

University of Tübingen
Faculty of Science
Department of Computer Science

Master's Thesis

Eliciting Code-modulated Visual Evoked Potentials by Non-Recognizable Presentation of Code Sequences

Submitted by: Lasse Schlör

November 24, 2021

First Supervisor

Prof. Dr. Oliver Bringmann

Faculty of Science
Department of Computer Science
Embedded Systems
University of Tübingen

Second Supervisor

Prof. Dr. Volker Franz

Faculty of Science
Department of Computer Science
Experimental Cognitive Science
University of Tübingen

Graduate Advisor

Alexander Blöck

Faculty of Science
Department of Computer Science
Computer Engineering
University of Tübingen

Schlör, Lasse:

Eliciting Code-modulated Visual Evoked Potentials by Non-Recognizable Presentation of Code Sequences

Master's Thesis

University of Tübingen

Processing period: May 25, 2021 - November 24, 2021

© 2021 Lasse Schlör. This work is licensed under the [Creative Commons Attribution 4.0 International License \(CC BY 4.0\)](https://creativecommons.org/licenses/by/4.0/).

Abstract

Recently, brain-computer interfaces (BCIs) based on visual evoked potentials (VEPs) have gained traction as a tool to improve quality of life in motor paralysis patients through applications such as gaze-controlled keyboards. However, avoiding perceptible flickering to mitigate side effects such as visual fatigue or seizures has been challenging on computer screens. The present thesis proposes a novel class of stimulation patterns employing digital modulation techniques to minimize perceptible flickering. The potentials elicited by these patterns, termed code-modulated carrier visual evoked potentials (CCVEPs), are capable of transmitting error correction codes (ECCs). In a pilot experiment with 7 test subjects, two incarnations of this paradigm were evaluated by means of a basic machine learning model, utilizing on-off keying (OOK) and binary phase-shift keying (BPSK) as modulation schemes, respectively. A mean information transfer rate (ITR) of $69.9^{\text{bit}}/\text{min}$ was achieved, and substantial room for improvement from modifications to the model is postulated. In addition, it is demonstrated that maximum length sequence (MLS) cross-correlation can be adapted to the CCVEP paradigm. 15-target MLS shift classification yielded a mean ITR of $124.6^{\text{bit}}/\text{min}$. While the stimuli presented in the experiment subjectively improve user comfort vastly compared to traditional approaches, achieving perfect non-recognizability with the CCVEP paradigm is a nontrivial optimization problem worthy of further study.

Abstract (deutsch)

In den letzten Jahren haben sich Hirn-Computer-Schnittstellen (BCIs), die auf visuell evozierten Potenzialen (VEPs) basieren, als Werkzeug zur Verbesserung der Lebensqualität von Patient:innen mit motorischer Lähmung durch Anwendungen wie blickgesteuerte Tastaturen durchgesetzt. Wahrnehmbares Flimmern zu vermeiden, um Nebenwirkungen wie visuelle Ermüdung oder Krampfanfälle zu mildern, hat sich auf Computerbildschirmen jedoch als Herausforderung herausgestellt. Die vorliegende Arbeit stellt eine neuartige Klasse von Stimulationsmustern vor, die digitale Modulationstechniken verwenden, um wahrnehmbares Flimmern zu minimieren. Die durch diese Muster hervorgerufenen Potentiale, die als codemodulierte visuell evozierte Trägerpotentiale (CCVEPs) bezeichnet werden, sind in der Lage, Fehlerkorrekturcodes (ECCs) zu übertragen. In einem Pilotversuch mit 7 Proband:innen wurden zwei Inkarnationen dieses Paradigmas mittels eines einfachen maschinellen Lernmodells evaluiert, wobei als Modulationsverfahren An-Aus-Umtastung (OOK) bzw. binäre Phasenumtastung (BPSK) verwendet wurden. Eine mittlere Informationsübertragungsrate (ITR) von $69.9^{\text{bit}}/\text{min}$ wurde erreicht und wesentliches Verbesserungspotenzial durch Modifikationen am Modell wird postuliert. Darüber hinaus wird gezeigt, dass Maximalfolgen (MLS)-Kreuzkorrelation auf das CCVEP-Paradigma adaptiert werden kann. 15-klassige MLS-Versatz-Klassifizierung ergab eine mittlere ITR von $124.6^{\text{bit}}/\text{min}$. Während die im Versuch präsentierten Stimuli den Benutzerkomfort im Vergleich zu herkömmlichen Ansätzen subjektiv erheblich verbessern, ist das Erreichen perfekter Nichterkennbarkeit mit dem CCVEP-Paradigma ein nicht-triviales Optimierungsproblem, das weiter untersucht werden sollte.

Contents

List of Figures	III
List of Tables	IV
List of Algorithms	V
List of Abbreviations	VI
1 Introduction	1
1.1 Background and Motivation	1
1.2 Organization of the following chapters	2
2 Fundamentals	3
2.1 Mathematical Notation	3
2.2 Ridge regression	4
2.3 Linear, shift-invariant (LSI) systems	5
2.4 Correlation	8
2.5 Visual evoked potentials (VEPs)	8
2.6 Brain-computer interfaces (BCIs)	9
2.7 Maximum length sequences (MLSs)	10
2.8 Information transfer rate (ITR)	12
2.9 Spatial filtering	13
2.10 The sliding window model	13
2.11 Flicker fusion thresholds	14
3 Related Work	15
4 Conceptualization	17
4.1 On stimulus recognizability	17
4.2 Code-modulated carrier visual evoked potentials (CCVEPs)	17
4.3 CCVEP challenges on computer screens	20

Contents

5	Materials and Methods	23
5.1	Hardware and Software	23
5.2	Stimulus design	23
5.3	Experiment design	26
5.4	Classifier design	29
5.5	Supplementary analyses	34
6	Results	35
6.1	Stimulation pattern recovery	35
6.2	Code recovery	36
6.3	Frame-wise MLS shift recovery	38
6.4	Segment-wise MLS shift recovery	40
6.5	Sliding window hyperparameters	40
6.6	Signal-to-noise ratio (SNR)	42
6.7	Jitter	42
7	Discussion	43
7.1	Data quality	43
7.2	Stimulus recognizability	44
7.3	Viability of the CCVEP paradigm for BCIs	44
7.4	Relevant signal content	45
7.5	Reflections on the machine learning model	47
8	Conclusion and Future Work	49
	Bibliography	51

List of Figures

2.1	Cyclic convolution in the time domain and in the frequency domain	6
2.2	Common IIR filter frequency responses	7
2.3	Sketches of a pattern reversal VEP and a steady-state VEP	9
2.4	Autocorrelation property of MLSs encoded by OOK and BPSK	12
4.1	Schematic illustration of naïve CCVEP stimulation patterns	18
4.2	Application of windowing/fading to segment transitions	20
4.3	Beating patterns in time-discrete signals	21
4.4	Flicker rendering artifacts on a typical monitor	21
5.1	The code sequence μ to be encoded as stimulus	24
5.2	Stimulation patterns presented in the experiment	25
5.3	Graphical layout for stimulus presentation	27
5.4	Architecture of the machine learning model	29
6.1	Frame-wise classification accuracy by encoding scheme and test subject	35
6.2	Segment-wise classification accuracy by encoding scheme and test subject	37
6.3	Run lengths of (in-)correctly classified segments by encoding scheme	38
6.4	Frame-wise MLS shift classification accuracy by encoding scheme and test subject	39
6.5	Segment-wise MLS shift classification accuracy by encoding scheme and test subject	40
6.6	Frame-wise classification accuracy across window hyperparameters .	41
6.7	Absolute jitter across all test subjects and takes	42
7.1	Magnitude responses of the learned kernels	46
7.2	Inter-channel phase relationships at 60Hz	46

List of Tables

5.1	Number and duration of trials by encoding scheme	28
6.1	Frame-wise classification accuracy by encoding scheme	36
6.2	Segment-wise classification accuracy and ITR by encoding scheme .	37
6.3	Frame-wise MLS shift classification accuracy and ITR by encoding scheme	39
6.4	Segment-wise MLS shift classification accuracy and ITR by encoding scheme	41
6.5	Mean SNR by test subject	42
7.1	Potential issues during each experiment	43

List of Algorithms

2.1	<code>lfsr</code>	10
5.1	<code>encode</code>	24
5.2	<code>frame_wise_features</code>	30
5.3	<code>ccccm</code>	31
5.4	<code>evaluation</code>	32

List of Abbreviations

ADC	analog-to-digital conversion
ASK	amplitude-shift keying
BCI	brain-computer interface
BFSK	binary frequency-shift keying
BPSK	binary phase-shift keying
CCA	canonical correlation analysis
CCVEP	code-modulated carrier visual evoked potential
CI	confidence interval
CNN	convolutional neural network
cVEP	code-modulated visual evoked potential
DAC	digital-to-analog conversion
DFT	discrete fourier transform
DSP	digital signal processing
ECC	error correction code
EEG	electroencephalogram
EMF	electromagnetic field
ERP	event-related potential
FFT	fast fourier transform
FRP	fixation-related potential
FSK	frequency-shift keying
IIR	infinite impulse response
ITR	information transfer rate
LCD	liquid crystal display
LED	light-emitting diode
LFSR	linear feedback shift register
LSI	linear, shift invariant
MLS	maximum length sequence
OOK	on-off keying
PC0	personal computer presenting the stimuli
PC1	personal computer recording the data
PCC	pearson correlation coefficient
PRVEP	pattern-reversal visual evoked potential

List of Abbreviations

PSK	phase-shift keying
PWM	pulse-width modulation
RGB	red, green, and blue color model
RMS	root mean square
SD	standard deviation
SNR	signal-to-noise ratio
SSVEP	steady-state visual evoked potential
VEP	visual evoked potential

1 Introduction

1.1 Background and Motivation

A brain-computer interface (BCI) is an apparatus translating a user’s brain activity into commands, allowing the control of external devices without muscle activity (Wolpaw, Birbaumer, McFarland, Pfurtscheller, & Vaughan, 2002). In recent decades, BCIs based on visual evoked potentials (VEPs) have gained traction as a tool to improve quality of life in motor paralysis patients. A common application is a gaze-controlled keyboard, where a set of distinct visual stimulation patterns, representing the individual keys, are concurrently presented to the user, allowing them to elicit a particular VEP by gazing at the respective stimulus. Using an electroencephalogram (EEG), the stimulation pattern can be identified and interpreted as pressing the corresponding letter on the keyboard.

The majority of research thus far has focused on stimulation patterns eliciting steady-state visual evoked potentials (SSVEPs) or code-modulated visual evoked potentials (cVEPs), with considerable success. As an example, Nagel and Spüler achieved a mean information transfer rate (ITR) of $701^{\text{bit}}/\text{min}$ and a spelling rate of 35min^{-1} error-free letters in 2019 using cVEPs. However, cVEP-based BCIs present the user with obviously flickering patterns, leading to physical side effects such as visual fatigue, migraine headaches, or epileptic seizures (Lin et al., 2014). SSVEP-based BCIs are able to mitigate this (Sakurada, Kawase, Komatsu, & Kansaku, 2015; Won, Hwang, Dähne, Müller, & Lee, 2015) by using light-emitting diodes (LEDs) to present high-frequency flickering patterns, which appear to the user as a steady color (Hecht & Shlaer, 1936; de Lange Dzn, 1954). Implementing high-frequency SSVEP-based BCIs on conventional computer displays is challenging, however, since the number of distinct stimulation patterns that can be accurately rendered from a time-discrete representation is severely limited by the refresh rate.

The central research goal of the present thesis is to realize a VEP-based BCI paradigm feasible for data transmission without recognizable flickering on computer displays. For this, a novel class of screen-suited stimulation patterns employing digital modulation techniques to encode arbitrary n -ary sequences while minimizing the perception of flickering is proposed. Potentials elicited by these stimulation patterns are termed code-modulated carrier visual evoked potentials (CCVEPs). In a pilot experiment with 7 test subjects, two incarnations of this approach were evaluated in terms of ITR, utilizing on-off keying (OOK) and binary phase-shift keying (BPSK) as modulation schemes, respectively. A basic offline machine learning model was constructed to decode the transmitted data from the EEG. In addition, the model is designed to provide an estimate of the original stimulation pattern as well as a second layer of stimulus classification based on maximum length sequence (MLS)

cross-correlation. A subset of the stimulation-classification tandem's parameters are characterized in terms of their impact on BCI performance. The system is furthermore analyzed in terms of signal-to-noise ratio (SNR) and jitter. Finally, some insights about the design of more sophisticated machine learning models for CCVEP decoding are presented.

1.2 Organization of the following chapters

The remainder of the thesis is organized as follows: Chapter 2 first defines some notation used throughout the following chapters. Subsequently, a set of general concepts that the thesis builds upon is introduced. Chapter 3 reviews the literature on previous approaches to achieving stimulus non-recognizability in BCIs and transmitting arbitrary digital codes. Chapter 4 presents the general framework of CCVEP stimulation and goes into some challenges and pitfalls surrounding the approach. Chapter 5 describes the experiment methodology, including the particular stimuli and classification models that were chosen for evaluation. Chapter 6 presents the experimental results. Chapter 7 discusses the methodology and results in terms of their implications and potential for improvement. Finally, chapter 8 summarizes the findings and provides suggestions for future research.

2 Fundamentals

This chapter goes into the foundational concepts which will be made use of throughout the following chapters. First, some mathematical notation will be defined (section 2.1). Next, some basic machine learning and signal processing techniques will be presented (sections 2.2, 2.3, and 2.4). Subsequently, the general idea of VEP-based BCIs will be introduced (sections 2.5 and 2.6). The remaining sections will go on to discuss some more specific concepts as they apply to BCIs (sections 2.7, 2.8, 2.9, 2.10, and 2.11).

2.1 Mathematical Notation

Let us introduce some notation for the handling of matrices which will be relied upon throughout the following chapters:

- Let $\mathbb{1}$ be a vector of all ones whose length can be deduced from the context, e.g. $(1, 2)^T + \mathbb{1} = (2, 3)^T$.
- Let $[\cdot]$ denote the 0-based indexing operator for matrices, such that, for instance,

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} [0, 1] = b \quad \text{and} \quad \begin{pmatrix} 2 \\ 3 \\ 4 \end{pmatrix} [\begin{pmatrix} 2 \\ 1 \end{pmatrix}] = \begin{pmatrix} 4 \\ 3 \end{pmatrix}$$

- Let $\cdot : \cdot$ denote a range vector, i.e. $i : j = (k)_{k=i}^{j-1}$, e.g. $2 : 5 = (2, 3, 4)^T$
- Let $[:]$ select all values of a dimension, e.g.

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} [1, :] = \begin{pmatrix} c \\ d \end{pmatrix}$$

- Let the $\text{vec}(\cdot)$ operator flatten a matrix to a 1-dimensional vector, e.g.

$$\text{vec} \left(\begin{pmatrix} a & b \\ c & d \end{pmatrix} \right) = \begin{pmatrix} a \\ c \\ b \\ d \end{pmatrix}$$

2 Fundamentals

- Let the $\text{cycshift}_i(\cdot)$ operator cyclically shift a vector's elements by $i \in \mathbb{Z}$ places, i.e. $\text{cycshift}_i(x) = (x[(j+i) \bmod n])_{j=0}^{n-1}$, e.g.

$$\text{cycshift}_2\left((2, 3, 4, 5, 6)^T\right) = (4, 5, 6, 2, 3)^T$$

- Let $\bar{\cdot}$ denote the arithmetic sample mean of a vector, e.g. $\overline{(1, 2, 3)^T} = 2$
- Let $\cdot^*$ denote the element-wise complex conjugate of a vector, e.g.

$$\left((1 + i, -2 - 2i)^T\right)^* = (1 - i, -2 + 2i)^T$$

- Let the $\cdot \bmod \cdot$ operator be defined as

$$x \bmod y = x - y \left\lfloor \frac{x}{y} \right\rfloor \quad \mathbf{2.1}$$

- Let

$$\delta_{i,j} = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j \end{cases} \quad \mathbf{2.2}$$

be the Kronecker delta function.

2.2 Ridge regression

Regression is a method of statistical analysis that can be regarded as a machine learning model. Given a sample of k feature vectors $x_i \in \mathbb{R}^n$ of independent variables and corresponding label scalars $y_i \in \mathbb{R}$ for the dependent variable, a model can be constructed by

$$y_i = f(x_i, \beta) + \epsilon_i, \quad i \in \{1, \dots, k\} \quad \mathbf{2.3}$$

where the assumption is that $y - \epsilon$, ϵ being the unmodeled error term, is a function of x . The goal of regression is to estimate the vector of unknown parameters β from the sample such that the model fits the underlying relationship between x and y as closely as possible.

Ridge regression is a method first introduced by Hoerl and Kennard in 1970 to estimate β in ill-posed problems where $f(x_i, \beta) = \beta x_i$ is a linear function. The method is a generalization of ordinary least squares regression, utilizing Tikhonov regularization (1943) to overcome ill-posedness and to prevent overfitting. β is estimated by minimizing the regularized sum of squared residuals between y_i and $f(x_i, \beta)$:

$$\hat{\beta} = \arg \min_{\beta} \|X\beta - Y\|_2^2 + \|\lambda\beta\|_2^2 = \left(X^T X + \lambda I\right)^{-1} X^T Y \quad \mathbf{2.4}$$

2 Fundamentals

where $X \in \mathbb{R}^{k \times n}$ is a matrix comprising k feature vectors, $Y \in \mathbb{R}^{k \times 1}$ are the labels of the k observations, I is an identity matrix, and $\lambda \geq 0$ is the regularizer weight. The resulting model is

$$\hat{y} = f(x, \hat{\beta}) = \hat{\beta}x \quad 2.5$$

Note that feature vectors may be transformed in arbitrary ways before performing the regression, including the use of a constant feature with a value of 1. This effectively introduces a bias term, such that f can be rewritten as

$$f(x, \hat{\beta}, \hat{\beta}_0) = \hat{\beta}x + \hat{\beta}_0 \quad 2.6$$

2.3 Linear, shift-invariant (LSI) systems

Any system that alters a signal in a manner which can be described by *convolution* is called linear, shift invariant (LSI) (Smith et al., 1997). While convolution is a general mathematical operation with applications in many fields, the present thesis will solely handle it in a digital signal processing (DSP) context. Consider two discrete, complex-valued signals $x: \mathbb{Z} \rightarrow \mathbb{C}, b: \mathbb{Z} \rightarrow \mathbb{C}$. The one-dimensional convolution $x * b$ can be defined as

$$y(k) = (x * b)(k) = \sum_{l \in \mathbb{Z}} x(k-l)b(l) \quad 2.7$$

In DSP jargon, b is usually called a *kernel*, and convolving x with a kernel is called *filtering*. In this sense¹, a filter is an LSI system that scales the magnitude and shifts the phase of a signal in a frequency-dependent manner, as determined by its *frequency response*:

$$H_b(e^{2\pi i \omega}) = \sum_{l \in \mathbb{Z}} b(l)e^{2\pi i \omega l} \quad 2.8$$

where ω is the normalized frequency such that 1 corresponds to the sampling frequency, and the result of H is the complex factor by which this frequency is multiplied.

