Extracting Eye Positions from Electrooculographic Recordings Using a Machine Learning Approach

Frederik Laubisch

Universität Osnabrück

Bachelor Thesis

First Supervisor: Kristoffer Appel, MSc.

Second Supervisor: Prof. Dr. Gordon Pipa

September 27 2016

Table of Contents

List of Figures

List of Tables

Abstract

The Traumschreiber is a sleep mask designed, amongst other things, to track a sleeping person's closed eye movements using EOG signals. This thesis tries to determine the Traumschreiber's eye tracking capabilities. This is done by using a linear regression algorithm in order to establish the relationship between the EOG data provided by the Traumschreiber and eye positions provided by a wearable eye tracking device. A single, wake subject was used for this purpose. It was determined that the data quality of the Traumschreiber is, as of yet, insufficient for reliable eye tracking using a linear regression model. Improvement of the data quality is recommended.

Acknowledgements

I would like to thank all the people who helped me in writing this thesis. I would like to thank Kristoffer Appel for providing me with the opportunity to write the thesis under his supervision, and all his support. I would also like to thank Prof. Dr. Gordon Pipa for being my second supervisor, and Johannes Leugering for effectively being my third supervisor. Furthermore, I would like to thank Inga Ibs for helping me with my writing, Ariane Mader for emotional support and especially Maike Hille for helping me trough this time in all its aspects.

Chapter 1

Introduction

1.1 Introduction

In this thesis, it was attempted to determine whether the Traumschreiber, a sleep mask with attached EOG electrodes, is capable of performing eye tracking. For this purpose, EOG data from the Traumschreiber and eye position data from a head mounted eye tracker were combined in a machine learning setting, using ordinary least squares and simple linear regression. Mentioned concepts and methods are explained, and the experiment described. Results are presented and later discussed, and a conclusion is drawn.

1.2 Eye Tracking

1.2.1 History and Mechanisms

Tracking a persons gaze has been of interest for cognitive scientists for decades (Wade and Tatler [2005]). Researchers first became aware of the importance of eye movements in the late 19th century while studying how people read (Javal [1878]).

To this day, eye tracking is still used for the study of reading, but has become important in other areas as well, such as visual perception, marketing research and recently also in human-computer interaction (Duchowski [2002]). Over the years many different approaches to eye tracking have been taken. While a lot of them have proven to be dead ends, some of the earliest ideas have been retained and improved, and are still used today (Holmqvist et al. [2011]).

The first eye tracking device, built by the psychologist Edmund Huey(Huey [original: 1908, reprint: 1968]), made use of scleral contact lenses. Scleral contact lenses cover the entire corneal surface and are attached to the sclera, the so called "white of the eye". Huey attached aluminum pointers to these lenses, whose movement could then be tracked by an outside observer. Today scleral contact lenses are still in use, though they are way less cumbersome. They can be outfitted with a number of tracking devices, these days most commonly magnetic ones. The most prominent technique is the search coil, where a magnetic coil is attached to a contact lens. The movement of the eye can then be tracked by observing the changes in a magnetic field created by magnetic field frames in which the subjects head is placed (Robinso [1963]). This allows for 3D tracking of the eye.

The first non-invasive eye tracking method was developed by Charles H. Judd and further developed by Guy Thomas Buswell(Buswell [1935]). This "eye movement camera" took motion pictures of subjects eyes which could later be analyzed. Photo- and video-oculography became one of the predominant forms of eye-tracking. Modern devices use small head mounted video-cameras that capture the entire eye. Two methods can be distinguished. Pupillo-graphic methods, where the image of the pupil is tracked, and Purkinje image based methods. Purkinje images are reflections on the boundaries of the cornea and the iris. Four different Purkinje images can be distinguished (see Figure 1.1). Eye movement causes a displacement of these reflections relative the pupil, the limbus and to the other Purkinje images. Usually either only the first Purkinje image (corneal reflection) or both the first and the fourth Purkinje image are used. The latter method has the advantage that these two images move relatively to one another. Thus, movements of the subjects head relative to the camera are not interpreted as eye movements, as it were the case if only one Purkinje image was used. Although head mounted cameras should also minimize this problem, small displacements of the head mount or the cameras can still influence the recording and tracking performance, with measurement errors

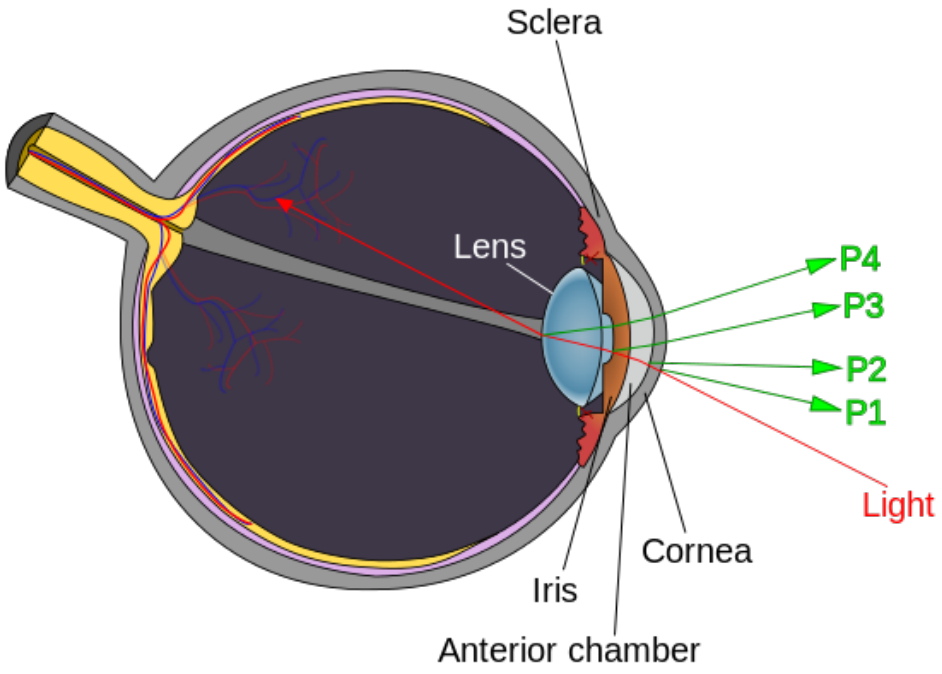


Figure 1.1: The four distinct Purkinje images (source: see ref.Z22 [retrived: Sept. 25 2016])

reported to be about 5 to 10 degree per 1 mm (Young and Sheena [1975]). Most modern devices use invisible infrared light to illuminate the eye, as this allows eye tracking in lower light environments and relative control of the amount of light used. Additionally infrared light does not blind the subject or cause discomfort.

Another technique that uses infrared light is infrared reflectance oculograhy (IROG). An outside light source illuminates the eye with infrared light, and photo-detectors record the reflected light. Pupil, iris and sclera reflect different amounts of light. The limbus and the boundary between iris and pupil are then tracked. IROG requires a constant distance between subject and recording device. Thus, most devices are head mounted. Stationary, table-mounted devices make constant supervision and resetting of the subjects head necessary. Eye blinks cause huge artifacts in the recording. IROG has a limited range for vertical measurements but has a good range and precision for horizontal measurements (Heide et al. [1999]).

Features of interest in eye tacking are saccades and fixations. Fixations are points in the subjects field of view the subject fixates on for a certain amount of time. Saccades describe the movement of the gaze from one fixation point to another. Another feature of interest is smooth tracking, in which the subject fixates on continuously moving feature (Richardson and Spivey).

1.2.2 Electrooculography

One of the earliest eye tracking techniques developed is electrooculography (EOG). The human eye acts as natural electrical dipole, with the anterior side at the retina being negative and the posterior side, at the cornea, being positive(Brown et al. [2006]). This corneo-retinal potential is not generated by excitable tissue in the eye, but is rather due to a metabolic difference between the retina and the rest of the eye, i.e. the retina has a higher metabolic rate(also Brown et al. [2006]). The potential difference along the eye is known to be between about 0.4 mV and 1 mV (Heide et al. [1999]).

Due to the nature of the corneo-retinal potential, it moves with the rotation of the eye. The retinal charge moves with the retina, and the corneal charge with the cornea, such that an eye rotation in one direction results in the negative retinal charge moving in the direction the eye is rotating to, and the positive corneal charge moving away, in the other direction. This affects the electric potential of the skin around the eye socket. Electrodes placed around the eye can detect these changes in the potential (Scott and Scott-Johnson [2002]). Typically two electrodes are placed on either side of the eyes at the outer canthi, to detect changes in the potential caused by horizontal movements of the eye. In order to detect vertical movement, electrodes are typically placed below the eye on the cheek and above the eye, either on the forehead or on the temple(Lypez et al. [2016]). When the eye is oriented directly forwards (central fixation), the difference between the electrodes is roughly zero. Movement in either direction will increase or decrease the difference, depending on the direction of movement. The relation between eye displacement and the EOG signal is known to be linear. However, this linearity remains stable only up to an angle of about 30 degrees from central fixation. (Young and Sheena [1988]). Within this linear interval, the EOG signal changes by about 15-20 µV per degree of

eye rotation(Duchowski [2007]). The precision of eye-movement detection is about 2 degrees and movements can be detected up to 70 degrees visual angle from central fixation (Andreassi [2000]; Stern and and [2001]) .

