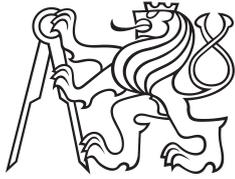


**Bachelor Project**



**Czech  
Technical  
University  
in Prague**

**F3**

**Faculty of Electrical Engineering  
Department of Cybernetics**

## **Automatic Estimation of hdEEG Electrode Positions**

**Radka Olyšarová**

**Supervisor: prof. Dr. Ing. Jan Kybic  
Field of study: Open informatics  
May 2021**



## I. Personal and study details

Student's name: **Olyšarová Radka** Personal ID number: **487014**  
Faculty / Institute: **Faculty of Electrical Engineering**  
Department / Institute: **Department of Cybernetics**  
Study program: **Open Informatics**  
Specialisation: **Artificial Intelligence and Computer Science**

## II. Bachelor's thesis details

Bachelor's thesis title in English:

**Automatic Estimation of hdEEG Electrode Positions**

Bachelor's thesis title in Czech:

**Automatická lokalizace hdEEG elektrod**

Guidelines:

Accurate position of EEG electrodes is important for the localization of cerebral activity, especially for high-density EEG, which uses a large number of electrodes. The goal of this work is to automatically localize the electrodes from images from 11 infrared cameras.

Perform the following subtasks:

- Manual annotation of electrodes in provided images.
- Finding the electrode positions and camera parameters based on manually identified correspondences.
- Creating an automatic method for electrode detection in images.
- Creating a robust method for finding correspondences between 2D images and between the 2D images and the 3D electrode model.
- Creating a complete automatic pipeline for finding camera parameters and electrode positions from the set of provided images.
- Experimental evaluation of the localization accuracy.
- [Optional] Experimental evaluation of the brain activity reconstruction improvement due to the electrode localization improvement.

Bibliography / sources:

- [1] H. Bauer, C. Lamm, S. Holzreiter, I. Holländer, U. Leodolter, and M. Leodolter, "Measurement of 3D electrode coordinates by means of a 3D photogrammetric head digitizer," NeuroImage, vol. 11, no. 5, p. S461, 2000.
- [2] Koessler et al: Spatial localization of EEG electrodes. Neurophysiology 2007.
- [3] G.S. Russel, K.J. Eriksen, P. Poolman, Luu. Phan, M. Tucker Don Geodesic photogrammetry for localizing sensor positions in dense-array EEG Clin Neurophysiol, 116 (2005), pp. 1130-1140.
- [4] Moreno-Noguer et al: Pose Priors for Simultaneously Solving Alignment and Correspondence. ECCV 2008.
- [5] Enqvist et al: Optimal correspondences from pairwise constraints. ICCV 2009.

Name and workplace of bachelor's thesis supervisor:

**prof. Dr. Ing. Jan Kybic, Biomedical imaging algorithms, FEE**

Name and workplace of second bachelor's thesis supervisor or consultant:

Date of bachelor's thesis assignment: **25.01.2021** Deadline for bachelor thesis submission: **21.05.2021**

Assignment valid until: **30.09.2022**

\_\_\_\_\_  
prof. Dr. Ing. Jan Kybic  
Supervisor's signature

\_\_\_\_\_  
prof. Ing. Tomáš Svoboda, Ph.D.  
Head of department's signature

\_\_\_\_\_  
prof. Mgr. Petr Páta, Ph.D.  
Dean's signature

### III. Assignment receipt

The student acknowledges that the bachelor's thesis is an individual work. The student must produce her thesis without the assistance of others, with the exception of provided consultations. Within the bachelor's thesis, the author must state the names of consultants and include a list of references.

\_\_\_\_\_  
Date of assignment receipt

\_\_\_\_\_  
Student's signature

## Acknowledgements

Firstly, I would like to express my gratitude to my supervisor, prof. Dr. Ing. Jan Kybic, who kindly and patiently guided me throughout this thesis.

Furthermore, I would like to thank Ing. Vlastimil Koudelka, Ph.D., for his expertise and valuable insights.

Finally, I would like to show my deepest gratitude to my family and boyfriend, for their continuous love and support.

## Declaration

I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instructions for observing the ethical principles in the preparation of university theses.

Prague, 21. May 2021

## Abstract

Electroencephalography (EEG) is one of the basic diagnostic methods in neurology, psychiatry and many more related fields. High density EEG captures the electrical activity of the brain using a large number of channels which makes it possible to determine a more precise location of the brain activity sources. But for correct results the exact location of each electrode is required.

This bachelor's thesis tries to develop an algorithm for automatic estimation of EEG electrode positions. Input data consist of eleven infrared camera images situated in a special construction. The partial 3D model is computed using a combination of different methods from computer vision and combinatorial optimization. Remaining electrode positions are subsequently computed from a statistical model.

The final algorithm is able to estimate the electrode positions with 3 mm error on average compared to the training dataset. The runtime of the algorithm is around three minutes.

**Keywords:** hdEEG, spatial localization of EEG electrodes, computer vision, neural network, epipolar geometry

**Supervisor:** prof. Dr. Ing. Jan Kybic  
Department of Cybernetics

## Abstrakt

Elektroencefalografie (EEG) patří mezi základní diagnostické metody v neurologii, psychiatrii a mnoha souvisejících oborech. High density EEG využívá ke snímání elektrické aktivity mozku vysoké množství kanálů, čímž lze zdroje mozkové aktivity přesněji lokalizovat. Pro správnou lokalizaci je ale potřeba znát přesné umístění jednotlivých elektrod.

Tato bakalářská práce se zabývá vývojem algoritmu pro automatický výpočet pozic EEG elektrod. Jako vstupní data slouží obrázky z jedenácti infrakamer umístěných ve speciální konstrukci. Navržený algoritmus využívá kombinaci různých metod počítačového vidění a kombinatorické optimalizace k nalezení částečného 3D modelu. Pozice zbylých elektrod jsou následně dopočítány ze statistického modelu.

Výsledný algoritmus je schopen nalézt pozice elektrod s průměrnou chybou 3 mm oproti trénovacímu datasetu. Průměrná doba běhu algoritmu je kolem tří minut.

**Klíčová slova:** hdEEG, lokalizace EEG elektrod, počítačové vidění, neuronové sítě, epipolární geometrie

**Překlad názvu:** Automatická lokalizace hdEEG elektrod

# Contents

<b>1 Introduction</b>	<b>1</b>		
Motivation and Goals	1		
Thesis Structure	2		
<b>2 Technological background</b>	<b>3</b>		
2.1 Electroencephalography	3		
2.1.1 Signals in human brain	3		
2.1.2 EEG signals	4		
2.1.3 Electrode placement systems	5		
2.1.4 Electrical source imaging	6		
2.2 Spatial localization of EEG electrodes	8		
2.2.1 State of the art	8		
2.3 3D reconstruction	10		
2.3.1 State of the art	10		
<b>3 Problem analysis</b>	<b>13</b>		
3.1 Solution outline	13		
3.2 Dataset analysis	14		
3.2.1 2D coordinates	14		
3.2.2 3D coordinates	14		
3.2.3 Electrode types	15		
<b>4 Electrode detection</b>	<b>17</b>		
4.1 Initial detection using Faster R-CNN	17		
4.1.1 Faster R-CNN architecture	17		
4.1.2 Dataset augmentation	18		
4.1.3 Neural network model optimization	19		
4.2 Improving detections using template matching	21		
4.2.1 Template matching	22		
4.2.2 Template creation	22		
4.2.3 Template matching optimization	24		
<b>5 Corresponding points estimation</b>	<b>27</b>		
5.1 Camera calibration	27		
5.1.1 Camera projection matrix	27		
5.1.2 Direct linear transformation	28		
5.1.3 Normalization	28		
5.1.4 Evaluation	28		
5.2 Corresponding points estimation	29		
5.2.1 Camera constraints	30		
5.2.2 Correspondences between two cameras	30		
5.2.3 Linear triangulation	32		
5.2.4 Consistent re-projected points	32		
5.2.5 Optimal subset	33		
5.2.6 Evaluation	36		
<b>6 3D model reconstruction</b>	<b>39</b>		
6.1 Statistical model	39		
6.1.1 Aligning the training dataset	39		
6.1.2 Deriving the model	40		
6.1.3 Final statistical model	41		
6.2 Final 3D model	42		
6.2.1 Overview of proposed algorithm	42		
6.2.2 Assignment matrix	43		
6.2.3 Optimal shape computation	44		
<b>7 End-to-end evaluation</b>	<b>45</b>		
7.1 Accuracy evaluation	45		
7.2 Time complexity	46		
7.3 Electrical source imaging	46		
7.3.1 Localization techniques	47		

7.3.2 Results .....	47
<b>8 Conclusion</b>	<b>49</b>
<b>Bibliography</b>	<b>51</b>
<b>A Technical details</b>	<b>57</b>

## Figures

2.1 Graph describing the formation of an action potential [4] . . . . .	3	3.2 Illustration of different types of electrodes (the type on the left is used in all computations) . . . . .	15
2.2 Comparison of normal EEG signal (A) to EEG signal including epileptic seizure (B) [6] . . . . .	4	4.1 Illustration of regions proposal process using the Region Proposal Network (RPN) [31] . . . . .	18
2.3 Traditional 10-20 electrode placement system [7] . . . . .	5	4.2 Example of an image from provided dataset before and after augmentation . . . . .	19
2.4 Distribution of the electrodes in Geodesic Sensor Net with 256 channels. Displayed are cardinal points (red outline), other electrodes (blue outline) and additional structural support points (without outline) [11]. . . . .	6	4.3 Computation of Intersection Over Union (IOU) [32] . . . . .	20
2.5 Illustration of Glassbrain visualization of Auditory Evoked Potentials measured with latency of 89-108 ms (provided by the Czech National Institute of Mental Health) . . . . .	7	4.4 Example of the Faster R-CNN output with confidence score of each predicted bounding box . . . . .	21
2.6 Illustration of stylus-based digitizers for sensor localization (ultrasound digitizer on the left, electromagnetic digitizer on the right) [14] . . . . .	8	4.5 Comparison of a original image and a equalized image using histogram equalization . . . . .	23
2.7 Illustration of camera dome and gantry used in geodesic photogrammetry system (right) [10] and example of image taken from the top camera (left) . . . . .	9	4.6 Final template (leftmost) and some of the images found in its cluster . . . . .	23
2.8 Basic diagram of 3D reconstruction based on multi-view [16] . . . . .	10	4.7 Example of the Template Matching output with match score of each bounding box . . . . .	25
2.9 Illustration of sparse (left) and dense (right) 3D reconstruction [27] . . . . .	11	5.1 Illustration of the worst (left) and best (right) case of the 3D coordinates re-projection . . . . .	29
3.1 Illustration of reprojected sensors (green and black circles) from provided dataset where the inaccuracies (red rectangles) and missing sensors can be seen . . . . .	14	5.2 Visualization of the main ideas of the epipolar geometry [22] . . . . .	31
		5.3 Example of epipolar lines on the images from training dataset . . . . .	31
		5.4 Example of a re-projected point at the bottom image, computed from 3D point estimated from corresponding points from the images at the top . . . . .	33

5.5 Example of optimization formulation problem. Consider four possible correspondences - three point in red and two point in blue. The formulation will prefer the two point correspondences as it maximizes the number of used points. ....	34
5.6 Comparison of pair correspondences before (top) and after (bottom) Leordeanu-Herbert algorithm. Note that the algorithm does not remove the top correspondence as the change of distance between images is similar to the correct correspondences.....	35
5.7 Example of the final correspondences found between two images .....	38
6.1 Comparison of referential model $\mathbf{x}^1$ and another model before and after aligning .....	40
6.2 Comparison of statistical models using different weight vectors $\mathbf{b}$ ...	41
6.3 Example of assignment re-computation from the assignment matrix. Only points 1 and B will be assigned as they share the maximal value. Point 2 cannot be matched with point A as the maximal value of point A is in the slack row. The slack row and column are not normalized.	43
6.4 Comparison of fitted and predicted model throughout the fitting process	44

## Tables

4.1 Evaluation of models trained on different datasets, each with its own optimized NMS and confidence score threshold .....	21
4.2 Evaluation of detections before and after using template matching algorithm (TM). Values are averaged across all images in training dataset.	25
5.1 Evaluation of camera matrices estimation using the leave-one-out cross validation .....	29
5.2 The average number of correspondences throughout the correspondence finding algorithm (LH refers to Leordeanu-Herbert algorithm) .....	36
5.3 Evaluation of found correspondences using different optimal subset finding methods ...	38
7.1 Evaluation of estimated coordinates using the leave-one-out cross validation technique .....	45
7.2 The overview of the time complexity of each part of the proposed algorithm .....	46
7.3 Dice coefficient values computed for different localization techniques	47

# Chapter 1

## Introduction

Electroencephalography is the most common method for brain signal recording and analysis. However, due to the small number of electrodes used in standard EEGs, it is not able to locate the precise source of the signal. High density EEG attempts to overcome this problem by covering the scalp with up to 256 electrodes. However, besides the high number of electrodes, it is also necessary to know the exact 3D position of hdEEG electrodes to achieve a high precision of source localization.

The task of this thesis is to develop a method for automatic estimation of hdEEG electrode positions from infrared camera images. These images are taken simultaneously using cameras placed around the patient in a polyhedron-based construction (see Figure 2.7). The focus of this thesis is the subsequent reconstruction of 3D model of hdEEG electrodes.

## Motivation and Goals

Human brain consists of more than 10 billion neural cells and more than 10 quadrillion connections between them [1]. Each neuron generates electrical impulses called action potentials, which transmit information between neurons. Over the years many methods for studying these communications have been developed and electroencephalography is one of the oldest and most widespread ones.

The determination of changes in the brain activity is useful in diagnosing various brain disorders such as epilepsy, sleep disorders and many others. EEG signals determining these changes are measured using the electrodes attached to the patient's scalp. In most routine clinical EEGs, electrodes are positioned on the head individually but in the case of hdEEG, where large number of sensors is applied, special caps with built-in electrodes are used.