Equation 2.7 is only computable if x and b have finite support, i.e. $(\exists k \in \mathbb{Z} \mid \forall l < k \mid x(l) = 0) \wedge (\exists k \in \mathbb{Z} \mid \forall l > k \mid x(l) = 0)$, and likewise for b . If this is the case, the nonzero regions of x and b can be seen as vectors $x \in \mathbb{C}^n$ and $b \in \mathbb{C}^m$. Furthermore, x may represent one period of an infinitely repeating sequence, leading to the cyclic convolution

¹The term *filter* is ambiguous. In the context of DSP, a (linear) filter usually implies an LSI system. In EEG applications, this is often called a temporal filter to distinguish it from the spatial filter (cf. section 2.9), which is merely a mixing matrix and does not take spatial frequencies into account. In addition, any non-LSI processing may certainly be called a (non-linear) filter.

2 Fundamentals

$$y = x * b = \left(\sum_{l=0}^{m-1} b[l] \text{cycshift}_{k-1}(x)[l \bmod n] \right)_{k=0}^{n-1} \quad 2.9$$

$$= n^{-1} \mathcal{F}(\mathcal{F}(x)^* \mathcal{F}(z)^*) \text{ where } z = \left(\sum_{l=0}^{\lfloor \frac{m-k-1}{n} \rfloor} b[l n + k] \right)_{k=0}^{n-1} \quad 2.10$$

Equation 2.10 states that the cyclic convolution can be written in terms of the discrete fourier transform (DFT) $\mathcal{F}: \mathbb{C}^n \rightarrow \mathbb{C}^n$. This is noteworthy, since it allows for efficient computation by means of fast fourier transform (FFT) algorithms (Cooley & Tukey, 1965). The DFT transforms a signal between its time and frequency domain representations. The definition used here is

$$\mathcal{F}(x) = \left(\sum_{k=0}^{n-1} x[k] e^{2\pi i \frac{kl}{n}} \right)_{l=0}^{n-1} = \left(H_x \left(e^{2\pi i \frac{l}{n}} \right) \right)_{l=0}^{n-1} \quad 2.11$$

The DFT has numerous applications. Of particular significance to the present thesis is the ability to inspect a signal in the frequency domain, providing insight into what frequencies are present at what magnitude and phase.

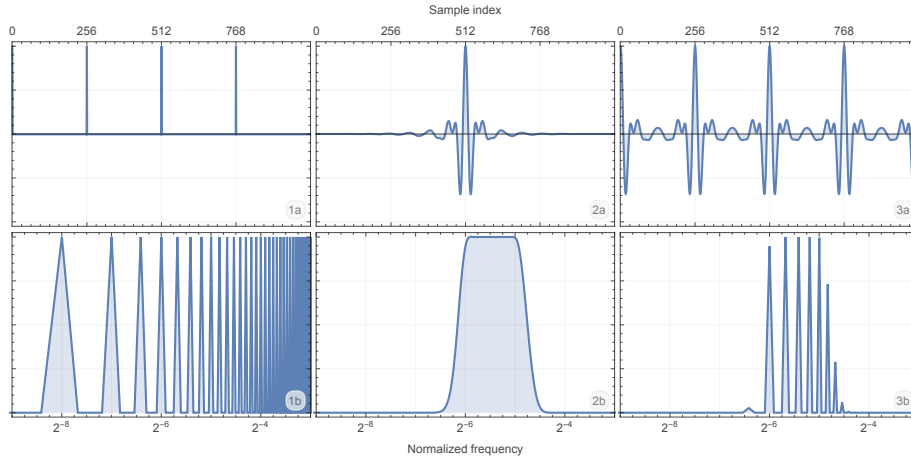


Figure 2.1: Cyclic convolution in the time domain and in the frequency domain.

A cyclic signal (1) is convolved with a kernel (2) to yield the convolved signal (3). Signals are shown in the time domain (a) and in the frequency domain (b). Converting between (a) and (b) is done via the DFT (equation 2.11). Signal (1) consists of four periods of an impulse train wave with a frequency of 2^{-8} . Its frequency domain representation reveals that it is made up of harmonics with equal magnitude at frequencies $(1 : 2^8 - 1)2^{-8}$. The kernel (2) is a so-called band-pass filter, alluding to the fact that its frequency response consists of a specific band of frequencies. Note how the convolution (3) in the frequency domain is simply a multiplication of (1b) and (2b), as stated by equation 2.10, while in the time domain (equation 2.9), each impulse of (1a) replicates the kernel (2a).

2 Fundamentals

Convolution (equation 2.7) is defined such that each value in the output signal is a weighted sum of values in the input signal. This definition can be generalized to include previous output signal values in the summation, yielding the *difference equation*,

$$y(k) = \sum_{l \in \mathbb{Z}} x(k-l)b(l) + \sum_{l \in \mathbb{N}} y(k-l)a(l) \quad 2.12$$

and its corresponding frequency response,

$$H_{b,a}(e^{2\pi i \omega}) = \frac{\sum_{l \in \mathbb{Z}} b(l)e^{2\pi i \omega l}}{\sum_{l \in \mathbb{Z}} a(l)e^{2\pi i \omega l}} \quad 2.13$$

If $a(l) = \delta_{0,l}$, equations 2.12 and 2.13 simplify to 2.7 and 2.8, respectively. (b, a) defines a so-called infinite impulse response (IIR) filter, since the result of the difference equation can always be written in terms of a convolution with a kernel of infinite support. Compared to regular convolution, IIR filters have the advantage of providing useful frequency responses with a relatively small number of coefficients (b, a) . Common examples include (discretized) Butterworth (1930) and Chebyshev (Weinberg & Slepian, 1960) filters. Figure 2.2 shows a selection of discretized Butterworth filter frequency responses.

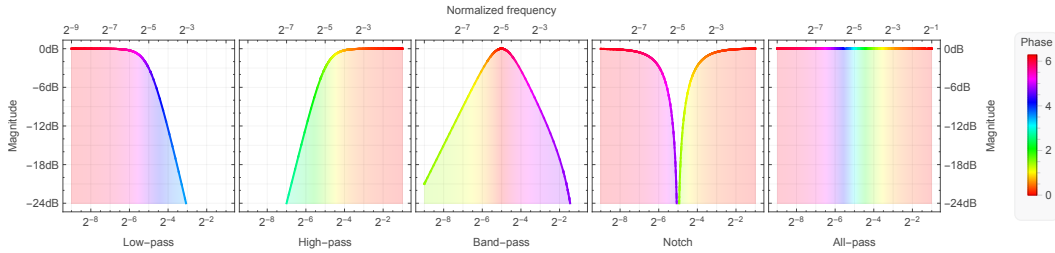


Figure 2.2: Common IIR filter frequency responses.

The frequency response of a 2-pole Butterworth filter with corner frequency 2^{-5} , discretized by the bilinear transform, is shown in each plot. The frequency response is given in polar representation, with the magnitude represented by the graph height and the phase represented by its hue. The filters are named by their pass-band, stop-band, and roll-off characteristics, e.g. a low-pass filter attenuates a band of frequencies above the corner frequency, while low frequencies are allowed to pass. The roll-off slope in the stop-band is proportional to the number of poles in the transfer function, in this case 2. An all-pass filter does not alter the magnitude of any frequency, but changes the phase around the corner frequency. Note that the discretization alters the original frequency response close to 2^{-1} , which is the reason for the asymmetric shape of the band-pass filter.

2.4 Correlation

Correlation is a measure of how linearly interrelated two variables are. While a variety of distinct cases and definitions exist, it suffices for the present thesis to only introduce the Pearson correlation coefficient (PCC) $r_{x,y}$ (Bravais, 1844; Pearson, 1895) for two discrete, real-valued sequences $x \in \mathbb{R}^n, y \in \mathbb{R}^n$ where $n \in \mathbb{N}, n > 1$, $\exists i \mid x_i \neq 0$, and $\exists i \mid y_i \neq 0$:

$$r_{x,y} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad 2.14$$

If the fraction of equation 2.14 is expanded by n^{-1} (or $(n-1)^{-1}$), the numerator becomes the biased (or unbiased) covariance between x and y and the denominator becomes the product of the biased (or unbiased) standard deviations of x and y , serving as a normalization factor. If $r_{x,y}$ is 1, 0, or -1, then x and y are said to be correlated, uncorrelated, or anticorrelated, respectively. The correlation of a sequence with itself is always 1, i.e. $r_{x,x} = 1$.

Since the value of $r_{x,y}$ indicates how similar x and y are, correlation can be used for template matching, that is, finding the occurrence of a signal in another signal. For this, one of the signals is shifted along its domain and the PCCs for a range of these shifts are computed. A high correlation indicates that one signal contains the other at the particular shift. This process is called *cross-correlation*. For the present thesis, the cyclic case is of particular interest: Given two periodic time sequences with the same period length, the cyclic cross-correlation finds the PCCs for all possible time shifts $i \in \mathbb{Z}_n$. It is defined as a function:

$$r_{x,y}(i) = r_{x,\text{cycshift}_i(y)} \quad 2.15$$

The (cyclic) cross-correlation of a sequence with itself, $r_{x,x}(i)$, is called the (cyclic) *autocorrelation*. Trivially, $r_{x,x}(0) = r_{x,x} = 1$.

Notice that the cyclic cross-correlation can be written in terms of cyclic convolution (equation 2.9):

$$r_{x,y}(i) = \frac{((y - \bar{y}) * (x - \bar{x}))_{i+2}}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad 2.16$$

2.5 Visual evoked potentials (VEPs)

Visual evoked potentials (VEPs) are electrophysiological responses elicited by visual stimuli that activate the visual cortex. When recorded by EEG electrodes placed over the posterior head regions of the observer, the general shape of these responses can be found by averaging out background activity over several recordings. A type of VEP commonly used as a reference in the literature is the pattern-reversal visual evoked potential (PRVEP). It is elicited by the color reversal of a black and white checkerboard pattern presented to the observer. It has a characteristic shape

2 Fundamentals

consisting of three peaks (N75, P100, and N135) occurring around 100ms after the pattern reversal. Potentials of similar shape are elicited by other light flash-type stimuli (cf. Markand, 2020, chapter 3).

In the context of BCIs, two types of VEPs are of particular interest: code-modulated visual evoked potentials (cVEPs) and steady-state visual evoked potentials (SSVEPs). cVEPs are elicited by complex flickering patterns of two or more colors, usually black and white. Their shape depends on the flickering pattern and can to some extent be regarded as a superposition of the PRVEP-type responses to the individual color changes (Yasinzai & Ider, 2020). SSVEPs are periodic potentials elicited by stimuli flickering at a constant rate. The SSVEP signal contains the flicker frequency and some harmonics (Herrmann, 2001). Figure 2.3 provides an illustration of common PRVEP and SSVEP shapes.

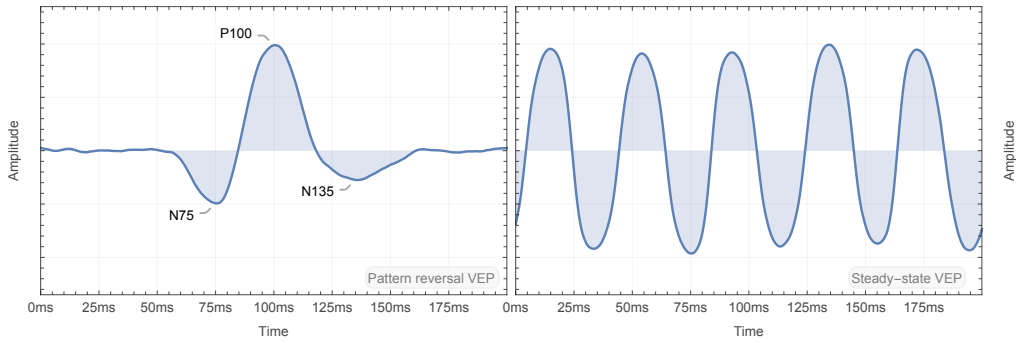


Figure 2.3: Sketches of a PRVEP and an SSVEP.

The plots show artificially created sketches of typical VEP shapes. The PRVEP consists of three characteristic peaks: a negative peak at roughly 75ms (N75), a positive peak at roughly 100ms (P100), and another negative peak at roughly 135ms (N135). The SSVEP has a steady sinusoidal pattern coinciding with the stimulation frequency, in this case 25Hz.

2.6 Brain-computer interfaces (BCIs)

Seeing as potentials evoked by behavior or external stimuli can be measured and distinguished, they can be used as a communication interface. Implementations of this idea, first described by Vidal in 1973, are called brain-computer interfaces (BCIs). A BCI defines an encoding of data in terms of distinct classes of potential patterns. If a particular potential pattern is elicited and recorded (e.g. by EEG), it can be decoded into the original data that it represents. This paradigm allows for the control of electronic devices through brain activity. Currently, applications of non-invasive BCIs mostly revolve around the assistance of patients in motor paralysis conditions such as stroke, spinal cord injury, cerebral palsy, and amyotrophic lateral sclerosis (Belkacem, Jamil, Palmer, Ouhbi, & Chen, 2020).

The present thesis focuses on VEP-based BCIs: A set of distinct visual stimuli, called targets, is presented to the user on a computer screen or by LEDs. If the

2 Fundamentals

user gazes at a particular target, its respective stimulus can be identified in the EEG signal, because the response to the other stimuli, located in the periphery of the visual field, will be smaller by comparison (Sutter, 1984). A command is assigned to each target and executed whenever the target is identified in the EEG. Usually, the BCI additionally needs to be able to detect the non-control state, i.e. when the user is not gazing at a target to issue a command. Hitherto, VEP-based BCIs commonly use stimuli eliciting SSVEPs or cVEPs to encode commands by flicker frequencies, phases, or pattern structure. A typical application would be a gaze-controlled keyboard.

2.7 Maximum length sequences (MLSs)

Maximum length sequences (MLSs), also called m-sequences, are a class of periodic pseudorandom k -ary code sequences that can be generated by linear feedback shift registers (LFSRs). Algorithm 2.1 represents a subroutine to compute one element of such a sequence at a time. Each iteration of the algorithm produces one **symbol** and returns the **state** to be used as the **seed** for the next iteration. If the *feedback polynomial* $\sum_{i=1}^n \mathbf{taps}_i^i + 1 \bmod k$ is primitive, the sequence generated is an MLS. It has a period of $k^n - 1$, where n is the order of the feedback polynomial.

Algorithm 2.1: **lfsr**

Computes one iteration of a k -ary LFSR.

Data:

with $k \in \mathbb{N}$, $n \in \mathbb{N}$

seed $\in \mathbb{Z}_k^n$

▷ Seed state for the current iteration

taps $\in \mathbb{Z}_k^n$

▷ Coefficient vector defining the LFSR and its feedback polynomial

Result:

symbol $\in \mathbb{Z}_k$

▷ Result of the current iteration

state $\in \mathbb{Z}_k^n$

▷ Seed state for the next iteration

symbol $\leftarrow \langle \mathbf{seed}, \mathbf{taps} \rangle \bmod k$

state $\leftarrow \text{cycshift}_1(\mathbf{seed})$

state[0] $\leftarrow \mathbf{symbol}$

MLSs exhibit several characteristic properties (Sarwate & Pursley, 1980; Golomb, 2017), including:

- (1) **Balance property:** The distribution of symbols contained in the sequence is approximately uniform.
- (2) **Run property:** The lengths of subsequences of identical symbols are distributed geometrically.
- (3) **Autocorrelation property:** The cyclic autocorrelation of one period of an MLS is a (shifted and scaled) Kronecker delta function (equation 2.2), that is, the autocorrelation has a value close to zero whenever the cyclic shift is nonzero.

2 Fundamentals

The autocorrelation property (3) is of particular interest to BCI applications, since it allows for the identification of targets based on cyclic shift: Up to $k^n - 1$ targets can present the same MLS with a unique time shift as their stimulation pattern. The resulting cVEPs can be cross-correlated with the MLS, only yielding a high correlation for the specific time shift corresponding to the target that the user gazed at. It should be noted that even though the cVEPs trace a nonlinear response to mostly the changes in the MLS stimulus, using the MLS itself for cross-correlation instead of an approximated response is viable. Initially proposed by Sutter in 1992, this technique has since been explored widely, including evaluations of the utilization of multiple MLSs, chromatic stimuli, higher frequencies, and higher arity (Hanagata & Momose, 2002; Bin, Gao, Wang, Hong, & Gao, 2009; Nezamfar et al., 2011; Aminaka, Makino, & Rutkowski, 2015; F. W. Gembler, Rezeika, Benda, & Volosyak, 2020).

Since it is relevant to the present thesis, let us briefly discuss to what extent the autocorrelation property is conserved when a (discrete) high-frequency carrier wave is modulated by a (binary) MLS $m \in \mathbb{Z}_2^n$. The result of such a modulation is a sequence $x \in \mathbb{R}^{nk}$ made up of two distinct types of segments: $s_0 \in \mathbb{R}^k$, representing the zeros in the MLS, and $s_1 \in \mathbb{R}^k$, representing the ones. For a reasonably constructed carrier, we can assume $\bar{s}_0 = \bar{s}_1 = 0$, which also implies $\bar{x} = 0$. Then, the autocorrelation in steps of k is

$$\begin{aligned} r_{x,x}(i) &= (nk - 1)^{-1} \sigma_x^{-2} \cdot \sum_{j=0}^{nk-1} x[j] x[(j + ik) \bmod (nk)] \\ &= \left(\sum_{l=0}^{n-1} (k - 1) \sigma_{s_m[l]}^2 \right)^{-1} \cdot \sum_{l=0}^{n-1} \sum_{j=lk}^{(l+1)k-1} x[j] x[(j + ik) \bmod (nk)] \\ &= \left(\sum_{l=0}^{n-1} \sigma_{s_m[l]}^2 \right)^{-1} \cdot \sum_{l=0}^{n-1} \sigma_{s_m[l]} \sigma_{s_m[l+i \bmod n]} r_{s_m[l], s_m[l+i \bmod n]} \end{aligned}$$

where $\sigma_x, \sigma_{s_0}, \sigma_{s_1}$ are the unbiased standard deviations of x, s_0, s_1 , respectively. It becomes apparent that the conservation of the autocorrelation property depends upon the correlation between s_0 and s_1 as well as the relationship between their standard deviations. If the two types of segments are anticorrelated, $r_{s_m[l], s_m[l+i \bmod n]} = (2m[l] - 1)(2m[l+i \bmod n] - 1)$, replicating the autocorrelation of an MLS. Thus, if $\sigma_{s_0} = \sigma_{s_1}$, the autocorrelation property will remain intact. The more correlated they become and the less similar their standard deviations, the stronger the autocorrelation for $i \neq 0$. Figure 2.4 illustrates this with two modulation schemes: on-off keying (OOK) and binary phase-shift keying (BPSK).

