29%	52.94%	

Table 23: Shows the accuracy of Subject 2 ranging from 6 seconds to 2 seconds in the initial testing of the speller.

Subject 3

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	2	2	1	2	2	2
5 Hz	5	3	4	2	3	6
5.5 Hz	2	2	1	1	0	2
6 Hz	3	3	1	2	2	3
7 Hz	0	0	1	0	1	1
7.4 Hz	2	1	1	2	0	3
Total	14	11	9	9	8	17
Accuracy	82.35%	64.71%	52.94%	52.94%	47.06%	

Table 24: Shows the accuracy of Subject 3 ranging from 6 seconds to 2 seconds in the initial testing of the speller.

Subject 4

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	2	2	2	1	2	2
5 Hz	4	2	1	2	2	6
5.5 Hz	0	1	1	0	1	2
6 Hz	0	1	0	1	0	3
7 Hz	0	0	1	0	0	1
7.4 Hz	1	0	1	0	0	3
Total	7	6	6	4	5	17
Accuracy	41.18%	35.29%	35.29%	23.53%	29.41%	

Table 25: Shows the accuracy of Subject 4 ranging from 6 seconds to 2 seconds in the initial testing of the speller.

Subject 5

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	2	2	1	2	2	2
5 Hz	6	6	4	0	3	6
5.5 Hz	2	1	2	1	1	2
6 Hz	2	2	2	0	0	3
7 Hz	0	1	1	0	0	1
7.4 Hz	2	3	1	0	0	3
Total	14	15	11	3	6	17
Accuracy	82.35%	88.24%	64.71%	17.65%	35.29%	

Table 26: Shows the accuracy of Subject 5 ranging from 6 seconds to 2 seconds in the initial testing of the speller.

Subject 6

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	1	2	2	2	1	2
5 Hz	6	4	3	6	2	6
5.5 Hz	2	2	2	1	1	2
6 Hz	2	2	1	1	0	3
7 Hz	1	0	0	0	0	1
7.4 Hz	1	0	1	0	1	3
Total	13	10	9	10	5	17
Accuracy	76.47%	58.82%	52.94%	58.82%	29.41%	

Table 27: Shows the accuracy of Subject 6 ranging from 6 seconds to 2 seconds in the initial testing of the speller.

Subject 7

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	2	2	2	2	2	2
5 Hz	6	6	5	4	2	6
5.5 Hz	2	2	2	2	2	2
6 Hz	2	3	2	2	0	3
7 Hz	1	1	0	0	0	1
7.4 Hz	2	3	1	2	1	3
Total	15	17	12	12	7	17
Accuracy	88.24%	100.00%	70.59%	70.59%	41.18%	

Table 28: Shows the accuracy of Subject 7 ranging from 6 seconds to 2 seconds in the initial testing of the speller.

D BCI Speller results retest

Subject 1

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	2	2	2	2	2	2
5 Hz	6	6	6	6	6	6
5.5 Hz	2	2	2	2	2	2
6 Hz	3	3	3	3	3	3
7 Hz	1	1	1	1	1	1
7.4 Hz	3	3	3	3	2	3
Total	17	17	17	17	14	17
Accuracy	100.00%	100.00%	100.00%	100.00%	82.35%	

Table 29: Shows the accuracy of Subject 1 ranging from 6 seconds to 2 seconds in the retesting of the speller.

Subject 2

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	1	2	2	0	0	2
5 Hz	3	2	2	3	4	6
5.5 Hz	1	0	1	0	1	2
6 Hz	0	0	1	0	0	3
7 Hz	0	0	0	0	0	1
7.4 Hz	3	1	0	1	0	3
Total	8	5	6	4	5	17
Accuracy	47.06%	29.41%	35.29%	23.53%	29.41%	

Table 30: Shows the accuracy of Subject 2 ranging from 6 seconds to 2 seconds in the retesting of the speller.

Subject 3

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	2	2	1	2	2	2
5 Hz	2	4	1	2	5	6
5.5 Hz	1	1	0	0	0	2
6 Hz	1	0	0	1	1	3
7 Hz	0	0	0	0	1	1
7.4 Hz	1	0	0	0	0	3
Total	7	7	2	5	9	17
Accuracy	41.18%	41.18%	11.76%	29.41%	52.94%	

Table 31: Shows the accuracy of Subject 3 ranging from 6 seconds to 2 seconds in the retesting of the speller.

Subject 4

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	1	2	2	1	1	2
5 Hz	1	2	3	1	0	6
5.5 Hz	0	0	0	1	1	2
6 Hz	1	1	1	0	0	3
7 Hz	1	1	0	0	0	1
7.4 Hz	0	0	1	0	0	3
Total	4	6	7	3	2	17
Accuracy	23.53%	35.29%	41.18%	17.65%	11.76%	

Table 32: Shows the accuracy of Subject 4 ranging from 6 seconds to 2 seconds in the retesting of the speller.

Subject 5

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	2	1	2	2	2	2
5 Hz	1	2	6	5	4	6
5.5 Hz	0	0	2	1	2	2
6 Hz	0	0	2	2	0	3
7 Hz	0	0	1	0	0	1
7.4 Hz	0	0	1	2	1	3
Total	3	3	14	12	9	17
Accuracy	17.65%	17.65%	82.35%	70.59%	52.94%	

Table 33: Shows the accuracy of Subject 5 ranging from 6 seconds to 2 seconds in the retesting of the speller.

Subject 6

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	0	1	0	2	2	2
5 Hz	1	1	2	0	0	6
5.5 Hz	0	0	0	0	0	2
6 Hz	0	0	1	0	0	3
7 Hz	0	0	0	0	0	1
7.4 Hz	1	0	0	0	0	3
Total	2	2	3	2	2	17
Accuracy	11.76%	11.76%	17.65%	11.76%	11.76%	

Table 34: Shows the accuracy of Subject 6 ranging from 6 seconds to 2 seconds in the retesting of the speller.

Subject 7

freq\sec	6 s	5 s	4 s	3 s	2 s	Total symbols
4 Hz	2	1	2	1	1	2
5 Hz	3	2	4	5	3	6
5.5 Hz	2	1	1	0	1	2
6 Hz	3	1	2	1	1	3
7 Hz	1	1	1	1	0	1
7.4 Hz	2	1	1	1	0	3
Total	13	7	11	9	6	17
Accuracy	76.47%	41.18%	64.71%	52.94%	35.29%	

Table 35: Shows the accuracy of Subject 7 ranging from 6 seconds to 2 seconds in the retesting of the speller.

E Preproject Plan

Pre-project plan

Olav F. P. Larsen, William G. Tresselt

May 5, 2023

Version 2.0

Revision History

Date	Version	Description	Author
19.01.2023	1.0	First draft	William Tresselt Olav Larsen
25.01.2023	1.1	Adjusted methodology	William Tresselt Olav Larsen
25.01.2023	2.0	Final pre-project plan	William Tresselt Olav Larsen

Contents

1 Goals and constraints

1.1	Background information
1.2	Project description and result goals
1.2.1	Performance / Process goals
1.2.2	Outcome goals
1.3	Constraints

2 Organisation

3 Implementation

3.1	Main activities
3.2	Milestones

4 Follow-up and quality assurance

4.1	Quality assurance
4.2	Reporting

5 Risk assessment

Bibliography

6 Appendix

6.1	Address list
6.2	Collaboration Agreement

1 Goals and constraints

1.1 Background information

The group initially discussed a project with DIPS AS, but NTNU had a more exciting project. When the projects gathered by NTNU were introduced, the group members sat down separately and read all their descriptions. The members noted which ones they thought were interesting and proceeded to compare. A project was considered if both members marked the same one. After gathering the relevant alternatives in a list, they were sorted by which seemed the most interesting and special. Only the top three were considered. At a later time, the group decided on a final order for the top three projects and submitted them. At the top of the list was the project about EEG and Eye Tracking.

1.2 Project description and result goals

The project is to combine eye tracking and EEG for a VR game. Initially, the group is going to set up, and synchronize the different pipelines from the eye tracking and the EEG. Students are expected to set up a functional pipeline, explore the accuracy of timing and jitter of the signals from the two data streams (EEG/eye tracking), as well as have some preliminary testing on at least a couple of persons to extract descriptive statistics of at least one biomarker. The students should also decide what software to use

Project description from original document: In this study, the students will combine EEG and eye tracking for a VR game. For a comprehensive analysis of bio-signals tracked by various sensors, eye and brain data need to be time-synchronized with VR contents. The pipelines that could be used are Neuropype or Timeflux. They should work with our EEG, Unity out of the box. Students are expected to set up a functional pipeline, explore the accuracy of timing and jitter of the signals from the two data streams (EEG/eye tracking), as well as have some preliminary testing on at least a couple of persons to extract descriptive statistics of at least one biomarker.

1.2.1 Performance / Process goals

- Deliver obligatory tasks on time.
- Achieve grade A.

-
- Put in at least five hundred hours of work each.
 - Complete the project at least one week before the deadline. Only write the report the final week and dedicate the last 48 hours to proofreading.
 - Publish the final report.
 - Create a product that can be used as a baseline for further research.
 - Utilize Design Science Research.

1.2.2 Outcome goals

- Collaborate efficiently.
- Communicate efficiently.
- Increase knowledge of EEG, Eye tracking, VR, and scientific methods.

Note that a more detailed list of how to achieve each goal can be found in the collaboration agreement.

1.3 Constraints

There are several limitations to the project. The team needs access to an EEG, the NTNU Vizlab, and eye-tracking technology. Since many groups are working at the NTNU Vizlab this semester, sharing of equipment may be required. Note that it is possible to book equipment when needed. The group is also given a monitor and a workstation in the Vizlab.