For precise electrical source localization, the exact position of each electrode in three-dimensional space has to be known. There are several methods for

electrode positions estimation (see Section 2.2), but many of them are time consuming for either the operator or both operator and subject.

Therefore, there has been a need for a fast, automated electrode localization system, which is the focus of this thesis. Thanks to the recent improvements in the field of computer vision and machine learning, this task can nowadays be accomplished just by using cameras mounted in a special construction designed for this purpose.

## ■ Thesis Structure

- In Chapter 2, the technological background is introduced, describing the EEG and hdEEG in greater depth as well as the current methods for electrode positions estimation and 3D reconstruction are introduced.
- In Chapter 3, quick analysis of the problem and provided dataset is conducted.
- In the three following chapters the proposed solution is presented and described in detail.
  - Electrode detection in Chapter 4
  - Corresponding point estimation in Chapter 5
  - Final 3D model reconstruction in Chapter 6
- In Chapter 7, the presented method is evaluated, and the results are discussed.
- In Chapter 8, the proposed solution is summarised, and future improvements are suggested.

## Chapter 2

### Technological background

#### 2.1 Electroencephalography

The first signs of electric activity in human brain appear around the 17-23 week of prenatal development [2]. The activity is a product of communication between neurons in the form of electrical signals which can be measured non-invasively using the electroencephalography.

##### 2.1.1 Signals in human brain

Neurons communicate using both the electrical and chemical signals. Signal transmission between neurons is performed using a chemical called a neurotransmitter. When this chemical signal is strong enough it triggers consequent electrical signal transmission within the neuron. This transmission is carried by a brief reversal of the resting membrane potential called an action potential [3].

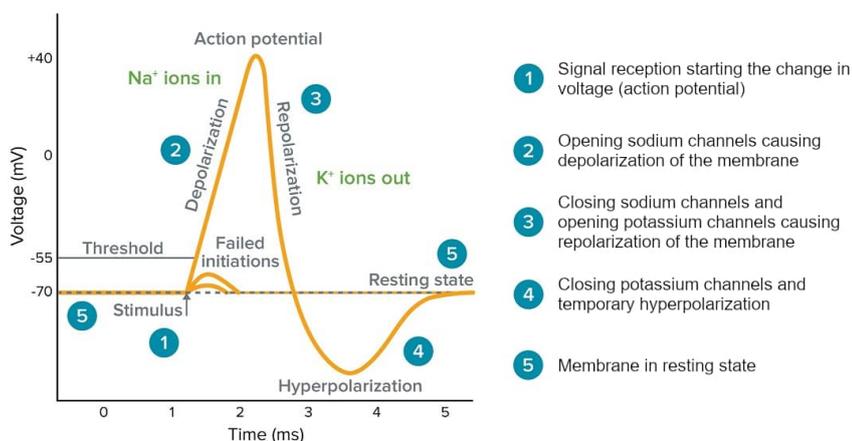


Figure 2.1: Graph describing the formation of an action potential [4]

Resting membrane potential is negatively charged to  $-70$  mV. Action potential occurs once the  $-55$  mV threshold is crossed due to reception of the chemical signal. Crossing the threshold causes  $\text{Na}^+$  channels to open which results in depolarization of the membrane and the potential to rise to about  $40$  mV. Once the depolarization is complete the  $\text{Na}^+$  channels are closed and at the same time the  $\text{K}^+$  channels open. With potassium ions leaving the cell the membrane potential once again becomes negatively charged [3].

### 2.1.2 EEG signals

When brain cells are activated due to the action potential, local current flows are produced. EEG measures the electrical activity using electrodes placed on the head surface. These are round shaped and around a centimetre in diameter. There are several different layers between electrode and neurons such as skin or skull, therefore only signal from large population of active neurons can be recorded. Electrical signals detected by the electrodes are weak and have to be massively amplified in order to be discernible [2].

There are several parameters that can be used to classify the EEG waveforms such as frequency, amplitude and shape of waveform as well as the position of the electrodes on the scalp [5]. Looking at the patterns of EEG signals a trained professional can recognize different states of alertness, behaviour and mental states of the brain.



**Figure 2.2:** Comparison of normal EEG signal (A) to EEG signal including epileptic seizure (B) [6]

### 2.1.3 Electrode placement systems

Routine EEG examination is performed non-invasively using electrodes placed on the patient's scalp. As the positions as well as the number of electrodes affect the accuracy of measurements, there are many different placement systems. Below we will describe two of them - the universally used 10-20 system and the hdEEG which is the focus of this thesis.

#### The International 10-20 System

The first standardized electrode positioning system known as the International 10-20 System with 21 electrodes in total was published by H. Jasper in 1958 and is still universally used [7].

The name of the 10-20 System comes from the placement of electrodes on the scalp (see Figure 2.3). Positions of the electrodes are calculated from the distance between nasion and inion and both earlobes, respectively. The placement is calculated by dividing the measured distance into sections of 10% and 20% [7]. Electrodes are labelled according to the brain area in which are placed: F (frontal), C (central), T (temporal), P (posterior), and O (occipital). The letters on electrodes at the left side of the brain are accompanied by odd numbers, even numbers are on the right [2].

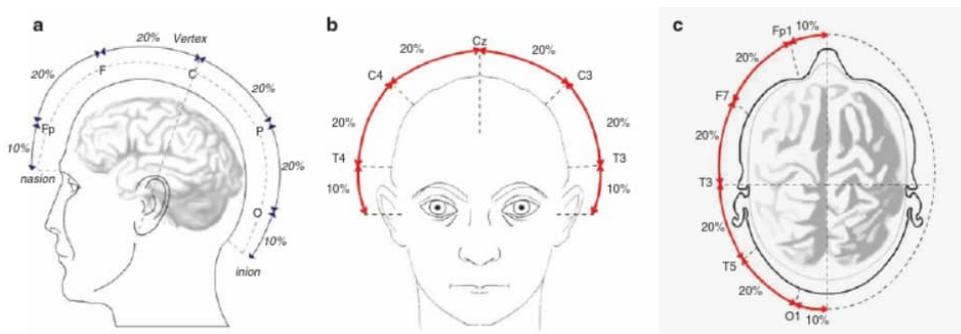


Figure 2.3: Traditional 10-20 electrode placement system [7]

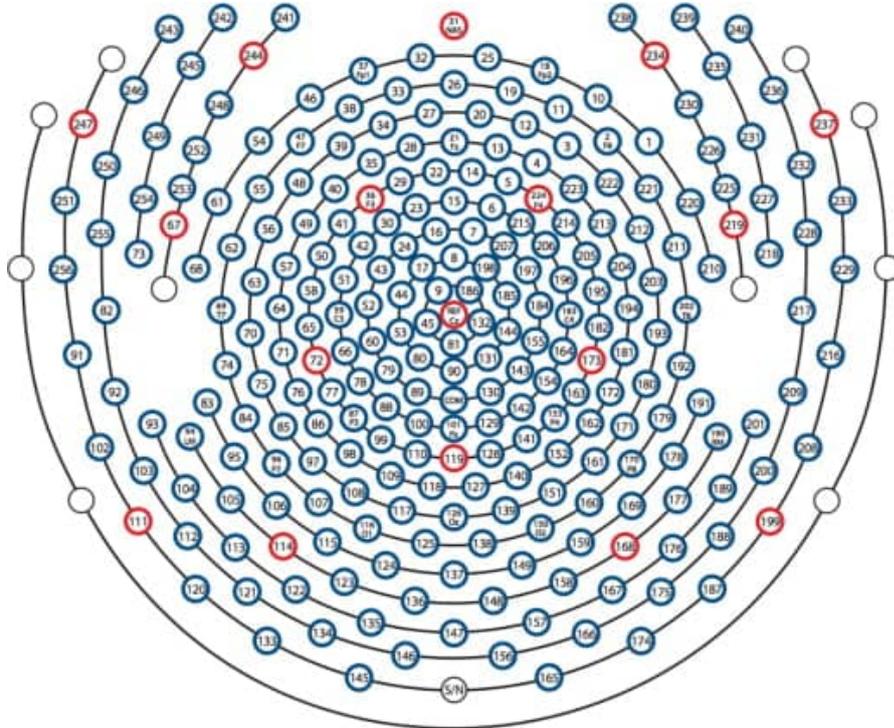
#### High density EEG

Even though the International 10-20 System is regularly used in routine EEG examination, it is not exempt from criticism. Not taking the asymmetry of the human head into account is one of the most commonly mentioned points [7]. Moreover, the 10-20 System does not cover the inferior part of the temporal lobe and generally has a small number of electrodes which may be problem when precise source localization is required [8].

The number of electrodes is an important factor in electrical source imaging and subsequent source localization, which is indispensable particularly in

diagnosing and treating epilepsy [9]. Theoretically, to accurately reconstruct the distribution of the electric potentials, one needs a spatial sampling with interelectrode distance of 2-3 cm. High density EEGs enable us to use electrode caps with up to 256 electrodes reducing the interelectrode distance to less than 2 cm [8].

Currently, one of the most commonly used electrode caps in high density EEGs is the Geodesic Sensor Net. The topography of this net is defined in terms of an icosahedron. Vertices of the icosahedron are known as cardinal points [10] and are designated as black electrodes in contrast to other white electrodes.



**Figure 2.4:** Distribution of the electrodes in Geodesic Sensor Net with 256 channels. Displayed are cardinal points (red outline), other electrodes (blue outline) and additional structural support points (without outline) [11].

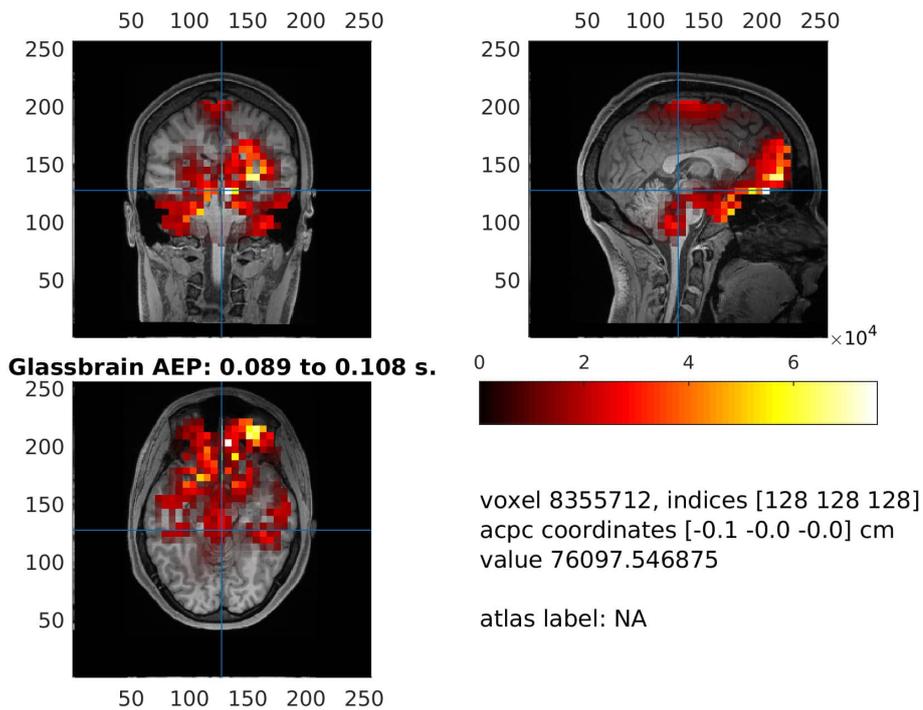
### 2.1.4 Electrical source imaging

The technique used for electrical source localization is known as the Electrical Source Imaging (ESI). This process includes many steps, but the main idea is splitting the problem into two parts - the forward and inverse problem [12].

The aim of the forward problem is to determine how much a given electrical source in the brain will impact each electrode on the scalp. For this the EEG electrode positions need to be known and fitted on an MRI head model. This can either be a real MRI scan or an MRI brain template.

The solution of the forward problem is then used to locate the electrical signal sources in the brain, which are responsible for the measured potentials at the EEG electrodes. This is what is known as the inverse problem.

The precision of electrode localization is vital for the electrical source imaging. However, the effect of electrode displacement is not well researched and therefore, there is no general guideline for the minimum recommended electrode localization precision. From various research articles, it seems the precision of localization is in millimetre order of magnitude and the best results were found to be for electrode localization error up to 5 millimetres [13].



**Figure 2.5:** Illustration of Glassbrain visualization of Auditory Evoked Potentials measured with latency of 89-108 ms (provided by the Czech National Institute of Mental Health)

## 2.2 Spatial localization of EEG electrodes

### 2.2.1 State of the art

There are several methods for measuring the 3D position of each electrode, based on manual measurement, ultrasound, MRI and others. This section aims to quickly summarize and compare some of these methods.

#### Manual methods

Manual methods measure the distance between electrodes or electrode and fixed landmarks (nasion, left and right pre-auricular points) using callipers. These measurements then enable calculation of 3D position of electrodes [14].

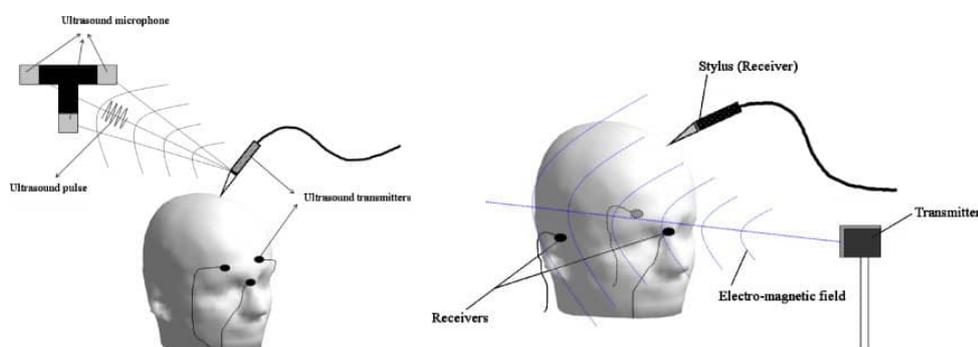
The advantage of these methods is low cost as they do not require any specific equipment with the exception of the callipers. However, these methods are time consuming for both the operator and the subject. Moreover, the manual methods are susceptible to human errors.