EOG comes with a number of problems. While the signal is linearly proportional to a certain degree (see above section), its linearity falls apart at higher degrees, i.e. when recording larger eye movements. Furthermore, the signal is vulnerable to variation due to a number of external factors such as luminance, skin resistance and the persons mental state, especially awareness. These and other factors also make the signal variable both across subjects and also across measurements of the same subject at different times or in different environments ()Davis and Shackel [1960]). This results in a non-constant resting potential. The problem can be addressed by filtering out the shift of the resting potential, typically using an amplifier that acts as a low pass filter. EOG is also incapable of detecting torsional eye movements, i.e. rotations of the eye along the corneo-retinal axis, as these do not change the orientation of the dipole (Young and Sheena [1975]).

Most eye tracking mechanisms require the eyes of the subject to be open, since certain features of the eye need to be visible in order for the mechanism to work. Exceptions are scleral contact lenses, where the subject could theoretically close the eyes with the lenses in, provided that the attached equipment does not prevent it. Another exception is EOG, where the eyes of the subject do not need to be directly accessed in any way. Both could theoretically be used to track eye movements in a sleeping person. Hence, they are of interest for sleep and dream researchers. The device used in this thesis, the 'Traumschreiber' utilizes EOG.

1.3 Traumschreiber

1.3.1 Sleep and Dream Research

Psychologists, cognitive scientists, and physiologists have been interested in human sleep and dreams for a long time. The technological advancements in monitoring tools especially neuro-imaging and electro-physiological methods has en-

abled sleep and dream researchers to perform cutting edge sleep studies. The most commonly used type of sleep study is polysomnography. Polysomnography uses a number of physiological characteristics of the subject, such as heart rate, muscle activity and brain waves, to identify the stages of sleep and other features of the subject's sleep. It is also used as a diagnostic tool in medicine. The most common tools used in Polysomnography are electroencephalography (EEG), electromyography (EMG), electrocardiography (ECG) and electrooculography (EOG, as defined in Chapter 1.1.2) (Bloch [1997]). Human sleep is characterized by an alternation between different phases of sleep, typically within a 90 min interval. Two of these phases are of interest in connection with EOG. Rapid eye movement sleep or REM, and the non-REM phase known as N-1 sleep. REM sleep is a phase of sleep characterized by rapid movements of the sleeper's eye (hence the name). It is also the phase of sleep in which dreams can occur. N-1 sleep, or light sleep is the transitional phase between sleep and wakefulness. During N-1 sleep the eyes of the sleeper tend to roll around slowly. Due to these characteristic eye-movement patterns, both phases can be identified using EOG(Carskadon and Dement [2011]).

Of particular interest in sleep and dream research are lucid dreams. Lucid dreams are dreams in which the dreamer is aware of the fact that he is dreaming. In some instances the dreamer may be able to actively and consciously influence the dream. In his groundbreaking PhD thesis Keith Hearne (Hearne [1978]) has shown that subjects experiencing a lucid dream can voluntarily control some of their body functions, such as eye movements. Due to this fact, EOG is an important tool for the study of lucid dreams.

1.3.2 The Traumschreiber

In this thesis the Traumschreiber was used. The Traumschreiber is a sleep mask designed to record and influence a subjects behavior during sleep. Attached to it are four EOG/EEG channels to measure ocular and cerebral activity. In addition it is equipped with four all-color LEDs in the periphery of the subjects eyes, and a speaker to provide visual and auditory stimuli, respectively. The Traumschreiber communicates with a computer or Android device via Bluetooth. The circuit board includes two filters. The first one is an active Sallen-Key high-pass

filter with a cut-off frequency of 0.04 Hz and a gain of 10. The second one is a passive first order low-pass filter with a cut-off frequency of 72 Hz. The analog to digital converter also acts as a low-pass filter, although with a much higher cut-off. The Traumschreiber was designed by Kristoffer Appel and Johannes Leugering at the neuroinformatics department at the University of Osnabrück, under the supervision of Prof. Dr. Gordon Pipa. It is currently still in an development stage. The current version number is 1.1. (see Appel and Leugering [retrieved: Sept 20. 2016]) In order to use the data from the Traumschreiber's EOG electrodes for eye tracking, a linear regression method was used to determine the relationship between EOG data and eye positions, and later predict new positions from new EOG data.

1.4 Background: Linear Regression

1.4.1 Regression

Regression tries to determine the relationship between a set of input variables $\{x_n\}$, and a set of target variables $\{t_n\}$. The variables are usually vectors. If the target variable is multivariate, it is called *multivariate regression*. The input can be multivariate in any case. The goal now is to find a function $f(x)$ that best describes the relationship between these variables. That is, ideally for any x, $f(x) = t$. With that function, new target values can than be found given any new input variables. In most cases where regression is used for this purpose, there is considerable noise in the given observed data. This results in the so called *error* (ϵ) which is the deviation of the observed target variable from the theoretical value of the target variable, as determined by the theoretical underlying function (i.e. the theoretical underlying relationship). Furthermore, said error is assumed to be of a random nature and usually to be distributed normally (i.e. that the error is Gaussian). Once a function $f(x)$ has been found, estimates $\hat{t}$ of the target variable for any input variable can be generated using the function. The difference between an observed target variable t for x and an estimate $\hat{t}$ for x is called the *residual*.

Regression is a form of supervised learning, since a set of target variables is known, the model is trained with the given data, and the performance of the model is than

evaluated based on the known data. There are a number of different methods for regression, some linear and some non-linear. Used in this thesis is *ordinary least squares*, a linear regression method, and the model's performance is evaluated using the *coefficient of determination* r^2. Thus, these concepts and methods will be further elaborated. For further information on regression and linear regression see Myers [2000]

1.4.2 Linear Regression

Linear regression is a special subset of regression in which the model is a linear combination of the input variables and parameters, i.e. coefficients. It is assumed that the underlying function is linear. Furthermore, it is usually assumed that the error is identically and independently distributed (i.i.d.), i.e that the error for all t_i comes from the same distribution (specifically it is usually assumed to be normally distributed), and to be independent from one another. If the input variable is not multivariate, the method is called *simple linear regression*.

1.4.3 Ordinary Least Squares

Ordinary least squares is a linear regression method, i.e. it models the relationship between the data by a function that is a linear combination of the input variables. The underlying function is thus assumed to be linear:

$$\hat{t_i} = w_0 + w_1 \cdot x_{i1} + ... + w_n \cdot x_{in}$$

With $w_0, ..., w_n$ being the coefficients. They can be written as a single vector w of length n. Equally the input variables can be written into a single vector of length n, with the constant term added, as $x_0 = 1$ to include the intercept of the linear function (w_0). Thus the function can be rewritten to: $\hat{t_i} = x_i^T \cdot w$, and can be generalized using matrix notation to: $\hat{t} = \mathbf{X} \cdot w$. The underlying function is assumed to be of the kind $t_i = x_i^T \cdot w + \epsilon_i$, or using matrix notation: $t = \mathbf{X} \cdot w + \epsilon$

It becomes apparent that $t = \hat{t} + \epsilon$ or $\epsilon = t - \hat{t}$.

For any arbitrary model, the residuals $t_i - x_i^T \cdot w$ can be computed. To measure the overall performance of that model, the sum of squared residuals SSR is used:

$$SSR = \sum_i (t_i - \hat{t}_i)^2$$

From this the estimators for w can be derived. The derivations presented here closely follows the derivation as done by [Federico [retrieved: Aug. 7 2016]]. Again by combining target variables (both observed and estimated) to vectors, the sum can be replaced:

$$SSR = (t - \hat{t})^2$$

and using vector notation:

$$SSR = (t - \hat{t})^T \cdot (t - \hat{t})$$

Substitute $\hat{t} = \mathbf{X} \cdot w$:

$$SSR = (t - \mathbf{X} \cdot w)^T \cdot (t - \mathbf{X} \cdot w)$$

Multiplying out the parantheses yields:

$$SSR = t^T \cdot t - w^T \cdot \mathbf{X}^T \cdot t - t^T \cdot \mathbf{X} \cdot w + w^T \cdot \mathbf{X}^T \cdot \mathbf{X} \cdot w$$

Since $w^T \cdot \mathbf{X}^T \cdot t = t^T \cdot \mathbf{X} \cdot w$, this simplifies to:

$$SSR = t^T \cdot t - 2t^T \cdot \mathbf{X} \cdot w + w^T \cdot \mathbf{X}^T \cdot \mathbf{X} \cdot w$$