2 Organisation

The main participants involved in the project are NTNU and the group's supervisors Alexander Holt and Emanuel Alexander Lorenz.

3 Implementation

3.1 Main activities

Meetings

During meetings, the group members and supervisors should gather and discuss relevant topics like project progression and roadblocks. If at least one of the two supervisors can participate, that is sufficient. The meetings are central for progression by helping guide the group in the right direction. A meeting should be initiated by a group member and sent to relevant participants with an agenda. During meetings, at least one group member should take notes of relevant information being discussed. For the time being, there is a verbal agreement between the group members and supervisors about the frequency of meetings, which is once every two weeks.

Documentation

Multiple documents are to be written during the project, including a collaboration agreement, timetables containing weekly reports, a main report summarising everything the group has done and learned during the project, and many more. A substantial part of the process demands documentation, meaning it has to be written. Both group members should take part in producing all documents to assure quality and avoid disagreements about how they turn out. Each document is to be written in a timely manner, e.g. the notes related to a meeting should be written as the meeting is being held and revised shortly after and the main report should be worked on continuously during the entire process.

Research

A large part of the work method "Design Science Research" is gathering relevant information by doing research. The group will look for resources on the internet through sites like Oria.no, NTNU Open, and Google Scholar, preferably ones that are peer-reviewed, and are otherwise trustworthy and critical of how they acquired their knowledge. To more efficiently collect information, the members should look for reading material individually, but both have to agree on which ones to actually use after careful consideration. The nature of the group's work will mainly be scientific, which means the information gathered must be of high quality.

The act of information gathering should be continuous and all potential sources should be saved to Zotero allowing for easy access to relevant information.

Create a pipeline for gathering EEG and eye tracking data

To gather data from the EEG and the eye tracker it is necessary to create pipelines for each technology. It is suggested to use software that is compatible with Unity. The two main software suggested were NeuroPype and TimeFlux. The group should find the most suitable software for their use, and create pipelines.

Synchronization of EEG and eye tracking data

The main part of the project is to create a synchronized pipeline between the EEG and the eye tracker. This could be a challenge since the eye tracker and EEG pipelines should be synchronized, such that is no more than a single millisecond difference, as stated by the supervisors. The EEG can be negatively affected by ocular artifacts, and the eye tracker will be used to control these artifacts, and make unusable data, usable (Hutton 2019).

Gather data from different test subjects

When conducting a scientific experiment different test subjects are needed to ensure that the scientific findings are valid. Therefore it is important to have a set of test subjects. Note that it is demanding to test the synchronization on a large group of individuals, therefore the number of test subjects is set to at least two, but this is the absolute minimum number of subjects. After synchronizing the EEG and eye tracking, sourcing subjects can commence. The data produced should be stored in a proper manner, and should preferably be backed up.

3.2 Milestones

Date	Milestone
27.01.2023	Deliver pre-project plan
24.02.2023	Setup of data gathering pipelines complete
27.03.2023- 02.04.2023	Poster presentation of research question(s)
06.04.2023	Managed to sync eye tracking and EEG data
10.04.2023	Gathered test subjects for data collection
15.05.2023	Self-imposed deadline for the final report and other documentation
22.05.2023	Absolute deadline for the final report and other documentation
26.05.2023	Digital presentation of the project

4 Follow-up and quality assurance

4.1 Quality assurance

To ensure the quality of the work conducted by the group, it should be revised a minimum of three times, extensively documented, and tested. The group must ensure that all parts of the projects are of high quality. The synchronization of the different pipelines should be thoroughly tested to ensure their sample rates are within one ms of each other to minimize errors. The group should follow scientific practices to ensure the findings and products are truly scientific. All documentation should be revised a minimum of three times and potential programs created by the group should be extensively documented and tested.

4.2 Reporting

The group should have a meeting every other week with the supervisors to report the group's current status, difficulties, or any other topics and or questions. The supervisors are also frequently at the NTNU Vizlab, making it simple to ask them questions if uncertainties or technical difficulties occur or update them on progress. Each meeting should be reported for later use and documentation.

5 Risk assessment

	Risk	Description
1	Few members	The group consists of few members and is vulnerable. If one member is gone, the workload on the other increases significantly.
2	Disagreement within the group that leads to conflicts	A disagreement might impact the workflow and the quality of work.
3	Underestimate time usage	If the group thinks they have more time than they actually do, it will lead to lots of work being moved to later stages of the project. If the remaining work is too great, they will not be guaranteed to finish in time.
4	Failing to follow work methodology	If the group does not use proper work methodology, the knowledge produced might be of less value.
5	A group member leaves or drops out	If a group member leaves or drops out the group will be reduced to one person. This increases the workload significantly, and the final product will reflect this.
6	No or little support from supervisor	If the supervisors are not available, or only to a small degree, this might influence the project negatively.
7	Equipment unavailable	Since the NTNU Vizlab's equipment is shared between many different groups, the equipment might not be available at all times. This can lead to difficulties if the group is not able to perform necessary tests etc.

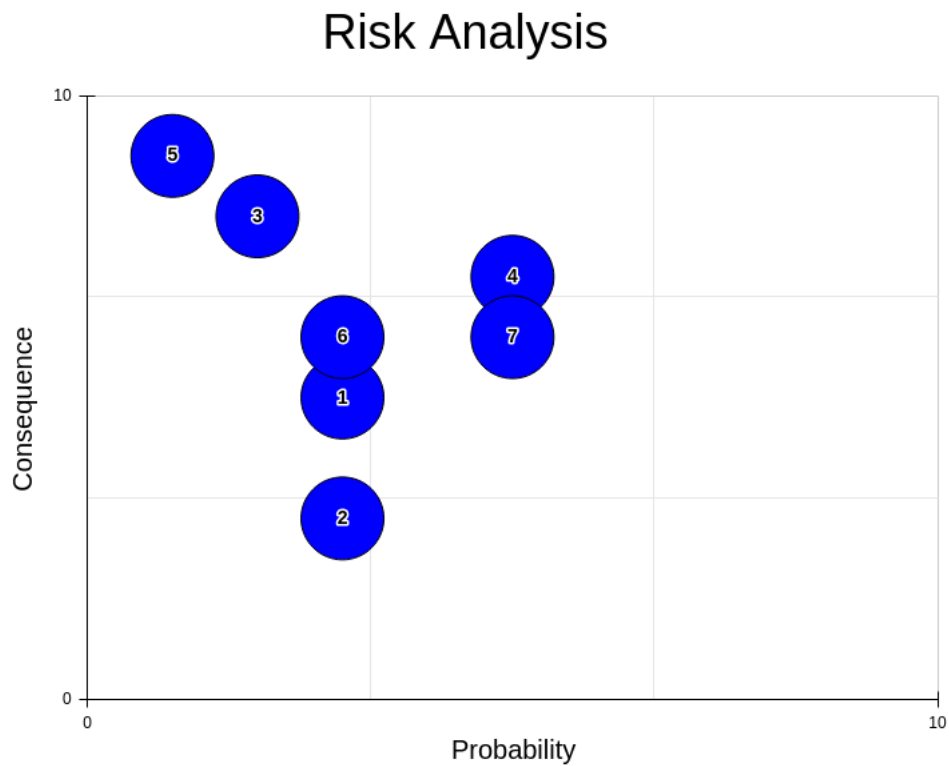


Figure 1: Risk analysis plotting probability and consequences

Prevention of risk

Risk factor 1: Few members

- Work consistently to prevent the build-up of unfinished work.

Risk factor 2: Disagreement within the group

- Make sure everyone follows the collaboration agreement.
- Talk about disagreements.
- Be willing to compromise.

Risk factor 3: Underestimate time usage

- Prioritize the different tasks.
- Always be aware of the time left, time spent, and deadlines.

Risk factor 4: Failing to follow work methodology

- Be aware of what is expected when following the methodology.
- Use the methodology continuously.

Risk factor 5: Group member leaves or drops out

- Work consistently to prevent the build-up of unfinished work.

Risk factor 6: No or little support from supervisors

- Talk to the supervisors about this issue.
- Avoid being overly dependent on their assistance.

Risk factor 7: Equipment unavailable

- Make sure to always book needed equipment.
- Tell supervisors that this is an issue.
- Ask the person who booked the equipment if possible to share.

Bibliography

Hutton, Sam (Apr. 2019). *Eye Tracking and EEG*. en-US. URL: <https://www.sr-research.com/eye-tracking-blog/background/eye-tracking-and-eeg/> (visited on Jan. 20, 2023).

6 Appendix

6.1 Address list

6.2 Collaboration Agreement

Address List

Name	Company	Address	Mail
Emanuel Alexander Lorenz	NTNU	Gløshaugen, 7034 Trondheim	emanuel.a.lorenz@ntnu.no
Alexander Holt	NTNU	Gløshaugen, 7034 Trondheim	alexander.holt@ntnu.no
Olav Finne Præsteng Larsen	NTNU	Urds veg 1, 7033, Trondheim	oflarsen@stud.ntnu.no
William Gjerberg Tresselt	NTNU	Munkegata 41, 7011 Trondheim	williagt@stud.ntnu.no

Collaboration Agreement

Members: Olav F. P. Larsen, William G. Tresselt

This collaboration agreement builds upon a set of typical goals, distribution of tasks, procedures, and guidelines for student interactions. The agreement is filled in with the group's own interpretations of what they think each section means and how they will achieve them.

Goals

Outcome goals

- Collaborate efficiently.
 - Meet frequently.
 - Be flexible in the sense that we are able to discuss potential problems, like disagreements, and be open to compromise.
- Communicate efficiently.
 - Be available 08-16 and preferably outside the given time frame.
 - Don't make assumptions and have a common understanding of what is being discussed.
- Increase knowledge of EEG, Eye tracking, VR, and scientific methods.
 - Research each topic thoroughly.
 - Test the equipment and use theory combined with practice.