#### Stylus-based digitization

Systems implementing stylus-based digitization such as ultrasound or electromagnetic digitization, use a system based on utilizing receiver and transmitter.

For the ultrasound digitizer, the transmitter is in the form of a stylus and the impulse generated is received by a set of microphones. On the other hand, the Electromagnetic digitizer generates an electromagnetic field, which is then measured with the stylus, which acts as the receiver [14].

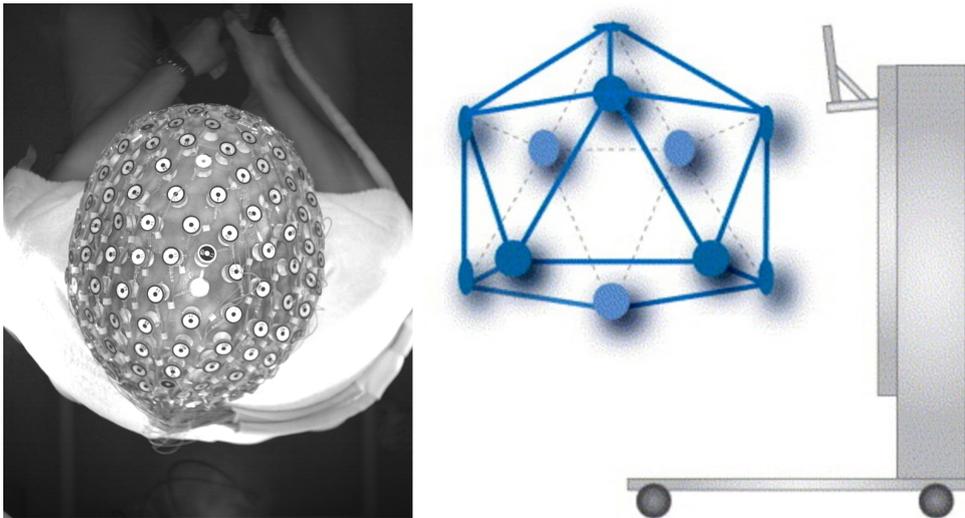
These methods tend to be sensitive to the environment and also require digitization of each electrode individually, which is again time consuming for both the operator and the subject.



**Figure 2.6:** Illustration of stylus-based digitizers for sensor localization (ultrasound digitizer on the left, electromagnetic digitizer on the right) [14]

### ■ The geodesic photogrammetry system

Currently there is one commercially available product, which uses similar approach to electrode position estimation proposed in this thesis. The subject wearing a special electrode cap called Geodesic Sensor Net is positioned in the centre of a polyhedron-based structure with 11 cameras, one in each vertex (see Figure 2.7). All of the cameras take a single picture simultaneously. The software then guides the operator to mark specific sensors and from them it estimates 3D position of all electrodes [10], [14].



**Figure 2.7:** Illustration of camera dome and gantry used in geodesic photogrammetry system (right) [10] and example of image taken from the top camera (left)

The idea behind the 3D coordinate estimation is as follows. First all possible pairings of points are computed across the cameras. From the marked sensors, the head surface shape is estimated and used for elimination of infeasible pairings [10].

As the patient is free to go immediately after the pictures are taken, this method is very fast and comfortable for the subject. The operator on the contrary has to manually detect some of the sensors, which is time consuming and leaves room for human errors.

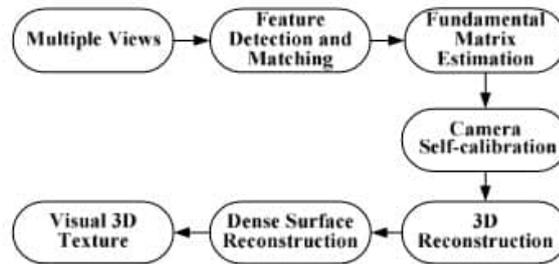
## 2.3 3D reconstruction

In computer vision the 3D reconstruction is a process of computing the three-dimensional model from a set of images. This technique has many applications outside the medical field such as art analysis and restoration, computer graphics or even criminal investigations.

### 2.3.1 State of the art

The currently available 3D reconstruction methods can be split into two categories. First are the active methods, which utilize a combination of image and active measuring devices such as lasers, microwaves or ultrasounds. Methods in the other category are called passive, as they reconstruct the 3D surface from the set of images only [15].

The problem of the 3D reconstruction in this thesis falls under the category of passive methods, as only a set of images is provided. Generally speaking, the 3D reconstruction process based on multi-view has its own established pipeline.



**Figure 2.8:** Basic diagram of 3D reconstruction based on multi-view [16]

Given a set of images, the first step is feature detection and matching. This problem can be solved using various algorithms. The first algorithms were based on edge and corner feature detection [17], [18]. However, these algorithms do not take affine deformation and scale into account. This led to the development of improved, scale and rotation invariant methods [19], [20]. In some of the feature matching algorithms methods for match refinement are also used [21].

The next step in the traditional pipeline is the fundamental matrix estimation and camera calibration. Naturally, there are many algorithms for these computations [16], however, both matrices have their standard computation approaches. For the fundamental matrix estimation, the seven-point or eight-point method is used. The camera calibration or camera matrix can be computed independently using the Direct Linear Transformation approach [22].

Using the results of the algorithms above, the 3D coordinates can be computed using triangulation. Afterwards, to refine the visual reconstruction bundle adjustment methods can be used [23]. These methods aim to improve the 3D structure and viewing parameters simultaneously.

The reconstruction as described so far generates only sparse 3D points and thus only roughly identifies the object shape. If the accurate visualization is needed, it is necessary to use the dense surface reconstruction [24]–[26].



**Figure 2.9:** Illustration of sparse (left) and dense (right) 3D reconstruction [27]



## Chapter 3

### Problem analysis

Thanks to the large number of electrodes the main purpose of high density EEG monitoring is electrical source localization. To achieve high precision of source localization, the exact 3D position of EEG electrodes is required. This bachelor thesis aims to develop a method for automatic estimation of these positions from 11 infrared camera images. It is assumed that the images are taken from the same camera dome as in the geodesic photogrammetry system (see Figure 2.7).

#### 3.1 Solution outline

Cameras being placed in the regular polyhedron-based structure ensures visibility of each sensor in at least two camera images. Due to this fact the solution can be designed in such a manner that no user interaction is required. The presented solution could be summarized in these steps:

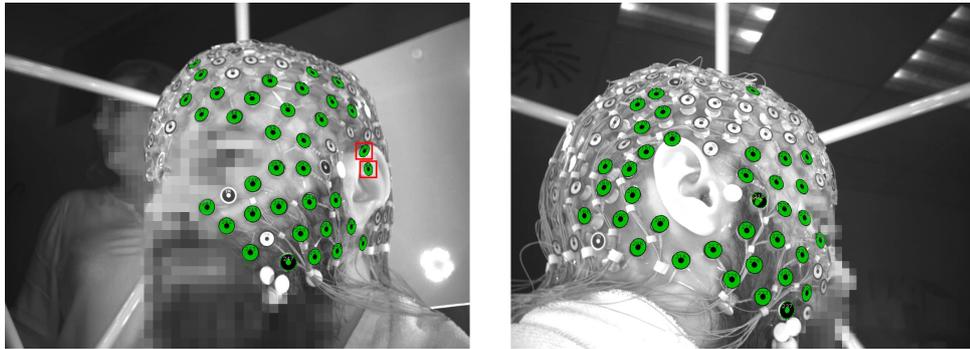
- Detecting the electrodes in all images (Section 4.1)
- Improving electrode detections using template matching (Section 4.2)
- Estimating corresponding points in camera images (Chapter 5)
- Computation of 3D coordinates from corresponding points (Section 5.2.3)
- Fitting statistical model to the computed model (Chapter 6)
- Computing the remaining electrode positions (Chapter 6)

## 3.2 Dataset analysis

For the purpose of this thesis a dataset of 18 subjects was provided by the Czech National Institute of Mental Health. Data of each subject contain 11 infrared camera images, 2D coordinates of detected sensors in each image, computed 3D model and other information not useful for the developed method. However, some issues with the provided dataset were discovered early in the method development and will be discussed below.

### 3.2.1 2D coordinates

The main issues of provided dataset were inaccuracies and missing 2D coordinates of detected sensors (see Figure 3.1). This could cause problems in parts of the proposed algorithm especially in the neural network training. Hence the annotations of the images had to be repaired.



**Figure 3.1:** Illustration of reprojected sensors (green and black circles) from provided dataset where the inaccuracies (red rectangles) and missing sensors can be seen

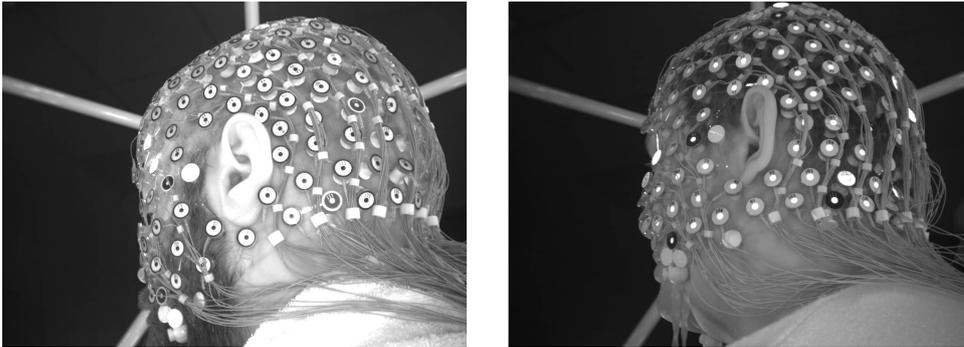
### 3.2.2 3D coordinates

Although, this may not be considered as an issue per se, the provided dataset consists of two different 3D coordinates. Neither of them has any description available. From manual inspection it became apparent, that the coordinates in the `coordinates.gpsc` file were very similar to the ACPC coordinate system [28], although measured in centimetres.

Furthermore, as inaccuracies in the 2D coordinates were discovered, it would seem logical to assume, that the 3D coordinates were flawed too.

### 3.2.3 Electrode types

The last issue was different type of electrodes throughout the images (see Figure 3.2). As there were only 2 subjects with a different type of electrodes, it was decided to omit these from neural network training and computation of templates for template matching. It did not affect any of the other parts of the presented method as they do not use input images.



**Figure 3.2:** Illustration of different types of electrodes (the type on the left is used in all computations)



## Chapter 4

### Electrode detection

Object detection is among the most researched topics in the field of computer vision. As a result, there are various approaches to solving this problem. Nowadays the main focus is on neural network utilization which is also the case in the proposed solution as the initial detection of hdEEG electrodes is performed using Faster R-CNN architecture. This process is described in Section 4.1. Afterwards, the detections are improved using template matching technique presented in Section Section 4.2.

#### 4.1 Initial detection using Faster R-CNN

There are many neural network architectures with their own specific use cases. However, for our purpose no specialized architecture is required. The detection problem is fairly straightforward as the electrodes have a simple circular shape and only two different colours are to be detected. Moreover, the size of the electrodes remains the same and also there are no strict requirements regarding speed of the detection.

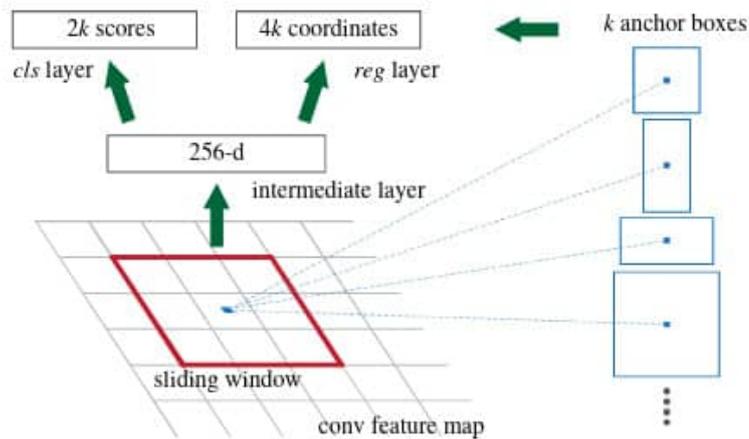
Because of the reasons stated above, we decided to use and slightly modify a ready to use implementation of Faster R-CNN architecture [29] as it is one of the most widely used object detection methods. The neural network was trained on various datasets, which are described in Section 4.1.2. Afterwards, the detections from trained model were optimized using different thresholds of confidence score and NMS score described in greater depth in Section 4.1.3.

##### 4.1.1 Faster R-CNN architecture

Faster R-CNN (Region-Based Convolutional Neural Networks) architecture is a two-stage detector as the detection process is divided into two stages - the region proposal and the classification [30].

In the first stage, several object candidates, known as regions of interest, are

proposed using the Region Proposal Network. To generate region proposals a sliding window approach is utilized on the feature map extracted by the backbone network. At each sliding window location, multiple regions called anchor boxes are proposed and subsequently passed through a two sibling convolutional layers. One layer computes bounding box regression, the other returns the objectness score, which determines whether proposed region contains an object or is part of the background. Afterwards, the proposed regions are classified using the Fast R-CNN architecture [30], [31].



**Figure 4.1:** Illustration of regions proposal process using the Region Proposal Network (RPN) [31]

#### 4.1.2 Dataset augmentation

Initially, as mentioned in Section 3.2, a dataset with 18 subjects each with 11 images was provided. For the purpose of neural network training it needed to be split into two parts. These parts are commonly referred to as training and validation datasets. Images of three of the subjects were randomly selected to be part of the validation dataset. For the training dataset various augmentations of the remaining images were tried with the aim of improving the precision of neural network<sup>1</sup>. The augmentations used in evaluation are described below.