Since the goal is to find the w such that SSR is minimal, the minimum of the SSR needs to be found. This is the case where the partial derivative with respect to w is 0. In accordance with the sum rule for derivatives, the derivative of a sum is the sum of the derivatives of the partial factors in the sum. Hence:

$$\frac{\delta}{\delta w} t^T \cdot t = 0,$$

$$\frac{\delta}{\delta w}[-2t^T \cdot \mathbf{X} \cdot w] = -2 \cdot \mathbf{X}^T \cdot t$$

, and

$$\frac{\delta}{\delta w}[w^T \cdot \mathbf{X}^T \cdot \mathbf{X} \cdot w] = 2 \cdot \mathbf{X}^T \cdot \mathbf{X} \cdot w$$

Therefore:

$$\frac{\delta}{\delta w}[SSR] = -2 \cdot \mathbf{X}^T \cdot t + 2 \cdot \mathbf{X}^T \cdot \mathbf{X} \cdot w$$

Setting the derivative to 0 yields:

$$0 = -2 \cdot \mathbf{X}^T \cdot t + 2 \cdot \mathbf{X}^T \cdot \mathbf{X} \cdot w$$

$$2 \cdot \mathbf{X}^T \cdot t = 2 \cdot \mathbf{X}^T \cdot \mathbf{X} \cdot w$$

$$\mathbf{X}^T \cdot t = \mathbf{X}^T \cdot \mathbf{X} \cdot w$$

$$\frac{\mathbf{X}^T \cdot t}{\mathbf{X}^T \cdot \mathbf{X}} = w$$

Now $\hat{w} = \frac{\mathbf{X}^T \cdot t}{\mathbf{X}^T \cdot \mathbf{X}}$ is the estimator for w.

1.4.4 Simple Linear Regression

Simple linear regression is a special case of the ordinary least squares method. It works when the input variables x_i are scalars, i.e. they are not multivariate. This renders the problem much simpler than the usual ordinary least squares. Instead of including the intercept in the weight vector and adding a constant term to the input vector, the model has the classical form of a linear equation: $t = a + b \cdot x$, with the underlying function assumed to be: $t_i = a + b \cdot x_i + \epsilon_i$. The SSR can be derived to:

$$SSR = \sum_i (t_i - a - b \cdot x_i)^2$$

Two partial derivatives need to be found. One with respect to a and one with respect to b. Again using the sum rule, this becomes:

$$\frac{\delta}{\delta_{a,b}}[SSR] = \sum_i (\frac{\delta}{\delta_{a,b}}[t_i - a - b \cdot x_i]^2)$$

Now deriving with respect to a:

$$\frac{\delta}{\delta a}[SSR] = -2 \sum_i (t_i - a - b \cdot x_i)$$

Setting to 0 yields:

$$0 = -2 \sum_i (t_i - a - b \cdot x_i))$$

divide by -2

$$0 = \sum_i (t_i - a - b \cdot x_i)$$

splitting up sums:

$$0 = \sum_i t_i - \sum_i a - \sum_i (b \cdot x_i)$$

since a and b are constant:

$$0 = \sum_i t_i - a \cdot n - b \cdot \sum_i x_i$$

isolatingS a:

$$a = \sum_i \frac{t_i}{n} - b \cdot \sum_i \frac{x_i}{n}$$

Since the sums are simply the mean of t and x over n observations:

$$a = \bar{t} - b \cdot \bar{x}$$

Now $\hat{a} = \bar{t} - \hat{b} \cdot \bar{x}$ is the estimator for a, with $\hat{b}$ being the estimator for b as derived in the following paragraph.

Similarly, for b:

$$\begin{aligned} \frac{\delta}{\delta b}[SSR] &= \sum_i -2 \cdot x_i \cdot (t_i - a - b \cdot x_i) \\ &= -2 \cdot (\sum_i (x_i \cdot (t_i - a - b \cdot x_i))) \end{aligned}$$

And setting to 0 yields

$$0 = -2 \cdot (x_i \cdot (\sum_i (t_i - a - b \cdot x_i)))$$

Dividing by -2

$$0 = x_i \cdot (\sum_i (t_i - a - b \cdot x_i)$$

Splitting up sums with x_i:

$$0 = \sum_i (t_i \cdot x_i) - \sum_i (a \cdot x_i) - \sum_i (b \cdot x_i^2)$$

Since a and b are constant:

$$0 = \sum_i (t_i \cdot x_i) - a \cdot \sum_i (x_i) - b \cdot \sum_i (x_i^2)$$

Isolating b:

$$b * \sum_i (x_i^2) = \sum_i (t_i \cdot x_i) - a \cdot \sum_i (x_i)$$

Now substituting a:

$$b \cdot \sum_i (x_i^2) = \sum_i (t_i \cdot x_i) - [a = (\sum_i \frac{(t_i)}{n}) - b \cdot (\sum_i \frac{x_i}{n})] \cdot \sum_i (x_i)$$

Now multiplying out the rightmost term yields:

$$b \cdot \sum_i (x_i^2) = \sum_i (t_i \cdot x_i) - (\sum_i (t_i) \cdot \frac{\sum_i (x_i)}{n}) - b \cdot \frac{\sum_i (x_i)^2}{n}$$

Adding rightmost term:

$$b \cdot \sum_i (x_i^2) + b \cdot \frac{\sum_i (x_i)^2}{n} = \sum_i (t_i \cdot x_i) - (\sum_i (t_i) \cdot \frac{\sum_i (x_i)}{n})$$

The right part of the equation is exactly the covariance between t and x. On the left side b is factored out:

$$b \cdot (\sum_i (x_i^2) + \frac{\sum_i (x_i)^2}{n}) = \sum_i (t_i \cdot x_i) - (\sum_i (t_i) \cdot \frac{\sum_i (x_i)}{n})$$

Finally, dividing by the left term to isolate b on the left-hand side of the equation yields:

$$b = \frac{\sum_i (t_i \cdot x_i) - (\sum_i (t_i) \cdot \frac{\sum_i (x_i)}{n})}{(\sum_i (x_i^2) + \frac{\sum_i (x_i)^2}{n})}$$

Now the the denominator on the right side is the variance of x. Therefore the equation can also be written as:

$$\hat{b} = \frac{Cov[t, x]}{Var[x]}$$

This gives the estimator for the parameter b ($\hat{b}$).

1.4.5 Coefficient of Determination

To evaluate the performance (or *goodness of fit*) of the model , the *coefficient of determination* r^2 is calculated. The coefficient of determination can be defined as the ratio between the variance explained by the model, and the total variance of the

target data, such that:

$$r^2 = \frac{\sum_i (\hat{t}_i - \bar{t})^2}{\sum_i (t_i - \bar{t})^2}$$

The numerator is exactly the sum of squared residuals, as mentioned before. The denominator is called the total sum of squares and is simply the variance of the data. The coefficient of determination r^2 is normally in the interval $[0, 1]$. A value of 0 means that the model explains none of the variance, and a model that would predict exactly the mean of the data ($\bar{t}$) for every input variable would perform just as good. A value of 1 would be considered a perfect fit, and the model explains all of the variance. The value can be negative under certain circumstances. The first would be the case if the intercept was not included in the model. This was done in this specific case of the derivation of $\hat{w}$. It can also be negative if the data used for computation of r^2 was not the same data as was used for the computation of $\hat{w}$. Thus, when using different sets of data for training and testing, the model can perform worse on the test data set than simply the mean as a constant predictor for all data points.

1.4.6 Cross-Validation

A common problem with regression, and machine learning in general is overfitting. Overfitting occurs when, given a set of training data, the algorithm finds a near perfect fit for that data set, but given any other data set from the same underlying function, would perform poorly. It is said that algorithm failed to generalize. One way to address this problem is to use cross-validation. For cross-validation, the given data is split into training and test data sets. This happens either by assigning each data point randomly to one of the k subsets, such that all k subsets are of equal size, or by simply splitting the data set into k equally sized subsets sequentially. This results either in shuffled or non-shuffled subsets. The training data is used to train the model, and performance is then evaluated on the test data. There are a number of different cross-validation approaches. A common method, and also the one used here is k-fold cross-validation. In this method the data is split into k subsets of equal size. Training of the model commences on the dataset of $k - 1$ subsets, and testing on the remaining subset. This process is done k times, always with one subset as the test sample, and the rest as the training sample.

Chapter 2

Methods

2.1 Subject

This thesis tries to establish whether the Traumschreiber at its current state is capable of performing eye tracking with the attached EOG electrodes at all, rather than optimizing performance. Therefore, only a single subject was used for the experiment. The subject was a 26 year old male, right handed, with normal vision on both eyes.

2.2 Stimulus

The stimulus used was a single moving dot stimulus. A single moving dot stimulus is a common stimulus in vision research, as it allows to direct the subjects gaze in a controlled manner, uninfringed by distractions and biases(Duchowski [2007]). The stimulus was created using *Pyplot* from the *Matplotlib* library for Python. Python version 3.5.1 and Anaconda 2.5.0 (64-bit) distribution were used. A scatter plot function was used to create a black dot of size 80^2 points, that changed positions based on pseudo-randomly generated continuous coordinates in a normalized square space on a white background. It changed position every 2 seconds, for

a total of 10 minutes resulting in a total of 200 data points which corresponds to a sampling rate of 0.5 Hz. For each data point a time stamp in ISO format was derived from system time. Coordinates with corresponding time stamp were saved in a Python matrix.