Performance / Process goals

- Deliver obligatory tasks on time.
 - Finish and deliver each obligatory task at least 3 days before their deadline.

-
- Keep track of deadlines.
 - Achieve grade A.
 - Fulfill all formal requirements defined for the main report.
 - Keep a common thread through the main report.
 - Deliver obligatory tasks on time.
 - Get and use input from supervisors.
 - Work in a structured manner.
 - Put in at least five hundred hours of work each.
 - Keep track of time spent on the project.
 - Prioritize spending time working on the project.
 - Complete the project at least one week before the deadline. Only write the report the final week and dedicate the last 48 hours to proofreading.
 - Plan ahead and work consistently.
 - Publish the final report.
 - Be satisfied with the final project.
 - Create a product that can be used as a baseline for further research.
 - The group needs to create a well-tested and documented product that is user-friendly.
 - Utilize Scientific methodology (Exploratory Research)
 - Research the different parts of the projects
 - Use research to gather data from the pipelines, and synchronize
 - Test and validate the data synchronization

Roles and division of tasks

Olav F. P. Larsen

- Main meeting reporter.
- Responsible for documents.
- Responsible for quality assurance.
- Responsible for returning the equipment after use.

William G. Tresselt

- Responsible for meetings.
- Supporting meeting reporter.
- Responsible for documents.
- Responsible for quality assurance.

Procedure

Schedule meetings

Every other week, Tuesday 15:00 starting January 31st via Teams.

Notification in case of absence or other events

In case of absence, notify William via mail at williagt@stud.ntnu.no or phone at +47 93823508.

Document management

Every document should be written in LaTeX. Each document should be shared with the other member, and a list of different templates should be in the Teams channel.

When a document is finished it should also be uploaded to the Teams channel.

Submission of assignments

Both members are responsible for quality assurance and proofreading. Also, they are both responsible for meeting deadlines. Documents and other files produced by the group should be delivered with all members present.

Interaction

Attendance and preparation

Before meeting in general, agree on what should be prepared and when/where to meet, with a leeway of 5 minutes. The entire process should be dynamic to allow for more flexibility. Note that the members must work a set amount of hours and they should always have that fact in mind.

Presence and engagement

The group members should be engaged while working and should keep distractions to a minimum. There should be breaks to keep the members motivated during work time. Usage of phones for entertainment purposes during meetings is forbidden.

Disagreement / Breach of Contract

The members should reflect on the matter at hand. The members could also inquire with the supervisors if there is a disagreement. Violating the agreement should result in the offender baking a cake of the other member's choice.

Signatures

Olav F. P. Jansen

Figure 1: Olav F. P. Larsen

William G. Tresselt

Figure 2: William G. Tresselt

F Vision Document

Vision Document

Olav F. P. Larsen, William G. Tresselt

Version: 2.0
02.05.2023

Revision History

Date	Version	Description	Author
24.01.2023	1.0	First draft	Olav Larsen William Tresselt
06.03.2023	1.1	Second draft	Olav Larsen William Tresselt
02.05.2023	2.0	Finished document	Olav Larsen William Tresselt

Contents

1 Introduction

2 Summary of the problem and the product

- 2.1 Summary of the problem
- 2.2 Summary of the synchronization
- 2.3 Summary of the speller

3 Description of stakeholders and users

- 3.1 Summary of stakeholders
- 3.2 General summary of potential user groups
- 3.3 User environment
- 3.4 Summary of the potential user groups' needs
- 3.5 Alternatives of our product

4 Product overview

- 4.1 The product's role in the user environment
- 4.2 Prerequisites and dependencies

5 Functional attributes

6 Non-functional attributes and other requirements

Bibliography

1 Introduction

This document is meant as a proposition for the current vision of the group concerning the bachelor project about EEG and eye tracking. After the first phase of the project, which had a heavy focus on setting up a pipeline for synchronizing the two mentioned data collection devices, the group should use this newfound setup to create something. This something will be a brain-computer interface (BCI) speller, which later should be tested on a set of participants to measure its accuracy.

2 Summary of the problem and the product

2.1 Summary of the problem

Problems with	unsynchronized equipment
Affects	the researchers at the Vizlab or others in the same field
This results in	little to no advancement on the field of EEG and eye tracking
A good solution would be	to synchronize the equipment as accurately as possible to make it usable in research and other applications

2.2 Summary of the synchronization

Made for	researchers
that	needs data streams from EEG and eye tracker to be synchronized
The synchronized EEG and eye tracker	is a pipeline
that	results in synchronized equipment
Opposite of	not taking synchronization into consideration
Our product can offer	a guarantee that the equipment is synchronized

2.3 Summary of the speller

Made as	a prototype
to	showcase an application for the synchronized equipment
The BCI speller	is a spelling system
that	enables communication
Opposite of	not being able to communicate at all
Our product can offer	a way to allow certain individuals to express themselves more freely using letters and symbols

3 Description of stakeholders and users

3.1 Summary of stakeholders

Name	Description	Role during development
Emanuel Alexander Lorenz	Represents NTNU. Stakeholder because he is writing a PhD about EEG, and could use the result in his future research.	Supervisor
Alexander Holt	Represents NTNU, and is well versed in VR and eye tracking, and could use the result in future research.	Supervisor
Vizlab	The entirety of the Vizlab at NTNU is of relevance, considering the research done by the group could be used for later work	Interested in the group producing some results

3.2 General summary of potential user groups

Name	Description	Role during development	Represented by
Researchers	It could be interesting to do research on the field of the brain in conjunction with the eyes	This is a general description of a user and will therefore not partake in the development	This is a general description of a user and will therefore not be represented by anyone
Individuals with disabilities	If someone difficulties communicating, but can use their eyes and has standard brain activity, they could be relevant users of the spelling system	No such individuals will be called in for testing or be conferred with, though it could be interesting	No one will represent this group of people

3.3 User environment

The system needs some way to record and stream both EEG and eye tracking data. During development, equipment such as an actiCAP and LiveAmp from Brain Products and an HTC VIVE Pro Eye will be used. For this setup, two computers are necessary. One of them with proprietary software for the Brain Products equipment used exclusively as the origin for streamed EEG data. The other one has to run an interface between the VIVE Pro and the individual using the system, a pipeline to manage different data streams, and some kind of

analyzing tool. And lastly, it should be mentioned that all of this is put in a lab at NTNU called the Vizlab.

3.4 Summary of the potential user groups' needs

Need	Priority	Affects	Today's solution	Proposed solution
Synchronized EEG and eye tracker	High	Researchers at the Vizlab that needs synchronized equipment	Utilize the equipment, but it is not synchronized	Find a way to synchronize the EEG and eye tracker
A BCI speller	High/medium	Individuals with disabilities that have difficulties communicating	Not using any speller or using one, such as a P300 or SSVEP speller	Use the synchronized EEG and eye tracker to create a BCI spelling system

3.5 Alternatives of our product

Originally, the EEG equipment and eye tracker was not synchronized, but some companies, such as EyeLink, already offer solutions for synchronizing EEG and eye tracker data with their proprietary equipment and recommended software. Also, other BCI spellers are already available for purchase. One example is the Unicorn Speller and Unicorn Hybrid Black, which is a P300 speller variant and an 8-channel EEG cap. This combination will cost a total of at least 1290 euro as per the date of writing this document (*Unicorn Hybrid Black* | *unicorn-bi.com* 2023; *Unicorn Speller Hybrid Black* | *unicorn-bi.com* 2023). The total of the equipment available for this project is estimated to be around 20 000-100 000 NOK (*EEG Headset Prices – An Overview of 15+ EEG Devices - iMotions* 2019; *HTC VIVE PRO FULL KIT VR-HEADSETT* 2023).

4 Product overview

4.1 The product's role in the user environment

The researcher should have all the necessary equipment and software in close proximity since they usually should sit at a lab with all the equipment available and in a highly controlled environment. This means that the researcher should have the Brain Products hardware and software, a computer for running the user interface, a VR headset to interact with the interface, analysis tools, and the pipeline managing data streams.

As for the other group which has been discussed, namely disabled individuals, it is another story. The system in its entirety is pretty large and not very portable, making it impractical to use. An individual that has any use for such a system likely has very limited mobility, but the discussed system is placed at a set location. It could all be done on one laptop to increase portability and flexibility, but that would require a very powerful machine. Therefore, if a user is a disabled person, the system is available to a small degree in their environment.

4.2 Prerequisites and dependencies

The system is planned to be built with specific equipment and software solutions, leading to a relatively rigid layout. Take for instance the Brain Products part. If one changes the EEG cap used, e.g. changes from one with fewer channels to one with more channels, it should pose no challenge. But if the VR headset with the built-in eye tracker is swapped out with another, they could have different APIs which could make it incompatible with the planned solution and ultimately make the entire system fail. The communication between each part of the planned system is based on a protocol called lab streaming layer (LSL) (*LabStreamingLayer submodules · GitHub 2023*), meaning if this is changed in the future in some way, it would most likely affect the system.

5 Functional attributes

Because this project not is a traditional development project, defining many functional attributes poses a problem. The only attributes the system should have are the ones stated in the research questions and listed up below.