#### Negative images

There are two classes to be detected, the white sensors and the cardinal black ones (see Figure 2.4). As the number of cardinal sensors is drastically lower than the number of white sensors, the first problem encountered was lower precision of black sensors detection. As a solution to this problem, negatives of all images were added to the dataset.

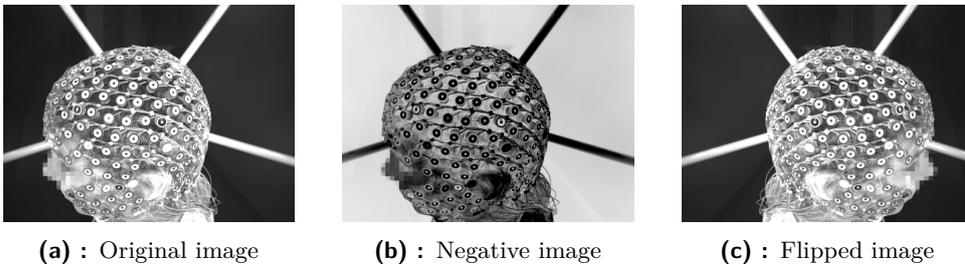
<sup>1</sup>All the augmentations were performed on the repaired dataset.

### ■ Flipped and negative images

Another proposed augmentation for improving the neural network results was to create larger dataset. All images were flipped horizontally and added to training dataset as well as the negative images from above.

### ■ Same electrode type only

As mentioned in Section 3.2, the provided dataset contained subjects with different types of electrodes. As this type of electrodes was present only in 2 subjects, the decision was made to exclude them to prevent interference in training process. The flipped and negative images were still included. However, the negative images were manually chosen as only images with correct looking electrodes were used. This is the final dataset used in neural network training.



**Figure 4.2:** Example of an image from provided dataset before and after augmentation

### ■ 4.1.3 Neural network model optimization

In computer vision each object detection can be evaluated based on its classification and correctness<sup>2</sup> of its bounding box position as follows

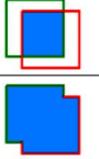
- True positive (TP): Correct classification and bounding box position
- False positive (FP): Incorrect classification or bounding box position
- False negative (FN): Object was not detected

Once the amount of true positive, false positive and false negative detections is calculated, additional metrics described below can be derived.

<sup>2</sup>The correctness of the bounding box position is evaluated using the IoU value.

### ■ Intersection over union

The correctness of the bounding box position is measured using the Intersection over Union (IoU) value computed from detected and ground truth bounding boxes. By comparing the IoU with a given threshold, each detection can be classified as correct or incorrect [32].

$$IOU = \frac{\text{area of overlap}}{\text{area of union}} = \frac{\text{area of overlap}}{\text{area of union}}$$


**Figure 4.3:** Computation of Intersection Over Union (IOU) [32]

### ■ Precision, Recall and Average Precision

The metric used to evaluate accuracy of a model during the training process is Mean Average Precision, which is defined as the mean of average precisions through all of the classes. Average precision estimates the area under the precision-recall curve. This curve can be seen as a trade-off between precision and recall [32]. The recall and precision values are defined as follows

$$\text{Recall} = \frac{TP}{TP + FN} = \frac{TP}{\text{all ground truths}} \quad (4.1)$$

$$\text{Precision} = \frac{TP}{TP + FP} = \frac{TP}{\text{all detections}} \quad (4.2)$$

### ■ Non-maximal suppression and confidence score

Performance of the trained model can be enhanced by optimizing the Non-maximal suppression (NMS) and confidence score thresholds. The confidence score is a part of the neural network detection output which specifies the probability of correct detection. Non-maximal suppression is a method used to filter overlapping bounding boxes. This algorithm selects the most confident detection and suppress all overlapping detections with IoU score larger than defined threshold having the same class. NMS threshold denotes the maximal IoU value between any two detected bounding boxes [33].

### ■ Results

Trained model can be optimized by choosing different NMS and confidence score thresholds. For the proposed solution it is beneficial to sacrifice some precision to achieve better recall.

We decided to fix the precision value on both classes to a minimum of 80%, while maximizing the recall using the confidence score and NMS thresholds. The results of this optimization are described in Table 4.1.

Dataset	Precision (%)		Recall (%)		Thresholds	
	White	Black	White	Black	NMS	Score
Original data	95.537	80.303	46.403	36.806	0.206	0.574
Repaired data	88.687	80.147	80.251	75.694	0.148	0.291
Negative images	86.424	81.452	78.996	70.139	0.333	0.519
Flipped and negative images	90.923	0.8	85.611	77.778	0.192	0.303
Same electrode type only	90.853	80.132	85.37	84.028	0.122	0.207

**Table 4.1:** Evaluation of models trained on different datasets, each with its own optimized NMS and confidence score threshold



**Figure 4.4:** Example of the Faster R-CNN output with confidence score of each predicted bounding box

## 4.2 Improving detections using template matching

As can be seen in Figure 4.4, the centres of the bounding boxes detected using the neural network do not always correspond to the true centre of the electrodes. As the following parts of the suggested solution require the most

possible accuracy, the effort to improve all of the electrode detections using template matching was made.

### ■ 4.2.1 Template matching

Template matching is a technique used to locate the part of an image that is most similar to the predetermined template. One of the most common way to find a template in an image is cross correlation. Its basic concept is measuring the correlation between the image and a template at each point of the image area. The image point with the maximal correlation then denotes the position where template matches the image the best [34].

Computation of correlation between template  $T(x, y)$  and image  $I(x, y)$  at point  $(i, j)$  is displayed in equation 4.3 below. Note that the image  $I$  has to be larger than the template  $T$ .

$$C(i, j) = \sum_{x,y} T(x, y)I(x + i, y + j) \quad (4.3)$$

As basic correlation is very sensitive to changes in intensity, the normalized correlation coefficient is often used instead. The computation is displayed in equation 4.4 [34], [35] and uses the average intensity value  $\bar{T}$  of the pixels in the template as well as the average value  $\bar{I}$  of the region of image  $I$  overlapping with the template.

$$R(i, j) = \frac{\sum_{x,y} T'(x, y) \cdot I'(x + i, y + j)}{\sqrt{\sum_{x,y} T'(x, y)^2} \cdot \sqrt{\sum_{x,y} I'(x + i, y + j)^2}} \quad (4.4)$$

$$I'(x + i, y + j) = I(x + i, y + j) - \bar{I}(i, j)$$

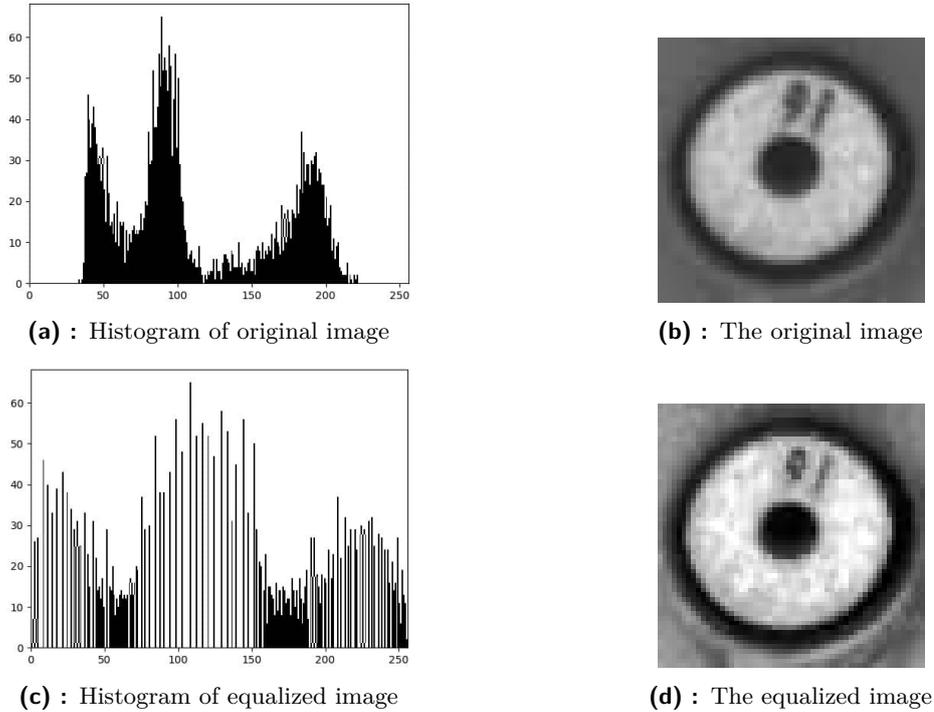
$$T'(x, y) = T(x, y) - \bar{T}$$

### ■ 4.2.2 Template creation

Templates for template matching method were created from repaired annotations of 16 subjects with the same type of electrodes. All ground truth boxes were cropped from the original images and subsequently split into twelve categories, denoted by electrode colour and its shape and size. Images in each category were resized and equalized using histogram equalization. This step was included for easier clustering that was performed immediately after. Eventually all of the images in each cluster have been averaged and contrast has been adjusted using histogram equalization. At the end, 16 black and 29 white templates were created.

### ■ Histogram equalization

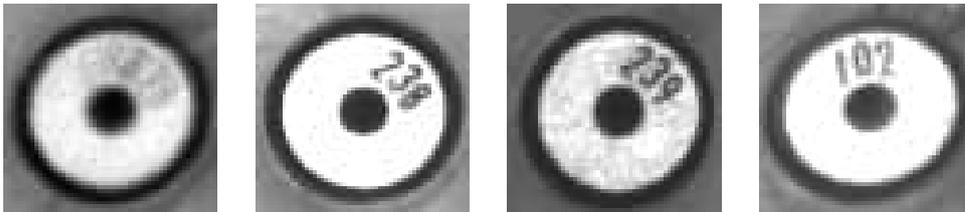
Histogram of an image is a graphical representation of frequency of occurrence of the various grey levels in the image [36]. The histogram equalization modifies the image histogram so that the full grey scale spectrum is used. This translates into an enhancement of the image contrast [37].



**Figure 4.5:** Comparison of a original image and a equalized image using histogram equalization

### ■ K-mean clustering

Clustering is a process of separating samples into groups of equal variances, minimizing the sum of squares [38]. Number of clusters is computed separately for each category such that the maximum number of images in each cluster is two hundred.



**Figure 4.6:** Final template (leftmost) and some of the images found in its cluster

### ■ 4.2.3 Template matching optimization

This section will describe the optimization of template matching algorithm using similar approach to neural network model optimization. The non-maximum suppression threshold and the match score are optimized in order to minimize the average distance between the detected and the ground truth bounding boxes.

#### ■ Average distance

This value denotes the average mean distance between ground truth and predicted bounding boxes through all images from the training dataset.

#### ■ False positive detections

As the inner circle of the electrodes measures approximately 10 pixels, if the distance from detection centre to its closest ground truth centre is larger than 10 pixels, the detection is considered incorrect. Such detections are marked as false positive.

#### ■ Undetected sensors

A sensor from ground truth annotation is considered undetected if the distance from its centre to closest prediction centre is larger than 10 pixels.

#### ■ Misclassified sensors

A sensor is misclassified if the predicted colour is different from its true colour.

#### ■ Match score

Match score is similar to confidence score in neural network models. Its value represents the maximum correlation.

#### ■ Results

Similarly to the neural network model optimization the results of template matching can be slightly enhanced by changing the NMS and match score thresholds. The number of undetected sensors was set to be 9 at the maximum and the average distance was minimized.

In the Table 4.2 below the results of different types of detections can be seen.  $TM_{all}$  as well as the  $TM_{opt. parameters}$  try to match all the templates regardless of the detected colour which results in a larger number of misclassified sensors. This problem was resolved by taking the colour into account while template matching. Note that the misclassification rate is higher in the R-CNN as it is common to detect the same electrode as both types, which in the case of  $TM_{R-CNN classes}$  and  $TM_{opt. parameters}$  is filtered out using the optimal parameters.

Detections	Average distance [px]	False positive	Undetected sensors	Misclassified sensors
R-CNN	6.1765	7.108	10.6364	0.7898
$TM_{all}$	3.5248	4.6364	8.2898	0.875
$TM_{opt. parameters}$	2.944	2.7727	8.9886	0.6193
$TM_{R-CNN classes}$	2.9087	2.7784	8.9716	0.1818

**Table 4.2:** Evaluation of detections before and after using template matching algorithm (TM). Values are averaged across all images in training dataset.



**Figure 4.7:** Example of the Template Matching output with match score of each bounding box



## Chapter 5

### Corresponding points estimation

As all the electrode positions are known, the next step will be estimation of corresponding points between camera images. However, for that the parameters of each camera, represented in the camera projection matrix, have to be known. Therefore, the first part of this chapter deals with the process of camera matrix estimation also known as camera calibration. In second part of the chapter the suggested method for corresponding points estimation will be presented.

#### 5.1 Camera calibration

The camera projection matrix for each camera was computed from the provided dataset. The matrices were computed separately for each of the 18 subjects and were averaged subsequently. The computation could be performed on all 18 subjects, including the 2 with different types of electrodes as the same set of cameras was used.

##### 5.1.1 Camera projection matrix

In computer vision the camera projection matrix refers to  $3 \times 4$  matrix  $P$  with 11 degrees of freedom. This matrix describes the mapping from a world point  $X$  to an image point  $x$  according to equation 5.1. It is important to note that the points are represented in homogeneous coordinates.<sup>1</sup>

$$\mathbf{x} = P\mathbf{X} \tag{5.1}$$

---

<sup>1</sup>Represented as vector  $(kx, ky, k)$ , alternatively  $(kx, ky, kz, k)$  for 3D coordinates.

### 5.1.2 Direct linear transformation

Given sufficiently many correspondences between 3D point  $X$  and its image point  $x$ , the camera matrix  $P$  can be estimated using the Direct Linear Transformation (DLT) algorithm [22].