2.3 Eye tracker

In addition to the Traumschreiber, a Pupil Labs eye tracker was used. The device is an infrared based dark pupil detection eye tracker in the form of a 3D-printed frame that is worn like glasses. It is designed to be modular so it can be customized for the users purposes. The device used, was equipped with a world camera mounted above the eye portion of the device, and two eye cameras with an infrared band-pass filter, mounted on rails on each side of the device such that they can capture the entire eye of the subject. They come with an attached infrared LED at 860nm wavelength, that illuminates the corresponding eye. All cameras are adjustable for calibration. The device has a 0.6 degree of visual angle accuracy, with a 0.08 degree precision and a sampling rate of up to 120 Hz for eye capturing (Kassner et al. [2014]). The eye tracker can be connected to a computer via USB 2.0. As the device works with infrared but does not depend on corneal reflections, it can be used with subjects wearing contact lenses or glasses.

Pupil Labs provides two software packages for the eye tracker, Pupil Capture for recording and Pupil Player for playback of the captured data. Only the Pupil Capture software was used. The algorithm used by the Pupil Capture software detects the dark pupil in the infrared image from the head mounted eye cameras (for specific details on the algorithm see Kassner et al. [2014]) The software also provides algorithms for calibration of the eye tracker and plug-ins that help with adjustment of the eye cameras. The calibration method used was a the ”screen marker calibration”, a succession of 9 animated points the subject was asked to fixate. The Pupil Capture software records the positions of the pupils in the eye camera space, together with corresponding timestamps. It also maps the pupil positions into the world camera space and saves them as gaze positions, also with corresponding timestamps.

2.4 Setup

The subject was placed in front of a flat screen monitor of height 36.1 cm and width 55.5 cm. The distance between the monitor and the subjects head was 40 cm. The area within which the stimulus was to be displayed was of height 23.6cm and width 38.5cm. A total of four electrodes of the Traumschreiber were attached to the subjects head. One slightly above the temple next to the each eye, one at the center of the forehead and one below the left eye on the cheek (See Figure 2.1) . The four electrodes where connected to the four channels on the Traumschreiber. Disposable Ag/AgCl electrodes with conductor gel were used. The rest of the Traumschreiber was placed behind the subjects head in a makeshift poach, and connection with the computer was established via Bluetooth. Figure 2.1 Shows the electrode placement and numbering. Electrodes 1 and 2 made up one channel (channel 2) and electrodes 2 and 3 another channel (channel 1). Electrode 4 acted as ground. A Python script ran from the computer received the electrooculographic data from the Traumschreiber and saved it with corresponding timestamps. The sampling rate of this process was 250 Hz.

The Pupil eye tracker was placed on the subjects head. The top-mounted world camera was adjusted such that it captured approximately the subjects field of view. Both eye cameras were adjusted such that they capture the eyes of the subject properly. For this purpose the Pupil Capture software was used to determine a good view of the subjects pupils for the eye cameras. The subject chose a comfortable sitting position, in which he stayed for the duration of each trial, and was told to keep his head as still as possible for the duration of each trial as well. Afterwards, the Pupil Capture software's built in calibration function was used to calibrate the eye tracker until a satisfactory performance was achieved. The confidence calculated by the Pupil Player for calibration never fell below 0.85. Calibration consists of a moving dot stimulus that the subject is asked to fixate on. Calibration was repeated after each trial. The pupil eye tracker was connected to the computer via USB, feeding its data directly into the Pupil Capture software. The computer ran the Ubuntu 12.04 LTS Linux distribution.

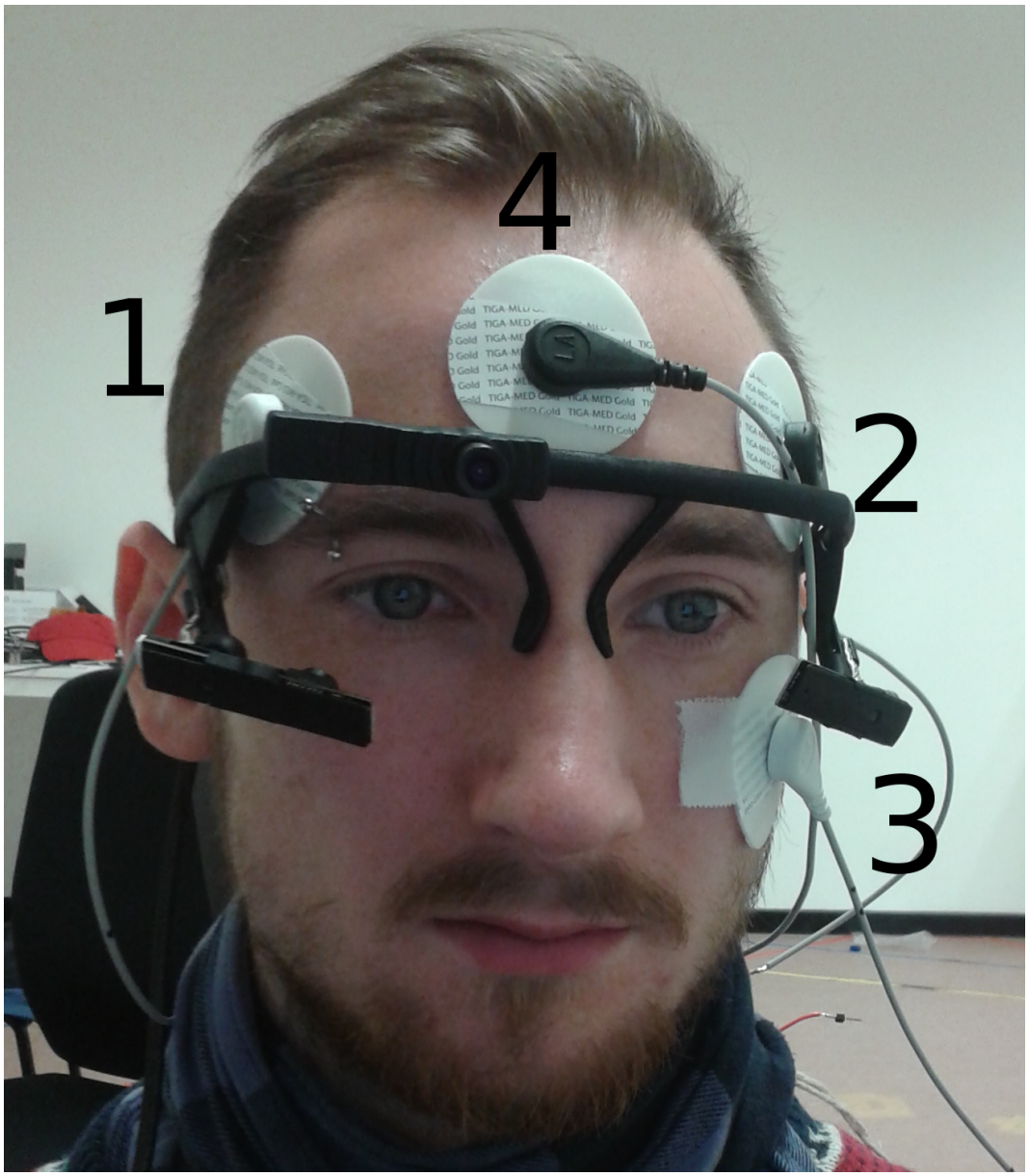


Figure 2.1: Placement of the electrodes and their assigned number labels

2.5 Procedure

Once calibration was done and the subject was ready, the Traumschreiber was activated. Since there was no practical way of synchronizing all three procedures

(i.e. stimulus, eye tracker recording and Traumschreiber recording) all three were started sequentially. First the Pupil Capture software started recording, since it does not require a running time to be set at the start. Then the Traumschreiber script was started, with an intended running time of 12 minutes. Finally the stimulus was started, which would run for 10 minutes. The subject started fixating on the stimulus at its activation. Once the stimulus was finishes, the pupil capture recording was terminated manually, and the Traumschreiber would keep recording for the remainder of its 12 minutes. Excess data was cut off later (see section 2.4) . The procedure was repeated two times in total, resulting in three data sets with 10 minutes of usable data each.

2.6 Pre- & Postprocessing

Since only the stimulus data and the electrooculographic data from the Traumschreiber had timestamps in ISO format, the eye-tracker timestamps, which were measured in seconds from the start of the epoch (floating point precision), needed to be converted. This was done using the start time saved by the Pupil Capture software. Using these timestamps, excess data from both the eye tracker and the Traumschreiber datasets were cut off, such that only data within the time intervals in which the stimulus was shown remained. Furthermore, the Pupil eye tracker does not write all of its data chronologically into the corresponding file. This is due to the fact that it has to integrate data from a number of different sources into a single file. Therefore the pupil data used for machine learning needed to be sorted. The *argsort()* function native to Python was used to sort the data based on the time stamps, utilizing the quicksort algorithm.