2 Fundamentals

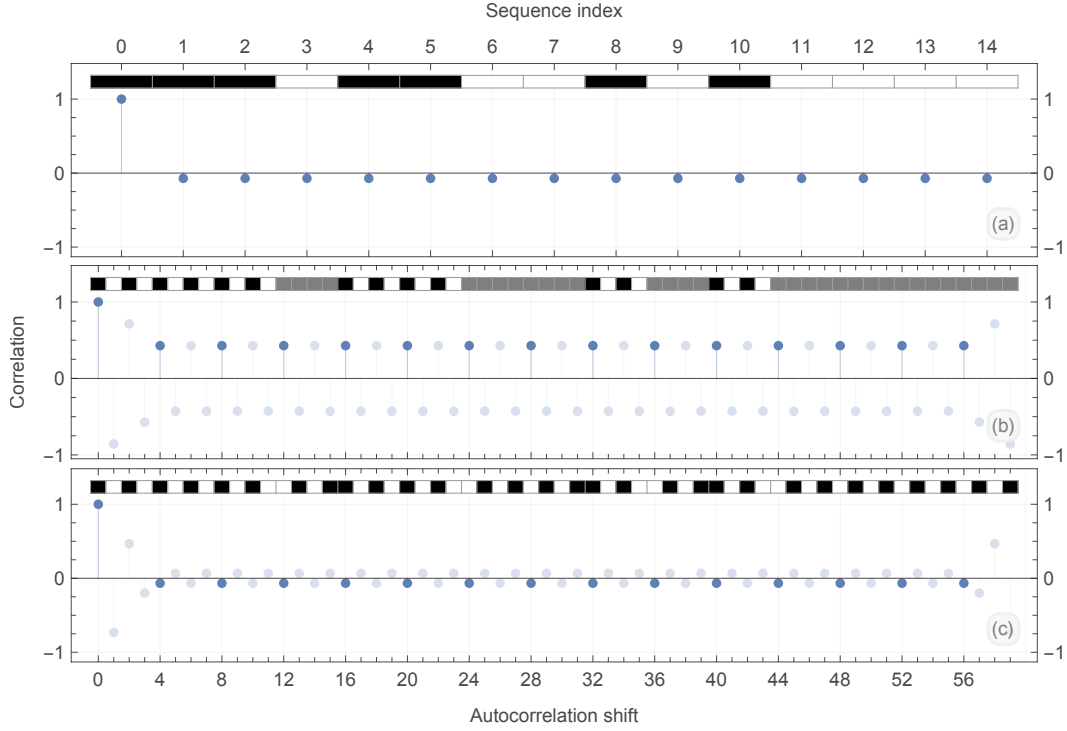


Figure 2.4: Autocorrelation property of MLSs encoded by OOK and BPSK.

Each plot shows a sequence (grayscale pattern) and its cyclic autocorrelation (blue dots). The top sequence (a) is an MLS of length 15. It can be seen that $\forall i \mid i \bmod 15 \neq 0 \iff r(i) \approx 0$. The middle sequence (b) is a carrier wave with a frequency of half the sampling rate (Nyquist frequency), OOK-modulated by the same MLS such that the resulting segments have an amplitude of either 0 or 1. Since the two types of segments are uncorrelated (and occur in roughly equal amounts due to the balance property (1)), the autocorrelation property loses about half of its power. For the bottom sequence (c), the segments have amplitudes 1 and -1 , effectively resulting in phase inversion. These two types of segments have equal standard deviation and are perfectly anticorrelated, thus the autocorrelation property remains intact.

2.8 Information transfer rate (ITR)

Given a communication channel that transmits n -ary symbols with a success probability of p at a frequency of f , the number of bits transmitted on average over time can be computed by

$$\text{ITR} = f \cdot \left(\log_2(n) + p \log_2(p) + (1 - p) \log_2\left(\frac{1 - p}{n - 1}\right) \right) \quad 2.17$$

This quantity is called the information transfer rate (ITR) and was first introduced by Shannon in 1948. In the context of BCIs, it serves as a metric for performance comparison and is usually given in bits per minute.

2.9 Spatial filtering

BCIs typically use multiple EEG electrodes, resulting in a multi-channel signal. This allows for the exploitation of inter-channel relationships in order to increase the SNR. One common method is (linear) spatial filtering, where one or several linear combinations of the channels are sought in order to reduce dimensionality while enhancing the signal of interest. Given a time-discrete EEG recording $X \in \mathbb{R}^{k \times n}$ with n time points and k channels, the spatially filtered signal $Y \in \mathbb{R}^{l \times n}$ is calculated by

$$Y = WX \tag{2.18}$$

where $W \in \mathbb{R}^{l \times k}$ is the spatial filter. While finding a W that maximizes the SNR can be seen as a general optimization problem solvable by arbitrary regression methods (Spüler, 2017), a common approach is to use canonical correlation analysis (CCA). The latter has been shown to improve performance in BCIs based on SSVEPs (Bin, Gao, Yan, Hong, & Gao, 2009), cVEPs (Bin et al., 2011), and other event-related potentials (ERPs) (Spüler, Walter, Rosenstiel, & Bogdan, 2013). Spatial filters may in theory be generalized to LSI and non-linear systems. However, the present thesis neither made use of generalized spatial filters nor CCA – instead, a composite temporal and spatial filter was learned by means of ridge regression.

2.10 The sliding window model

The sliding window model, first proposed by Nagel, Rosenstiel and Spüler in 2017, is a paradigm for recovering arbitrary stimulation patterns from their elicited VEP response. Each time point of the stimulation pattern is predicted from its local temporal neighborhood in the EEG signal by a machine learning model. Specifically, given a time-discrete, possibly pre-processed, k -channel EEG response signal $x \in \mathbb{R}^{k \times n}$ and a corresponding stimulation pattern $y \in \Omega^n$, the relationship between x and y is modeled by a function $f: \mathbb{R}^{k \times b} \rightarrow \Omega$ such that

$$y[i] = \epsilon(f(x[:, i+a : i+a+b]), \alpha) \tag{2.19}$$

where $a \in \mathbb{Z}$ and $b \in \mathbb{N}$ are the window offset and size, respectively, and ϵ is an error function with unknown influences α , whose impact is to be minimized. f can be learned by an arbitrary machine learning model, e.g. Nagel (2019) has successfully constructed sliding window models employing ridge regression and convolutional neural networks (CNNs).

Analogously, a backward model can be trained on windows into the stimulation pattern to predict the elicited VEP response (Nagel & Spüler, 2018). Ideal backward and forward models would hence allow to perfectly simulate the brain's VEP response to stimuli and convert back and forth between stimulus and response.

2.11 Flicker fusion thresholds

Consider a flickering visual stimulus whose luminance or chrominance follows a sinusoidal pattern. At high flicker frequencies, the stimulus appears to be completely steady. This phenomenon is called *flicker fusion*. The *flicker fusion threshold* is defined as the lowest frequency where flicker fusion takes place. The flicker fusion threshold differs between people and furthermore depends on a variety of variables, including viewing distance, stimulus size, modulation amplitude, maximum luminance, color spectra at the sinusoid's extrema, and retinal location (Brenton, Thompson, & Maxner, 1989). Thresholds are commonly reported to be in the range of 35 Hz to 60 Hz for achromatic flickering (Hecht & Shlaer, 1936; de Lange Dzn, 1954) and 10 Hz to 15 Hz for chromatic flickering (Wisowaty, 1981; Schiller, Logothetis, & Charles, 1991). A distinction between the perception of a steady luminance/chrominance and the complete imperceptibility of flickering is generally not made, potentially blurring the measurements.

3 Related Work

The aim of the present thesis is to pave the way for ergonomic computer screen-based BCIs with adequate performance by introducing a novel class of stimulation patterns. Previous work has tended to result in a less than satisfactory trade-off between ITR and stimulus non-recognizability, with approaches usually not being able to fully achieve flicker fusion and instead focusing on altering the structure of SSVEP and cVEP stimulation patterns to reduce user discomfort.

With LEDs, eliminating stimulus recognizability through SSVEP stimulation above the flicker fusion threshold is trivial and has shown considerable success: Sakura et al. (2015) and Won et al. (2015) both demonstrated that high-frequency LED flicker reduced visual fatigue in users of an SSVEP-based BCI, while also yielding a higher classification accuracy and stability compared to lower stimulation frequencies. Implementing high-frequency SSVEP-based BCIs on computer screens, however, proves more difficult, since the frame rate and aliasing limits the frequencies and phase shifts available for use (cf. section 4.3). Benda et al. (2019) proposed a custom-made monitor to solve this problem. Beyond that, few studies exist on BCIs employing imperceptible SSVEP stimulation on computer screens. Blöck (2018) created 60Hz stimuli that appeared to the observer as steady pictures (e.g. a photograph) and was able to distinguish 2 classes of SSVEPs. Jiang et al. achieved a mean ITR of $19.95^{\text{bit}}/\text{min}$ in a 2019 study using a refresh rate of 240Hz to elicit four-class phase-coded SSVEPs at 60Hz. As an alternative to flicker fusion, stimulation patterns eliciting sub-flicker fusion threshold SSVEPs may be adapted to being less obviously flickering. For instance, a sampled sinusoid (Manyakov et al., 2013) produces more subtle flickering than a black and white square wave pattern, although still being recognizable. More notable, Armengol Urpi and Sarma (2018) achieved a low perception of flickering and beating from stimulation at 42, 43, 44, and 45Hz by lowering the stimulus amplitude.

Eliminating the perception of flickering from screen-based cVEP stimulation is no less challenging, because the stimulation patterns tend to contain low frequencies, even at high presentation rates. A paradigm known as quasi-steady-state VEP, introduced by Kaya et al. in 2019, may be a candidate for non-recognizable presentation of code sequences, but has thus far only been studied with LEDs and frequency content below the flicker fusion threshold (Kaya, Bohorquez, & Ozdamar, 2019 and 2021; Shirzhiyan et al., 2020). Other studies have evaluated the use of various cVEP stimulation patterns and presentation rates to increase user comfort. In 2015, Aminaka et al. found that using chromatic (green and blue) stimuli resulted in less discomfort and similar performance compared to achromatic stimulation. Aminaka and Rutkowski also stated in 2017 that MLS stimulation patterns presented at 40Hz by LEDs are “very perceptually friendly” and can successfully be used for

3 Related Work

BCI control. In 2019, Gembler et al. evaluated the presentation of MLS on a computer screen at 30, 60 and 120Hz and found that the 120Hz stimuli resulted in more subtle flickering as well as higher ITRs. Shirzhiyan et al. demonstrated in 2019 that chaotic codes as stimulation patterns result in less visual fatigue compared to MLSs while not sacrificing performance. Gembler et al. obtained similar results with the use of quintary MLSs in 2020.

The class of stimulation patterns proposed by the present thesis is designed to constitute a general communication channel, capable of transmitting n -ary digital codes of arbitrary length. This paradigm has already been applied with success by Nagel and Spüler in 2018 and 2019: The authors were able to predict arbitrary binary sequences, presented as traditional (recognizable) black and white flickering stimuli, with high accuracy.

4 Conceptualization

The purpose of this chapter is to introduce the idea of CCVEP stimulation. First, the motivation for, as well as interpretation of, stimulus non-recognizability will be discussed (section 4.1). Next, a characterization of the CCVEP paradigm will be given (section 4.2). Finally, some caveats around the implementation of the paradigm on computer displays are elucidated (section 4.3).

4.1 On stimulus recognizability

The particular class of stimuli evaluated by the present thesis is geared towards minimizing conscious recognizability as a proxy for cognitive and physical adverse effects. In particular, it utilizes flicker fusion (cf. section 2.11) as a means to obfuscate stimuli. It is important to note that the assumption that this non-recognizability of high-frequency stimuli implies absence of adverse effects is already known to not be universally true. While the alleviation of discomfort compared to traditional cVEP-based BCIs should be obvious to any user, high-frequency flicker may still lead to complications (Wunsch, 2006; Wilkins, Veitch, & Lehman, 2010). No clinical outcomes or subjective measures of comfort were examined over the course of the study. However, there do exist promising evaluations of user comfort for high-frequency SSVEP-based BCIs in the literature (Armengol Urpi & Sarma, 2018; Jiang et al., 2019).

Another important point is that the term *conscious recognizability* is used in a loose fashion here. Untangling the complexities of true conscious and subconscious (im-)perceptibility is nontrivial (Meyen, Zerweck, Amado, von Luxburg, & Franz, 2021) and arguably not necessary for this work – the goal is simply to minimize the user’s awareness of and physiological response to the stimulus while still eliciting VEPs for use as a communication channel. Imperceptibility of the stimuli is neither proven nor even fully achieved. Section 4.3 discusses some of the reasons why completely eliminating recognizability of the stimuli is nontrivial and goes beyond the scope of the thesis. However, subjectively, the stimuli are only marginally perceptible and will hopefully serve as a basis for more ergonomic BCIs.

4.2 Code-modulated carrier visual evoked potentials (CCVEPs)

The present thesis proposes a novel class of stimuli and corresponding evoked potentials, namely code-modulated carrier visual evoked potentials (CCVEPs). Borrowing concepts from baseband and passband modulation, the idea is to present a chain of

4 Conceptualization

brief segments of high-frequency SSVEP-type stimuli, thereby transmitting a digital code. cVEP- and CCVEP-based BCIs can both be conceived of as a communication channel capable of transmitting error correction codes (ECCs). However, while cVEP-type stimuli transmit the code as is, CCVEPs encode the data using a carrier frequency, ideally achieving flicker fusion. To date, it has been shown that the absence and presence of an SSVEP can be distinguished, and that distinct frequencies and phase shifts can be recovered from the EEG signal (Armengol Urpi & Sarma, 2018; Jiang et al., 2019). Furthermore, it is conceivable that distinct duty cycles of pulse waves may be distinguishable as well, either based upon their signature spectrum or their time domain representation, if it is the case that recovery of arbitrary binary codes from cVEPs (cf. Nagel & Spüler, 2019) generalizes to higher temporal resolutions. This leaves us with a framework for non-recognizable stimuli that utilizes frequency, phase, amplitude, and/or pulse width characteristics of a high-frequency carrier to encode symbols of a digital code.

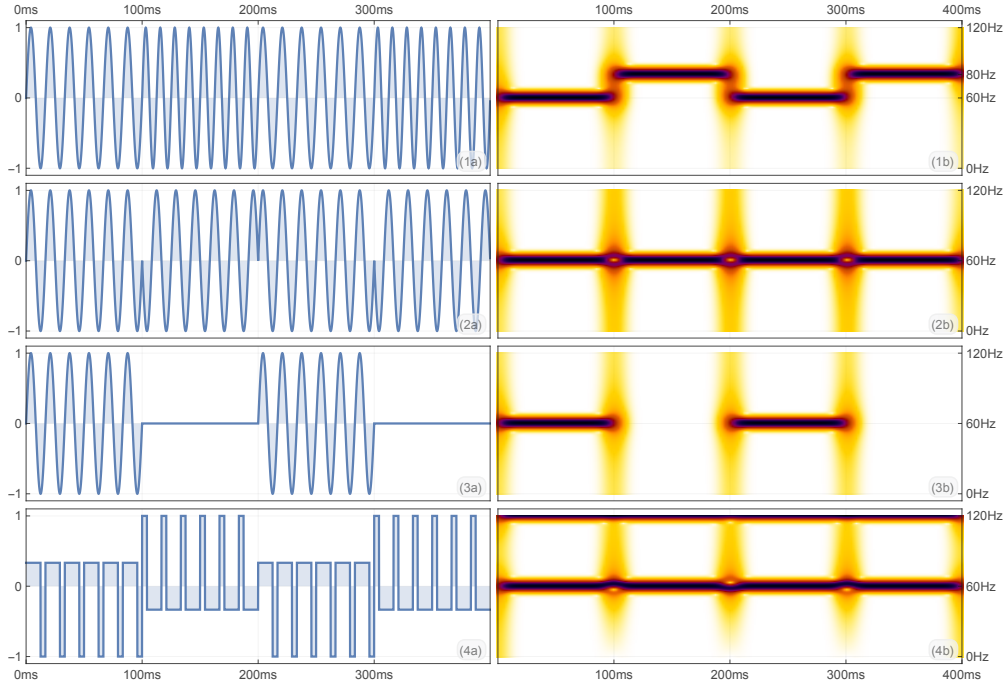


Figure 4.1: Schematic illustration of naïve CCVEP stimulation patterns.

From top to bottom: segments encoding symbols by varying the carrier’s frequency (1), phase (2), amplitude (3), and pulse width (4). The four modulation schemes can be called binary frequency-shift keying (BFSK), binary phase-shift keying (BPSK), on-off keying (OOK), and pulse-width modulation (PWM), respectively. The left column (a) depicts the signals in the time domain and the right column (b) depicts their spectrograms (i.e. magnitude of each frequency at each point in time). Note that since the segments are naïvely concatenated without any additional boundary handling, significant amounts of low frequency content exists at the transition between segments.

4 Conceptualization

Figure 4.1 shows four schemes of stimulation patterns designed to elicit CCVEPs. Each stimulation patterns is comprised of a potentially infinite stream of segments with a fixed length. In the figure, four such segments are shown per plot. In order for the stimuli to be non-recognizable as a whole, each segment must appear to the observer as the same steady color. Calibrating the exact colors presented by each segment type may be necessary. Given the segment length and the correct time offset, a classifier should be able to decode each individual segment from the evoked potential into an n -ary symbol based on its frequency (1), phase (2), amplitude (3) or pulse width (4) characteristics. In this case, $n = 2$, because only two segment types exist per stimulation pattern. n could be increased by including additional distinct carrier frequencies, phase shifts, etc. Furthermore, it may as well be possible to combine the schemes (1) through (4) to yield even more complex signals, further increasing n . Note that much of this is a purely hypothetical proposal at this point. The present thesis will merely go on to evaluate schemes (2) and (3) separately, each with $n = 2$.

The right column (b) of figure 4.1 shows the spectrograms for the respective stimulation patterns. Note that even though the frequency content of the stimulation patterns is mostly concentrated above a reasonable flicker fusion threshold, low frequencies do occur in minor but still significant amounts, particularly at the transitions between segments. This is a reason for concern when non-recognizability is the goal. Indeed, if these stimulation patterns were presented on a screen, the segment transition points would appear as light or dark flashes to the viewer. Hence, there is a need for more sophisticated handling of the transitions in order to conceal such artifacts.

The stimulation patterns as shown in the figure can be understood as a carrier modulated by a stepped function, allowing only discrete jumps. Smoothing this step function would be an obvious approach to minimize artifacts. For instance, in scheme (1), this would mean replacing the hard transitions by brief frequency sweeps, interpolating between 60Hz and 80Hz. However, this proves challenging on a computer screen (cf. section 4.3).

Another possibility would be to apply a window function to each segment, such that the carrier fades from zero to its full amplitude and back. Typical window functions from signal processing such as the Tukey (Bloomfield, 2004) or Kaiser (1966) windows would serve as a starting point, since these aim at optimizing the trade-off between time and frequency resolution while being adjustable to match the psychophysical constraints. Figure 4.2 shows how windowing is able to virtually eliminate low frequencies around segment transitions. This can further be generalized to simply fading to zero between distinct segments and not fading between segments of the same type, since there is no apparent transition in the latter case. An example of CCVEP stimulation patterns utilizing this technique can be seen in section 5.2, figure 5.2.

4 Conceptualization

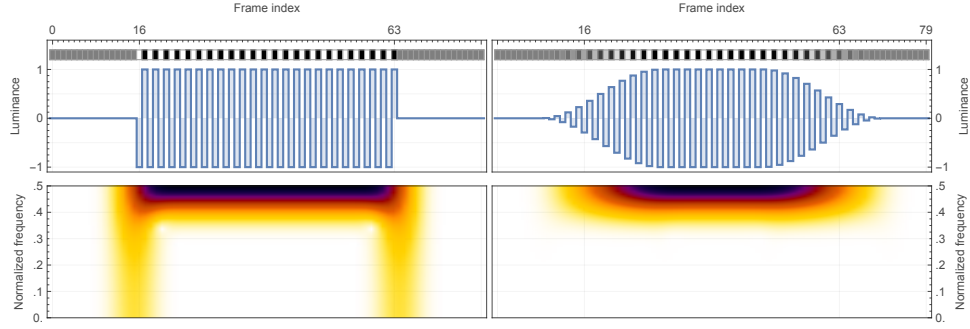


Figure 4.2: Application of windowing/fading to segment transitions.