Functional attribute	Description
Synchronized equipment	The EEG and eye tracker data should be synchronized within a 1ms margin of error
Working speller	The synchronized equipment should be used to create a working BCI speller
Little to no discrimination	The BCI speller should work for multiple individuals
Accuracy of acceptable degree	The BCI speller should be at least 80% accurate

6 Non-functional attributes and other requirements

Non-functional attribute	Description
Follow standard scientific conventions	During synchronization and testing of the BCI speller, standard scientific conventions, such as controlling sources of error, should be utilized
Maintainable and readable code	The code produced should be maintainable and readable with comments throughout to explain what's happening and properly named variables, methods, and files to make the project as easy to continue for other people as possible
Usability	The BCI speller should follow general usability guidelines according to the WCAG 2.1 Standard Perceivable (<i>Understanding Success Criterion 1.4.3: Contrast (Minimum)</i> WAI W3C 2023; <i>Understanding WCAG 2.1</i> WAI W3C 2023)
The speller should be robust	The underlying implementation of the speller should be robust and work as expected, leading to few or no crashes and intuitive controls
The speller should be relatively efficient	Using the speller should not be slow as to increase the amount of information one can convey over a short period of time

Bibliography

- EEG Headset Prices – An Overview of 15+ EEG Devices - iMotions* (July 2019). en-US. Section: Consumer Insights. URL: <https://imotions.com/blog/learning/product-guides/eeg-headset-prices/> (visited on May 5, 2023).
- HTC VIVE PRO FULL KIT VR-HEADSETT* (2023). no. URL: <https://www.power.no/gaming/vr-og-simulation/vr-briller/htc-vive-pro-full-kit-vr-headsett/p-1063464/> (visited on May 5, 2023).
- LabStreamingLayer submodules · GitHub* (2023). URL: <https://github.com/labstreaminglayer> (visited on Feb. 9, 2023).
- Understanding Success Criterion 1.4.3: Contrast (Minimum) | WAI | W3C* (2023). URL: <https://www.w3.org/WAI/WCAG21/Understanding/contrast-minimum.html> (visited on May 5, 2023).
- Understanding WCAG 2.1 | WAI | W3C* (2023). URL: <https://www.w3.org/WAI/WCAG21/Understanding/> (visited on May 5, 2023).
- Unicorn Hybrid Black | unicorn-bi.com* (2023). en-US. URL: <https://www.unicorn-bi.com/product/unicorn-hybrid-black/> (visited on May 5, 2023).
- Unicorn Speller Hybrid Black | unicorn-bi.com* (2023). en-US. URL: <https://www.unicorn-bi.com/product/unicorn-speller-hybrid-black/> (visited on May 5, 2023).

G Project Manual

Project Manual

May 20, 2023

Contents

1 Collaboration Agreement

2 Milestones

2.1 Planned Milestones

2.2 Actual Milestones

3 Meeting notices and Meeting reports

4 Timesheets and Status reports

1 Collaboration Agreement

Collaboration Agreement

Members: Olav F. P. Larsen, William G. Tresselt

This collaboration agreement builds upon a set of typical goals, distribution of tasks, procedures, and guidelines for student interactions. The agreement is filled in with the group's own interpretations of what they think each section means and how they will achieve them.

Goals

Outcome goals

- Collaborate efficiently.
 - Meet frequently.
 - Be flexible in the sense that we are able to discuss potential problems, like disagreements, and be open to compromise.
- Communicate efficiently.
 - Be available 08-16 and preferably outside the given time frame.
 - Don't make assumptions and have a common understanding of what is being discussed.
- Increase knowledge of EEG, Eye tracking, VR, and scientific methods.
 - Research each topic thoroughly.
 - Test the equipment and use theory combined with practice.

Performance / Process goals

- Deliver obligatory tasks on time.
 - Finish and deliver each obligatory task at least 3 days before their deadline.

-
- Keep track of deadlines.
 - Achieve grade A.
 - Fulfill all formal requirements defined for the main report.
 - Keep a common thread through the main report.
 - Deliver obligatory tasks on time.
 - Get and use input from supervisors.
 - Work in a structured manner.
 - Put in at least five hundred hours of work each.
 - Keep track of time spent on the project.
 - Prioritize spending time working on the project.
 - Complete the project at least one week before the deadline. Only write the report the final week and dedicate the last 48 hours to proofreading.
 - Plan ahead and work consistently.
 - Publish the final report.
 - Be satisfied with the final project.
 - Create a product that can be used as a baseline for further research.
 - The group needs to create a well-tested and documented product that is user-friendly.
 - Utilize Scientific methodology (Exploratory Research)
 - Research the different parts of the projects
 - Use research to gather data from the pipelines, and synchronize
 - Test and validate the data synchronization

Roles and division of tasks

Olav F. P. Larsen

- Main meeting reporter.
- Responsible for documents.
- Responsible for quality assurance.
- Responsible for returning the equipment after use.

William G. Tresselt

- Responsible for meetings.
- Supporting meeting reporter.
- Responsible for documents.
- Responsible for quality assurance.

Procedure

Schedule meetings

Every other week, Tuesday 15:00 starting January 31st via Teams.

Notification in case of absence or other events

In case of absence, notify William via mail at williagt@stud.ntnu.no or phone at +47 93823508.

Document management

Every document should be written in LaTeX. Each document should be shared with the other member, and a list of different templates should be in the Teams channel.

When a document is finished it should also be uploaded to the Teams channel.

Submission of assignments

Both members are responsible for quality assurance and proofreading. Also, they are both responsible for meeting deadlines. Documents and other files produced by the group should be delivered with all members present.

Interaction

Attendance and preparation

Before meeting in general, agree on what should be prepared and when/where to meet, with a leeway of 5 minutes. The entire process should be dynamic to allow for more flexibility. Note that the members must work a set amount of hours and they should always have that fact in mind.

Presence and engagement

The group members should be engaged while working and should keep distractions to a minimum. There should be breaks to keep the members motivated during work time. Usage of phones for entertainment purposes during meetings is forbidden.

Disagreement / Breach of Contract

The members should reflect on the matter at hand. The members could also inquire with the supervisors if there is a disagreement. Violating the agreement should result in the offender baking a cake of the other member's choice.

Signatures

Olav F. P. Jansen

Figure 1: Olav F. P. Larsen

William G. Tresselt

Figure 2: William G. Tresselt

2 Milestones

2.1 Planned Milestones

Date	Milestone
27.01.2023	Deliver pre-project plan
24.02.2023	Setup of data gathering pipelines complete
27.03.2023- 02.04.2023	Poster presentation of research question(s)
06.04.2023	Managed to sync eye tracking and EEG data
10.04.2023	Gathered test subjects for data collection
15.05.2023	Self-imposed deadline for the final report and other documentation
22.05.2023	Absolute deadline for the final report and other documentation
26.05.2023	Digital presentation of the project

2.2 Actual Milestones

Date	Milestone
27.01.2023	Deliver pre-project plan
13.02.2023	Setup of data gathering pipelines complete
01.03.2023	Managed to sync eye tracking and EEG data
07.03.2023	Presenting results of synchronization to supervisors
07.03.2023	Decided to use synchronized EEG and Eye Trackers to create a BCI speller
29.03.2023	Poster presentation of research question(s)
30.03.2023	POC presentation of BCI speller
14.04.2023	A working implementation of a BCI speller
24.04.2023	Testing BCI speller for the first time with test subjects
26.04.2023	Showcase product at the stand at Sensor Tech Day at NTNU Øya
28.04.2023	Testing BCI speller second time with the same test subjects as the first time
15.05.2023	Self-imposed deadline for the final report and other documentation
22.05.2023	Absolute deadline for the final report and other documentation
26.05.2023	Digital presentation of the project

3 Meeting notices and Meeting reports

Meeting invitation: Bachelor thesis 92

Date/time: 17.01.2023, 15:00-15:30

Location: Meeting room 201, 2nd floor of IT-building

Mazemap: <https://link.mazemap.com/pJk9fSUx>

Participants:

- Alexander Holt (supervisor)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 Contract
- 0.2 Weekly meetings/guidance
- 0.3 Room? Vizlab?
- 0.4 Discuss assignment, where should we take it?
- 0.5 Working method
- 0.6 Should we have Vision Document and Requirements Document?
- 0.7 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 12.01.2023

Report 17.01.2023

Date/time: 17.01.2023, 15:00-15:30

Location: Meeting room 201, 2nd floor of IT-building

Present: Alexander Holt, Emanuel Alexander Lorenz, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: None

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

0.1	Contract	
0.2	Weekly meetings/guidance	
0.3	Room? Vizlab?	
0.4	Discuss assignment, where should we take it?	
	0.4.1 Main Task	
	0.4.2 Report	
0.5	Working method	
0.6	Should we have Vision Document and Requirements Document?	
0.7	Miscellaneous	

0.1 Contract

Since NTNU is our respondent there is no need to write a contract between us, NTNU, and the respondent. We own what is written and created, while NTNU has the right to use our finds and creations.

0.2 Weekly meetings/guidance

There should be a meeting every other week starting Tuesday, the 31. January, 15:00. This meeting will be mainly on Tuesdays from 15:00. It should only be written Meeting reports for every official meeting, not for casual questions, etc.

0.3 Room? Vizlab?

The group is granted access to the Vizlab and is assigned a designated spot to work there. This spot is beside Emanuel. Note that there are many other groups, so the groups might need to share equipment. Equipment can be booked through Google Docs. Note, one computer, one screen, and one TV are "ours".