The Traumschreiber has a sampling rate of 250 Hz while the Pupil eye tracker's sampling rate is only 120 Hz. For the machine learning algorithm it is necessary that both correlated data sets have the same number of data points. Furthermore it is of convenience if these data points occur at the same time steps. Therefore, more data points needed to be created for the eye tracker data. This was done using linear interpolation. Linear interpolation is an interpolation method where new data points are created, given a discrete finite set of data points. The process makes the

assumption that in between any of the given data points the underlying function behaves linearly. This is of course an approximation of the original function and in no way represents the actual underlying function. It is, however, sufficient for this purpose. *SciPy's* interpolation library was used to create an interpolation function from the Pupil eye tracker data. This function is an approximation of the underlying function, that is piecewise linear between the given data points. The function was then used to determine new data points given the Traumschreiber data points (i.e new data points were created at the time steps where the Traumschreiber data has values). Now each Traumschreiber data point had one Pupil eye tracker data point associated with it, at a given time point.

Later the data was smoothed using a one-dimensional Gaussian filter. The corresponding function found in the SciPy filter library was used (scipy.ndimage.filters.gaussian_filter1d). Filter parameters were determined during the cross-validation together with the model parameters of the machine learning model (see 3.2). The only parameter that needed to be determined was the standard deviation of the Gaussian kernel (σ).

2.7 Regression

After post-processing of all the data, data from the Traumschreiber and from the eye tracker were used for the regression. For the Traumschreiber as it was later intended to be used, the position of the pupils is more of interest, rather than the position of the gaze in the world camera space. This is due to the fact that in the final version of the Traumschreiber, the subject is supposed to be asleep with their eyes closed. Thus, the pupil positions from the Pupil eye tracker were used in the regression. The pupil positions given by the Pupil eye tracker are positions in eye camera space coordinates. The electrode data from the Traumschreiber was used as the regressor, and the pupil positions as the regressands. Specifically, the regressor was either a single channel, or both channels, and the regressand either the vertical or the horizontal coordinate of the pupil position from the Pupil eye tracker. This results in separate models for the vertical and the horizontal movement prediction. Variables for the filter parameters were set as sigma_t for the filter parameter for

the Traumschreiber data, and sigma_e for the filter parameter for the eye tracker data. The data was split into 10 subsets for 10-fold cross-validation. This was done without shuffling the data. Shuffling would result in what could be considered unnatural data. Each set of computations was repeated for each tuple of filter parameters $sigma\_t$ and $sigma\_e$. The *scikit learn* library (see Buitnich et al. [2013]) was used for regression, specifically the class *sklearn.linear_model.LinearRegression*. The class implements an ordinary least squares linear regression by instantiating a model object which then can be trained using data (method: *fit()*), resulting in a set of coefficients that are saved in a instance variable (*coef_*). New data can be predicted using the *predict()* method, and a score can be computed for any prediction (method: *score()*) As mentioned in chapter 1.1.1 the relationship between EOG activity and eye movements has been shown to be linear for up to 30 degrees visual angle from central fixation. Since the stimuli of the experiment were only displayed within a small window on the screen, most of the eye movement should be within the 30 degree of linearity. For each combination of sigmas, the coefficients were computed and an average value was obtained via cross-validation. Afterwards a score was computed using the ordinary least squares error function, by the *score()* function of the linear model. All parameters and the corresponding score were saved in a matrix. The best combination of filter parameters was then picked based on the score calculated by the score method of the model instance. The function can perform both simple linear regression and multiple linear regression.

As mentioned before, the Traumschreiber has two channels. One channel measures the potential difference between the electrode placed on the left temple and the one on the left cheek (channel 1). The other channel measures the potential difference between the electrodes placed on the temples (channel 2). Since channel 1 is measured by electrodes above and below the eyes, it is assumed that channel 1 would be best utilized for measuring vertical eye movements. By the same reasoning channel 2 should perform best in predicting horizontal eye movements. Since both channels should carry almost only information regarding the one eye movement that they should measure, it is possible that data from only that one channel performs much better in predicting that particular eye movements, than data from both channels together. To test this hypothesis, the regression was performed for each channel separately, and finally again using both channels together. All computations were done for each of the three 10 min sessions separately. In the single channel condition, the algorithm is a simple linear regression, since only a single

regressor is used. In the multi channel condition it works according to the ordinary least squares method, as described in chapters 1.3.4 and 1.3.5.

Chapter 3

Results

3.1 Single Channel

The *score()* function of the *sklearn* model computes the *coefficient of determination* r^2 of the prediction, as defined in chapter 1.3.5. Figure 3.1 and 3.2 show the line fitted by the model plotted together with the test data set for each condition for session 1 (for plots of session 2 and 3 results, see appendix). Tables 3.1 - 3.3 show the scores for all conditions. In all three session, relatively good scores are archived for the prediction of the horizontal position of eye 0. While these scores are still fairly low, they are the best scores in the single channel condition, with a mean of .17 . For the other eye (eye 1), scores for this condition are significantly worse, as almost all of them are negative with a mean of -.17. Interestingly data from channel 1 achieves much higher scores for prediction of eye 0 than channel 2, but also much lower scores for prediction for eye 1, where the lowest values of the experiment can be found. The prediction of the vertical eye position yields relatively low score over both eyes and all three session. All scores are below .05 with the exception of eye 1 in session 2 with a score of .12. The mean score for vertical predictions over both eyes is only 0.04. Generally performance was much better on eye 0 than on eye 1 with means of .05 and -.05, respectively.

	E0		E1	
	H	V	H	V
channel 1	0.18	0.01	-0.27	0.06
channel 2	0.005	0.05	0.008	0.03
both channels	0.34	0.11	0.35	0.07

Table 3.1: Session 1: Scores computed by score() method of sklearn model
E0 = eye 0, E1 = eye 1, H = horizontal, V = vertical

	E0		E1	
	H	V	H	V
channel 1	0.16	-0.005	-0.3	0.12
channel 2	-0.003	0.06	0.004	0.05
both channels	0.27	0.06	0.25	0.12

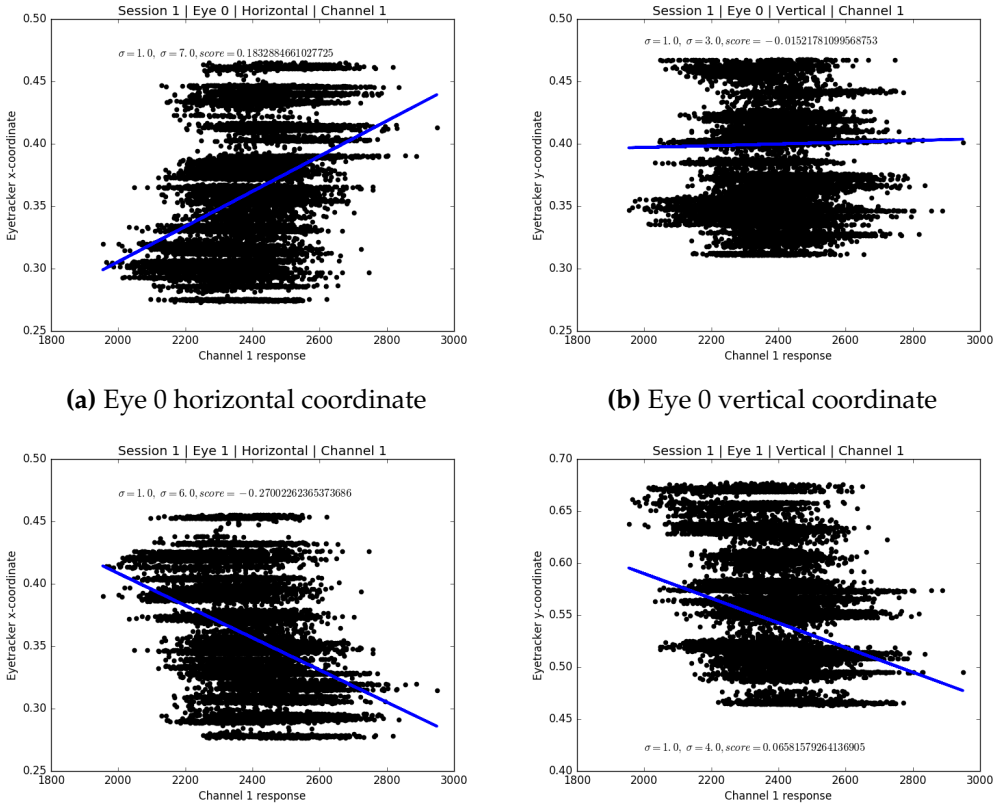
Table 3.2: Session 2: Scores computed by score() method of sklearn model
E0 = eye 0, E1 = eye 1, H = horizontal, V = vertical

3.2 Multi Channel

Using the same score function, but with both channel 1 and channel 2 as the regressors, results were obtained. Generally the model performs much better when trained using data from both channels simultaneously. The mean performance is .198 compared to -0.15 and 0.015 for channel 1 and channel 2 alone, respectively.