The top row shows a flickering stimulation pattern as time domain signal (blue curve) and represented as achromatic frame-wise colors (grayscale squares above). On the left, the stimulation pattern contains hard segment transitions (equivalent to a rectangular window). On the right, the stimulation pattern is blended in and out smoothly by a Tukey window function. The bottom row shows the corresponding spectrograms. Note the lack of low frequency content when windowing is applied.

4.3 CCVEP challenges on computer screens

While a CCVEP-based BCI could absolutely be implemented using LEDs, the present thesis focuses on computer displays to present the stimuli on. The obvious advantage to this is that the BCI can be seamlessly integrated into a software environment. However, digital screens come with several limitations, complicating the accurate display of stimuli:

- (1) time-discrete signal with limited frame rate and poor digital-to-analog conversion (DAC),
- (2) non-linear relationship between digital color values and actual irradiance,
- (3) dependency of pixel irradiance on viewing angles,
- (4) non-uniform raster latency,
- (5) other nontrivial artifacts, caused by a variety of mechanisms such as motion blur reduction.

Issues (1) and (2) result in the inability to properly render frequencies that do not have an integer multiple equal to the monitor refresh rate. Such frequencies will create aliasing in the form of visible beating patterns, illustrated by figure 4.3. Given common monitor refresh rates such as 120Hz or 144Hz, this greatly limits the number of viable frequencies to use for non-recognizable CCVEP stimuli.

4 Conceptualization

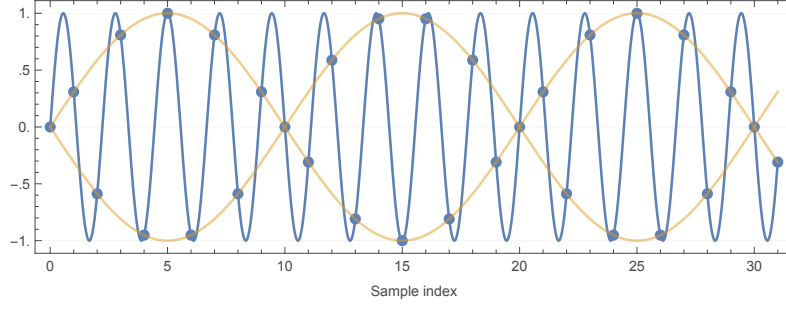


Figure 4.3: Beating patterns in time-discrete signals.

A time-discrete sampling (blue dots) of a time-continuous signal (blue curve) contains patterns of other frequencies (yellow curves) if the sample rate is not an integer multiple of the sampled signal's frequency. Nonlinearities and subideal DAC can create signals that actually contain such unwanted frequencies.

Issues (3), (4) and (5) complicate the proper rendering of signals with varying amplitudes. Figure 4.4 illustrates what different amplitudes of high-frequency grayscale flicker may look like on a typical monitor. This poses a substantial problem when CCVEP segments are to be concealed by fading.

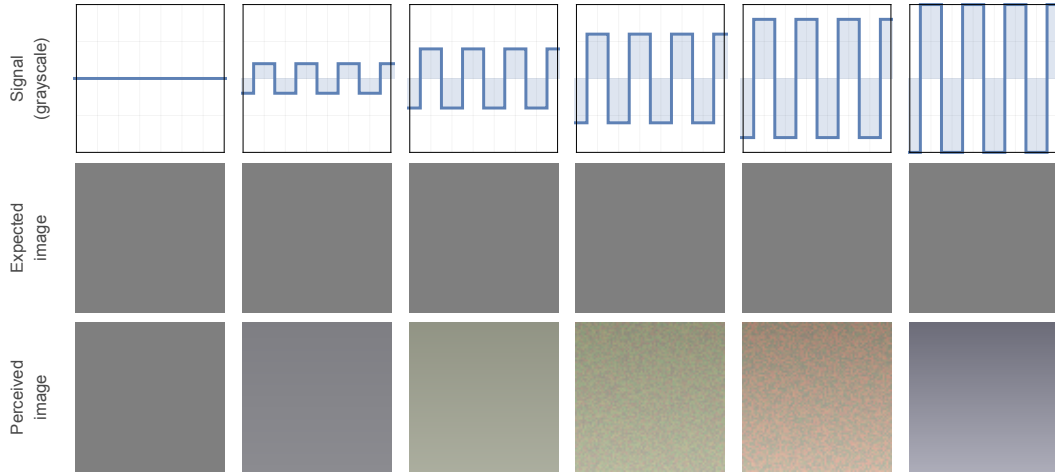


Figure 4.4: Flicker rendering artifacts on a typical monitor.

The blue curves in the top row plots represent grayscale values over time. Each column represents a specific flicker amplitude, increasing from constant gray on the left to full black and white flicker on the right. On an ideal monitor, the resulting perceived image should look the same for all amplitudes if the flicker frequency is above the flicker fusion threshold. The bottom row is a somewhat exaggerated illustration of what the different amplitudes may actually look like on a real-world monitor. Note the changes in chrominance, the noise artifacts, the dependence of the luminance on the vertical viewing angle for high amplitudes, as well as the fact that the general luminance of maximum flickering may not correspond to the constant gray color, depending on monitor calibration.

4 Conceptualization

A compounding issue is that flicker fusion thresholds as well as the perceived flicker fusion color may vary from person to person and likely even from time to time. Furthermore, stimulus perception itself is a nonlinear system. Thus, while ideal intra-segment steady-state signals and ideal fades from one CCVEP segment to the next for a given person, time and monitor may exist in theory, determining them is certainly a nontrivial optimization problem which goes beyond the scope of the present thesis. At the same time, however, it is manageable to calibrate monitor colors and inter-segment fades to produce a CCVEP stimulus that approaches non-recognizability to a reasonable degree.

5 Materials and Methods

This chapter will detail the acquisition and evaluation of the experimental data. First, the equipment used for the experiment will be listed (section 5.1). Next, the particular stimuli chosen for evaluation and the experiment execution will be described (sections 5.2 and 5.3). Lastly, the design of the machine learning model as well as the evaluation procedures will be explained (sections 5.4 and 5.5).

5.1 Hardware and Software

- Two personal computers were used: one to display the stimuli (PC0) and one to record EEG data (PC1).
- The LCD monitor used for displaying stimuli was a BenQ XL2430-B. It was set to its native resolution of 1920×1080 pixels with a color resolution of 8 bit per RGB channel and a refresh rate of 120Hz.
- The stimuli were presented by a custom-made BCI framework based on SFML (Gomila, 2021) in C++17 (ISO, 2017).
- The EEG was recorded using a Brainproducts Acticap system with 32 channels, a g.USBamp (g.tec, Austria) amplifier, and the software BCI2000 (“BCI2000”, 2021; Schalk, McFarland, Hinterberger, Birbaumer, & Wolpaw, 2004). The sampling rate was 600Hz.
- The classification and evaluation routines were implemented in Julia 1.6.2 (The Julia Project, 2021; Bezanson, Edelman, Karpinski, & Shah, 2017).

5.2 Stimulus design

The stimuli evaluated by the present thesis were designed to be presented on a display with a refresh rate of 120Hz. This places the carrier frequency of choice at 60Hz, i.e. half the refresh rate. This is sufficiently high to achieve flicker fusion, while still being adequately detectable in the EEG compared to even higher stimulation frequencies. The stimuli were achromatic, that is, a calibrated grayscale coloring was used.

One particular binary code sequence of length 15, shown in figure 5.1, was chosen for encoding as a CCVEP-eliciting stimulation pattern. Let this sequence be denoted as a vector $\mu = (1, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0)^T$. μ was deliberately chosen to be an MLS for its pseudorandom nature, balance and run properties, as well as to allow for an alternative classification strategy based on its autocorrelation property (cf. sections 2.7 and 5.4).

5 Materials and Methods

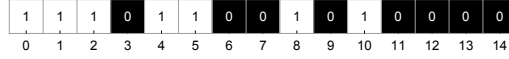


Figure 5.1: The code sequence μ to be encoded as stimulus.

μ was encoded by means of 2 CCVEP modulation schemes and 3 segment lengths per scheme, yielding 6 distinct encodings to be evaluated during the experiment. The first scheme was on-off keying (OOK), that is, segments either had a peak-to-peak amplitude of 0, i.e. constant gray, or the maximum amplitude of 1, i.e. full black and white flicker (cf. figure 4.1, row (3)). The second scheme was binary phase-shift keying (BPSK), that is, segments were consistently flickering at maximum amplitude, but were distinguished by a phase shift of either 0 or π (cf. figure 4.1, row (2)). For both schemes, three segment lengths were tested: 15, 30 or 45 frames, corresponding to $1/8$, $2/8$, or $3/8$ of a second at a refresh rate of 120Hz, respectively. Algorithm

Algorithm 5.1: encode

Encodes a ternary word as a stimulation pattern using OOK and/or BPSK.

Data:

with $n \in \mathbb{N}$

▷ Length of the input word

word $\in \{-1, 0, 1\}^n$

▷ The input word to be encoded

segment_length $\in \mathbb{N}$

▷ Number of frames to encode a single symbol

Result:

pattern $\in [0, 1]^{n \cdot \text{segment_length}}$

▷ Lightness for each frame

foreach **i**, **symbol** in $0 : n$, **word** **do**

start = **i** · **segment_length**

▷ First index,

stop = **start** + **segment_length**

▷ index one-past-the-end,

range = **start** : **stop**

▷ ...and index range of the current segment

switch **symbol** **do**

case 0 **do**

 | **pattern**[**range**] = $(\frac{1}{2})_{k=1}^{\text{segment_length}}$

▷ ..., $\frac{1}{2}$, $\frac{1}{2}$, $\frac{1}{2}$, $\frac{1}{2}$, ...

end

case 1 **do**

 | **pattern**[**range**] = $(k \bmod 2)_{k=\text{start}}^{\text{stop}-1}$

▷ ..., 0, 1, 0, 1, ...

end

case -1 **do**

 | **pattern**[**range**] = $(k \bmod 2)_{k=\text{start}+1}^{\text{stop}}$

▷ ..., 1, 0, 1, 0, ...

end

end

 ...create adequate fading in the beginning and end of the segment...

end

5 Materials and Methods

5.1 provides a sketch of how these encodings were calculated.¹ The set of resulting stimulation patterns (vectors of lightness values for each frame) can be denoted as $\{\text{encode}(\text{word}, \text{segment_length}) \mid \text{word} \in \{\mu, 2\mu - 1\}, \text{segment_length} \in \{15, 30, 45\}\}$. Figure 5.2 shows plots of the resulting stimulation patterns for a segment length of 15.

The codomain of the stimulation patterns was mapped to a calibrated grayscale using the color function $\text{cf}: [0, 1] \rightarrow [0, 1]^3$ with a parameter $c_{\text{mid}} \in [0, 1]^3$ that corresponds to the constant color of non-flickering segments. Given $c_{\text{low}} = (0, 0, 0)^T$

¹Note that while `encode` is designed for the general case of ternary symbols and thus able to produce stimulation patterns that include all three segment types (non-flickering, flickering, and phase-inverted flickering), in the case of the present thesis it is only used to encode binary symbols using either pure OOK or pure BPSK.

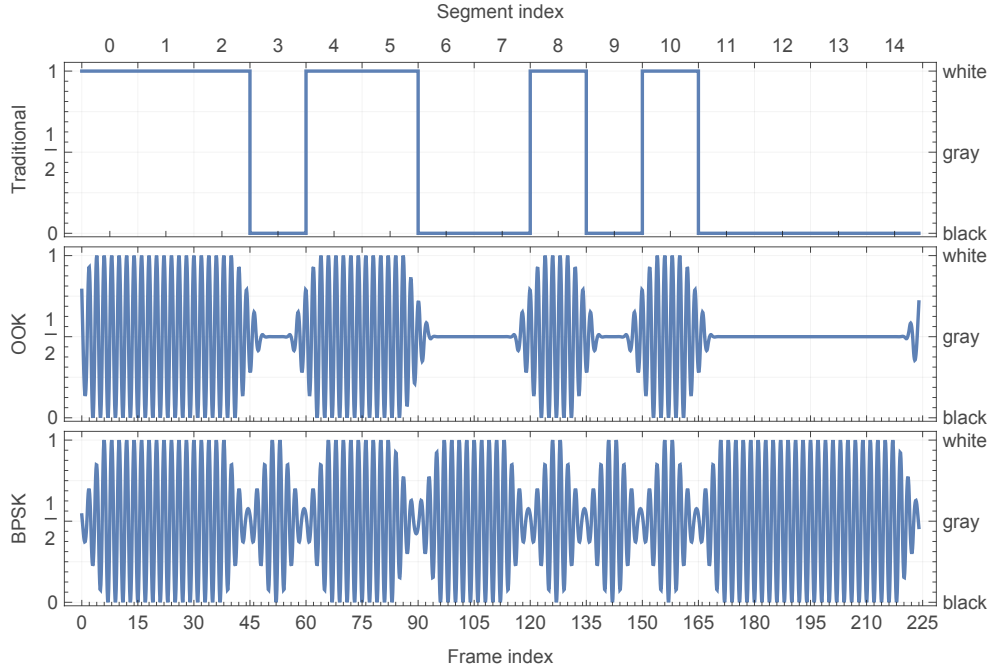


Figure 5.2: Stimulation patterns presented in the experiment.

The top plot shows the code sequence μ as it would be presented by a traditional cVEP-based BCI. The middle and bottom plots show μ encoded as a CCVEP-eliciting stimulation pattern with a segment length of 15, using OOK and BPSK, respectively. The stimulation patterns are designed to run in a loop. Note, however, that since the length of the whole stimulation pattern is odd in this case, the phase at the end (frame index 224) does not match with the phase at the start (frame index 0). Consequently, when looping, every other iteration must have its phase inverted. Also note that for the sake of better illustration, the shape of the fades differs slightly from what was used in the actual experiment.

5 Materials and Methods

and $c_{high} = (1, 1, 1)^T$ as RGB colors for black and white, respectively, cf can be defined as follows:

$$cf(x) = \begin{cases} 2x \cdot (c_{mid} - c_{low}) + c_{low} & \text{if } x \leq \frac{1}{2} \\ (2x - 1) \cdot (c_{high} - c_{mid}) + c_{mid} & \text{if } x > \frac{1}{2} \end{cases} \quad 5.1$$

cf simply interpolates linearly between black, c_{mid} , and white. If c_{mid} is adjusted properly, flickering and non-flickering segments will appear the same to the observer. However, to also fully obfuscate the fades between segments, more extensive calibration would be needed (cf. section 4.3).

Finally, individual color channel values were discretized to 8 bit resolution by $d: [0, 1] \rightarrow \mathbb{Z}_{255}$:

$$d(x) = \lfloor 255x + \frac{1}{2} \rfloor \quad 5.2$$

5.3 Experiment design

To evaluate the viability and performance of the stimuli described in section 5.2, 7 test subjects (age 22 ± 2.3 years, range 19 to 26 years, 4 male and 3 female by biological sex) were recruited. All subjects had normal or corrected-to-normal vision, no known neurological conditions and no prior history of epilepsy. Some of the subjects had participated in other VEP-related EEG experiments before.

The EEG layout comprised 32 electrodes which were positioned in and around the vicinity of the visual cortex at Fz, T7, C3, Cz, C4, T8, CP3, CPz, CP4, P5, P3, P1, Pz, P2, P4, P6, PO9, PO7, PO3, POz, PO4, PO8, PO10, O1, POO1, POO2, O2, OI1h, OI2h, Iz, PPO5h, and PPO6h. The ground electrode (GND) was positioned at FCz, the reference electrode (REF) at OZ. The 32 channels were each sampled at a rate of 600Hz and subsequently processed by an 8-pole Butterworth high-pass filter at .1Hz, an 8-pole Butterworth low-pass filter at 100Hz, and a notch filter around 50Hz, consisting of 4-pole Butterworth low-pass and high-pass filters at 48Hz and 52Hz, respectively, to eliminate interference from the European power line frequency.

Subjects were seated approximately 70cm before the screen and asked to avoid blinking and other movements during the relevant parts of the recording. For the duration of the experiment, all ambient light was either turned off or largely blacked out, with the only major remaining light source being the screen itself. While sounds from inside the test chamber were kept to a minimum, ambient noise from outside was merely attenuated by the building walls, door, and windows. Ambient electromagnetic fields (EMFs) were not addressed.

As an initial step, c_{mid} was determined to achieve a basic color calibration by means of the color function cf (equation 5.1): Two horizontally adjacent squares were presented in the center of the screen, the left one having a constant gray color, the right one flickering between black and white at 60Hz. The RGB values of the left square's color were adjustable in steps of $1/255$ using the keyboard. For convenience, two additional keys were assigned to increment and decrement all channels at once.

5 Materials and Methods

Subjects were asked to adjust the color of the left square to match the perceived color of the right square. The resulting color was then used as c_{mid} for the remainder of the experiment.

For the main phase of the experiment, a row of five flicker fields² was presented on the screen. A screenshot of this can be seen in figure 5.3. The task of the participants was to gaze at these fields while stimuli were presented to obtain a total of 300 takes over a time frame of roughly 55min. Each take consists of either 2 or 3 repetitions of the code sequence μ , encoded with one particular encoding scheme (cf. section 5.2) per take. Each of these repetitions is called a trial, i.e. a trial comprises the recording of one full 15-segment CCVEP, and a take is a concatenation of several trials. Table 5.1 lists the number of trials that were shown for each of the six encoding schemes.

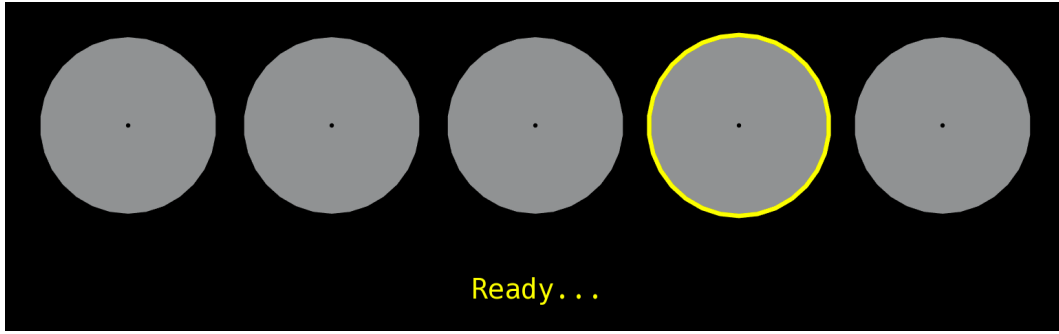


Figure 5.3: Graphical layout for stimulus presentation.