0.4 Discuss assignment, where should we take it?

0.4.1 Main Task

1. Get training in how to use equipment
2. Get both the EEG and the eye-tracking pipelines running
3. Synchronize both pipelines
 - Test and evaluate to ensure the pipelines are **actually** synchronized. There should be less than a 1 ms difference.
 - If the pipelines aren't synchronized well enough, they could be used for testing, but not research
 - Note that synchronizing can also be checked through the different artifacts (closing the eyes, blinking, etc)
 - Synchronize through software (easiest) not hardware
4. Getting access to old projects (for inspiration)
5. When the pipelines are synchronized, we can decide what's next

0.4.2 Report

- The most important part of the project
- External sensors are the ones setting the grade and they only look at the main report
- It's important that there is a common thread throughout the report. If it doesn't fit the thread, think about removing the part
- Every part should be of relevance to the research question
- The supervisor can't read the entire report, but we can ask them to read parts, like a chapter
- Every choice made should be justified.
- When choosing methods and technology every choice **must** be justified
 - Find a fitting method (Scrum may be too much for two people)
 - Not that big of a choice when it comes to technology (Etc why we use NeuroPype instead of Timeflux (should test both))
- Study the report template and see what is needed. It's important to write notes about what each chapter should contain, and if there is anything that is important to remember. If this is not done, we might have some problems.
- Gather sources and references early and use Zotero to keep track

0.5 Working method

After getting started with the project the group should make a milestone plan with five to six milestones. This should not be as descriptive as GANT. "This should be mentioned why used instead of GANT, since an exception to Grethes Template" (Sander). It is also important to read about the technology and test different software, instead of instantly starting to write code.

0.6 Should we have Vision Document and Requirements Document?

- Vision Document
 - This should be done to have a common understanding of what is to be done in the project
 - Approved by the supervisor when done
- Requirements Specification
 - "Since the task is mainly research-oriented this is not that relevant and should/could be skipped" (Sander)

0.7 Miscellaneous

- Add who was invited to the meeting
- Add who is present in each meeting, and why a person wasn't
- Biomarker
 - A biological mark, either from EEG or eye tracker. Could be different EEG frequencies etc
- Use Jitzy for digital meetings (<https://meet.idi.ntnu.no/>)

Meeting invitation: Bachelor thesis 92

Date/time: 31.01.2023, 15:00-15:30

Location: Meeting room 201, 2nd floor of IT-building

Mazemap: <https://link.mazemap.com/pJk9fSUx>

Participants:

- Alexander Holt (supervisor)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 Ready to start Eye Tracking?
- 0.2 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 28.01.2023

Report 31.01.2023

Date/time: 31.01.2023, 15:00-15:30

Location: Meeting room 201, 2nd floor of IT-building

Present: Alexander Holt, Emanuel Alexander Lorenz, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: None

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

0.1	Ready to start Eye Tracking?
0.2	Miscellaneous

0.1 Ready to start Eye Tracking?

The group has achieved a connection between the EEG and NeuroPype and is therefore ready to integrate the eye tracker into the pipeline. The group explores the use SRanipal as an API for the HTC Vive, to create an eye-tracking stream. Note that the API could be demanding to use.

Unity also has an LSL solution, where the git-repo LSL4Unity can transfer eye-tracking data through LSL, an already familiar protocol. The group should be able to start with eye tracking without formal training, besides reading about Unity, SRanipal, and HTC Vive.

The group should be aware that this project is a POC (Proof of Concept), and therefore an environment or a prototype should be created to show the potential of the technology.

The group should be aware of sharing equipment with other groups and book the equipment if needed.

0.2 Miscellaneous

It would be nice if the group is able to create an Eye Brain-Computer Interface (EBCI), with some sort of demo.

Meeting invitation: Bachelor thesis 92

Date/time: 14.02.2023, 15:00-15:30

Location: Meeting room 201, 2nd floor of IT-building

Mazemap: <https://link.mazemap.com/pJk9fSUx>

Participants:

- Alexander Holt (supervisor)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 EEG Neuropype Visualisation
- 0.2 Are blinks an acceptable marker for synchronization?
- 0.3 EEG electrode fragility
- 0.4 Line noise from the VR headset?
- 0.5 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 12.02.2023

Report 14.02.2023

Date/time: 14.02.2023, 15:00-15:30

Location: Meeting room 201, 2nd floor of IT-building

Present: Alexander Holt, Emanuel Alexander Lorenz, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: None

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

0.1	EEG Neuropype Visualisation
0.2	Are blinks an acceptable marker for synchronization?
0.3	EEG electrode fragility
0.4	Line noise from the VR headset?
0.5	Miscellaneous

0.1 EEG Neuropype Visualisation

It would be cool to include a graph that shows the jitter and latency of the different LSL streams of the EEG and eye tracker while running in real time. This is however most important before choosing the biomarker for the synchronization.

0.2 Are blinks an acceptable marker for synchronization?

Eyeblinks are generally easy to see in the EEG signals, and also easy to determine using the eye trackers. It would be best to include an EMG for the recording of the blinks, luckily Brain Products already has this available, and the EMG is also synchronized with the EEG.

To synchronize the data, offline data analysis is suggested. This is done because live streaming of the data would be harder to find due to the streams being live. There are also many different types of jitter, both from hardware, software, and the network. It would be nice to also gather data during high CPU load, to see if there would be any significant delay compared to the normal streams.

To find the common marker for synchronization, the EMG should peak, at the start of eye closure of the eye trackers. Based on this, the streams should be synchronized. Note that this should be done over multiple blinks and recordings and the Standard Deviation (STD) and Standard Error (SE) should be computed.

0.3 EEG electrode fragility

The EEG electrodes are robust and should not break when equipping both the EEG and then the VR headset on top of this. The group should instead be careful with the electrode wires since these can easily break.

0.4 Line noise from the VR headset?

Earlier research studies done by Emanuel have found little to no noise on the EEG from the VR headset. The wet EEG was however used in these studies so a smoke test should be performed. A smoke test is when the EEG starts recording, then seeing if any of the values change when equipping the VR Headset. Signal filtering will also be implemented later in the project, so this shouldn't be an issue.

0.5 Miscellaneous

- Record and bring some blinking data for the next meeting to analyze together
- If the group were to use the results from the blink recordings, these should be added to the report in the appendix
- The meeting should be moved forward to 14 pm for the next, and all upcoming meetings
- After the synchronization there might be time to implement a BCI Speller, or maybe something that recognizes neglect in patients. This is however not the group's main concern at this point in time

Meeting invitation: Bachelor thesis 92

Date/time: 06.03.2023, 14:30-15:00

Location: Meeting room 201, 2nd floor of IT-building

Mazemap: <https://link.mazemap.com/pJk9fSUx>

Participants:

- Alexander Holt (supervisor)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 Show graphs and tables
- 0.2 FIR or IIR
- 0.3 What's next?
- 0.4 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 05.03.2023

Report 06.03.2023

Date/time: 06.03.2023, 14:30-15:00

Location: Meeting room 201, 2nd floor of IT-building

Present: Alexander Holt, Emanuel Alexander Lorenz, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: None

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

0.1	Show graphs and tables	
0.2	FIR or IIR	
0.3	What's next?	
0.4	Miscellaneous	

0.1 Show graphs and tables

The group found some problems with sampling. Unity's FPS was inconsistent, and therefore the theoretical frequency of 120 Hz was more like 50-80 Hz. The group either gets a better PC or labels this as a limitation. It is recommended to use a fixed FPS for consistent data.

0.2 FIR or IIR

The IIR Butterworth is preferred, because of its speed. Zero-phase Butterworth is preferred when analyzing but is not possible when streaming data. The group should use the Butterworth from NeuroPype instead of the Python version.

0.3 What's next?

Next for the group is creating a speller game POC. This should be ready by Thursday 30th March (week 13). The speller game is a VR game where the user is able to spell using both eye trackers and the EEG. The eye tracker should create a gaze that shows a cluster of letters. Each letter is blinking with different frequencies, which should be possible to determine what letter the user is looking at by the EEG.

0.4 Miscellaneous

Remember to thoroughly explain graphs, tables, and figures with text. Axes and columns are not enough

Meeting invitation: Bachelor thesis 92

Date/time: 15.03.2023, 14:00-14:30

Location: Room 043 (Vizlab), Floor S of IT-building

Mazemap: <https://link.mazemap.com/QG1KRwur>

Participants:

- Alexander Holt (supervisor) (optional)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 Analysis of EEG data
- 0.2 How much theory is relevant?
- 0.3 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 14.03.2023

Report 15.03.2023

Date/time: 15.03.2023, 14:00-15:00

Location: Room 043 (Vizlab), Floor S of IT-building

Present: Emanuel Alexander Lorenz, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: Alexander Holt

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

0.1	Analysis of EEG data	
0.2	How much theory is relevant?	
0.3	Miscellaneous	

0.1 Analysis of EEG data

The relevant channels should be selected based on earlier studies and implementations of BCI Spellers. For feature extraction, it should be taken into consideration the need for training or data to correct the extracting method. It is suggested to use Canonical Correlation Analysis (CCA) due to no need for training or training data. This is already implemented in NeuroPype, and might be easy to implement. The CCA algorithm needs segmentation of the EEG data. This could be implemented using a Hamming window in NeuroPype. The speller should only need to be tested by two different subjects to rule out any biological factors. The testing of the speller is not the main focus of the project. The group should track accuracy, and not bother with other metrics like ITR.

0.2 How much theory is relevant?

As Sander has already mentioned, the group should strive to make the theory both comprehensible and comprehensive, allowing the reader to gain a nuanced understanding of the project if desired.

0.3 Miscellaneous

- The group should inform in the report the nodes and frequencies used in NeuroPype
- It is common for outliers in the dataset. This could be due to other delays or jitters, like from the OS, network, etc. Outliers with an STD higher than 3x could be removed, but this should be justified in the report.
- The group should remember to save data from the final pipeline.

-
- When testing the speller, the project should be reported to NSD due to EEG being sensitive information
 - The group and supervisors also discussed writing a condensed paper for the research done.