All the corresponding points have to satisfy the equation 5.1. From this assumption the relationship 5.2 for each correspondence  $X_i \leftrightarrow x_i$  can be derived, writing  $x_i = (x_i, y_i, w_i)$  and  $P^{iT}$  representing the  $i$ -th row of matrix  $P$ .

$$\begin{bmatrix} 0^T & -w_i X_i^T & y_i X_i^T \\ w_i X_i^T & 0^T & -x_i X_i^T \end{bmatrix} \begin{pmatrix} P^1 \\ P^2 \\ P^3 \end{pmatrix} = 0 \quad (5.2)$$

Given a set of  $n$  correspondences, the matrix  $A$  of size  $2n \times 12$  can be created by stacking up the equations from 5.2 for each correspondence. By solving the equation  $Ap = 0$ , where  $p$  is the vector containing the entries of the matrix  $P$ , the projection matrix  $P$  can be computed.

### 5.1.3 Normalization

In order to achieve better numerical stability of DLT estimation, the use of data normalization is essential. The normalization consists of two steps. First, the points are translated so that their mean position (centroid) is at the origin. Afterwards, the points are scaled so that their root mean squared distance from the origin is  $\sqrt{2}$ , alternatively  $\sqrt{3}$  for 3D points. It should be noted that the camera matrix computed from normalized data has to be denormalized in order to obtain the camera matrix for the original data [22].

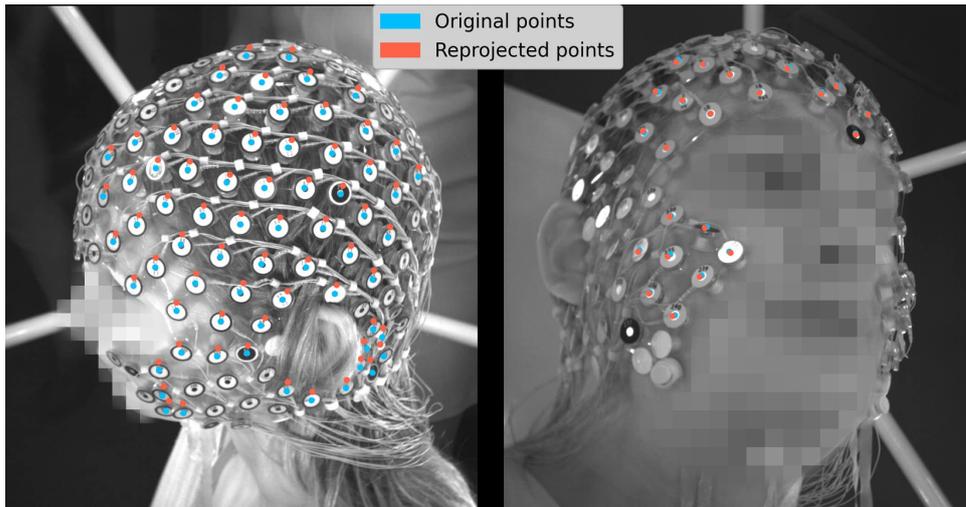
### 5.1.4 Evaluation

The evaluation was performed using the leave-one-out cross validation method. This means that one subject was left out from camera matrices estimation and used as the test subject. This process was then repeated for every subject in the dataset.

As the evaluation metric the reprojection error was used. The reprojection error is obtained by computing the re-projections of 3D coordinates using the computed camera matrix  $P$  and measuring the average euclidean distance between ground truth and re-projected points.

Camera	Reprojection error across all subjects [px]		
	Minimum	Maximum	Average
Camera 1	3.6039	13.8467	8.1270
Camera 2	3.0821	16.1087	8.5947
Camera 3	4.1248	16.8658	8.2112
Camera 4	4.3706	12.7024	8.0698
Camera 5	4.061	15.4016	8.9685
Camera 6	4.1472	17.2836	8.5358
Camera 7	2.9519	11.7884	8.2485
Camera 8	4.6069	14.4847	8.5055
Camera 9	5.5044	14.3455	8.7627
Camera 10	3.2112	14.7826	9.0207
Camera 11	4.6398	14.9952	8.0231

**Table 5.1:** Evaluation of camera matrices estimation using the leave-one-out cross validation



**Figure 5.1:** Illustration of the worst (left) and best (right) case of the 3D coordinates re-projection

## 5.2 Corresponding points estimation

For estimation of corresponding points, a method consisting of several steps was suggested. These steps are outlined below and described in greater depth in following subsections.

- Find all possibly corresponding points between two cameras
- From corresponding pairs compute the 3D coordinates
- Re-project 3D coordinates to other camera images
- From re-projected points select only the consistent ones
  - This creates the set of all possibly corresponding points<sup>2</sup>
- Find the optimal subset of corresponding points

### ■ 5.2.1 Camera constraints

As mentioned above, the first step is the computation of possibly corresponding points from different cameras pairs. As there are many possible pairs, some of which have no overlap, the improvement in the form of camera constraints was implemented.

From the provided dataset the visible sensors for each camera were extracted. Afterwards, the camera pairings were deemed possible if they had at least one sensor in common.

### ■ 5.2.2 Correspondences between two cameras

Suppose a point  $X$  in space is imaged in two views as point  $x$  and  $x'$ . The relation between these corresponding points can be expressed using epipolar geometry. The point  $x$  from one image can be mapped to its corresponding epipolar line  $l'$  in the other image, on which the corresponding point  $x'$  lies [22].

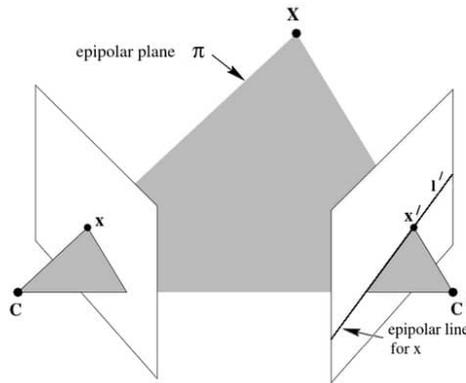
For each detection in one camera image the corresponding epipolar line in second camera image is computed. All points from the second camera with the same colour as the detection and maximum distance of 8 pixels from the epipolar line is considered as possibly corresponding point.

### ■ Epipolar geometry

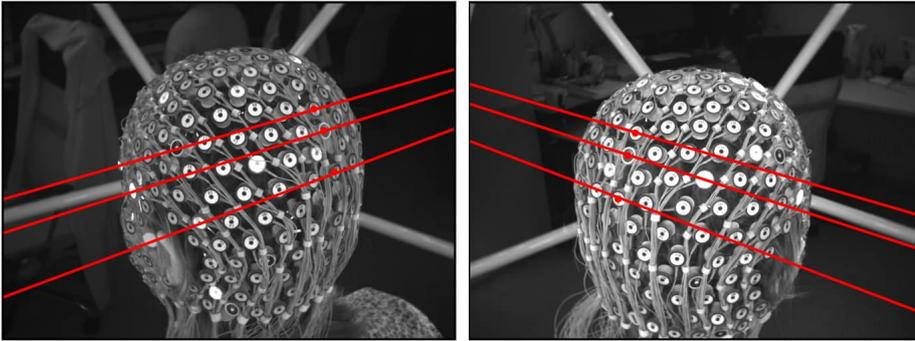
Epipolar geometry describes the geometric relationship between two perspective cameras. As mentioned above the relation between  $x$  and  $x'$  can be expressed using epipolar lines. The points  $X$ ,  $x$  and  $x'$  as well as the camera centres  $C$  and  $C'$  lie on the epipolar plane. Epipolar line is defined as the intersection of the epipolar plane with the image plane [22].

---

<sup>2</sup>Various initial pairs of cameras result in multiple sets of corresponding points.



**Figure 5.2:** Visualization of the main ideas of the epipolar geometry [22]



**Figure 5.3:** Example of epipolar lines on the images from training dataset

### ■ Fundamental matrix

As the fundamental matrix is the algebraic representation of epipolar geometry, it is the base for epipolar line computation. Assuming that  $\mathbf{x}$  is a point in the first image, the computation of corresponding epipolar line  $l'$  in the second image using the fundamental matrix  $\mathbf{F}$  is defined in equation 5.3.

$$l' = \mathbf{F}\mathbf{x} \quad (5.3)$$

There are several methods for fundamental matrix computation. In this thesis the computation using the camera matrices shown in equation 5.4 was used. This method was included in the source files that were provided with the Multiple View Geometry book [22], [39]. The  $P_j^i$  in equation 5.4 represent the  $i$ -th row of camera matrix  $j$ .

$$\mathbf{F} = \begin{bmatrix} \det \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} & \det \begin{pmatrix} x_2 \\ y_1 \end{pmatrix} & \det \begin{pmatrix} x_3 \\ y_1 \end{pmatrix} \\ \det \begin{pmatrix} x_1 \\ y_2 \end{pmatrix} & \det \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} & \det \begin{pmatrix} x_3 \\ y_2 \end{pmatrix} \\ \det \begin{pmatrix} x_1 \\ y_3 \end{pmatrix} & \det \begin{pmatrix} x_2 \\ y_3 \end{pmatrix} & \det \begin{pmatrix} x_3 \\ y_3 \end{pmatrix} \end{bmatrix} \quad (5.4)$$

$$x_1 = \begin{pmatrix} P_1^2 \\ P_1^3 \end{pmatrix} \quad x_2 = \begin{pmatrix} P_1^3 \\ P_1^1 \end{pmatrix} \quad x_3 = \begin{pmatrix} P_1^1 \\ P_1^2 \end{pmatrix}$$

$$y_1 = \begin{pmatrix} P_2^2 \\ P_2^3 \end{pmatrix} \quad y_2 = \begin{pmatrix} P_2^3 \\ P_2^1 \end{pmatrix} \quad y_3 = \begin{pmatrix} P_2^1 \\ P_2^2 \end{pmatrix}$$

### ■ 5.2.3 Linear triangulation

Suppose two corresponding points  $\mathbf{x} \leftrightarrow \mathbf{x}'$  and a pair of camera matrices  $\mathbf{P}$  and  $\mathbf{P}'$ . From equation 5.1 and the fact that  $\mathbf{x} \leftrightarrow \mathbf{x}'$  are corresponding points, both equations 5.5 and 5.6 must be satisfied simultaneously. This is the key idea to the 3D point estimation.

$$\mathbf{x} = \mathbf{P}\mathbf{X} \quad (5.5)$$

$$\mathbf{x}' = \mathbf{P}'\mathbf{X} \quad (5.6)$$

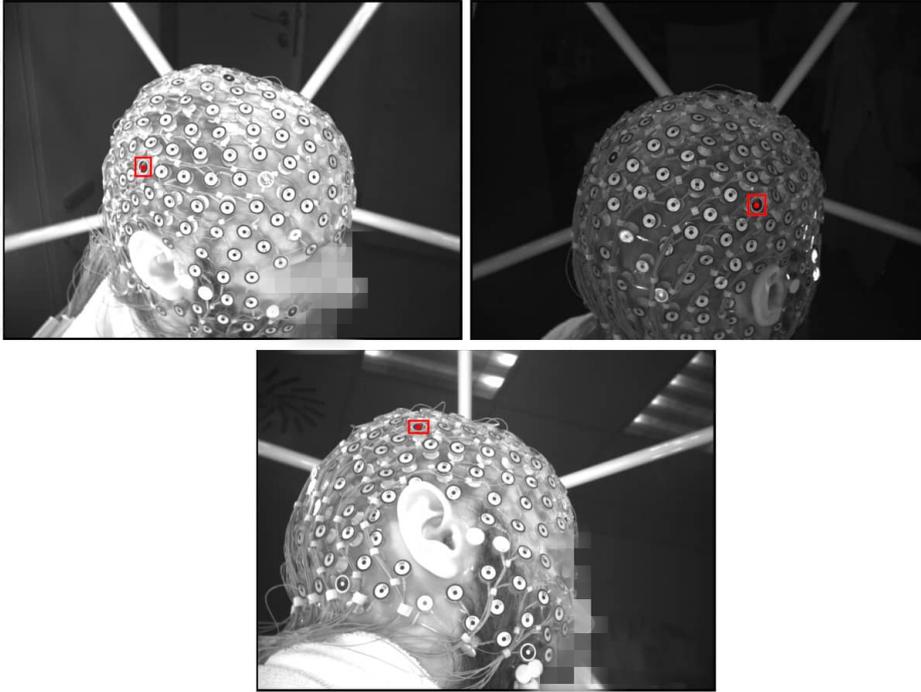
The linear triangulation method is the direct analogue of the DLT method described in Section 5.1.2. From the equations above the relationship 5.7 can be derived, writing  $\mathbf{x} = (x, y)$ ,  $\mathbf{x}' = (x', y')$ ,  $\mathbf{X} = (kx, ky, kz, k)$  and  $\mathbf{p}^{iT}$  and  $\mathbf{p}'^{iT}$  being the rows of matrix  $\mathbf{P}$  and  $\mathbf{P}'$ . By solving the equation 5.7 the 3D point  $\mathbf{X}$  can be estimated [22].

$$\begin{bmatrix} x\mathbf{p}^{3T} - \mathbf{p}^{1T} \\ y\mathbf{p}^{3T} - \mathbf{p}^{2T} \\ x'\mathbf{p}'^{3T} - \mathbf{p}^{1T} \\ y'\mathbf{p}'^{3T} - \mathbf{p}^{2T} \end{bmatrix} \begin{pmatrix} kx \\ ky \\ kz \\ k \end{pmatrix} = 0 \quad (5.7)$$

### ■ 5.2.4 Consistent re-projected points

At this moment, it is assumed that the 3D point for each possibly corresponding pair is computed. In the next step, the 3D point is re-projected into other cameras satisfying the camera constraints from Section 5.2.1. By selecting the

consistent re-projected points, a set of points which could possibly represent the same 3D point is obtained. The re-projected point is considered to be consistent if its distance to the closest detected centre is smaller than 5 pixels and the predicted classes match.