The layout consists of a row of five circular flicker fields and an area underneath for displaying textual instructions. The fields are laid out horizontally so as to mitigate flicker fusion luminance shifts due to a varying vertical viewing angle, as well as raster latency dependent phase shifts (cf. Nagel, Dreher, Rosenstiel, & Spüler, 2018). Each field has a diameter of 256 pixels, corresponding to approximately 71mm on the screen. In the center of every field, a small black dot aids the observer in maintaining a steady focus. The screenshot was taken just before the start of the next take, hence, one of the fields' outlines is highlighted in yellow, indicating that the test subject is supposed to gaze at this field next.

After the EEG recording was started, subjects had to press the space bar to signal readiness. Then, the following three steps were repeated 8 times:

- (1) Reset all flicker fields to show the constant gray color c_{mid} . Pause for 1s to allow subjects to blink.
- (2) Randomly select one of the flicker fields, excluding the field that was selected previously. Highlight the outline of the field in yellow for 1s, indicating that the subject should gaze at this field next.
- (3) Present the full stimulus for one take on the selected flicker field while highlighting its outline in green. Present cyclically shifted (by 3, 6, 9, and 12

²The term *flicker field* shall be used here to describe an area on the screen that presents a stimulus by changing its color on a frame-by-frame basis.

5 Materials and Methods

Mod. scheme	<code>segment_length</code>	Loops	Takes	Total trials	Total time
OOK	$1/8$ s	3	50	150	281.25s
OOK	$2/8$ s	2	50	100	375s
OOK	$3/8$ s	2	50	100	562.5s
BPSK	$1/8$ s	3	50	150	281.25s
BPSK	$2/8$ s	2	50	100	375s
BPSK	$3/8$ s	2	50	100	562.5s
Total:			300	700	~ 40 min

Table 5.1: Number and duration of trials by encoding scheme.

The left two columns, *modulation scheme* and *segment length*, form the encoding scheme. For each encoding scheme, 50 takes were recorded. A take consists of 2 or 3 repetitions of the respective stimulation pattern. The right two columns, *total trials* and *total time*, list the number of full 15-segment stimulation patterns shown per encoding scheme, as well as their total duration. The bottom row lists the total number of takes and trials across all encoding schemes as well as the total time of stimulation throughout the experiment.

segments) versions of the stimulus on the other four fields. Let the subject gaze at the center of the highlighted field.

After 8 of these takes, the subjects had to press the space bar again, allowing them to take a break of arbitrary length before continuing with the next set of 8 takes. This was repeated until all 300 takes were recorded. The order of the takes was randomized to mitigate the impact of potential confounding variables. No mechanism for control of subject adherence was implemented, i.e. subjects were merely trusted, but not verified, to gaze at the correct field at the correct time without blinking or moving. Reasons for displaying the stimuli on multiple flicker fields instead of a single one include:

- simulation of interference from peripheral flicker fields in a real-world BCI,
- simulation of interference from eye fixation-related potentials (FRPs),
- keeping participants alert.

For quality analysis as well as the subsequent offline evaluation, some additional data were acquired during each experiment: Supplementary to the 32 EEG channels, digital labeling data were output via the parallel port of PC0 and recorded synchronously to the EEG data. This included, among other things, the label of the currently presented segment and a display frame clock signal. Furthermore, PC0 also logged some auxiliary information such as the random permutation of the takes and the duration of every display frame.

5.4 Classifier design

To decode the CCVEP segments from the EEG data, a relatively basic machine learning model was designed. Each of the 6 trialed CCVEP encoding schemes was treated as a separate dataset, i.e. the model was individually trained and tested on the data of the 50 takes corresponding to one scheme without any cross-talk between the schemes. Figure 5.4 gives an overview of the model architecture. The main model consists of two stages: a regressor which estimates the stimulus lightness on a frame-by-frame basis from the EEG data, followed by a classifier which predicts the type of a segment based on the respective lightness estimates³. While this is

³The term *estimate* will be used here to describe the (continuous) result of a regression, while a *prediction* will signify the (discrete) result of a classification. The frame-wise and segment-wise stages both consist of a regressor, providing an estimate of the target value, followed by an optional thresholding step which classifies this estimate as a corresponding target label, i.e. a prediction. Nevertheless, since the main part of the machine learning model produces a segment-wise prediction from frame-wise estimates, the corresponding stages were named frame-wise *regressor* and segment-wise *classifier*. Algorithm 5.4 clarifies the various usages of the thresholded and non-thresholded regression results of the two stages.

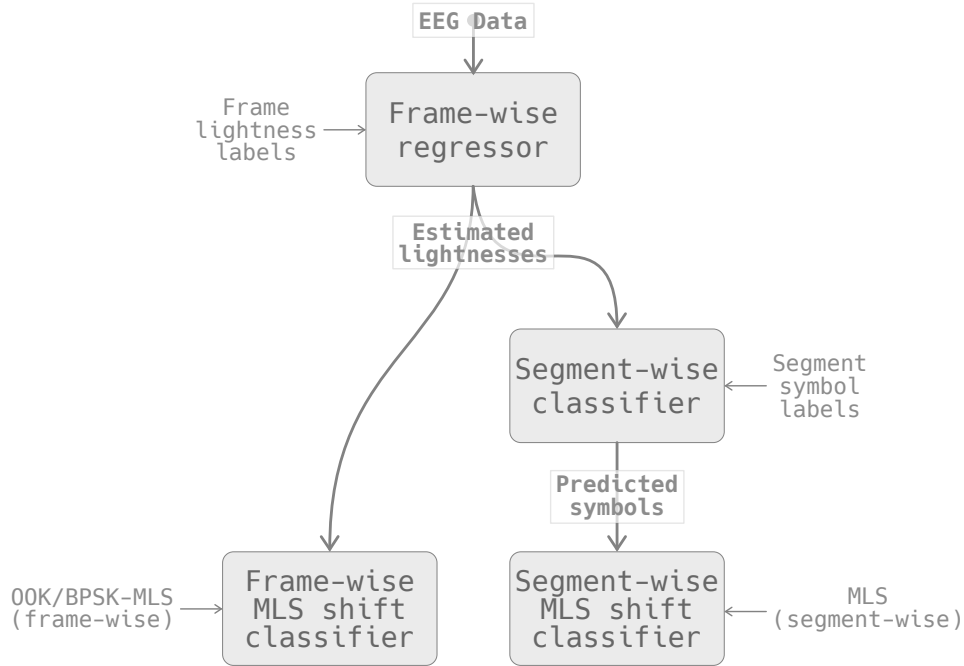


Figure 5.4: Architecture of the machine learning model.

Starting with the EEG data and the respective frame-wise labels, a regressor is trained to estimate the stimulation pattern. This estimate, together with the segment-wise symbol labels, is then used to train a classifier to predict the transmitted code. If the code is an MLS its cyclic shift can be determined by correlating the frame- or segment-wise predictions with the respective label vectors, yielding an alternative classification layer.

5 Materials and Methods

sufficient to transmit an ECC, an optional, additional classification step based on MLS cross-correlation was also implemented. MLS shift classification was performed both on the predicted stimulation patterns as well as on the estimated segment types.

For the frame-wise regressor, a sliding window model based on ridge regression was used (cf. sections 2.2 and 2.10). That is, a window into each of the 32 channels of EEG data was positioned around the start of each display frame. Together, these 32 windows were then used as the feature vector for that frame. Algorithm 5.2 details the construction of the feature matrix. The corresponding label vector simply consists of the respective elements of the stimulation pattern vector. The position and size of the windows can be regarded as hyperparameters for the frame-wise regressor. The mean and standard deviation per feature was learned in order to perform a standardization before the ridge regression. A bias term was not included. No separate spatial filtering stage was implemented, i.e. a composite spatial and temporal filter was learned. Differences in monitor raster latency between the flicker fields were not taken into account, since they should be negligible. Clock jitter was partially compensated for by aligning the window positions with the local clock signal. The frame-wise regressor includes an optional thresholding step to create a discrete prediction which is used for the accuracy evaluation and frame-wise MLS shift classification, but not for the segment-wise classification.

Algorithm 5.2: `frame_wise_features`

Computes the feature matrix of a set of trials for frame-wise regression.

Data:

with $n \in \mathbb{N}$,	▷ Number of sampled time points in the EEG recording
$k \in \mathbb{N}$,	▷ Number of EEG channels
$m \in \mathbb{N}$	▷ Number of frames of interest
<code>signal</code> $\in \mathbb{R}^{n \times k}$	▷ Full EEG recording
<code>frames</code> $\in \{i \in \mathbb{N}_0 \mid i < n\}^m$	▷ Indexes in <code>signal</code> of the frames of interest
<code>window_size</code> $\in \mathbb{N}$	▷ Window size in samples
<code>signal_offset</code> $\in \mathbb{Z}$	▷ Relative window start in samples

Result:

<code>features</code> $\in \mathbb{R}^{m \times \text{window_size} \cdot k}$	▷ Rows: one feature vector per frame
---	--------------------------------------

```

foreach i, row in frames, features do
    start ← i + signal_offset
    row ← flatten(signal[start : start + window_size, :])
end

```

The frame-wise regressor produces an estimate of the stimulus lightness for each frame. This estimate can be split into its CCVEP segments, since the position and size of each segment are a given. Each estimated segment then serves as a feature vector for the segment-wise classifier, while the symbol it encodes constitutes the corresponding label. The segment-wise classifier was implemented as a cascade of

5 Materials and Methods

standardization, ridge regression without bias term, and thresholding. Only for the segment-wise MLS shift classification, the non-thresholded estimates were used.

The segment-wise classifier recovers each code symbol individually from the EEG response without any knowledge of the surrounding symbols. This allows for the transmission of code sequences with arbitrary length and content. In the context of a BCI, the number of possible targets is thus only limited by stimulus duration and sufficient error correction. While ECCs represent an option worthy of further exploration, the present thesis instead evaluated the use of cyclically shifted MLS as a means to identify targets. As discussed in section 5.2, the stimulation patterns presented during the experiment encoded an MLS of length 15, namely μ . Two methods of MLS shift classification were evaluated: correlating 15-segment subsections of the (thresholded) frame-wise predictions with cyclically shifted versions of the stimulation pattern, and correlating 15-symbol subsections of the (non-thresholded) segment-wise estimates with cyclically shifted versions of μ . Section 2.7 discussed why both approaches are viable – in particular, the stimulation patterns conserve the MLS autocorrelation property. Algorithm 5.3 details the classification procedure of the MLS classifiers. It accepts a **sequence** that consists of the concatenated frame-wise predictions or segment-wise estimates of all trials, i.e. a repeating, possibly modulated reconstruction of μ from the EEG data, and returns a **prediction** of the cyclic shift of μ for all 15-segment subsequences.

Algorithm 5.3: ccccm

Classification by cyclic cross-correlation with a 15-segment MLS-like pattern.

Data:

with $k = 15$, $n \in \mathbb{N}$

sequence $\in \mathbb{R}^n$

▷ Full sequence within which to classify MLS shifts

pattern $\in \mathbb{R}^{k \cdot \text{stride}}$

▷ MLS-like reference pattern

stride $\in \mathbb{N}$

▷ Length of segments encoding one MLS symbol

Result:

prediction $\in \mathbb{Z}_k^{n \cdot \text{stride}^{-1}}$

▷ Predicted cyclic shifts

labels $\in \mathbb{Z}_k^{n \cdot \text{stride}^{-1}}$

▷ True shift labels

$$\mathbf{A} \leftarrow \left(\text{cycshift}_{i \cdot \text{stride}}(\text{pattern})^T \right)_{i=0}^{k-1}$$

▷ $\mathbf{A} \in \mathbb{R}^{k \times k \cdot \text{stride}}$

$$\text{subsequences} \leftarrow (\text{sequence}[(i : i + k) \cdot \text{stride}] \bmod n)_{i=0}^{n \cdot \text{stride}^{-1} - 1}$$

$$\text{prediction} \leftarrow \left(\arg \max_{j \in \mathbb{Z}_k} (\mathbf{A} \cdot \text{subsequences}[i])_{j+1} \right)_{i=0}^{n \cdot \text{stride}^{-1} - 1}$$

$$\text{labels} \leftarrow (i \bmod k)_{i=0}^{n \cdot \text{stride}^{-1} - 1}$$

The thresholding steps in the frame-wise regressor and the segment-wise classifier serve the purpose of mapping regression estimates to two discrete label predictions, $\{-1, 1\}$. Thresholds yielding the highest accuracy were learned on the training set

Algorithm 5.4: evaluation

Training and evaluation of the four classification stages.

Data:

pattern ▷ Computed by **encode**
word ▷ The input word passed to **encode** (i.e. the MLS μ)
segment_length ▷ The segment length passed to **encode**
features₀ ▷ Computed by **frame_wise_features**
 $\lambda_0, \lambda_1 \in \mathbb{R}_+$ ▷ Regularizer weights for frame- and segment-wise ridge regressions
threshold ▷ Function mapping lightness/symbol labels to $\{-1, 1\}$

Result:

acc₀, acc₁ $\in [0, 1]$ ▷ Accuracies of frame- and segment wise classifiers
acc_{m0}, acc_{m1} $\in [0, 1]$ ▷ Accuracies of MLS shift classifiers

labels₀ $\leftarrow$ get appropriate frame-wise lightness labels from **pattern**
labels₀ $\leftarrow$ **labels₀** $\cdot 2 - \mathbb{1}$ ▷ Centers **labels₀** to span $[-1, 1]$
do cross-validation using features₀, labels₀
 Standardize features to zero mean and unit variance
 Perform ridge regression with $\lambda = \lambda_0$
 Optimize thresholds to map estimates to discrete predictions
yielding estimate₀, prediction₀ ▷ (Non-)thresholded frame lightness estimates

labels₁ $\leftarrow$ get appropriate segment-wise labels from **word**
labels₁ $\leftarrow$ **threshold(labels₁)** ▷ Centers **labels₁** to span $[-1, 1]$
features₁ $\leftarrow$ split **estimate₀** into corresponding segments
do cross-validation using features₁, labels₁
 Standardize features to zero mean and unit variance
 Perform ridge regression with $\lambda = \lambda_1$
 Optimize thresholds to map estimates to discrete predictions
yielding estimate₁, prediction₁ ▷ (Non-)thresholded segment symbol estimates

accuracy($\mathbf{v} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^n$) $\leftarrow \overline{(\delta_{\mathbf{v}_i, \mathbf{u}_i})_{i=1}^n}$ ▷ Fraction of elements $v_i = u_i$
acc₀ $\leftarrow$ **accuracy**(**threshold(labels₀)**, **prediction₀**)
acc₁ $\leftarrow$ **accuracy**(**labels₁**, **prediction₁**)
acc_{m0} $\leftarrow$
 accuracy(**cccm**(**prediction₀**, **threshold(pattern)**, **segment_length**))
acc_{m1} $\leftarrow$ **accuracy**(**cccm**(**estimate₁**, **threshold(word)**, 1))

5 Materials and Methods

and applied to the testing set. For the OOK schemes in particular, a design choice was made to map black and white frames to the same label, such that the label -1 represents frames classified as gray and 1 represents frames classified as non-gray. This was deemed sensible because non-gray frames belong to the same segment type. Furthermore, it makes an ideal frame-wise prediction be a perfect (non-modulated) MLS, fully conserving the autocorrelation property. The frame- and segment wise label vectors need to be matched to this type of thresholding. This was achieved by a function `threshold`: $\mathbb{R}^n \rightarrow \{-1, 1\}^n$, which was matched to the modulation scheme. Specifically, for OOK,

$$\text{threshold}(x) = \left(\begin{cases} 1 & \text{if } x_i > \frac{1}{2} \\ -1 & \text{if } -\frac{1}{2} \leq x_i \leq \frac{1}{2} \\ 1 & \text{if } x_i < -\frac{1}{2} \end{cases} \right)_{i=1}^n \quad 5.3$$

while for BPSK,

$$\text{threshold}(x) = \left(\begin{cases} 1 & \text{if } x_i > 0 \\ -1 & \text{if } x_i \leq 0 \end{cases} \right)_{i=1}^n \quad 5.4$$

The four stages of the machine learning model were each evaluated in terms of their prediction accuracy. Algorithm 5.4 gives an overview of the full evaluation procedure. In particular, for each test subject and encoding scheme,

- (1) the frame-wise accuracy was computed by measuring the fraction of frames where the estimated lightness fell into the same interval as the true lightness,
- (2) the segment-wise accuracy was computed by measuring the fraction of segments where the predicted symbol coincided with the true symbol,
- (3) the frame-wise and segment-wise MLS shift classification accuracies were computed by measuring the fraction of correctly identified cyclic shifts across all 15-segment subsequences of the frame-wise predictions and segment-wise estimates, respectively.

The subspace spanned by the hyperparameters `window_size` and `signal_offset` was sampled in terms of classification accuracy. More detailed results will be given for the specific hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, $\lambda_0 = 1$, and $\lambda_1 = 10$. For each accuracy value, Pearson-Clopper confidence intervals (CIs) were computed. Furthermore, mean accuracies across all test subjects along with p -values and CIs were computed by means of t -tests⁴ The variance between test subjects as well as between encoding schemes was analyzed by Kruskal-Wallis (1952) H -tests. Moreover, the lengths of consecutive runs of segment-wise misclassifications were gathered. Their distribution will be illustrated through histograms.

⁴Note that using t -tests for classification accuracies, while commonly being done in the literature (Hamarashid, 2021), is technically not applicable here, since the accuracy is not quite normally distributed. This applies to the results presented in sections 6.1, 6.2, 6.3, and 6.4. The t -test should be seen as a rough confidence estimate only.

5.5 Supplementary analyses

Alongside the evaluation of the machine learning model, the data were analyzed in terms of SNRs and jitter. Broadband (5 Hz to 300 Hz) SNR values were calculated for each test subject, encoding scheme, and EEG channel separately in the following manner:

- (1) Apply a temporal 4-pole Butterworth high-pass filter with a cutoff frequency of 5Hz to all EEG data.
- (2) From the 50 takes for the given encoding scheme and test subject, crop the first and last $\frac{1}{3}$ s to discard any transient artifacts.
- (3) For the given EEG channel, take the root mean square (RMS) across all 50 takes, yielding a_{full} .
- (4) Compute the mean signal across the 50 takes and take the RMS of the given channel, yielding a_{mean} .
- (5) Compute the SNR from a_{full} and a_{mean} as defined by equation 5.7.

The idea is to reduce the noise by averaging across takes in order to isolate the CCVEP. Since on average, $\frac{1}{50}$ of the RMS of the noise will still be present in a_{mean} and a_{full} is the RMS of the sum of the CCVEP and noise, the RMSs of the CCVEP and the noise can be estimated as follows:

$$a_{\text{noise}} = \sqrt{\frac{50}{49}} \cdot (a_{\text{full}} - a_{\text{mean}}) \quad 5.5$$

$$a_{\text{signal}} = a_{\text{mean}} - \sqrt{\frac{1}{50}} \cdot a_{\text{noise}} \quad 5.6$$

From there,

$$\text{SNR} = \frac{a_{\text{signal}}^2}{a_{\text{noise}}^2} = \frac{(a_{\text{full}} - 8 \cdot a_{\text{mean}})^2}{50 \cdot (a_{\text{full}} - a_{\text{mean}})^2} \quad 5.7$$

The jitter between the display frame clock and the EEG's analog-to-digital conversion (ADC) clock was analyzed by means of an absolute jitter histogram (Galton & Weltin-Wu, 2018). For this, the display frame clock phase was estimated from the parallel port data at every sampling point across all takes and all test subjects. The absolute jitter was calculated by

$$\Delta(i) = \frac{(\phi_1(i) - \phi_0(i) + \frac{1}{2}) \bmod 1 - \frac{1}{2}}{\omega_0} \quad 5.8$$

where i is a sample index, $\phi_1(i) - \phi_0(i)$ is the phase error relative to the start of the take, with 1 corresponding to one full period, and $\omega_0 = \frac{120\text{Hz}}{600\text{Hz}} = \frac{1}{5}$ is the normalized nominal frequency.