Meeting invitation: Bachelor thesis 92

Date/time: 21.03.2023, 14:00-14:30

Location: Meeting room 201, 2nd floor of IT-building

Mazemap: <https://link.mazemap.com/pJk9fSUx>

Participants:

- Alexander Holt (supervisor)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 Interesting Theory?
- 0.2 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 18.03.2023

Report 21.03.2023

Date/time: 21.03.2023, 14:00-14:30

Location: Meeting room 201, 2nd floor of IT-building

Present: Alexander Holt, Emanuel Alexander Lorenz, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: None

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

0.1	Interesting Theory?
0.2	Miscellaneous

0.1 Interesting Theory?

The group should include theory about the different types of BCI Spellers, and different paradigms. The group should choose the best Speller based on the theory in the report. The group should also include a section in the theory about eye tracking.

0.2 Miscellaneous

Meeting invitation: Bachelor thesis 92

Date/time: 30.03.2023, 14:00-14:30

Location: Room 043 (Vizlab), Floor S of IT-building

Mazemap: <https://link.mazemap.com/QG1KRwur>

Participants:

- Xiaomeng Su (external)
- Alexander Holt (supervisor)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 Presentation of Proof of Concept (POC)
- 0.2 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 16.03.2023

Report 30.03.2023

Date/time: 30.03.2023, 14:00-15:00

Location: Room 043 (Vizlab), Floor S of IT-building

Present: Alexander Holt, Emanuel Alexander Lorenz, Xiaomeng Su, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: None

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

0.1	Presentation of Proof of Concept (POC)
0.2	Miscellaneous

0.1 Presentation of Proof of Concept (POC)

The group presented the BCI POC. The group was able to make the speller work using a C program on a laptop. They were however not able to recreate this in Unity. It was agreed that the BCI pipeline was working and that there was something wrong with the flickering implementation in Unity, or something wrong with the display of the VR Headset. This was supported by the fact that the Pipeline worked with flickering from a 2D screen with a 60Hz refresh rate, but not when wearing the headset with a 90hz refresh rate.

It was decided that the group was to find the reason Unity wasn't able to reproduce the flickering frequencies. Some things to check out were the flickering rate in Unity and in the VR headset. This could be done by using light sensors.

0.2 Miscellaneous

- The group was to take a week off during the easter holiday

Meeting invitation: Bachelor thesis 92

Date/time: 18.04.2023, 13:30-14:00

Location: Room 043 (Vizlab), Floor S of IT-building

Mazemap: <https://link.mazemap.com/QG1KRwur>

Participants:

- Alexander Holt (supervisor)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 Discussion of the current process and showcase demo
- 0.2 Change of letter layout? (Currently QWETRY)
- 0.3 Should the code be refactored to use fewer dependencies?
- 0.4 ISTES Mail
- 0.5 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 14.04.2023

Report 18.04.2023

Date/time: 18.04.2023, 13:30-14:00

Location: Room 043 (Vizlab), Floor S of IT-building

Present: Alexander Holt, Emanuel Alexander Lorenz, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: None

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

- 0.1 Discussion of the current process and showcase demo
- 0.2 Change of letter layout? (Currently QWETRY)
- 0.3 Should the code be refactored to use fewer dependencies?
- 0.4 ISTES Mail
- 0.5 Miscellaneous

0.1 Discussion of the current process and showcase demo

The speller worked as expected, and all participants were happy with the result. The next step is to test the speller with other test subjects than the one when developing it. The group should apply to NSD before doing the tests, due to the privacy of the subjects and ethical research. The group should aim to test 5-10 people. This is not a large sample size, but good enough when taking the little time left into consideration. The more subjects, the better.

0.2 Change of letter layout? (Currently QWETRY)

The group should read related research to see if there are any better, but a QWERTY setup will be fine, due to the little time left.

0.3 Should the code be refactored to use fewer dependencies?

The speller is currently running using Unity to display the speller visually, NeuroPype to dejitter and store the different recordings, then a Python script is used to perform the CCA on the signals. When discussing if the Python script for example should be moved into C#, it was decided that this would be time-consuming, and not worth it. It would be better to test more subjects. Python is also a lot easier to work with, due to implementations like Pandas DataFrames. This is not available in C#. It was also discussed to move the Python script doing the CCA into NeuroPype, but NeuroPype has such little documentation that this also would be time-consuming.

The most important aspect is to carefully document each part of the BCI Speller, for the readers of the paper to be able to recreate the results.

0.4 ISTES Mail

This mail was considered spam, and therefore not relevant.

0.5 Miscellaneous

- The group should aim to include more test subjects
- All results from the BCI speller should be recorded, even the bad ones
- The theory should contain formulas and mathematical explanations, but also be understandable for non-experts

Meeting invitation: Bachelor thesis 92

Date/time: 02.05.2023, 14:00-14:30

Location: Meeting room 201, 2nd floor of IT-building

Mazemap: <https://link.mazemap.com/pJk9fSUx>

Participants:

- Alexander Holt (supervisor)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 Theory: Review of sections and subsections
- 0.2 Research Methodology: Review of sections and subsections
- 0.3 Method: Review of sections and subsections
- 0.4 Results: Review of sections and subsections
- 0.5 Societal Impact: Review of sections and subsections
- 0.6 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 27.04.2023

Report 02.05.2023

Date/time: 02.05.2023, 14:00-14:30

Location: Meeting room 201, 2nd floor of IT-building

Present: Alexander Holt, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: Emanuel Alexander Lorenz

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

0.1	Theory: Review of sections and subsections	
0.1.1	EEG	
0.1.2	Signal Filtering	
0.1.3	LSL	
0.1.4	TLL/TCP	
0.1.5	BCI Speller	
0.1.6	Research Participants	
0.2	Research Methodology: Review of sections and subsections	
0.2.1	Data sampling design/collection	
0.3	Method: Review of sections and subsections	
0.3.1	Computer Hardware	
0.4	Results: Review of sections and subsections	
0.4.1	Scientific Results	
0.4.2	Engineering results	
0.4.3	Administrative Results	
0.5	Societal Impact: Review of sections and subsections	
0.6	Miscellaneous	

0.1 Theory: Review of sections and subsections

This section contains everything related to the theoretical part of the report Each section and subsection was reviewed by the group and the supervisor. The general feedback was to explain things easily but also to retain the complexity of the entire project.

0.1.1 EEG

The subsections about different sleep stages and different brain waves should be removed due to not being relevant information in this project.

0.1.2 Signal Filtering

The theory about signal filtering should be easy to understand but also include mathematical explanations to keep the level of knowledge high. It's important for the reader to understand some, and the most important aspects, but be able to understand the mathematical principles behind it if needed. It is important to explain what signal filtering does, how it is doing it, and what this means in practice.

0.1.3 LSL

It is important to make the reader understand what LSL is, and this can be done by including a sentence at the start of the subsection about how the group uses it. This is to make the reader able to place LSL in the project and understand why it is important to know about it.

0.1.4 TLL/TCP

This subsection should be removed due to not being relevant information in this project.

0.1.5 BCI Speller

It should be written about other speller types than just the SSVEP-based spellers. Different paradigms, feature extraction methods, etc should also be included. This is to be able to justify why the group chose the combination of feature extraction method, paradigm, and speller type. Note that this should be limited to only include the most common spellers, paradigms, and feature extraction methods.

0.1.6 Research Participants

This section should be placed under the research methodology chapter, instead of theory. The current amount of information looks good. It is important to not be too careful when writing about this, due to other studies not being completely like this one.

0.2 Research Methodology: Review of sections and subsections

This section contains everything related to the research methodology part of the report.

0.2.1 Data sampling design/collection

Different types are interviews, surveys, observations, etc. The sources the group uses do not have to have a completely identical setup as themselves, there shouldn't always be a 1-1 relation. The group should aim to have less theory about data gathering and more about the planning of the different research experiments. This means less theory and more about how the group has followed research principles. The group should also include why there was such a small sample size of subjects.

0.3 Method: Review of sections and subsections

This section contains everything related to the methodical part of the report. The Overall feedback on this part was that it was nicely built up.

0.3.1 Computer Hardware

The computer used had a Ryzen 9 5900X CPU, and 32 GB of DDR 4 RAM.

0.4 Results: Review of sections and subsections

This section contains everything related to the results of the report. This part should be split into three different sections; scientific results, engineering results, and administrative results.

0.4.1 Scientific Results

This section should show the data results from the different experiments. Some of the current tables are a bit redundant and should be added to the appendix instead. This applies to the different cutting strategies and the individual results of the different subjects during the BCI testing. The summary of these results should be included in the results section.

0.4.2 Engineering results

In this section, the implementation of the project should be added. What did the group do to make the speller work, what didn't work, etc? The different approaches to figuring out what was incorrect with the speller should be moved to this section. Other things the group has learned during the process of the project should be added in this section.

0.4.3 Administrative Results

This section should include results like the Vision Document, the Milestones, and the Time Sheet.

0.5 Societal Impact: Review of sections and subsections

This section should include all the societal benefits of research combining EEG and eye trackers. This shouldn't be associated with global warming and other environmental factors, but rather societal factors. A BCI Speller could be used by paralyzed individuals, and other BCI systems can be used by people with neglect, etc. The group should read about the Global Milestones created by the United Nations, and add these to the section. The group should however not write about what doesn't exist, like less emission due to the synchronization.

0.6 Miscellaneous

- The group does not need to add the code as an appendix, but it is possible to print it from Gitlab. This will Sander take care of
- The group should finish the Vision Document
- The group should move tables with similar, repeating, and or redundant info to the appendix
- The group could limit the statistic used in the report to STD and SE due to only being at bachelor level.