	E0		E1	
	H	V	H	V
channel 1	0.19	-0.006	-0.4	0.07
channel 2	-0.001	0.05	-0.07	0.05
both channels	0.27	0.06	0.25	0.0027

Table 3.3: Session 3: Scores computed by score() method of sklearn model
E0 = eye 0, E1 = eye 1, H = horizontal, V = vertical



(a) Eye 0 horizontal coordinate

(b) Eye 0 vertical coordinate

(c) Eye 1 horizontal coordiante

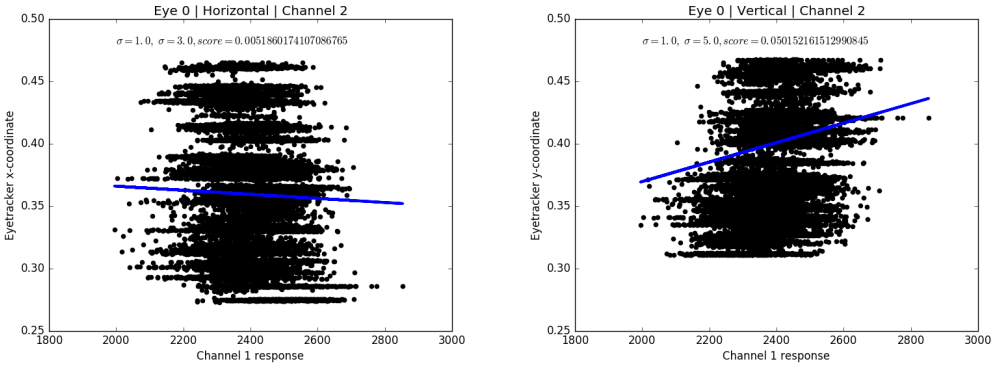
(d) Eye 1 vertical coordinate

Figure 3.1: Regression lines for session 1 single channel (channel 1) condition

Although again, just as in both other conditions, performance is much better for the prediction of the horizontal eye position ($\mu = 0.305$) than it is for the prediction of the vertical eye position ($\mu = 0.09$). It outperforms the single channel trained models in all conditions. Still the score never surpasses .35. In this condition there is no significant difference between the eyes ($\mu_{e0} = 0.19, \mu_{e1} = 0.2$).

3.3 Overall Performance

Overall, the model performed much poorer for prediction of the vertical eye position. Similarly, the scores are much lower for eye 1. The models performance did remain relatively stable over the course of the three sessions.



(a) Eye 0 horizontal coordinate

(b) Eye 0 vertical coordinate

(c) Eye 1 horizontal coordiante

(d) Eye 1 vertical coordinate

Figure 3.2: Regression lines for session 1 single channel (channel 2) condition

Chapter 4

Discussion

The results clearly show that using data from both channels results in a much better performance for both directions and on both eyes. Even tough the single channels should only contain the relevant information for the particular direction it is measuring, the model performs much better when the data from the other channel is also used. This would suggest that the other channel also contains information about the particular direction.

The model performed much better in predicting horizontal eye positions than in predicting vertical eye positions. This is in accordance with previous findings that EOG performs much better in measuring horizontal movements than vertical movements (Acuña et al. [2014]). Therefore, the problem is not with the algorithm itself, since poorer performance on the vertical position can be expected due to the nature of the EOG signal. Of course this only explains the difference in performance between the two directions, rather than the actual overall performance level.

The performance also differed significantly between the two eyes. Table 4.1 shows the confidence score averages for the two eyes, as they are computed by the Pupil Capture software. Since there is no significant difference in the confidence between the two eyes, it can be assumed that the eye tracker performed equally well in tracking of both eyes. Therefore, the source of the difference in the performance must be assumed to be in the Traumschreiber data. This could possibly be due to the place-

Confidence	Eye 0	Eye 1
Session 1	0.919592553593	0.947961047066
Session 2	0.933082608005	0.923733363983
Session 3	0.922947485277	0.873983091204

Table 4.1: Eye tracker confidence scores

ment of the electrodes. Channel 2 consists of the electrodes 1 and 2 on the temples (see Figure 2, chapter 2). Their placement relative two the two eyes is the same for both eyes. The difference in performance between the eyes on data from channel 2 only is minimal (see Tables 3.1-3.3). However, the difference is much more significant on data from channel 1 only. Channel 1 consists of electrodes 2 and 3, which are located much closer to the left eye. It is conceivable that this somehow impacted the data and therefore, the performance of the algorithm, especially for eye 1. Although, this is unprecedented in the literature, and the electrode placement used here is a common one for EOG studies.

There was a significant problem concerning the EOG data delivered by the Traumschreiber. Figures of the raw data can be found in the appendix. It can also be seen in Figure 3.1 and 3.2 that there is a great variance in the EOG data at every eye position. Even after applying two hardware filters during recording and a Gaussian filter afterwards, the data from the Traumschreiber remains fairly noisy. This is a great problem since in classical linear regression it is assumed that the regressor is measured without noise, and that only the regressand is noisy. That is, the dependent variable is a linear combination of the independent variables plus a noise term, while the independent variable is assumed to be the exact underlying value. There are models that can deal with noise in the regressor. These models are called *errors-in-variable models*. However, most of these methods make certain assumptions about the variance of the nose terms. The simplest method, *orthogonal regression*, is an expansion of the ordinary least squares approach used here. This method, for example, assumes that the variance of both noise term (i.e of the dependent and of the independent variable) is the same. Other methods, such as *Deming regression*, also make such assumptions (in the case of Deming regression, the ratio of the two variances is assumed to be known). Therefore it is usually necessary to know the variances. Since the data was measured in fundamentally different ways

and is of completely different kinds (although technically from the same underlying source), it can not be assumed that they have the same error variance, neither can other necessary assumptions about the variances be made. Using orthogonal or Deming regression might still yield slightly better results than ordinary least squares as used here. This is of course due to the fact that for the ordinary least squares approach a major assumption was violated. It is also important to consider that the Traumschreiber is still in a fairly early development stage, and hence is prone to errors. The Pupil eye tracker on the other hand is much further developed and thus, it could even be assumed that the Traumschreiber delivers data that is even more noisy than the eye tracker data.

In this thesis the model used was a linear regression model. Due to the nature of the experimental design, the stimulus presented was only in a relatively small part in the center of the subjects visual field. Previous studies have shown that the relationship between the EOG signal and the eye position is linear within thirty degree from central fixation (see chapter 1.2.2). For this stimulus the assumption was made that most of the stimulus is only within this linear part. Such assumptions can not be made in general. Since the Traumschreiber needs to be able to detect eye movements of any degree, it would probably be useful to entertain the possibility of using a non-linear model, in order to account for the non-linear parts. Of course the actual necessity can only be established once the device delivers data of better quality and with a stimulus that covers a bigger part of the subjects visual field.

Linear models for regression require a number of assumptions to be made about the data. One assumption simply concerns the linearity of the underlying model. When using linear regression to model the relationship of the data, it is naturally assumed that the relationship is actually linear. This assumption was made in this case given the fact that a linear relationship between EOG data and eye movement has been shown before, for a certain part of the visual field, as mentioned before (see chapter 1.2.2).
Another assumption is that the noise term of the dependent variable is i.i.d (independently and identically distributed), i.e. that the noise of an observation is not dependent of any other observation (i.e. no autocorrelation), and that they all come from the same distribution with a constant variance and an expected value of 0. The data from the Pupil eye tracker has many sources of error. Some of them are systematic, and therefore are not considered to be part of the error term. These

includes the calibration error. Of the other errors, some errors, mostly those caused by human errors (e.g slight misplacement of the device during the experiment) are more problematic. Some error sources can be considered to be i.i.d., others not. Due to the high number and variability of error sources in the signal, the assumption may need to be reconsidered.

Overall, using the provided data with the linear regression method herein used, reliable prediction of the eye positions is not possible. This is particularly true for the vertical eye position. Also, with the current electrode placement, prediction of the position of eye 1 alone is completely impossible. This may be practically unimportant for the intended use of the Traumschreiber. It also can be addressed easily by changing the placement of the electrodes.

Chapter 5

Conclusion

At the current state, the quality of the data delivered by the Traumschreiber is insufficient to predict eye positions to a satisfactory degree. It is recommended to improve the quality of the EOG data before further attempting to implement the eye tracking features of the device. Furthermore, in future experiments, it would be beneficial to considered a regression method that can a) deal with noise in the independent variable and b) allows for non-linearity. This is due to the fact that the relationship between the EOG data and the eye positions is only linear within a window of 30 degree visual angle. It is advisable to use data from both channels as the regressor, since the algorithm performed much better using both channels.