6 Results

In the following, the experimental results will be presented. Sections 6.1, 6.2, 6.3, and 6.4 are similarly structured and give an overview of the performance of each of the four machine learning model stages. Sections 6.5, 6.6, and 6.7 present the results of the hyperparameter space sampling, SNR computation, and jitter analysis, respectively.

6.1 Stimulation pattern recovery

The frame-wise classification accuracy (acc_0 in algorithm 5.4) was evaluated for the model hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, and $\lambda_0 = 1$. Figure 6.1 shows the obtained accuracies for each test subject and encoding scheme. Table 6.1 gives a summary of the results across the group. All encoding schemes yielded accuracies significantly better than chance ($p < 10^{-3}$). Kruskal-Wallis H -tests revealed significant differences both between test subjects

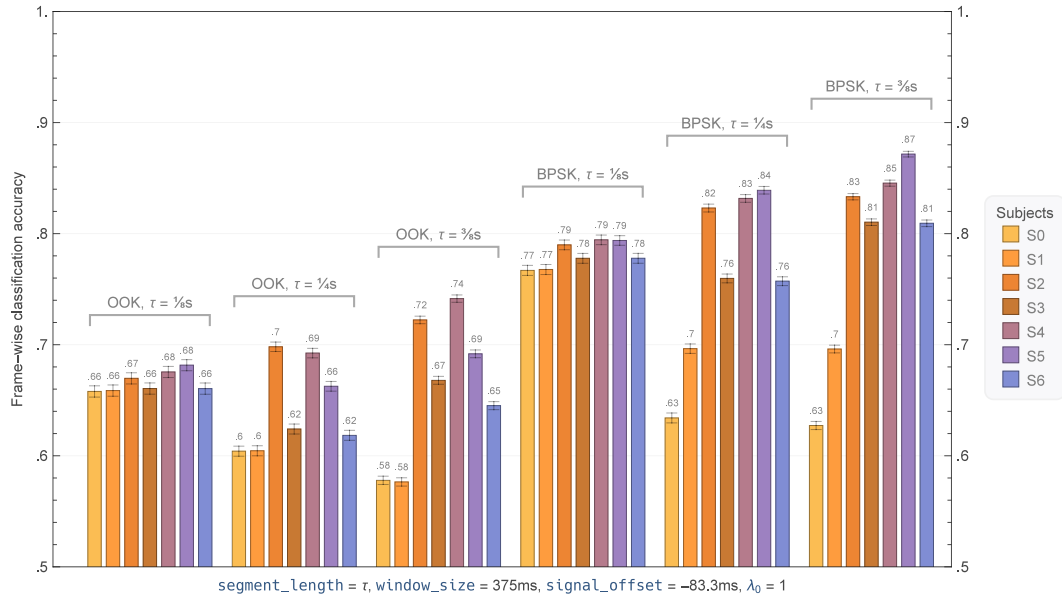


Figure 6.1: Frame-wise classification accuracy by encoding scheme and test subject.

Values were obtained by 3-fold cross-validation over the 33750, 45000, or 67500 frames per test subject and encoding scheme for the segment lengths of $1/8s$, $2/8s$, or $3/8s$, respectively. Error bars indicate the 95% Pearson-Clopper confidence interval. Since two types of segments are distinguished for each encoding scheme, a value of .5 is equivalent to chance.

6 Results

($p = .037$) and between encoding schemes ($p = .00050$). Overall, frame lightnesses were classified more accurately for the BPSK stimulation patterns.

For the OOK schemes, frames classified as either black or white by the thresholding are treated as the same label and distinguished from frames classified as gray. For the BPSK schemes, the thresholding only distinguishes between black and white frames. Thus, for all encoding schemes, a value of .5 is equivalent to chance. Distinguishing between black and white frames for the OOK schemes is unnecessary, since both belong to the same type of flickering segment. If the distinction were made, however, the accuracies would decrease only marginally, and 3^{-1} would be equivalent to chance.

Modulation scheme <code>segment_length</code>	OOK $\frac{1}{8}$ s	OOK $\frac{2}{8}$ s	OOK $\frac{3}{8}$ s	BPSK $\frac{1}{8}$ s	BPSK $\frac{2}{8}$ s	BPSK $\frac{3}{8}$ s
Accuracy	.67	.64	.66	.78	.76	.78
p -value	$< 10^{-8}$	$< 10^{-4}$	$< 10^{-3}$	$< 10^{-8}$	$< 10^{-4}$	$< 10^{-3}$
CI	[.66, .67]	[.61, .68]	[.6, .72]	[.77, .79]	[.69, .83]	[.7, .87]

Table 6.1: Frame-wise classification accuracy by encoding scheme.

The top two columns list the encoding scheme. Below, the mean accuracies across all test subjects are given, along with the t -test p -values for the null hypothesis that the population mean is equal to chance ($H_0: \mu = .5$) and the corresponding confidence intervals (CIs) for the means. All values were obtained from a model with hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, and $\lambda_0 = 1$.

6.2 Code recovery

The segment-wise classification accuracy (`acc1` in algorithm 5.4) was evaluated for the model hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, $\lambda_0 = 1$, and $\lambda_1 = 10$. Figure 6.2 shows the obtained accuracies for each test subject and encoding scheme. Table 6.2 gives a summary of the results across the group. All encoding schemes yielded accuracies significantly better than chance ($p < 10^{-2}$). Kruskal-Wallis H -tests revealed significant differences both between test subjects ($p = .0034$) and between encoding schemes ($p = .0015$). Overall, segment symbols were classified more accurately and yielded higher ITRs for the BPSK stimulation patterns.

6 Results

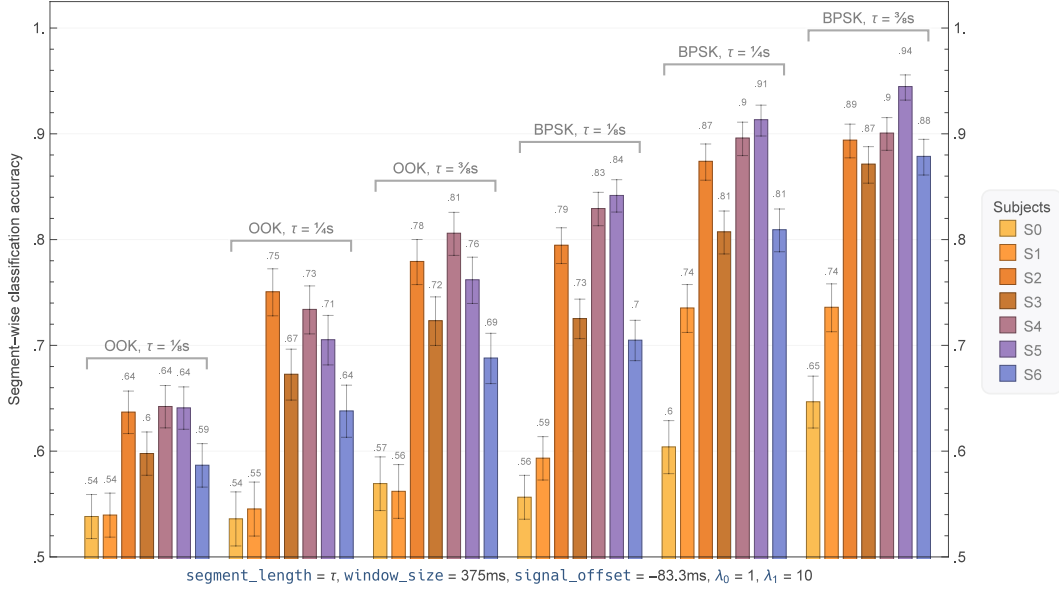


Figure 6.2: Segment-wise classification accuracy by encoding scheme and test subject.

Values were obtained by 6-fold cross-validation over the 4500, 3000, or 3000 segments per test subject and encoding scheme for the segment lengths of $1/8$ s, $2/8$ s, or $3/8$ s, respectively. Error bars indicate the 95% Pearson-Clopper confidence interval. Since two types of segments are distinguished for each encoding scheme, a value of .5 is equivalent to chance.

Modulation scheme	OOK	OOK	OOK	BPSK	BPSK	BPSK
<code>segment_length</code>	$1/8$ s	$2/8$ s	$3/8$ s	$1/8$ s	$2/8$ s	$3/8$ s
Accuracy	.6	.65	.7	.72	.81	.84
p -value	$< 10^{-2}$	$< 10^{-2}$	$< 10^{-2}$	$< 10^{-2}$	$< 10^{-3}$	$< 10^{-3}$
CI	[.56, .64]	[.57, .73]	[.61, .79]	[.62, .82]	[.71, .91]	[.74, .94]
ITR	13.2	16.8	18.7	69.9	69.5	58.1

Table 6.2: Segment-wise classification accuracy and ITR by encoding scheme.

The top two columns list the encoding scheme. Below, the mean accuracies across all test subjects are given, along with the t -test p -values for the null hypothesis that the population mean is equal to chance ($H_0: \mu = .5$) and the corresponding confidence intervals (CIs) for the means. The bottom row lists the respective information transfer rates (ITRs) in bit/min as defined by equation 2.17. All values were obtained from a model with hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, $\lambda_0 = 1$, and $\lambda_1 = 10$.

Figure 6.3 shows the relative frequencies of lengths of consecutive runs of correctly and incorrectly classified segments per encoding scheme. For all encoding schemes, runs of more than five consecutive incorrectly classified segments were rare, with single-segment misclassifications being the most prevalent by far. Longer runs of

6 Results

correctly classified segments were generally more common for the BPSK stimulation patterns.

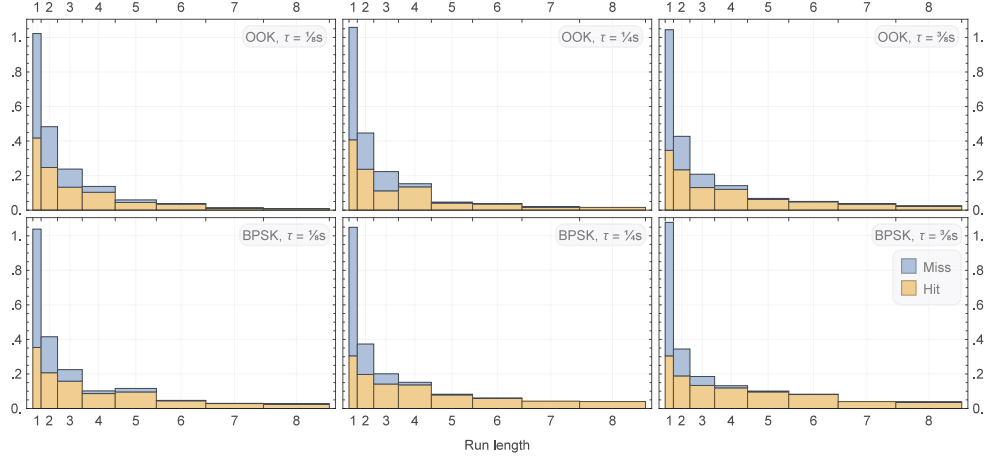


Figure 6.3: Run lengths of (in-)correctly classified segments by encoding scheme.

The bars represent consecutive runs of correctly (yellow) or incorrectly (blue) classified segments within a take. Bar height indicates the empirical conditional probability of obtaining a specific run length, given it being a hit or a miss. The width is scaled by the run length, such that the area represents the relative frequency of (in-)correctly classified segments. Probability estimation was done across all test subjects and takes. Although not shown here, longer runs than those of length 8 did exist, the vast majority being hits. All values were obtained from a model with hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, $\lambda_0 = 1$, and $\lambda_1 = 10$.

6.3 Frame-wise MLS shift recovery

The frame-wise MLS shift classification accuracy (`acc_m0` in algorithm 5.4) was evaluated for the model hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, and $\lambda_0 = 1$. Figure 6.4 shows the obtained accuracies for each test subject and encoding scheme. Table 6.3 gives a summary of the results across the group. All encoding schemes yielded accuracies significantly better than chance ($p < 10^{-4}$). Kruskal-Wallis H -tests revealed significant differences between encoding schemes ($p = .00026$), but not between test subjects ($p = .98$). Most notable, perhaps, are the accuracies for the BPSK scheme with a segment length of $1/8$ s, where the classification was almost universally correct, yielding a mean ITR of 124.6^{bit}/min.

6 Results

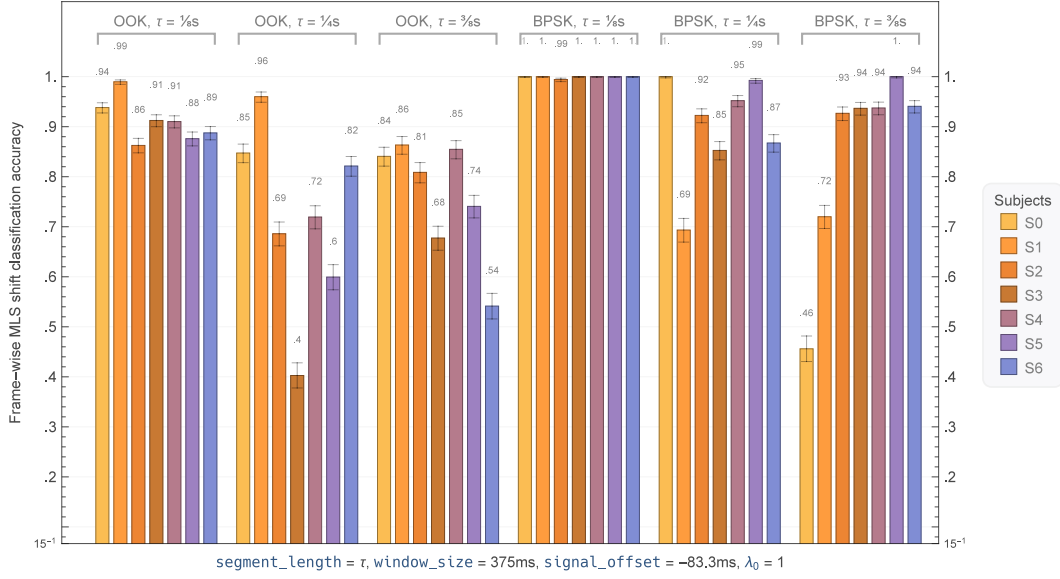


Figure 6.4: Frame-wise MLS shift classification accuracy by encoding scheme and test subject.

Values were obtained by 15-class MLS cyclic cross-correlation on the 4500, 3000, or 3000 subsequences per test subject and encoding scheme for the segment lengths of $1/8$ s, $2/8$ s, or $3/8$ s, respectively. Error bars indicate the 95% Pearson-Clopper confidence interval. Since 15 unique shifts are distinguished, a value of 15^{-1} is equivalent to chance.

Modulation scheme <code>segment_length</code>	OOK $1/8$ s	OOK $2/8$ s	OOK $3/8$ s	BPSK $1/8$ s	BPSK $2/8$ s	BPSK $3/8$ s
Accuracy	.91	.72	.76	> .99	.9	.85
p -value	$< 10^{-8}$	$< 10^{-4}$	$< 10^{-5}$	$< 10^{-16}$	$< 10^{-6}$	$< 10^{-4}$
CI	[.87, .95]	[.55, .89]	[.65, .87]	> .99	[.8, 1.]	[.67, 1.]
ITR	100.3	31.7	23.5	124.6	48.6	28.8

Table 6.3: Frame-wise MLS shift classification accuracy and ITR by encoding scheme.

The top two columns list the encoding scheme. Below, the mean accuracies across all test subjects are given, along with the t -test p -values for the null hypothesis that the population mean is equal to chance ($H_0: \mu = 15^{-1}$) and the corresponding confidence intervals (CIs) for the means. The bottom row lists the respective information transfer rates (ITRs) in bit/min as defined by equation 2.17. All values were obtained from a model with hyperparameters `window_size = 375ms`, `signal_offset = -83.3ms`, and $\lambda_0 = 1$.

6.4 Segment-wise MLS shift recovery

The segment-wise MLS shift classification accuracy (`acc_m1` in algorithm 5.4) was evaluated for the model hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, $\lambda_0 = 1$, and $\lambda_1 = 10$. Figure 6.5 shows the obtained accuracies for each test subject and encoding scheme. Table 6.4 gives a summary of the results across the group. All encoding schemes yielded accuracies significantly better than chance ($p < .02$). Kruskal-Wallis H -tests revealed significant differences both between test subjects ($p = .0031$) and between encoding schemes ($p = .0025$). Overall, MLS shifts were classified more accurately and yielded higher ITRs for the BPSK stimulation patterns.

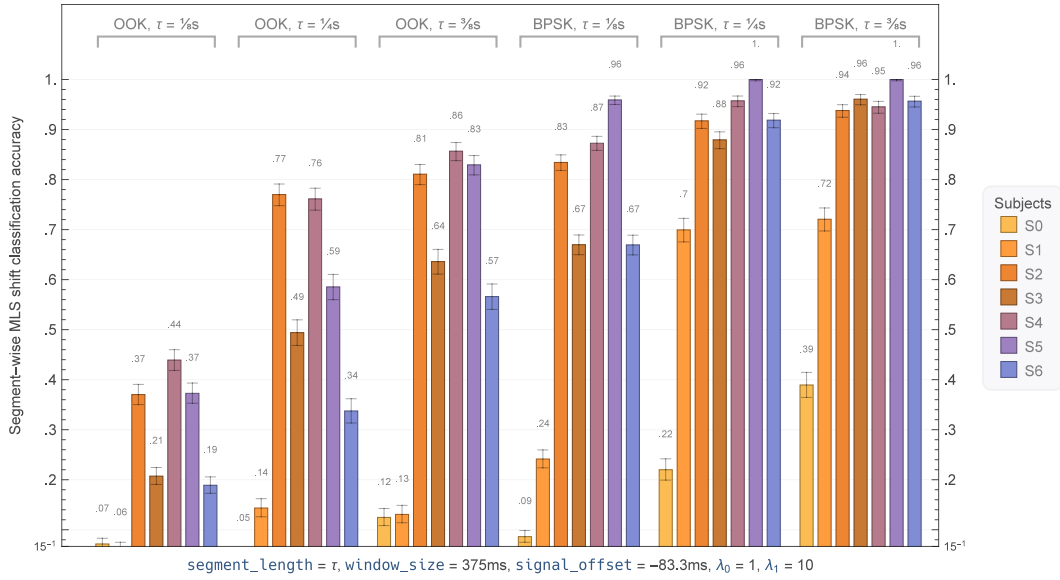


Figure 6.5: Segment-wise MLS shift classification accuracy by encoding scheme and test subject.