Meeting invitation: Bachelor thesis 92

Date/time: 09.05.2023, 12:00-12:30

Location: Meeting room 201, 2nd floor of IT-building

Mazemap: <https://link.mazemap.com/pJk9fSUx>

Participants:

- Alexander Holt (supervisor)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 Review EEG-related table of contents with Emanuel. Need more? Less?
- 0.2 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 03.05.2023

Report 09.05.2023

Date/time: 09.05.2023, 12:00-12:30

Location: Meeting room 201, 2nd floor of IT-building

Present: Alexander Holt, Emanuel Alexander Lorenz, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: None

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

0.1	Review EEG-related table of contents with Emanuel. Need more? Less? . . .
0.2	Miscellaneous
0.2.1	Vision Document Review
0.2.2	Research Methodology
0.2.3	Scientific Rigor
0.2.4	Presentation
0.2.5	Report Length
0.2.6	Related Work
0.2.7	Acronyms
0.2.8	Sources
0.2.9	Tabels

0.1 Review EEG-related table of contents with Emanuel. Need more? Less?

It is crucial for the group to discuss the reasons behind choosing a particular approach and to weigh different options. The EEG component is acceptable, but it should be explained more thoroughly, with examples and the addition of some images. The visual system should be described, though perhaps not in extensive detail. When discussing the synchronization, the group should ensure that the following bullet points are addressed:

- Limitations
- The effectiveness of synchronization
- What can and cannot be measured
- Reasons for it not working
- Elimination of other potential errors
- Hardware latency - even when not measurable

The retest of the speller should primarily focus on noise and power line noise-related discussions.

0.2 Miscellaneous

0.2.1 Vision Document Review

The vision document was examined, and modifications, such as updating the stakeholders, were agreed upon. Apart from these changes, the document was deemed satisfactory.

0.2.2 Research Methodology

The group was unsure about the methodology they employed, as different approaches were taken for various aspects of the assignment. Consequently, they were uncertain about how to discuss this topic. It was decided that the group utilized a mixed-method approach.

0.2.3 Scientific Rigor

The group faced challenges in representing their research methodology in a figure due to the aforementioned reasons. It was agreed that they would divide the assignment into two parts and modify the image as needed.

0.2.4 Presentation

The thesis presentation is scheduled for Friday, May 26th. Since the censor might not be available in person, a video recording of the presentation should be prepared, either during the presentation itself or in advance.

0.2.5 Report Length

The report should prioritize quality over quantity. Images should be incorporated to make the large volume of text more accessible to the censor. It is crucial not to overwhelm the censor, thus allowing them to maintain focus throughout the entire report. Information that is not referenced later in the report, such as theory, should be removed as it is not relevant. Consequently, all theory chapters should be addressed in the discussion.

0.2.6 Related Work

The related work section will be eliminated, as it is not suitable for this report.

0.2.7 Acronyms

Acronyms should be employed due to the extensive information provided. Acronyms that appear only once in the text should not be included in the acronym list. The group should

also adjust the format to ensure that page numbers are clearly associated with the acronyms and not random figures.

0.2.8 Sources

Ensure that sources are accurately listed with the author's information, and refrain from including random blogs or similar content.

0.2.9 Tabela

Make sure each table is correctly formatted, and that the figure and table text is in a smaller case in cursive.

Meeting invitation: Bachelor thesis 92

Date/time: 16.05.2023, 13:00-13:30

Location: Meeting room 201, 2nd floor of IT-building

Mazemap: <https://link.mazemap.com/pJk9fSUx>

Participants:

- Alexander Holt (supervisor)
- Emanuel Alexander Lorenz (co-supervisor)
- Olav Finne Prestæng Larsen
- William Gjerberg Tresselt

Agenda:

- 0.1 Closing thoughts
- 0.2 Preproject plan goals not mentioned in the report?
- 0.3 Data as an appendix?
- 0.4 Background and motivation
- 0.5 Miscellaneous

If you can't show up, be sure to notify William G. T. at +47 93823508 or williagt@stud.ntnu.no.

Best regards
William

Trondheim 12.05.2023

Report 16.05.2023

Date/time: 16.05.2023, 13:00-13:30

Location: Meeting room 201, 2nd floor of IT-building

Present: Alexander Holt, Emanuel Alexander Lorenz, Olav Finne Præsteng Larsen, William Gjerberg Tresselt

Absent: None

Chair: William Gjerberg Tresselt

Secretary: Olav Finne Præsteng Larsen

Contents

0.1	Closing thoughts	
0.2	Preproject plan goals not mentioned in the report?	
0.3	Data as an appendix?	
0.4	Background and motivation	
0.5	Miscellaneous	

0.1 Closing thoughts

The group should read the entire report to ensure no unnecessary mistakes.

0.2 Preproject plan goals not mentioned in the report?

The group should mention these goals in the administrative results.

0.3 Data as an appendix?

The group should add the blinking data used as an appendix. This should however not include any names or other things that can identify the subject used.

0.4 Background and motivation

There is no need for this subsection, as long it is mentioned in the report.

0.5 Miscellaneous

The societal impact should include both the synchronization part of the assignment, and the BCI speller.

4 Timesheets and Status reports

Summary of timetables in project nr: 92

Week nr.	William	Olav
2	14	15
3	25.5	27.5
4	22.5	23
5	20.5	18
6	18.5	18.5
7	15	16.5
8	15.5	20.5
9	14.5	14
10	15	13
11	16.5	16.5
12	31	29
13	32	32
14	0	0
15	32	30
16	45	45
17	49	49
18	45	45
19	45	45
20	45	45
Sum amount of hours pr person/total	501.5	502.5

Timetable with status reports

Timetable		William
Activity	Category	Hours
General set up and reading about EEG and VR	Set up	5.0
Reading about EEG, VR and how to write a bachelor's thesis	Research	5.0
Attended bachelor information lecture	Lecture	4.0
Sum week 2		14.0

Timetable		Olav
Aktivitet	Category	Hours
General set up and reading about EEG and VR	Set up	3.5
Gathering information about EEG and VR	Research	4.5
Reading about EEG and BCI	Research	3.0
Attended bachelor information lecture	Lecture	4.0
Sum week 2		15.0

Weekly report

Both group members started doing some light reading on EEG, VR and eye tracking. They also attended lectures where formal information about the bachelor thesis should be written and how to think when writing.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Got info about scientific theory and method	Lecture	8.0
Initial meeting with supervisors	Meeting	0.5
Wrote pre-project plan and collaboration agreement	Documentation	6.5
Research about different brain waves and EEG	Research	1.5
Preparation for lecture next week	Research	2.0
Continued writing pre-project plan and some main report	Documentation	7.0
Sum week 3		25.5

Timetable		Olav
Activity	Category	Hours
Got info about scientific theory and method	Lecture	8.0
Initial meeting with supervisors	Meeting	0.5
Revised and combined both meeting reports to common report	Meeting Report	1.5
Research about different brain waves and EEG	Research	2.0
Wrote Pre-project plan and collaboration agreement	Documentation	6.5
Preparation for lecture next week	Research	2.0
Continued writing pre-project plan and some main report	Documentation	7.0
Sum week 3		27.5

Weekly report

Initial start up meeting with supervisor, more research and attended lectures. The team also wrote the first drafts of the pre-project plan and the collaboration agreement. Olav also summarized the report after the meeting with the supervisors

Timetable with status reports

Timetable		William
Activity	Category	Hours
Started working on vision document	Documentation	2.0
Got info about scientific theory and method	Lecture	3.0
Finished Pre-project plan	Documentation	4.0
Training in using EEG	Equipment training	2.0
Testing EEG and neurotype	Experimenting, Research	6.0
Wrote theory in the main report, sat up a test LSL from Unity	Documentation, Research	5.5
Sum week 4		22.5

Timetable		Olav
Activity	Category	Hours
Started working on vision document	Documentation	2.0
Got info about scientific theory and method	Lecture	3.0
Finished Pre-project plan	Documentation	4.0
Training in using EEG	Equipment training	2.0
Testing EEG and neurotype	Experimenting, Research	6.0
Wrote theory in the main report	Documentation, Research	6.0
Sum week 4		23.0

Weekly report

Both members got a lecture in how to use the EEG. Both finished the Pre-Project plan, and wrote more theory about EEG and eye tracking. Other than that, the members experimented with the EEG

Timetable with status reports

Timetable		William
Activity	Category	Hours
Meeting with supervisors	Meeting	0.5
Setup of eye tracker	Eye tracker	3.0
Started work on capturing Unity LSL stream in neuropype	Eye tracker	2.5
LSL and Eye Tracker in Unity	Eye tracker pipeline	9.0
Worked on the final report	Documentation	5.5
Sum week 5		20.5

Timetable		Olav
Activity	Category	Hours
Meeting with supervisors	Meeting	0.5
Setup of eye tracker	Eye tracker	2.5
Revision of meeting report	Documents	1.0
LSL and Eye Tracker in Unity	Eye tracker pipeline	9.0
Worked on the final report	Documentation	5.0
Sum week 5		18.0

Weekly report

The group had a meeting with the supervisors and startedf working on the part of the project where they use Unity to stream eye tracking data via LSL. They also did some work on the final report.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Writing report	Main Report	7.5
Working on Eye Tracker	Eye Tracker	3.0
Tried to fix pc	Fixing Pc	1.5
Creating NeuroPype pipeline	NeuroPype	6.5
Sum week 6		18.5

Timetable		Olav
Activity	Category	Hours
Writing report	Main Report	7.5
Working on Eye Tracker	Eye Tracker	3.0
Tried to fix pc	Fixing Pc	1.5
Creating NeuroPype pipeline	NeuroPype	6.5
Sum week 6		18.5

Weekly report

Spent most of the week setting up the NeuroPype pipeline for gathering data from EEG and eye tracker and trying to fix the PC the group was assigned.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Worked more on NeuroPype pipeline	NeuroPype	3.5
Meeting	Meeting	1.0
Data gathering	Data gathering	2.5
Migrated setup	Setup	8.0
Sum week 7		15.0