Appendix A

Appendix

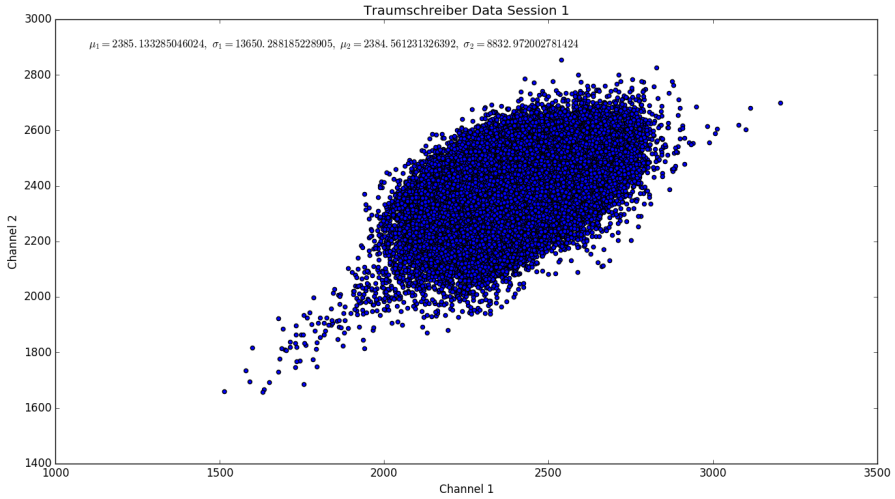


Figure A.1: Session 1: raw Traumschreiber EOG data

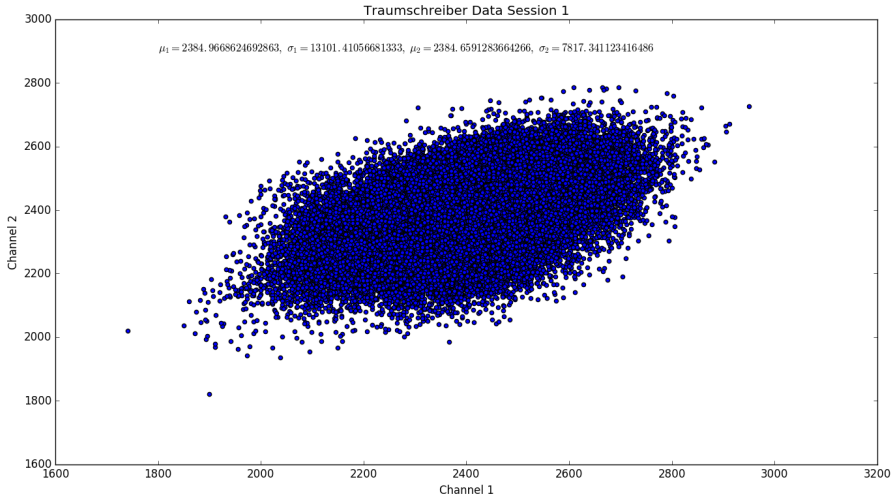


Figure A.2: Session 2: raw Traumschreiber EOG data

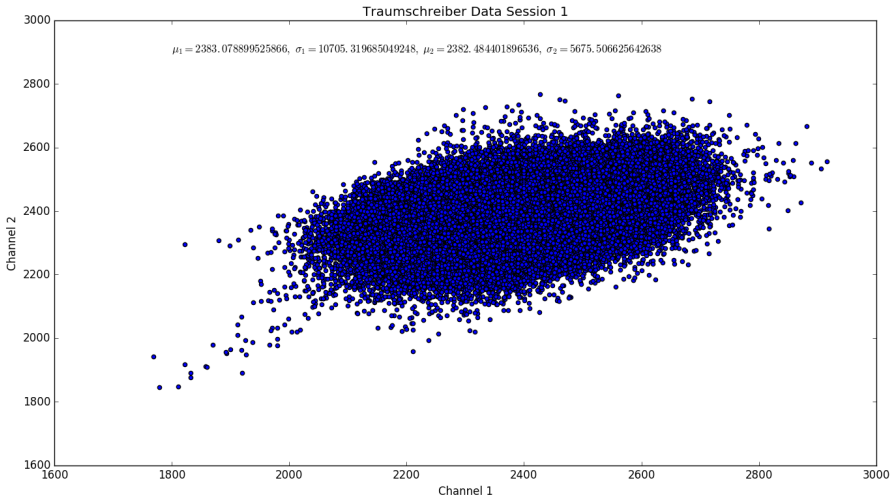
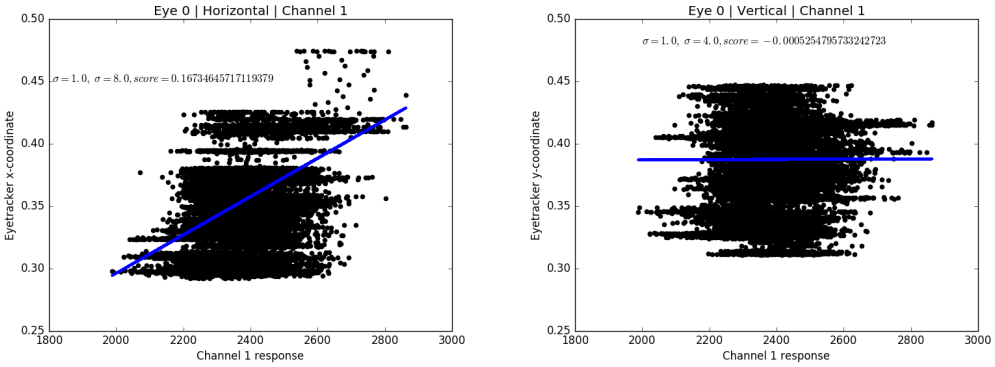