Values were obtained by 15-class MLS cyclic cross-correlation on the 4500, 3000, or 3000 subsequences per test subject and encoding scheme for the segment lengths of $1/8$ s, $2/8$ s, or $3/8$ s, respectively. Error bars indicate the 95% Pearson-Clopper confidence interval. Since 15 unique shifts are distinguished, a value of 15^{-1} is equivalent to chance.

6.5 Sliding window hyperparameters

Figure 6.6 shows the mean frame-wise classification accuracy as a function of `window_size` and `signal_offset` for two encoding schemes. Regions of optimal performance varied somewhat with the `segment_length`. Generally, the best performance was attained in the approximate region of $(\text{window_size}, \text{signal_offset}) \in [300\text{ms}, 400\text{ms}] \times [-120\text{ms}, -50\text{ms}]$.

6 Results

Modulation scheme <code>segment_length</code>	OOK $\frac{1}{8}$ s	OOK $\frac{2}{8}$ s	OOK $\frac{3}{8}$ s	BPSK $\frac{1}{8}$ s	BPSK $\frac{2}{8}$ s	BPSK $\frac{3}{8}$ s
Accuracy	.24	.45	.56	.62	.8	.84
p -value	.02	.01	$< 10^{-2}$	$< 10^{-2}$	$< 10^{-3}$	$< 10^{-4}$
CI	[.11, .38]	[.19, .71]	[.27, .86]	[.31, .93]	[.55, 1.]	[.64, 1.]
ITR	7.3	13.1	13.5	47.9	38.7	28.7

Table 6.4: Segment-wise MLS shift classification accuracy and ITR by encoding scheme.

The top two columns list the encoding scheme. Below, the mean accuracies across all test subjects are given, along with the t -test p -values for the null hypothesis that the population mean is equal to chance ($H_0: \mu = 15^{-1}$) and the corresponding confidence intervals (CIs) for the means. The bottom row lists the respective information transfer rates (ITRs) in bit/min as defined by equation 2.17. All values were obtained from a model with hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, $\lambda_0 = 1$, and $\lambda_1 = 10$.

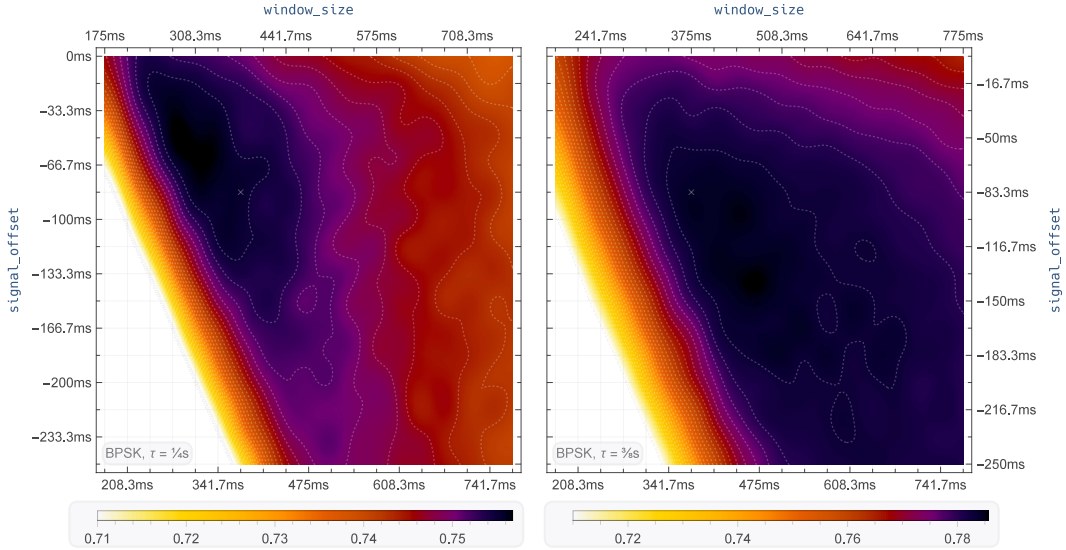


Figure 6.6: Frame-wise classification accuracy across window hyperparameters.

The plots show a cross-section of the hyperparameter space of the frame-wise regressor at $\lambda_0 = 1$. This subspace was sampled in terms of the mean frame-wise classification accuracy (acc_0 in algorithm 5.4) across test subjects on the BPSK takes with a `segment_length` of $\frac{2}{8}$ s (left) and $\frac{3}{8}$ s (right), represented by the color map. The dashed lines are contours of constant accuracy. The cross (x) marks the `window_size` and `signal_offset` used for the evaluations in the previous sections.

6.6 Signal-to-noise ratio (SNR)

Table 6.5 lists the mean SNRs for each test subject. The overall mean SNR was -28.7dB , with subject S4 having the highest and subject S0 having the lowest SNR.

Subject	S0	S1	S2	S3	S4	S5	S6
SNR	-31.7	-30.1	-27.9	-30.2	-24.2	-27.3	-29.5
SD	1.8	3.6	3.2	2.8	3.7	3.9	2.6

Table 6.5: Mean SNR by test subject.

SNRs as defined by equation 5.7 were calculated for each test subject, encoding scheme and EEG channel separately. Given here are the mean and standard deviation (SD) in dB per test subject over all schemes and channels.

6.7 Jitter

Figure 6.7 shows a histogram of the absolute jitter. Jitter was relatively common in the takes – in particular, a second mode exists at -1 , corresponding to a phase error of $1/5$. Apart from the irregularities introduced by random jitter, imperfect clock signal frequencies were a source of phase error. The ratio of the monitor refresh rate to the EEG sampling frequency should ideally be $120\text{Hz}/600\text{Hz} = 1/5 = .2$, but the actual (harmonic) mean ratio across all experiments was roughly .1999770.

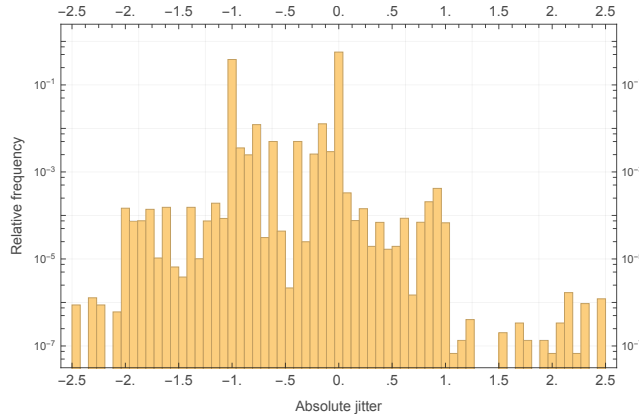


Figure 6.7: Absolute jitter across all test subjects and takes.

The yellow bars indicate the relative frequency of absolute jitter values (equation 5.8) contained in the respective interval. An absolute jitter value of zero is ideal.

7 Discussion

In this chapter, the experiment’s methodology, results, and implications for the CCVEP paradigm are discussed. First, the general quality and applicability of the data are inspected (section 7.1). Subsequently, the CCVEP paradigm is discussed in terms of its performance as a flicker-free communication channel (sections 7.2 and 7.3). Finally, some insights into the decoding of CCVEPs and potential for improvement of the machine learning model are expounded (sections 7.4 and 7.5).

7.1 Data quality

This section briefly discusses the experiment conditions as they apply to the quality of the data obtained. While some external factors such as the lighting and the order of the takes were controlled or randomized, other influences may have had a negative impact, introduced biases, or at best, provided more real-world-like conditions. All test subjects wore surgical face masks during the experiment due to hygiene regulations. The test chamber was not shielded against EMFs, and in fact, several EMF-emitting devices, such as smartphones and the experiment hardware, were present inside. Moderate ambient noise from outside the test chamber, e.g. chatter, was a frequent occurrence.

The SNRs in the EEG recordings was relatively low. This was expected, since the amplitude of VEPs is known to decrease with higher stimulation frequencies (Pastor,

Subject	Potential issues
S0	Foggy glasses due to face mask throughout the experiment
S1	Blinking due to face mask throughout the experiment
S2	None
S3	Ocular fatigue in the second half of the experiment
S4	Somnolence towards the end of the experiment
S5	Somnolence towards the end of the experiment
S6	A person trying to enter the test chamber during the experiment

Table 7.1: Potential issues during each experiment.

Participants were briefly asked about their subjective experience and assessment of adherence to the instructions. Given here are the test subjects’ reports and the supervisor’s observations of issues potentially impacting the quality of the data.

Artieda, Arbizu, Valencia, & Masdeu, 2003). The variation in SNR between test subjects generally mirrored the variation in classification accuracies. Table 7.1 lists issues experienced by the test subjects during the experiment. These were obtained through observation by the experiment supervisor as well as a brief, and by no means rigorous, conversation with the participants after the experiment. Several participants reported somnolence and not being able to completely avoid blinking during the takes. Most notably, perhaps, was the fact that S0’s vision was impaired by their glasses being foggy throughout the duration of the experiment. This might explain the subpar SNR and classification accuracies obtained from S0’s data.

Although the alignment of the frame-wise classifier’s windows with the display clock likely mitigated the majority of the impact, jitter between the display and EEG sampling clocks may still have skewed the temporal positions of some data points. It is conceivable that a higher temporal sampling resolution and less jitter may have improved phase shift recognition marginally. Fortunately, no display frames were dropped completely during any of the experiments. In addition, since the display viewing angle was not controlled and colors were calibrated on a per-subject basis, the peak-to-peak amplitude of the stimuli’s irradiance may have differed slightly between experiments.

7.2 Stimulus recognizability

Rigorously evaluating the recognizability of the stimuli shown in the experiments was not part of the present thesis. Subjectively, it should be obvious to any user that the improvement in comfort compared to traditional cVEP-eliciting stimuli is overwhelming. Nevertheless, the stimuli were still recognizable to a minor degree. Future research may improve upon this by investigating techniques to further minimize recognizability on a per-user basis by tuning parameters such as the carrier frequency, flicker coloring, and most importantly, the specific implementation of segment transitions. Attention has to be paid to the peculiarities of the screen presenting the stimuli as well as to the variation in perception of flicker fusion colors across users and sessions.

7.3 Viability of the CCVEP paradigm for BCIs

It is clear from the results that transmission of data through nearly non-recognizable CCVEP-type stimuli is possible both with a BPSK- as well as OOK-modulated carrier frequency of 60Hz. Given the current literature on SSVEPs, it is likely that these results generalize to other carrier frequencies and conceivably also to other modulation schemes, such as frequency-shift keying (FSK).

Generally, BPSK performed better than OOK. This is presumably because the classification of distinct phase shifts is more robust against noise than distinguishing between amplitudes and the lightness gap between black and white frames is approximately twice as large as the gap between non-gray and gray frames. Although not perfectly applicable here, theoretical results about digital modulation

7 Discussion

corroborate this (Modabbes, 2004). Furthermore, neurons in the human visual cortex synchronize their firing rate to the frequency of flickering light, producing harmonic frequencies, suggesting that the elicitation of VEPs is a non-LSI system (Müller-Putz, Scherer, Brauneis, & Pfurtscheller, 2005) that may override the phase of noise to some degree.

Longer CCVEP segments decrease the ITR by definition, but tended to yield higher classification accuracies, likely because they provide a higher number of features to base the segment-wise classification on and because there is more time for a steady state to be established in the VEP. This creates a trade-off when optimizing for ITR. The present thesis achieved reasonable ITRs for BCI applications such as on-screen keyboards. Given that this was the first trial of CCVEP-based data transmission ever done, it is certain that there exists room for improvement in the high-dimensional space of open parameters (modulation schemes, classification techniques, etc.) that this approach has to offer.

With sensible hyperparameters, the approach is able to build upon the established use of MLSs for fairly robust BCI target classification. Since long runs of misclassified segments seem to be rare, CCVEP-based data transmission is also a suitable candidate for the leverage of ECCs to ensure data integrity. Furthermore, the present thesis did not implement any classification confidence thresholding or non-control state detection, which may further improve the reliability of the communication channel.

7.4 Relevant signal content

The frame-wise regressor estimates the stimulation pattern by applying a composite spatial and temporal LSI filter to the EEG data. Given that the stimuli presented in the experiment encode their data using a carrier frequency of 60Hz, it stands to reason to assume that the relevant information in the EEG data is contained within a narrow frequency band around 60Hz, and the temporal component of the filter rejects most of the remaining frequency content. This is indeed the case: Figure 7.1 shows the median magnitude responses of the temporal kernels learned from the experiment data. The filters essentially isolate the frequency 60Hz. Since the frame-wise regressor predicts frame lightnesses at 120Hz from the 600Hz EEG signal, it effectively performs downsampling by a factor of 5, folding the frequency 180Hz onto 60Hz. This is presumably the reason for the second peak at 180Hz, which then serves as a correcting term for the frame lightness estimates. The frequency domain representation of the kernels gives another perspective on the reason why a substantial window size is needed to yield satisfactory regression accuracy: A larger kernel allows for a more surgical frequency response. In this sense, the frame- and segment-wise regression may be regarded as a type of custom adapted wavelet transform.

Figure 7.2 goes into further detail about the frequency responses at 60Hz, specifically. The magnitude and phase relationships between channels are shown for the 3 kernels produced during the cross-validation procedure for a single subject and encoding scheme. Notice how similar phase shifts are learned by each fold, espe-

7 Discussion

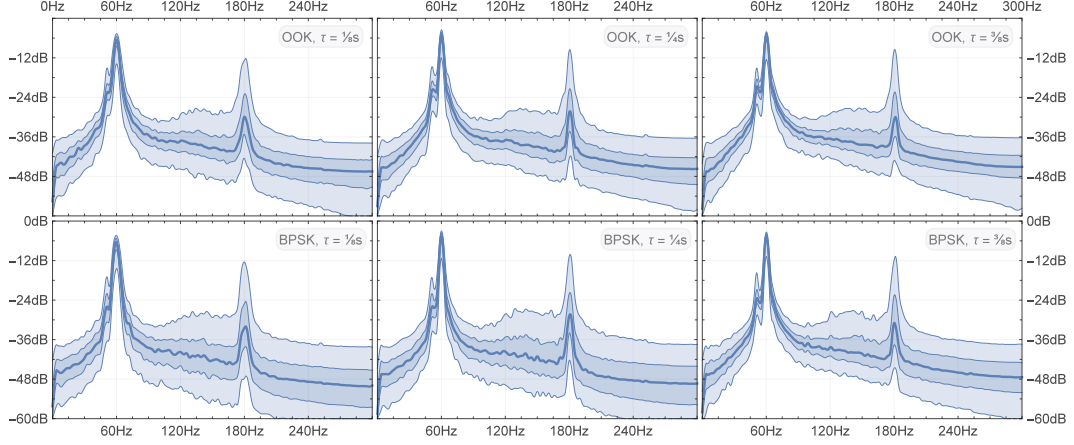


Figure 7.1: Magnitude responses of the learned kernels.

The plots show the median magnitude responses (thick blue line) along with the $[.3, .7]$ - and $[.1, .9]$ -interquartile ranges (regions shaded light blue) of the RMS-normalized kernels learned by the frame-wise regressor, over all channels, test subjects, and cross-validation folds, for each encoding scheme. All values were obtained from a model with hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, and $\lambda_0 = 1$. The frequency responses exhibit high similarity between encoding schemes. Prominent peaks exist at 60Hz and 180Hz, while the remainder of the frequency range is markedly attenuated.

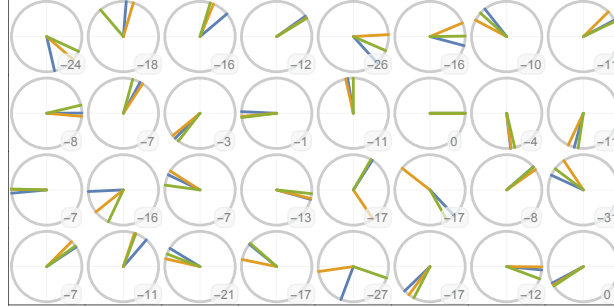


Figure 7.2: Inter-channel phase relationships at 60Hz.

Each circle represents one of the 32 EEG channels. The frequency response of the frame-wise regressor’s kernels, learned by the 3 cross-validation folds for BPSK with a segment size of $\frac{3}{8}$ s from test subject S5’s data, was evaluated at 60Hz. The blue, yellow, and green radial lines represent the phase, and the mean magnitude is given in dB in the lower right corner. Values are given in relation to the 14th channel. All values were obtained from a model with hyperparameters `window_size` = 375ms, `signal_offset` = -83.3ms, and $\lambda_0 = 1$. For all three folds, similar phase responses at 60Hz were learned, especially when the corresponding magnitude was high.

cially for the channels with the highest overall contribution (i.e. magnitude). This suggests that the VEP has a unique phase shift at every EEG electrode.

Seeing as the relevant information in the EEG signal largely revolves around the magnitudes and phases at 60Hz, it may be sensible to perform pre-processing by a temporal band-pass filter and enhance the representation of the signal’s phase in the feature vectors passed to the frame-wise regressor. This was tested in a post-hoc analysis: EEG data were filtered by 2-pole Vicanek (2016) band-pass filters with a quality factor of $1/16$ at 60Hz and 180Hz. These signals were then augmented with imaginary parts by applying 1-pole all-pass filters matched to produce a phase shift of $\pi/2$ at 60Hz and 180Hz, respectively, yielding a complex-valued EEG signal with $2 \cdot 32 = 64$ channels¹. The ridge regression stage of the frame-wise classifier was substituted by a complex least-squares regression. For window sizes such as a mere 30ms, the classification accuracies `acc0`, `acc1`, `acc_m0`, and `acc_m1` (cf. algorithm 5.4) were nearly identical to, but not significantly better than, the results presented in sections 6.1, 6.2, 6.3, and 6.4, respectively. This suggests that CCVEP decoding model design should focus on the magnitude and phase of the carrier frequency in the case of amplitude-shift keying (ASK) or phase-shift keying (PSK). Similar findings have been published in the context of SSVEP-based BCIs (Falzon, Camilleri, & Muscat, 2012; Ravi, Heydari, & Jiang, 2019).

7.5 Reflections on the machine learning model

Since the present thesis represents a pilot study on CCVEPs, the machine learning model was kept relatively primitive in order to simplify subsequent analyses. This section goes into some of the aspects of the model, including potential pitfalls in the interpretation of the results as well as design choices worth improving upon.

The frame-wise regressor generally performed best when windows were centered around a time point roughly 100ms after the respective frame was shown on the monitor. This coincides with the main peaks (N75, P100, and N135) of a typical VEP. However, optimal window sizes tended to cover a substantially greater duration than the body of a typical VEP. As discussed in section 7.4, one reason for this may be the increased frequency resolution. Another, less fortunate reason may be that the frame-wise regressor learned the structure of the MLS μ to some degree, particularly when the window size was significantly larger than the segment length. The impact of including approximately one additional segment within the sliding window should be minor, since the distribution of binary symbols within the additional range is pseudorandom with a roughly uniform distribution. However, if more segments are included, the segment type that the current frame to be estimated belongs to becomes more predictable based on the other segments’ types, consequently providing clues about the current frame. This may be the reason why the frame-wise accuracies were much more consistent between test subjects for the shortest segment length of $1/8$ s, compared to the longer segment lengths. Note that while

¹Generally, the Hilbert transform (Hardy, Littlewood, Pólya, et al., 1952) is the method of choice to produce a complex-valued analytical signal from a real-valued signal. Here, a Hilbert transform was approximated by filtering the relevant frequencies with appropriately chosen all-pass filters.