Timetable		Olav
Activity	Category	Hours
Worked more on NeuroPype pipeline	NeuroPype	3.5
Meeting	Meeting	1.0
Data gathering	Data gathering	2.5
Migrated setup	Setup	8.0
Analysed data in python	Data analysis	1.5
Sum week 7		16.5

Weekly report

Tried gathering data from EEG and eye tracker at once from a subject. Also got information about external electrode placement for blinking.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Data gathering	Data gathering	5.5
Researched definitions for blinking	Research	3.0
Modified Unity project and NP pipelines, then gathered data	Data gathering	2.0
Analysed data in Python and NP	Data analysing	3.0
Researched definitions for blinking	Research	2.0
Sum week 8		15.5

Timetable		Olav
Activity	Category	Hours
Data gathering	Data gathering	5.5
Researched definitions for blinking	Research	3.0
Modified Unity project and NP pipelines, then gathered data	Data gathering	2.0
Analysed data in Python and NP	Data analysing	3.0
Analysed data in Python and NP	Data analysing	4.0
Analysed data in Python and NP	Data analysing	3.0
Sum week 8		20.5

Weekly report

Kept on setting up data streams and gathering data, also early analysis in Python.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Gathered EMG and eye tracking data	Data gathering	1.0
Analysed gathered data	Data analysis	2.0
Tried figuring out problem with data gathering	Troubleshooting	1.0
Gathered EMG and eye tracking data	Data gathering	4.5
Written Main report	Main Report	6.0
Sum week 9		14.5

Timetable		Olav
Activity	Category	Hours
Gathered EMG and eye tracking data	Data gathering	1.0
Analysed gathered data	Data analysis	2.0
Tried figuring out problem with data gathering	Troubleshooting	1.0
Gathered EMG and eye tracking data	Data gathering	4.0
Written Main report	Main Report	6.0
Sum week 9		14.0

Weekly report

Kept on gathering data and analysing. The amount of data allowed for some analysis resulting in an indication of how out of sync the EEG and eye tracking was. Of course, there was some hardware complications out of the blue one day which explains the "Troubleshooting" category. Kept working on the main report.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Meeting with supervisors	Meeting	0.5
Presentation of Bachelor Thesis to East European Doctors	Presentation	1.0
Worked in Unity and NP	Project work	2.0
Started working on ERP speller	Unity	6.5
Worked on ERP speller	Unity	5.0
Sum week 10		15.0

Timetable		Olav
Activity	Category	Hours
Meeting with supervisors	Meeting	0.5
Presentation of Bachelor Thesis to East European Doctors	Presentation	1.0
Worked in Unity and NP	Project work	2.0
Started working on ERP speller	Unity	6.5
Working on ERP speller	Unity	3.0
Sum week 10		13.0

Weekly report

Meeting with supervisors -> Further work -> Presented the current state of the project and syncing setup to doctors -> Started working on ERP speller (SSVEG)

Timetable with status reports

Timetable		William
Activity	Category	Hours
Data sanking	Data sanking	2.0
Main report	Main report	8.0
PC setup	Set up of new PC for use in project	1.0
Meeting	Meeting with Emmanuel talking about pipeline for speller	1.0
Gathered data	Gathered data with new PC	1.0
Pipeline	Worked on pipeline for speller	3.5
Sum week 11		16.5

Timetable		Olav
Activity	Category	Hours
Data sanking	Data sanking	2.0
Main report	Main report	6.0
Meeting	Meeting with Emmanuel talking about pipeline for speller	1.0
Gathered data	Gathered data with new PC	1.0
Analyzed data	Analyzed data	3.0
Pipeline	Worked on pipeline for speller	3.5
Sum week 11		16.5

Weekly report

Got the new PC and set it up. Also started working on the NP pipeline for the speller after a meeting with Emanuel.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Tried to figure out more about CCA	Research	2.0
Data gathering	Data gathering	2.0
Analyzed gathered data	Data analysis	2.5
Doubted previous methods	Research	4.0
Gathered data for ERP and analyzed it	Data analysis, Data gathering	3.0
Gathered data for ERP and analyzed it	Data analysis, Data gathering	5.5
Gathered data for Synch and found offset it	Data analysis, Data gathering	3.0
Tested BCI speller and implemented spelling in Unity	BCI Speller	5.0
Continued writing the main report	Documentation	4.0
Sum week 12		31.0

Timetable		Olav
Activity	Category	Hours
Data gathering	Data gathering	2.0
Analyzed gathered data	Data analysis	2.5
Doubted previous methods	Research	4.0
Gathered data for ERP and analyzed it	Data analysis, Data gathering	3.0
Gathered data for ERP and analyzed it	Data analysis, Data gathering	5.5
Gathered data for Synch and found offset it	Data analysis, Data gathering	3.0
Tested BCI speller and implemented spelling in Unity	BCI Speller	5.0
Continued writing the main report	Documentation	4.0
Sum week 12		29.0

Weekly report

Was doubting that pipeline was synchronized, resulting in gathering more accurate data for synching purposes. Created a Python script for performing CCA and connected it to NP instead of using NP's CCA node and prepared BCI speller interface in Unity.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Tested BCI speller and implemented spelling in Unity	BCI Speller	7.0
Tested BCI speller and implemented spelling in Unity	BCI Speller	7.0
Tested BCI speller and implemented spelling in Unity	BCI Speller	7.0
Discussion with supervisor (Emanuel)	BCI Speller	1.0
Poster presentation	Bachelor degree	1.0
POC presentation	Bachelor degree	1.0
Tested BCI speller and implemented spelling in Unity	BCI Speller	8.0
Sum week 13		32.0

Timetable		Olav
Activity	Category	Hours
Tested BCI speller and implemented spelling in Unity	BCI Speller	7.0
Tested BCI speller and implemented spelling in Unity	BCI Speller	7.0
Tested BCI speller and implemented spelling in Unity	BCI Speller	7.0
Discussion with supervisor (Emanuel)	BCI Speller	1.0
Poster presentation	Bachelor degree	1.0
POC presentation	Bachelor degree	1.0
Tested BCI speller and implemented spelling in Unity	BCI Speller	8.0
Sum week 13		32.0

Weekly report

The group worked towards creating a POC for the BCI speller. Did not manage a working speller in 3D, but 2D. Identified many possible reasons why it did not work and started ruling them out one by one.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Gathered synchronization data	Data gathering	6.0
Analyzed gathered data, both with NP and Python	Analyze	3.0
Meeting with supervisors, showing poc and talk about tests	Meeting	1.0
Some adjustments of speller	Speller	15.0
Main Report	Main Report	20.0
Sum week 16		45.0

Timetable		Olav
Activity	Category	Hours
Gathered synchronization data	Data gathering	6.0
Analyzed gathered data, both with NP and Python	Analyze	3.0
Meeting with supervisors, showing poc and talk about tests	Meeting	1.0
Some adjustments of speller	Speller	15.0
Main Report	Main Report	20.0
Sum week 16		45.0

Weekly report

Both members gathered and analyzed more synchronization data, had a meeting with supervisors finally showing a working speller, did some adjustments on the speller to make it ready for testing next week and continued with the report.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Tested speller on participants round 1	Testing	8.0
Wrote formal documentation for testing	Documentation	1.0
Continued writing main report	Documentation	9.0
Tested speller on participants round 2	Testing	8.0
Put results in graphs	Documentation	1.0
Continued writing main report	Documentation	18.0
Stand representing the Vizlab	Stand	4.0
Sum week 17		49.0

Timetable		Olav
Activity	Category	Hours
Tested speller on participants round 1	Testing	8.0
Wrote formal documentation for testing	Documentation	1.0
Continued writing main report	Documentation	9.0
Tested speller on participants round 2	Testing	8.0
Put results in graphs	Documentation	1.0
Continued writing main report	Documentation	18.0
Stand representing the Vizlab	Stand	4.0
Sum week 17		49.0

Weekly report

This week, the group made preparations and conducted tests for finding accuracy of the speller. They also attended a stand which the supervisors said they should join to represent the Vizlab

Timetable with status reports

Timetable		William
Activity	Category	Hours
Continued writing documentataion	Documentation	44.0
Meeting with supervisor discussing report	Meeting	1.0
Sum week 18		45.0

Timetable		Olav
Activity	Category	Hours
Continued writing documentataion	Documentation	44.0
Meeting with supervisor discussing report	Meeting	1.0
Sum week 18		45.0

Weekly report

The group finished testing and such last week, this week was therefore only used for writing the bachelor thesis. Also had a meeting with supervisor to discuss structure of report and other relevant items

Timetable with status reports

Timetable		William
Activity	Category	Hours
Continued with thesis and other documentation	Documentation	44.0
Meeting with supervisors	Meeting	1.0
Sum week 19		45.0

Timetable		Olav
Activity	Category	Hours
Continued with thesis and other documentation	Documentation	44.0
Meeting with supervisors	Meeting	1.0
Sum week 19		45.0

Weekly report

The group spent the second last week writing the final report and had a meeting with the supervisors for general guidance related to the document.

Timetable with status reports

Timetable		William
Activity	Category	Hours
Finished thesis and other documentation	Documentation	44.0
Had final meeting with supervisors	Meeting	1.0
Sum week 20		45.0

Timetable		Olav
Activity	Category	Hours
Finished thesis and other documentation	Documentation	44.0
Had final meeting with supervisors	Meeting	1.0
Sum week 20		45.0

Weekly report

The group had a final meeting with the supervisors and completed the final report.

H NeuroPype Pipelines and Python Repository

Repository: <https://gitlab.stud.idi.ntnu.no/group-92/neuropype-pipeline>

I Unity Repository

Repository: <https://gitlab.stud.idi.ntnu.no/group-92/eye-tracking-unity-lsl>



 **NTNU**

Norwegian University of
Science and Technology