**Figure 5.4:** Example of a re-projected point at the bottom image, computed from 3D point estimated from corresponding points from the images at the top

### ■ 5.2.5 Optimal subset

From the previous steps we have obtained all sets of possibly corresponding points. Note that at this stage one point can be included in multiple correspondences. The last step focuses on finding the optimal subset of consistent corresponding points - meaning that each image point can be included only once. Several approaches were evaluated and will be discussed below.

### ■ Greedy algorithm

The first attempt was to utilize a basic greedy algorithm as the correspondences with more points are more probable to be correct. The main idea of the greedy algorithm is to take the correspondence with the most points and suppress all other correspondences with the same points. This step is repeated until there are no more correspondences to choose from.

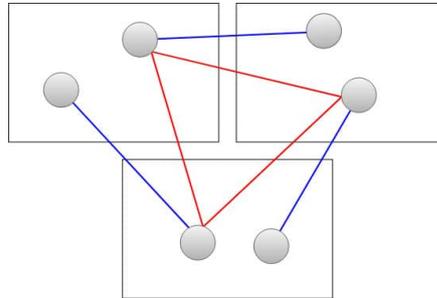
Because of the camera layout, the sensors usually are visible on three cameras at most. Due to this, the set of corresponding points consists mainly of two and three point correspondences. Even though the greedy algorithm prefers the three point correspondences it selects them randomly. Therefore, it is unable to achieve satisfactory results.

### Integer linear programming (ILP)

Another idea was considering our problem as the maximum independent set problem, which can be solved using the integer linear programming. Let  $S$  be the optimal subset of corresponding points. The problem of finding  $S$  can be formulated as in equation 5.8, where  $C_i$  is the  $i$ -th correspondence and  $x_i$  is binary variable denoting that  $C_i \in S$ .

$$\begin{aligned} \max \quad & \sum_i |C_i| x_i \\ \text{s.t.} \quad & x_j + x_k \leq 1 \quad \text{iff } C_j \text{ and } C_k \text{ have} \\ & \quad \quad \quad \text{a common point} \end{aligned} \quad (5.8)$$

This formulation tries to maximize the number of used image points such that each point can be used only once. However, this formulation was found to be unfortunate for our problem as in practise it prefers correspondences with smaller number of points. Hence, it was common, that more than 300 correspondences were chosen, although there are only 256 electrodes.



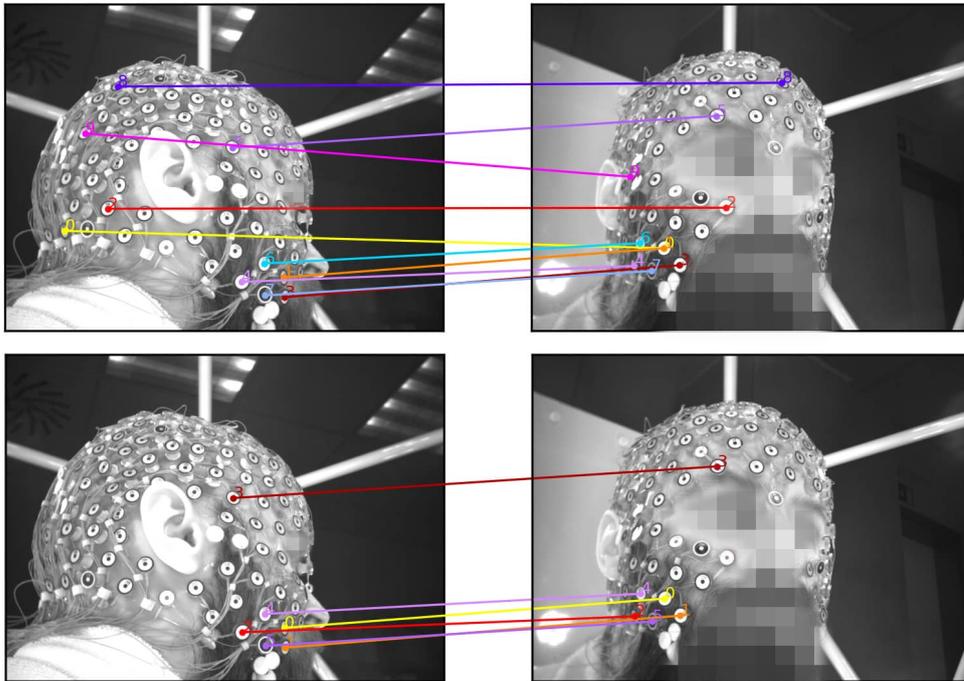
**Figure 5.5:** Example of optimization formulation problem. Consider four possible correspondences - three point in red and two point in blue. The formulation will prefer the two point correspondences as it maximizes the number of used points.

It is important to note, that the set of possibly corresponding points consists of about 2600 two point correspondences and around 400 three point correspondences on average. Therefore, there are many incorrect correspondences, pairs especially, that the algorithm can choose from.

### ■ Leordeanu-Herbert algorithm

In previous approaches, we did not take the relationship between the individual correspondences into consideration. Approach described in this section is based on the article [40], which focuses on finding consistent correspondences between two images. The idea behind this algorithm is that the distance between corresponding points in two images should not be too different. With this in mind, a score for each correspondence is computed and subsequently used in a greedy algorithm thus solving the problem. For the purpose of this thesis, this algorithm is slightly modified so that it can remove obviously incorrect correspondences as well. The modification consists of using thresholds for score and similar distance change.

The score of all correspondences is computed as a principal eigenvector of non-negative, symmetrical matrix  $M$  [40]. In this thesis, matrix  $M$  is computed from differences of distances between each pair of corresponding points. Value of  $M(a, b)$  refers to the number of correspondence pairs with similar distance change as correspondences  $a$  and  $b$ .



**Figure 5.6:** Comparison of pair correspondences before (top) and after (bottom) Leordeanu-Herbert algorithm. Note that the algorithm does not remove the top correspondence as the change of distance between images is similar to the correct correspondences.

### ■ The final method

The final method for finding optimal subset of possibly corresponding points combines the approaches described above. As the three point correspondences are more probable to be correct, we decided to divide the algorithm into two parts. At first the three point correspondences are chosen. Afterward, we can reduce the number of two point correspondences by considering only the ones that are consistent with previously selected three point correspondences. Then the optimal pair correspondences will be easier to find.

For the three point correspondences selection the approach utilizing ILP is used. As only three point correspondences are used the problem with smaller set of corresponding points preference does not apply. Afterward, we use the Leordeanu-Herbert algorithm to remove the apparently incorrect correspondences. After removing the non-consistent pair correspondences its number is reduced to approximately  $\frac{1}{5}$  of the original set. As this still leave us with a large set of two point correspondences the Leordeanu-Herbert algorithm is used again for further reduction. Then the approach taking advantage of ILP is to find optimal consistent set of pairs.

Correspondence type	Algorithm step	Number of possible correspondences
3-points	initially	391.625
3-points	after ILP	137.9375
3-points	after LH	133.6875
2-points	initially	2613.25
2-points	only consistent	465.75
2-points	after LH	151.125
2-points	after ILP	96.6875

**Table 5.2:** The average number of correspondences throughout the correspondence finding algorithm (LH refers to Leordeanu-Herbert algorithm)

### ■ 5.2.6 Evaluation

Evaluation of this part of code is quite challenging as the algorithms output cannot be easily validated against the dataset. The algorithm outputs the 3D points and its corresponding 2D coordinates across the images. However, the correctness of the computed correspondences cannot be automatically decided by either 3D or 2D coordinates. Therefore, the following evaluation metrics were proposed.

### ■ Number of correspondences

The first and easiest metric is the average number of found correspondences across the images. If there are more than 256 correspondences, apparently some of them are incorrect.

### ■ Average distance to the closest point

This metric utilizes the knowledge of 3D computed and ground truth coordinates. However, as the algorithm does not provide information about the label of the coordinate, the computed point cannot be matched with its ground truth position and subsequently evaluated.

This proposed metric computes the average distance from 3D computed point to the closest ground truth point with the same electrode colour.

### ■ Right, Wrong and Unknown correspondences

Before computing this metric, we extracted information about electrode label and position in all images for each subject from the training dataset. Afterwards, based on this information we can divide the computed correspondences into three groups using their 2D coordinates. For each 2D coordinate in the computed correspondence, we find the label of the closest ground truth point. For the computed point to be matched with the ground truth point, their distance has to be smaller than 15 pixels, otherwise we assume that the computed point is not included in the ground truth points. According to the labels of estimated corresponding points we then decide their correctness.

The correspondence is marked as unknown if the number of found labels is smaller than two. This means, that no or only one point from correspondence was found in ground truth points, due to which, the correctness of correspondence cannot be decided.

The second possibility is, that the found labels (at least two) are the same. In this case the correspondence is marked as correct. Note that this definition is not optimal as we assume that two matching labels automatically mean correct correspondence. However, this does not have to always be true. The problem is that training dataset consist almost exclusively of two point correspondences, therefore all of the computed three point correspondences would be marked as unknown.

Finally, the correspondence is marked as wrong if none of the previous rules matched.

## Results

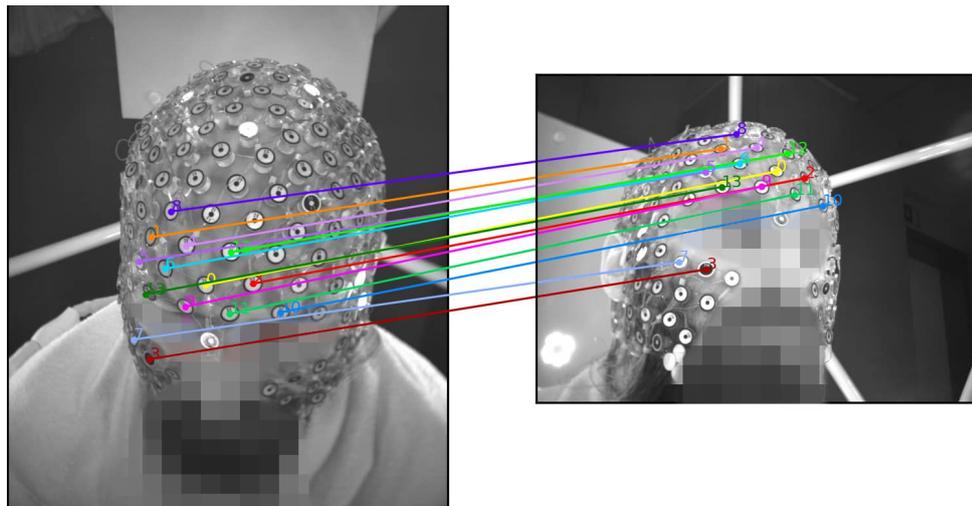
Using the metrics described above, the algorithm for correspondence estimation was evaluated. Different methods for optimal subset finding were used. The *Greedy*, *ILP* and *Final* methods were described in Section 5.2.2. Methods *ILP 3+ only* and *ILP sequential* are proposed improvements for the ILP algorithm, which attempt to solve the problem of correspondence with smaller number of points preference.

*ILP 3+ only* method selects the optimal subset from three and more point correspondences only. Although, the preference problem is solved with a good accuracy, the number of found correspondences rapidly decreases.

*ILP sequential* attempts to solve the preference problem by splitting the problem into two steps. First the optimal three and more point correspondences are found. The number of two point correspondences is then reduced by selecting only the consistent ones and afterwards, the optimal two point correspondences are found.

Subset finding method	Number of corresp.	Closest point distance [cm]	Right	Wrong	Unknown
Greedy	243.125	1.228	137.25	50.188	55.688
ILP	323.875	2.12	47.8	114.375	161.688
ILP 3+ only	137.938	0.351	127.938	4.938	5.063
ILP sequential	260.125	1.091	155.5	44.125	60.5
Final	230.375	0.568	168.5	17.938	43.938

**Table 5.3:** Evaluation of found correspondences using different optimal subset finding methods



**Figure 5.7:** Example of the final correspondences found between two images

## Chapter 6

### 3D model reconstruction

At this point, we have determined a partial 3D model using the correspondences found in previous chapter. The final step of the 3D model reconstruction is the computation of the remaining points. In this chapter, the suggested solution consisting of fitting a statistical model to the computed one as well as the statistical model computation, will be described.

#### 6.1 Statistical model

In this section the computation of the statistical model in a form of the Point Distribution Model (PDM) [41] is described. This approach assumes we have a set of model examples from which the statistical description of the shape and its variations can be derived [42].

##### 6.1.1 Aligning the training dataset

In order to develop the PDM model, it is necessary to align the training shapes first. Considering we have a set of  $M$  training shapes  $\{\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3 \dots, \mathbf{x}^M\}$  we can align the shapes using the steps bellow [43].