7 Discussion

this may imply that the results do not necessarily generalize to the transmission of ECCs, it does not invalidate the accuracies obtained for the MLS shift classifiers. On the other hand, the plots of the hyperparameter space in section 6.5 seem to suggest that increasing the window size beyond the segment length does not improve performance in general. Another reason for the more homogeneous results for the segment length of $1/8$ may be that more training examples per number of relevant model degrees of freedom were provided (3 trials per take instead of 2), yielding a better fit.

Since the frame-wise regression and the segment-wise classification are both based on a composition of two linear functions, standardization and ridge regression, by associativity and distributivity, it is possible to formulate them in tandem as a single cascade of one standardization and one ridge regression. When disregarding the centering of the mean, the tandem kernel is simply the convolution of their respective kernels. The reason this tandem was not learned as a single model are that the amount of training data would be reduced greatly, and the results provided by the frame-wise regressor are relevant in their own right.

One possible modification of the machine learning model would be to use the thresholded frame-wise predictions as input to the segment-wise classifier instead of the non-thresholded estimates. Given a frame-wise classification accuracy $\text{acc}_0 > 1/2$, this would ensure that the worst-case segment-wise classification accuracy would be at least as high as the frame-wise accuracy, and likely substantially higher in the average case. An illustration of this are the solid frame-wise MLS shift classification accuracies, where the thresholded frame-wise predictions were used as input. A post-hoc analysis of including the intermediary thresholding before the segment-wise classification was not done, however, due to time restrictions. Note that thresholding the frame-wise estimates introduces a non-linearity which breaks the tandem interpretation of the frame- and segment-wise regressions mentioned above.

Self-evidently, the standardization, ridge regression, and thresholding stages in the machine learning model may be replaced by any other regression or classification methods. Notably, CNNs have proven superior compared to ridge regression in the context of BCIs previously (Nagel, 2019) and may be a supreme candidate to implement segment-wise classification with multiple intermediary non-linear steps, similar to the frame lightness thresholding mentioned above.

Based on the previous two paragraphs, achieving a high segment-wise classification accuracy seems realistic with the appropriate modifications to the model. This would further raise the appeal of using ECCs instead of MLSs to refine the trade-off between word length and robustness against errors. Moreover, segment classification errors may also be detected through classification confidence scores. Finally, a robust general non-control state detection needs to be incorporated into the model to ensure optimal data integrity.

8 Conclusion and Future Work

The present thesis involved the design, implementation, and evaluation of the novel CCVEP paradigm for BCIs as a gateway to screen-based visual stimulation for data transmission without perceptible flickering. The paradigm is an adaptation of digital modulation techniques to BCI stimulation patterns, utilizing segmented modulation of an imperceptible carrier frequency to encode an n -ary sequence. Special attention has to be paid to visually conceal the transitions between segments, which is a nontrivial problem for which a solution was only approximated. Hence, while the particular stimuli chosen for evaluation subjectively resulted in a vast improvement in observer comfort compared to traditional cVEP-eliciting stimuli, they were still recognizable to a minor degree. No rigorous evaluation of stimulus recognizability or user comfort was performed.

The thesis evaluated OOK and BPSK, each with 3 distinct segment lengths, as CCVEP encoding schemes in a group of 7 test subjects. A basic machine learning model based on two ridge regression stages was used for offline recovery of the transmitted code sequences from the EEG. Classification accuracies were significantly ($p < .01$) above chance for all 6 encoding schemes, with BPSK at a segment length of $1/8$ s yielding the highest mean ITR of $69.9^{\text{bit}}/\text{min}$. Long runs of misclassifications were rare, making the CCVEP paradigm a suitable candidate for the transmission of ECCs. Major room for improvement in ITR from modifications to the model was postulated.

In addition to the bitwise decoding of the recorded CCVEPs, the thesis demonstrated that the established use of MLS cross-correlation for identifying BCI targets can be adapted to the CCVEP paradigm. 15-target MLS shift classification accuracies were significantly ($p < 10^{-4}$) above chance for all 6 encoding schemes, with BPSK at a segment length of $1/8$ s again yielding the highest mean ITR, namely $124.6^{\text{bit}}/\text{min}$.

In summary, the CCVEP paradigm is a promising candidate for screen-based BCIs without perceptible flickering. It can be regarded as a general communication channel capable of transmitting ECCs, but is also suitable for MLS-based target identification. CCVEP classification models based on ASK or PSK should focus on the magnitude and phase response of the carrier frequency and incorporate intermediary nonlinearities, which may yield ITRs substantially higher than $70^{\text{bit}}/\text{min}$.

Seeing as this was a pilot study of a small subset of potentially viable CCVEP modulation schemes, opportunities for future research are plentiful. A large parameter space of modulation schemes, segment lengths, stimulus chrominances, carrier frequencies, classification models, etc. is waiting to be investigated. The construction of perfectly concealed segment transitions is a subject in particular need of further study. Moreover, since the aim of the CCVEP paradigm is to address BCI

8 *Conclusion and Future Work*

ergonomics, subjective measures of user comfort as well as clinical endpoints should be tested for – ideally in larger groups as well as motor paralysis patients.

Bibliography

- Aminaka, D., Makino, S., & Rutkowski, T. M. (2015). Chromatic and high-frequency cVEP-based BCI paradigm. In *2015 37th annual international conference of the IEEE engineering in medicine and biology society (embc)* (pp. 1906–1909). IEEE.
- Aminaka, D., & Rutkowski, T. M. (2017). A sixteen-command and 40 Hz carrier frequency code-modulated visual evoked potential BCI. In *Brain-computer interface research* (pp. 97–104). Springer.
- Armengol Urpi, A., & Sarma, S. (2018, May). Sublime: a hands-free virtual reality menu navigation system using a high-frequency SSVEP-based brain-computer interface. (pp. 1–8).
- BCI2000. (2021). Retrieved October 23, 2021, from <https://www.bci2000.org>
- Belkacem, A. N., Jamil, N., Palmer, J. A., Ouhbi, S., & Chen, C. (2020). Brain computer interfaces for improving the quality of life of older adults and elderly patients. *Frontiers in Neuroscience*, *14*, 692.
- Benda, M., Gembler, F., Stawicki, P., Ben-Salem, S., Islam, Z., Vogelsang, A., & Volosyak, I. (2019). Custom-Made Monitor For Easy High-Frequency SSVEP Stimulation. In *International work-conference on artificial neural networks* (pp. 382–393). Springer.
- Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM review*, *59*(1), 65–98.
- Bin, G., Gao, X., Wang, Y., Hong, B., & Gao, S. (2009). VEP-based brain-computer interfaces: time, frequency, and code modulations [Research Frontier]. *IEEE Computational Intelligence Magazine*, *4*(4), 22–26.
- Bin, G., Gao, X., Wang, Y., Li, Y., Hong, B., & Gao, S. (2011). A high-speed BCI based on code modulation VEP. *Journal of neural engineering*, *8*(2), 025015.
- Bin, G., Gao, X., Yan, Z., Hong, B., & Gao, S. (2009). An online multi-channel SSVEP-based brain-computer interface using a canonical correlation analysis method. *Journal of neural engineering*, *6*(4), 046002.
- Blöck, A. (2018). *The Steady-State Visual Evoked Potential under Unperceived Stimulation as Control Signal in a Brain-Computer Interface*. (Master’s thesis, University of Tübingen).
- Bloomfield, P. (2004). *Fourier analysis of time series: an introduction*. John Wiley & Sons.
- Bravais, A. (1844). *Analyse mathématique sur les probabilités des erreurs de situation d’un point*. Impr. Royale.
- Brenton, R. S., Thompson, H. S., & Maxner, C. (1989). Critical flicker frequency: a new look at an old test. In *New methods of sensory visual testing* (pp. 29–52). Springer.

Bibliography

- Butterworth, S. et al. (1930). On the theory of filter amplifiers. *Wireless Engineer*, 7(6), 536–541.
- Cooley, J. W., & Tukey, J. W. (1965). An algorithm for the machine calculation of complex Fourier series. *Mathematics of computation*, 19(90), 297–301.
- de Lange Dzn, H. (1954). Relationship between critical flicker-frequency and a set of low-frequency characteristics of the eye. *JOSA*, 44(5), 380–389.
- Falzon, O., Camilleri, K., & Muscat, J. (2012). Complex-valued spatial filters for SSVEP-based BCIs with phase coding. *IEEE Transactions on Biomedical engineering*, 59(9), 2486–2495.
- Galton, I., & Welton-Wu, C. (2018). Understanding phase error and jitter: Definitions, implications, simulations, and measurement. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 66(1), 1–19.
- Gembler, F., Stawicki, P., Rezeika, A., & Volosyak, I. (2019). A comparison of cVEP-based BCI-performance between different age groups. In *International work-conference on artificial neural networks* (pp. 394–405). Springer.
- Gembler, F. W., Rezeika, A., Benda, M., & Volosyak, I. (2020). Five shades of grey: Exploring quintary m-sequences for more user-friendly c-VEP-based BCIs. *Computational intelligence and neuroscience*, 2020.
- Golomb, S. W. (2017). *Shift register sequences: Secure and limited-access code generators, efficiency code generators, prescribed property generators, mathematical models*. World Scientific.
- Gomila, L. (2021). Simple and Fast Multimedia Library. Retrieved October 23, 2021, from <https://www.sfm-dev.org>
- Hamarashid, H. K. (2021). Utilizing Statistical Tests for Comparing Machine Learning Algorithms. *Kurdistan Journal of Applied Research*, 6.
- Hanagata, J., & Momose, K. (2002). A method for detecting gazed target using visual evoked potentials elicited by pseudorandom stimuli. In *Proc. 5th asia pacific conf. medical and biological engineering and 11th int. conf. biomedical engineering (icbme)*.
- Hardy, G. H., Littlewood, J. E., Pólya, G., et al. (1952). *Inequalities*. Cambridge university press.
- Hecht, S., & Shlaer, S. (1936). Intermittent stimulation by light V. The relation between intensity and critical frequency for different parts of the spectrum. *Journal of General Physiology*, 19(6), 965–977.
- Herrmann, C. S. (2001). Human EEG responses to 1–100 Hz flicker: resonance phenomena in visual cortex and their potential correlation to cognitive phenomena. *Experimental Brain Research*, 137(3), 346–353.
- Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1), 55–67.
- ISO. (2017, December). *ISO/IEC 14882:2017 Information technology — Programming languages — C++* (Fifth). Geneva, Switzerland: International Organization for Standardization.
- Jiang, L., Wang, Y., Pei, W., & Chen, H. (2019, July). A Four-Class Phase-Coded SSVEP BCI at 60Hz Using Refresh Rate. In *2019 41st annual international conference of the IEEE engineering in medicine and biology society (embc)* (pp. 6331–6334).

Bibliography

- Kaya, I., Bohorquez, J., & Özdamar, Ö. (2021). BCI Performance Improvement by Special Low Jitter Quasi-Steady-State VEP Paradigm. In *Modern approaches to augmentation of brain function* (pp. 121–139). Springer.
- Kaya, I., Bohorquez, J. E., & Ozdamar, O. (2019). Brain Computer Interface Switch Based on Quasi-Steady-State Visual Evoked Potentials. In *2019 9th international ieee/embs conference on neural engineering (ner)* (pp. 1175–1178). IEEE.
- Kaya, I., Bohórquez, J., & Özdamar, Ö. (2019). A BCI gaze sensing method using low jitter code modulated VEP. *Sensors*, 19(17), 3797.
- Kruskal, W. H., & Wallis, W. A. (1952). Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, 47(260), 583–621.
- Lin, F.-C., Chien, Y.-Y., Zao, J. K., Chou, C.-C., Huang, Y.-P., Wang, Y., ... Shieh, H.-P. D. (2014). Imperceptible polychromatic visual stimuli for brain-display interfaces. In *Jsap-osa joint symposia* (20a_C4_5). Optical Society of America.
- Manyakov, N. V., Chumerin, N., Robben, A., Combaz, A., Van Vliet, M., & Van Hulle, M. M. (2013). Sampled sinusoidal stimulation profile and multichannel fuzzy logic classification for monitor-based phase-coded SSVEP brain-computer interfacing. *Journal of neural engineering*, 10(3), 036011.
- Markand, O. N. (2020). *Clinical Evoked Potentials: An Illustrated Manual*. Springer Nature.
- Meyen, S., Zerweck, I. A., Amado, C., von Luxburg, U., & Franz, V. H. (2021). Advancing research on unconscious priming: When can scientists claim an indirect task advantage? *Journal of Experimental Psychology: General*.
- Modabbes, M. S. (2004). Performance analysis of shift keying modulation in the presence of noise. In *Proceedings. 2004 international conference on information and communication technologies: From theory to applications, 2004.* (pp. 271–272). IEEE.
- Müller-Putz, G. R., Scherer, R., Brauneis, C., & Pfurtscheller, G. (2005). Steady-state visual evoked potential (SSVEP)-based communication: impact of harmonic frequency components. *Journal of neural engineering*, 2(4), 123.
- Nagel, S. (2019). *Towards a home-use BCI: fast asynchronous control and robust non-control state detection* (Doctoral dissertation, Universität Tübingen).
- Nagel, S., Dreher, W., Rosenstiel, W., & Spüler, M. (2018). The Effect of Monitor Raster Latency on VEPs, ERPs and Brain-Computer Interface Performance. *Journal of neuroscience methods*, 295, 45–50.
- Nagel, S., Rosenstiel, W., & Spüler, M. (2017). Random Visual Evoked Potentials (RVEP) for Brain-Computer Interface (BCI) Control.
- Nagel, S., & Spüler, M. (2018). Modelling the brain response to arbitrary visual stimulation patterns for a flexible high-speed Brain-Computer Interface. *PLOS ONE*, 13(10), e0206107–.
- Nagel, S., & Spüler, M. (2019). World’s fastest brain-computer interface: Combining EEG2Code with deep learning. *PLOS ONE*, 14(9), e0221909–.
- Nezamfar, H., Orhan, U., Erdogmus, D., Hild, K., Purwar, S., Oken, B., & Fried-Oken, M. (2011). On visually evoked potentials in EEG induced by multiple pseudorandom binary sequences for brain computer interface design. In *2011*

Bibliography

- ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 2044–2047). IEEE.
- Pastor, M. A., Artieda, J., Arbizu, J., Valencia, M., & Masdeu, J. C. (2003). Human cerebral activation during steady-state visual-evoked responses. *Journal of neuroscience*, 23(37), 11621–11627.
- Pearson, K. (1895). Notes on Regression and Inheritance in the Case of Two Parents. In *Proceedings of the royal society of london*, 58, 240-242.
- Ravi, A., Heydari, N., & Jiang, N. (2019). User-independent SSVEP BCI using complex FFT features and CNN classification. In *2019 ieee international conference on systems, man and cybernetics (smc)* (pp. 4175–4180). IEEE.
- Sakurada, T., Kawase, T., Komatsu, T., & Kansaku, K. (2015). Use of high-frequency visual stimuli above the critical flicker frequency in a SSVEP-based BMI. *Clinical Neurophysiology*, 126(10), 1972–1978.
- Sarwate, D. V., & Pursley, M. B. (1980). Crosscorrelation properties of pseudorandom and related sequences. *Proceedings of the IEEE*, 68(5), 593–619.
- Schalk, G., McFarland, D. J., Hinterberger, T., Birbaumer, N., & Wolpaw, J. R. (2004). BCI2000: a general-purpose brain-computer interface (BCI) system. *IEEE Transactions on biomedical engineering*, 51(6), 1034–1043.
- Schiller, P. H., Logothetis, N. K., & Charles, E. R. (1991). Parallel pathways in the visual system: their role in perception at isoluminance. *Neuropsychologia*, 29(6), 433–441.
- Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal*, 27(3), 379–423.
- Shirzhiyan, Z., Keihani, A., Farahi, M., Shamsi, E., GolMohammadi, M., Mahnam, A., ... Jafari, A. H. (2019). Introducing chaotic codes for the modulation of code modulated visual evoked potentials (c-VEP) in normal adults for visual fatigue reduction. *PLoS One*, 14(3), e0213197.
- Shirzhiyan, Z., Keihani, A., Farahi, M., Shamsi, E., GolMohammadi, M., Mahnam, A., ... Jafari, A. H. (2020). Toward New Modalities in VEP-Based BCI Applications Using Dynamical Stimuli: Introducing Quasi-Periodic and Chaotic VEP-Based BCI. *Frontiers in neuroscience*, 14.
- Smith, S. W. et al. (1997). The scientist and engineer’s guide to digital signal processing.
- Spüler, M. (2017). Spatial filtering of EEG as a Regression Problem. In *Gbic*.
- Spüler, M., Walter, A., Rosenstiel, W., & Bogdan, M. (2013). Spatial filtering based on canonical correlation analysis for classification of evoked or event-related potentials in eeg data. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(6), 1097–1103.
- Sutter, E. E. (1984). The visual evoked response as a communication channel. In *Proceedings of the ieee symposium on biosensors* (pp. 95–100).
- Sutter, E. E. (1992). The brain response interface: Communication through visually-induced electrical brain responses. *Journal of Microcomputer Applications*, 15(1), 31–45.
- The Julia Project. (2021). The Julia Programming Language. Retrieved October 23, 2021, from <https://julialang.org>

Bibliography

- Tikhonov, A. N. (1943). On the stability of inverse problems. In *Dokl. akad. nauk sssr* (Vol. 39, pp. 195–198).
- University, P., KAISER, J. F., & KUO, F. F.-K. (1966). *System Analysis by Digital Computer. Edited by Franklin F. Kuo... James F. Kaiser.*[Based on Notes Prepared for a Summer Conference in Computer Science, Sponsored Jointly by Princeton University and the Committee on Computer Science in Electrical Engineering of the Commission on Engineering Education. With Illustrations.]. John Wiley & Sons.
- Vicanek, M. (2016). Matched Second Order Digital Filters.
- Vidal, J. J. (1973). Toward direct brain-computer communication. *Annual review of Biophysics and Bioengineering*, 2(1), 157–180.
- Weinberg, L., & Slepian, P. (1960). Takahasi's results on Tchebycheff and Butterworth ladder networks. *IRE Transactions on Circuit Theory*, 7(2), 88–101.
- Wilkins, A., Veitch, J., & Lehman, B. (2010). LED lighting flicker and potential health concerns: IEEE standard PAR1789 update. In *2010 ieee energy conversion congress and exposition* (pp. 171–178).
- Wisowaty, J. J. (1981). Estimates for the temporal response characteristics of chromatic pathways. *JOSA*, 71(8), 970–977.
- Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., & Vaughan, T. M. (2002). Brain-computer interfaces for communication and control. *Clinical neurophysiology*, 113(6), 767–791.
- Won, D.-O., Hwang, H.-J., Dähne, S., Müller, K.-R., & Lee, S.-W. (2015). Effect of higher frequency on the classification of steady-state visual evoked potentials. *Journal of neural engineering*, 13(1), 016014.
- Wunsch, A. (2006). Artificial lighting and health. *J. Optometr. Photother*, 2, 4–5.
- Yasinzai, M. N., & Ider, Y. Z. (2020). New approach for designing cVEP BCI stimuli based on superposition of edge responses. *Biomedical Physics & Engineering Express*, 6(4), 045018.