Figure A.3: Session 3: raw Traumschreiber EOG data



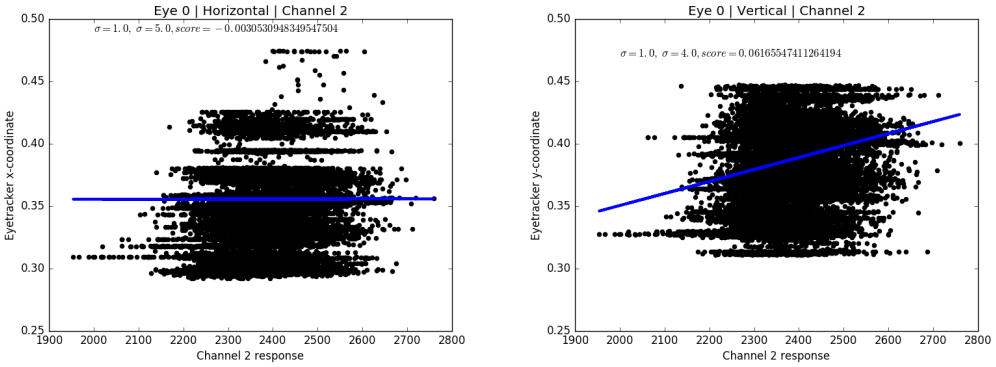
(a) Eye 0 horizontal coordinate

(b) Eye 0 vertical coordinate

(c) Eye 1 horizontal coordiante

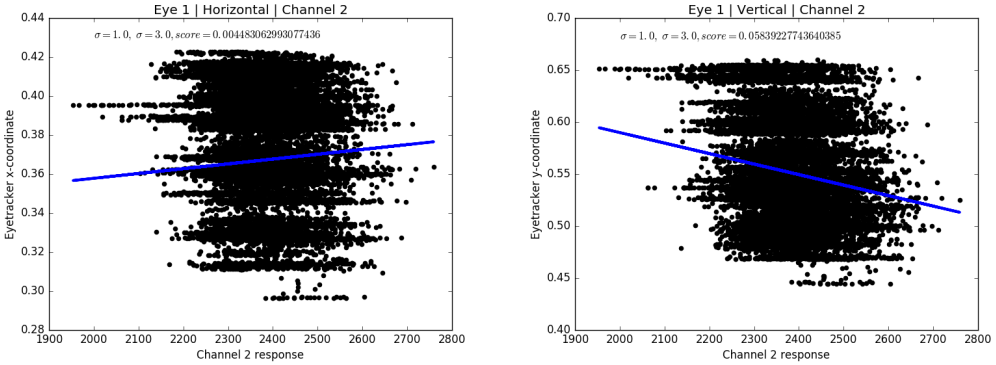
(d) Eye 1 vertical coordinate

Figure A.4: Regression lines for session 2 single channel (channel 1) condition



(a) Eye 0 horizontal coordinate

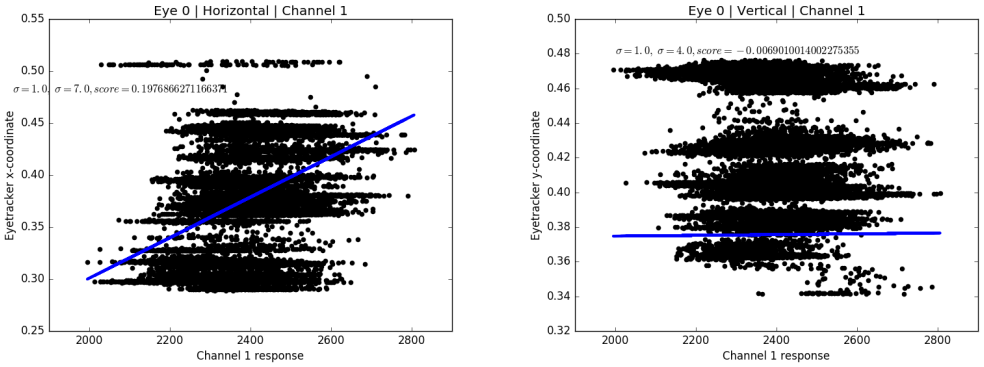
(b) Eye 0 vertical coordinate



(c) Eye 1 horizontal coordiante

(d) Eye 1 vertical coordinate

Figure A.5: Regression lines for session 2 single channel (channel 2) condition



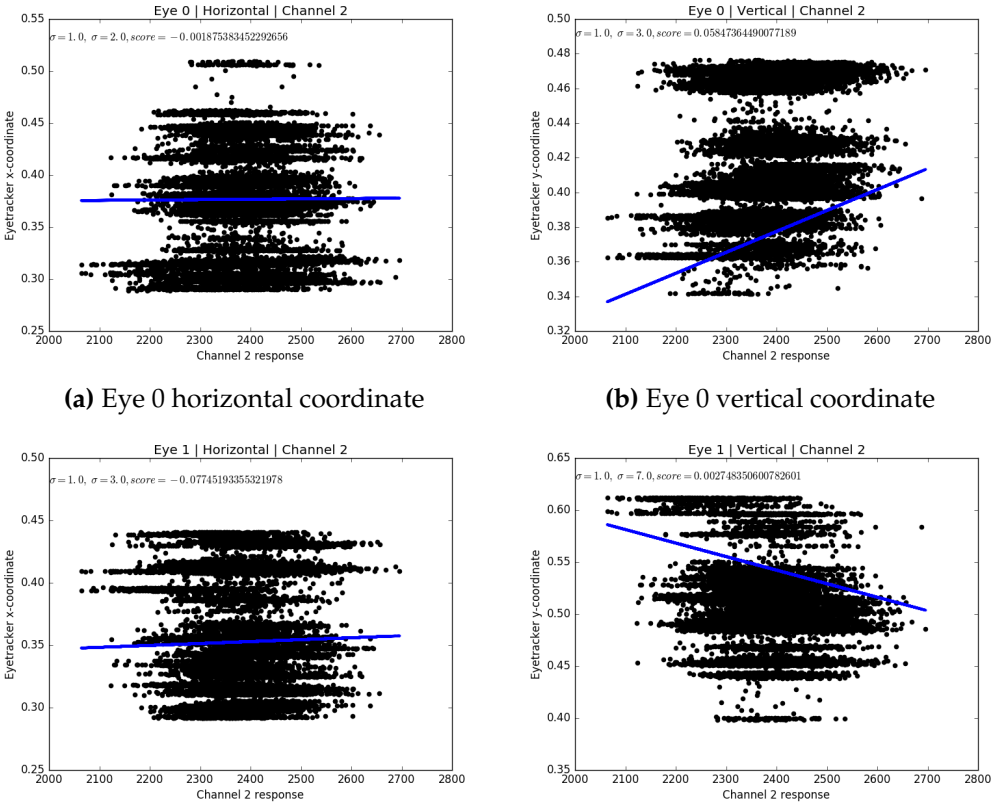
(a) Eye 0 horizontal coordinate

(b) Eye 0 vertical coordinate

(c) Eye 1 horizontal coordiante

(d) Eye 1 vertical coordinate

Figure A.6: Regression lines for session 3 single channel (channel 1) condition



(a) Eye 0 horizontal coordinate

(b) Eye 0 vertical coordinate

(c) Eye 1 horizontal coordiante

(d) Eye 1 vertical coordinate

Figure A.7: Regression lines for session 3 single channel (channel 2) condition

Bibliography

O.V. Acuña, P. Aqueveque, and E.J. Pino. Eye-tracking capabilities of low-cost eog system. *Conf Proc IEEE Eng Med Biol Soc.*, 610(3), 2014. doi: 10.1109/EMBC.2014.6943665.

J. L. Andreassi. *Human Behavior Physiological Response*. Lawrence Erlbaum Associates., 2000.

Krisstoffer Appel and Johannes Leugering. Traumschreiber, retrieved: Sept 20. 2016. URL `https://www.traumschreiber.uni-osnabrueck.de/`.

K.E. Bloch. Polysomnography: a systematic review. *Technol. Health Care*, 5(4):285–305, 1997.

M. Brown, M. Marmor, Vaegan, E. Zrenner, M. Brigell, and M. Bach. Iscev standard for clinical electro-oculography (eog). *Documenta Ophthalmologica*, 113(3):205–212, 2006. doi: 10.1007/s10633-006-9030-0.

Lars Buitnich, Gilles Louppe, Matthieu Blondel, Fabian Pedrogesa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake Vanderplas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. Machine learning software: experiences from the scikit-learn project. *European Conference on Machine Learning and Principles and Practices of Knowledge Discovery in Databases*, 2013.

Guy Thomas Buswell. *How People Look at Pictures: A Study of the Psychology of Perception in Art*. University of Chicago Press, 1935.

M.A. Carskadon and W.C. Dement. Monitoring and staging human sleep. In M.H. Kryger, T. Roth, and W.C. Dement, editors, *Principles and practice of sleep medicine*, chapter 2, pages 16–26. Elsevier Saunders, St. Louis, 2011.

Jean R. Davis and B. Shackel. Changes in the electro-oculogram potential level. *Brit. J. Ophtal.*, 44:606–620, 1960.

Andrew Duchowski. *Eye Tracking Methodology: Theory and Practice*. Springer Science Business Media, 2007.

Andrew T. Duchowski. A breadth-first survey of eye-tracking applications. *Behavior Research Methods, Instruments, Computers*, 34(4):445–470, 2002.

Christopher M. Federico. The mathematical derivation of least squares, retrieved: Aug. 7 2016. URL `https://isites.harvard.edu/fs/docs/icb.topic515975.files/OLSDerivation.pdf`.

K. M. Hearne. *Lucid dreams: an electrophysiological and psychological study*. PhD thesis, University of Liverpool, 1978.

W. Heide, E. Koenig, P. Trillenberg, D. Kömpf, and D.S Zee. Electrooculography: technical standards and applications. *Electroencephalogr Clin Neurophysiol*, 52:223–240, 1999.

Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost van de Weijer. *Eye Tracking: A comprehensive guide to methods and measures*. OUP Oxford, 2011.

Edmund Huey. *The Psychology and Pedagogy of Reading*. reprint: MIT Press, original: 1908, reprint: 1968.

E. Javal. Essai sur la physiologie de la lecture. *Annales d'ocullistique*, 80:61–73, 1878.

Moritz Kassner, William Patera, and Andreas Bullig. Pupil: An open source platform for pervasive eye tracking and mobile gaze-based interaction. 2014. URL `http://arxiv.org/pdf/1405.0006v1.pdf`.

A. Lypez, F.J. Ferrero, M. Valledor, J.C. Campo, and O. Postolache. A study on electrode placement in eog systems for medical applications. *2016 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, 2016. doi: 10.1109/MeMeA.2016.7533703.

Raymond H. Myers. *Classical and Modern Regression with Applications*. Duxbury Press, 2000.

Daniel C. Richardson and Michael J. Spivey. Eye-tracking: Characteristics and methods. In Gary E. Wnek and Gary L. Bowlin, editors, *Encyclopedia of Biomaterials and Biomedical Engineering, Second Edition*. Marcel Dekker, Inc.

Davis A. Robinso. A method of measuring eye movemnent using a scieral search coil in a magnetic field. *IEEE Transactions on Bio-medical Electronics*, 10(4):137–145, 1963.

J.W. Scott and P.E. Scott-Johnson. The electroolfactogram: a review of its history and uses. *Microsc Res Tech*, 58(3):152–160, 2002.

R. M. Stern and W. J. Ray and. *Psychophysiological Recording*. Oxford University Press, 2001.

N. Wade and B. W Tatler. *The moving tablet of the eye: The origins of modern eye movement research*. Oxford University Press, 2005.

L.R. Young and D. Sheena. Survey of eye movement recording methods. *Behav. Res. Methods Instrum.*, 7:397–429, 1975.

L.R. Young and D. Sheena. Eye-movement measurement techniques. *Encyclopedia of Medical Devices and Instrumentation*, pages 1259–69, 1988.

Z22. Diagram of light and four purkinje images, retrived: Sept. 25 2016. URL `https://en.wikipedia.org/wiki/Purkinje_images#/media/File:Diagram_of_four_Purkinje_images.svg`.

Decleration of authorship

I hereby declare that I have created this work completely on my own and used no other sources or tools than the ones listed, and that I have marked any citations accordingly.

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Berlin, September 2016
Frederik Laubisch