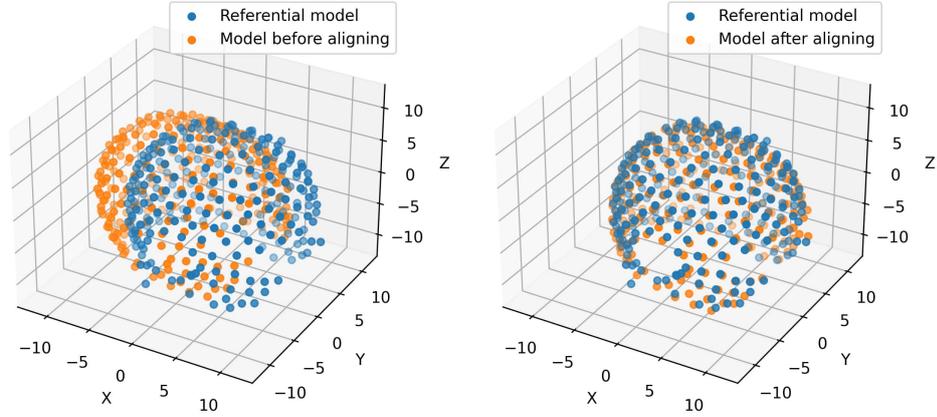
1. Align each  $\mathbf{x}^i$  with  $\mathbf{x}^1$ , for  $i = 2, 3, \dots, M$ 
  - This gives us the set  $\{\mathbf{x}^1, \hat{\mathbf{x}}^2, \hat{\mathbf{x}}^3 \dots, \hat{\mathbf{x}}^M\}$
2. Calculate the mean  $\bar{\mathbf{x}}$  of the aligned shapes
3. Align the mean shape  $\bar{\mathbf{x}}$  with  $\mathbf{x}^1$
4. Align the set  $\{\mathbf{x}^1, \hat{\mathbf{x}}^2, \hat{\mathbf{x}}^3 \dots, \hat{\mathbf{x}}^M\}$  to match to the adjusted mean
5. Repeat steps 2. - 4. until convergence

### ■ Aligning two shapes

Let  $P = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n\}$  and  $Q = \{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n\}$  be two sets of corresponding points in  $\mathbb{R}^d$ . Our aim is to find an optimal rigid transformation  $T$  of shape  $P$  composed of rotation  $R$  and translation  $t$  using standard techniques (equation 6.1). The optimal transformation should minimize the sum of squared distance between the shapes (equation 6.2), which is a routine application of the least-squares approach [44].

$$T_{R,t}(P) = R\mathbf{p}_i + t \quad (6.1)$$

$$(R, t) = \operatorname{argmin}_{R,t} \sum_{i=1}^n \|(R\mathbf{p}_i + t) - \mathbf{q}_i\|^2 \quad (6.2)$$



**Figure 6.1:** Comparison of referential model  $\mathbf{x}^1$  and another model before and after aligning

### ■ 6.1.2 Deriving the model

The alignment from previous step gives us the set of mutually aligned shapes  $\{\hat{\mathbf{x}}^1, \hat{\mathbf{x}}^2, \hat{\mathbf{x}}^3 \dots, \hat{\mathbf{x}}^M\}$  and their mean  $\bar{\mathbf{x}}$ . This knowledge allows us to compute the covariance matrix  $S$ .

$$S = \frac{1}{M} \sum_{i=1}^M \delta \mathbf{x}^i (\delta \mathbf{x}^i)^T \quad (6.3)$$

$$\delta \mathbf{x}^i = \hat{\mathbf{x}}^i - \bar{\mathbf{x}}$$

The most significant modes of variation in the shapes can be described using the eigenvectors of covariance matrix  $S$  corresponding to the largest eigenvalues. In this thesis, the number of eigenvectors used in further computations is set such that the sum of neglected eigenvalues does not exceeds 5%.

The process of computation and using the eigenvalues as described above is known as the Principal Component Analysis (PCA).

Assuming that  $\Phi$  contains the eigenvectors described above, each shape from the training set can be approximated as in equation 6.4. The  $\mathbf{b}$  is a vector of weights for each eigenvector and thanks to the eigenvectors orthogonality, it can be easily derived, see 6.5. Each weight  $\mathbf{b}_i$  should be constrained using the corresponding eigenvalue  $\lambda_i$  as in equation 6.6 in order for the shape to be plausible [45].

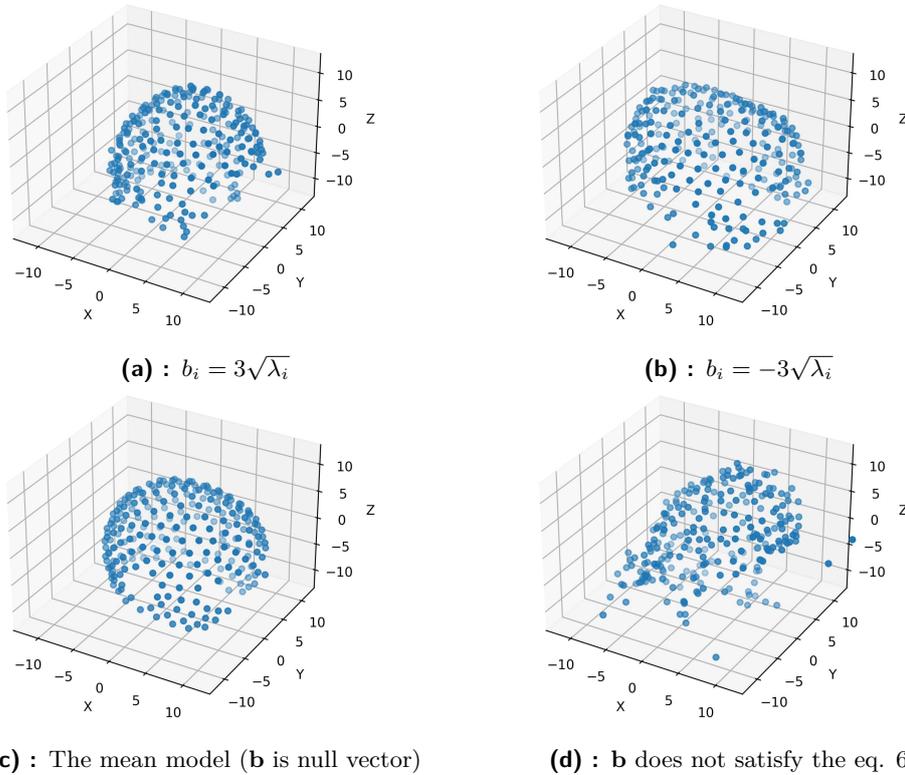
$$\mathbf{x} = \bar{\mathbf{x}} + \Phi \mathbf{b} \quad (6.4)$$

$$\mathbf{b} = \Phi^T (\mathbf{x} - \bar{\mathbf{x}}) \quad (6.5)$$

$$|\mathbf{b}_i| \leq 3\sqrt{\lambda_i} \quad (6.6)$$

### 6.1.3 Final statistical model

The final statistical model was computed from all 18 provided subjects, thus including the 2 subjects with different electrode type. This was possible because even though the electrodes have different appearance the underlying system remains the same.



**Figure 6.2:** Comparison of statistical models using different weight vectors  $\mathbf{b}$

## 6.2 Final 3D model

Fitting a statistical model to corresponding set of new points is a well-known problem complete with working solutions. The aim is to find the optimal pose  $(R, t)$  and shape  $(b)$  parameters such that the sum of square distances between the model and the new points is minimized [42], [43], [45]. However, this algorithm assumes the same number of points in both the model and the set of the new point as well as the knowledge of assignments between them. In our case, neither is applicable.

To compute the assignments, one possible approach could be based on the Iterative Closest Point algorithm [46]. We could assign each new point the nearest corresponding model point and based on the determined assignments transform the model. However, in our case, the closest point assignment is hardly ever correct. Another possible approach is to use the Softassign algorithm [47], where the assignments are determined using the assignment matrix with values in range from 0 to 1.

The problem is that both the ICP as well as the Softassign algorithm are made to fit a predefined model using affine or rigid transformation. In other words, these algorithms do not take the shape parameter  $b$  into account.

This section aims to describe the proposed algorithm for 3D model reconstruction. First the statistical model is fitted on predicted partial one utilizing a combination of beforementioned methods. Afterwards, the 3D positions of remaining points are taken from the fitted statistical model.

### 6.2.1 Overview of proposed algorithm

For this algorithm we assume the knowledge of the predicted partial model  $\mathbf{x}$  and statistical model consisting of the mean model  $\bar{\mathbf{x}}$  and covariance matrix  $S$ .

The proposed algorithm consists of two loops. The first loop aims to find the optimal pose and shape parameters for the mean model  $\bar{\mathbf{x}}$  given the assignment matrix  $\mathbf{A}$ . Using that, we can determine the assignments between points and therefore, the fitting algorithm mentioned at the beginning of this chapter can be used.

However, the assignment matrix still needs to be optimized. This is handled by the inner loop, which is based on the Softassign algorithm.

It is important to note, that these two loops cannot be merged. This is because the Softassign algorithm assumes the model to be fixed in order to work properly and at the same time the same holds for the outer loop and point assignments. This became apparent when this approach was first evaluated.

**Algorithm 1:** Proposed solution for 3D reconstruction

---

**Result:** assignment matrix  $\mathbf{A}$ , fitted model  $\mathbf{M}$   
 $\bar{\mathbf{x}}$  = shift  $\bar{\mathbf{x}}$  to  $\mathbf{x}$  such that their centroids match;  
 $\mathbf{A}$  = compute assignment matrix for  $\mathbf{x}$  and  $\bar{\mathbf{x}}$ ;  
 $\Phi, \lambda$  = compute eigenvectors and eigenvalues using  $\text{PCA}(S)$ ;  
**while** *Distance between fitted and predicted model does not converge* **do**  
     $\mathbf{R}, \mathbf{t}, \mathbf{b}$  = compute optimal parameters given  $\mathbf{A}$ ;  
     $\mathbf{M}$  = compute model using optimal parameters as  $T_{\mathbf{R},\mathbf{t}}(\bar{\mathbf{x}} + \Phi\mathbf{b})^a$ ;  
    **while** *assignment matrix does not converge* **do**  
         $\mathbf{A}$  = compute assignment matrix for  $\mathbf{x}$  and  $\mathbf{M}$ ;  
         $\mathbf{R}, \mathbf{t}$  = compute optimal pose parameters using *Walker et al.'s method*;  
         $\mathbf{M}$  = transform model using optimal parameters as  $T_{\mathbf{R},\mathbf{t}}(\mathbf{M})$ ;  
    **end**  
**end**

---

<sup>a</sup> $T_{\mathbf{R},\mathbf{t}}$  refers to transformation using the rotation and translation parameters (see 6.1).

### 6.2.2 Assignment matrix

The assignment matrix specifies the matching between two sets of points. Assume image points  $J$  and model points  $K$ , assignment matrix  $\mathbf{A}$  is then an  $(J + 1) \times (K + 1)$  matrix with values in range from 0 to 1. The value  $A(j, k)$  defines the probability that points  $j$  and  $k$  are matching. The last row and column consist of slack variables, which are equal to one if relevant points do not have their correspondences in the other set.

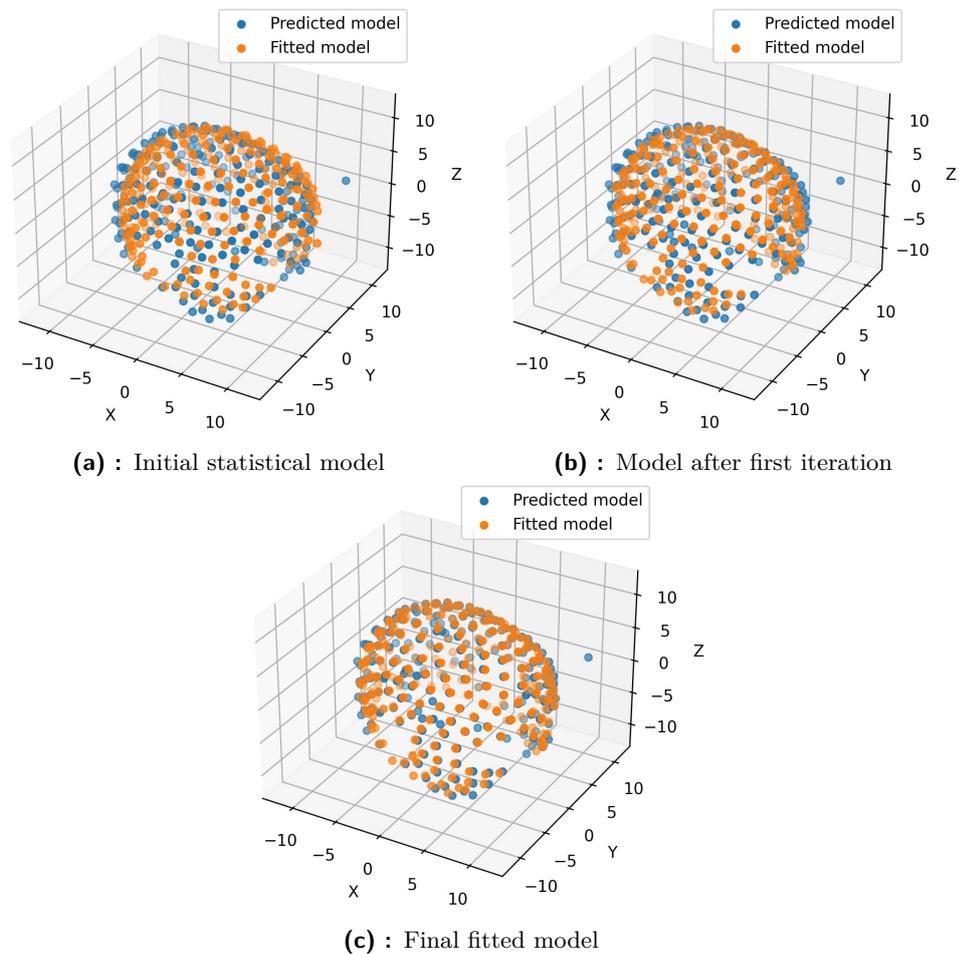
The computation of assignment matrix is based on Softassign and Softposit algorithms [47], [48]. In this thesis the computation is slightly modified. We use the knowledge of points class (black or white sensor) to forbid the assignment of different class. Two points are then deemed as a correct assignment if their maximal values in assignment matrix corresponds.

	A	B	C	Slack
1	0.1	0.7	0.1	0.1
2	0.4	0.1	0.3	0.2
Slack	0.5	0.2	0.6	-

**Figure 6.3:** Example of assignment re-computation from the assignment matrix. Only points 1 and B will be assigned as they share the maximal value. Point 2 cannot be matched with point A as the maximal value of point A is in the slack row. The slack row and column are not normalized.

### 6.2.3 Optimal shape computation

This section refers to the first line of the outer while loop from Algorithm 1. The algorithm is based on the computation of optimal pose and shape parameters from [45]. However, as the algorithm is described in [45], it uses sets of 2D points and assumes the exact assignments between them are known, thus the algorithm had to be modified such that it can use the assignment matrix and our data. The main problem was in computation of optimal rotation and translation given the assignment matrix, for which the Walker et al.'s method [47], [49] was used.



**Figure 6.4:** Comparison of fitted and predicted model throughout the fitting process

# Chapter 7

## End-to-end evaluation

In this Chapter, we will evaluate the proposed algorithm for electrode position estimation as a whole. The evaluation will be divided into three sections. In the first and second section, we will discuss the precision of computed results and the time complexity of the proposed algorithm. In the third section, the results of the electrical source localization from the computed coordinates, will be described.

### 7.1 Accuracy evaluation

The evaluation results were obtained using the leave-one-out cross validation technique. In our implementation this translates to the following approach. We used all except one subject for the camera calibration and statistical model computation. Afterwards, the position of electrodes for the excluded subject were estimated and compared to the ground truth coordinates. This process was then repeated for all other subjects from the training dataset.

As the evaluation metric the average euclidean distance between the estimated and ground truth coordinates is used. On average, the distance between computed and ground truth coordinates is 3.3 mm, whereas the Geodesic Photogrammetry System can achieve accuracy up to 1 mm [50]. However, it is important to remember, that the training dataset used also in this evaluation as ground truth contained some inaccuracies (as mentioned in Section 3.2), which might skew the results.

Average distance between estimated and GT coordinates [cm]		
Minimum	Maximum	Average
0.2108	0.4914	0.3292

**Table 7.1:** Evaluation of estimated coordinates using the leave-one-out cross validation technique

## 7.2 Time complexity

Overall most parts of the algorithms run fast, however, considering its runtime, the statistical model fitting stands out the most. Due to its iterative nature, it largely depends on the correctness of the estimated correspondences. Therefore, this would be the best candidate for future runtime improvement.

The proposed algorithm completes the computation in about 3 minutes whereas the process of electrode estimation using the Geodesic Photogrammetry System software takes up to 30 minute [50].

Part of the algorithm	Average runtime [s]
Sensors detection using R-CNN	9.414
Template Matching	2.081
Correspondence estimation	4.169
Statistical model fitting	174.053
Computation of final 3D model	0.065
In total	189.782

**Table 7.2:** The overview of the time complexity of each part of the proposed algorithm

## 7.3 Electrical source imaging

An optional part of this thesis was the experimental validation of the improvement in electrical source localization using the proposed algorithm. Our contact in the Czech National Institute of Mental Health kindly agreed to help with this part of the evaluation. Five subjects were selected for the validation based on an initial EEG data quality check. There were various issues with the rejected datasets including blurred images, no corresponding MRI scans in database or different electrode types to the trained ones. However, due to the time constraints and time consuming creation of reference electrode positions performed by NIMH employees, only one subject could be evaluated in time. On account of that, the results in this section have more illustrative than informative purpose.

The aim of the evaluation was to compare the auditory evoked potentials with the latency of 90-110 ms using three different electrode localization techniques.

### 7.3.1 Localization techniques

#### Manual localization

The manual localization, includes manual selection of each electrode in at least three images and afterwards, using a specialized software for the electrode position computation via triangulation. This process can take more than an hour for one subject and was used as the ground truth for the further evaluation.

#### Localization using template

The Geodesic sensor net used for EEG signals measurement has an official placement template. It can be used to obtain a rough estimate of the electrode positions. The fiducial sensors (nasion, left and right preauricular points) are marked in MRI which enable co-registration of electrode template model and the brain model from MRI. Each sensor from template is then assigned to the closest point from MRI model.

#### Automatic localization

Automatic localization refers to the proposed algorithm for electrode position estimation.

### 7.3.2 Results

After the solution for forward and inverse problem (see Section 2.1.4) was obtained, the signal threshold was set to be the half of the maximum signal value. Using this threshold, each voxel can be evaluated as true positive (TP), false positive (FP) or false negative (FN). This evaluation can be subsequently used for dice coefficient computation from equation 7.1.

$$\text{Dice coefficient} = \frac{2TP}{2TP + FP + FN} \quad (7.1)$$

Localization technique	Dice coefficient
Manual localization	1.00
Automatic localization	0.77
Template localization	0.68

**Table 7.3:** Dice coefficient values computed for different localization techniques





## Chapter 8

### Conclusion

The goal of this thesis was to develop an algorithm for automatic estimation of hdEEG electrode positions from the set of infrared cameras, which was completed successfully. The proposed algorithm seems to have achieved satisfying results in both the accuracy and time complexity from the electrode position localization standpoint. However, the training as well as the evaluation was made on a small dataset that contained inaccuracies, which might have skewed the results.

Therefore, evaluating the real-world performance would be highly beneficial. Unfortunately, due to time constraints, only one real world example was evaluated by our contact in the Czech National Institute of Mental Health during this thesis. Thus, we cannot draw conclusions about the effect on the electrical source localization accuracy.

One of the optional tasks in the assignment of this bachelor thesis is implementation of a text recognition method. That would have been used to recognize the electrode labels and thus improve correspondence estimation. This part was not implemented due to time constraint, however the proposed algorithm for correspondence estimation is accurate enough as it is.

Even though, the goal of the thesis was achieved, there is plenty of room for improvement. The inaccurate training dataset could be considered the biggest issue as it affects not only the evaluation but also the computation of the camera matrices and statistical model. If a larger dataset could be obtained, it would be also possible to train algorithm for different electrode types. From the algorithm standpoint, the best candidate for improvement is the statistical model fitting.





## Bibliography

- [1] M. Penhaker, M. Imramovský, P. Tiefenbach, and F. Kobza, “Elektroencefalografie”, in *Lékařské diagnostické přístroje: učební texty*. Ostrava: VŠB - Technická univerzita Ostrava, 2004, pp. 149–159, ISBN: 8024807513.
- [2] M. Teplan, “Fundamental of EEG Measurement”, *Measurement science review*, vol. 2, Jan. 2002.
- [3] C. Rye, R. Wise, V. Jurukovski, J. DeSaix, J. Choi, and Y. Avissar, “How neurons communicate”, in *Biology*. OpenStax, Oct. 2016. [Online]. Available: <https://openstax.org/books/biology/pages/35-2-how-neurons-communicate>.
- [4] Molecular Devices, *What is an action potential?* [Online]. Available: <https://www.moleculardevices.com/applications/patch-clamp-electrophysiology/what-action-potential>.
- [5] J. S. Kumar and P. Bhuvanewari, “Analysis of Electroencephalography (EEG) Signals and Its Categorization – A Study”, *Procedia Engineering*, vol. 38, Jan. 2012, ISSN: 1877-7058. DOI: 10.1016/j.proeng.2012.06.298.
- [6] F. Fürbass, “EEG monitoring based on automatic detection of seizures and repetitive discharges”, Ph.D. dissertation, Dec. 2017. [Online]. Available: <https://www.researchgate.net/publication/324720694>.
- [7] O. Mecarelli, “Electrode Placement Systems and Montages”, in *Clinical Electroencephalography*, O. Mecarelli, Ed. Springer International Publishing, 2019, pp. 35–52, ISBN: 9783030045722. DOI: 10.1007/978-3-030-04573-9\_4. [Online]. Available: [http://link.springer.com/10.1007/978-3-030-04573-9\\_4](http://link.springer.com/10.1007/978-3-030-04573-9_4).
- [8] S. Beniczky and P. Sharma, “Electromagnetic Source Imaging, High-Density EEG and MEG”, in *Clinical Electroencephalography*, O. Mecarelli, Ed. Springer International Publishing, 2019, pp. 329–343, ISBN: 9783030045722. DOI: 10.1007/978-3-030-04573-9\_20. [Online]. Available: [http://link.springer.com/10.1007/978-3-030-04573-9\\_20](http://link.springer.com/10.1007/978-3-030-04573-9_20).



- [20] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-Up Robust Features (SURF)”, *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, Jun. 2008, ISSN: 10773142. DOI: 10.1016/j.cviu.2007.09.014.
- [21] M. Brown, R. Szeliski, and S. Winder, “Multi-Image Matching Using Multi-Scale Oriented Patches”, in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 1, IEEE, 2005, pp. 510–517, ISBN: 9780769523729. DOI: 10.1109/CVPR.2005.235. [Online]. Available: <http://ieeexplore.ieee.org/document/1467310/>.
- [22] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, 2nd ed. Cambridge University Press, May 2004, ISBN: 9780521540513.
- [23] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, “Bundle Adjustment — A Modern Synthesis”, in *Vision Algorithms: Theory and Practice*, B. Triggs, A. Zisserman, and R. Szeliski, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, ISBN: 978-3-540-44480-0.
- [24] M. Lhuillier and L. Quan, “Match propagation for image-based modeling and rendering”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1140–1146, 2002. DOI: 10.1109/TPAMI.2002.1023810.
- [25] Y. Furukawa and J. Ponce, “Accurate, Dense, and Robust Multiview Stereopsis”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362–1376, 2010. DOI: 10.1109/TPAMI.2009.161.
- [26] M. Lhuillier and L. Quan, “A quasi-dense approach to surface reconstruction from uncalibrated images”, *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 3, pp. 418–433, Mar. 2005, ISSN: 0162-8828. DOI: 10.1109/tpami.2005.44.
- [27] S. El Hazzat, N. El Akkad, M. Merras, A. Saaidi, and K. Satori, “Fast 3D reconstruction and modeling method based on the good choice of image pairs for modified match propagation”, *Multimedia Tools and Applications*, vol. 79, no. 11–12, pp. 7159–7173, Mar. 2020, ISSN: 1380-7501, 1573-7721. DOI: 10.1007/s11042-019-08379-2.
- [28] FieldTrip toolbox contributors, *How are the different head and mri coordinate systems defined?*, 2021. [Online]. Available: [https://www.fieldtriptoolbox.org/faq/how\\_are\\_the\\_different\\_head\\_and\\_mri\\_coordinate\\_systems\\_defined/](https://www.fieldtriptoolbox.org/faq/how_are_the_different_head_and_mri_coordinate_systems_defined/).
- [29] Y. Chen, *Chenyuntc/simple-faster-rcnn-pytorch*, Jul. 2020. [Online]. Available: <https://github.com/chenyuntc/simple-faster-rcnn-pytorch>.



- [41] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, “Training Models of Shape from Sets of Examples”, in *Proceedings of the British Machine Vision Conference 1992*, Springer-Verlag London Limited, 1992, pp. 2.1–2.10, ISBN: 9783540197775. DOI: 10.5244/C.6.2. [Online]. Available: <http://www.bmva.org/bmvc/1992/bmvc-92-002.html>.
- [42] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, analysis, and machine vision*, Fourth edition. Cengage Learning, 2015, ISBN: 9781133593607.
- [43] J. Kybic, *Point Distribution Models*, 2007. [Online]. Available: <http://cmp.felk.cvut.cz/cmp/courses/33DZ0zima2007/slidy/pointdistributionmodels.pdf>.
- [44] O. Sorkine-Hornung and M. Rabinovich, “Least-Squares Rigid Motion Using SVD”, Jan. 2017. [Online]. Available: [https://igl.ethz.ch/projects/ARAP/svd\\_rot.pdf](https://igl.ethz.ch/projects/ARAP/svd_rot.pdf).
- [45] T. Cootes and C. Taylor, “Statistical Models of Appearance for computer vision”, pp. 12–23, Apr. 2004.
- [46] P. Besl and N. D. McKay, “A method for registration of 3-D shapes”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992. DOI: 10.1109/34.121791.
- [47] S. Gold, A. Rangarajan, C.-P. Lu, S. Pappu, and E. Mjolsness, “New algorithms for 2D and 3D point matching: pose estimation and correspondence”, *Pattern Recognition*, vol. 31, no. 8, pp. 1019–1031, Aug. 1998, ISSN: 00313203. DOI: 10.1016/S0031-3203(98)80010-1.
- [48] D. DeMenthon, P. And, and H. Samet, “SoftPOSIT: An Algorithm for Registration of 3D Models to Noisy Perspective Images Combining Softassign and POSIT”, May 2001.
- [49] M. W. Walker, L. Shao, and R. A. Volz, “Estimating 3-D location parameters using dual number quaternions”, *CVGIP: Image Understanding*, vol. 54, no. 3, pp. 358–367, Nov. 1991, ISSN: 10499660. DOI: 10.1016/1049-9660(91)90036-0.
- [50] Electrical Geodesics, Inc., *Sensor Digitization: Fast, accurate sensor digitization for source estimation and neuromodulation*, Brochure, Oct. 2016. [Online]. Available: [https://www.egi.com/images/stories/research/documents/Sensor\\_Digitization\\_Brochure\\_Oct\\_2016\\_MM\\_482.pdf](https://www.egi.com/images/stories/research/documents/Sensor_Digitization_Brochure_Oct_2016_MM_482.pdf).





## Appendix A

### Technical details

The implementation of the proposed algorithm is accessible on Gitlab<sup>1</sup> using your CTU account.

The repository contains the following directories

- [CameraCalibration](#) - implementation of camera calibration (5.1)
- [Config](#) - contains precomputed configuration files
- [Dataset](#) - contains dataset annotations and images
- [FasterRCNN](#) - implementation of the Faster R-CNN (4.1)
- [GeneratedData](#) - contains results of all training data
- [HelperScripts](#) - scripts used for dataset creation
- [PointDistributionModels](#) - implementation of all algorithms from 3D model reconstruction (6)
- [PointMatching](#) - implementation of corresponding points estimation (5.2)
- [TemplateMatching](#) - implementation of Template Matching (4.2)
- [VisualizationScripts](#) - scripts for visualization of predicted bounding boxes and correspondences

The algorithm can be run using the [run.py](#) file in the root directory. More detailed description can be found in [README.md](#).

---

<sup>1</sup><https://gitlab.fel.cvut.cz/olysarad/localization-of-eeg-electrodes-using-a-system-of-cameras>