FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

Fast and Accurate Automatic Localization of Anatomical Landmarks on Medical Images.

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Computational Intelligence

eingereicht von

Martin Trapp
Matrikelnummer 0925809

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung: o.Univ.-Prof. Dipl.-Ing. Dr. techn. Robert Sablatnig
Mitwirkung: Dipl-Math. Dr. Katja Bühler

Wien, 04.12.2013

(Unterschrift Verfasser) (Unterschrift Betreuung)

FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

Fast and Accurate Automatic Localization of Anatomical Landmarks on Medical Images.

MASTER'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Computational Intelligence

by

Martin Trapp
Registration Number 0925809

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: o.Univ.-Prof. Dipl.-Ing. Dr. techn. Robert Sablatnig
Assistance: Dipl-Math. Dr. Katja Bühler

Vienna, 04.12.2013

(Signature of Author) (Signature of Advisor)

Erklärung zur Verfassung der Arbeit

Martin Trapp
Pfeilgasse 5/19, 1080 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

(Ort, Datum) (Unterschrift Verfasser)

Acknowledgements

I would like to express my gratitude to my supervisors Prof. Dr. Robert Sablatnig and Dr. Katja Bühler for their support and advice. I also want to thank David Major and the rest of Biomedical Visualization group of the VRVis Research Center for the inspiring conversations and critical comments. Very special thanks go to Rene Donner from the Computational Imaging Research Lab of the Medical University of Vienna for providing annotations on Whole Body Morphometry Project images and helpful comments. I would also like thank my family, friends and my partner for their support and encouragement.

Abstract

Methods for medical image acquisition have rapidly evolved and the amount of digital images acquired in the daily image production of hospitals has exponentially increased in the last 30 years. Therefore, methods for the efficient automatic localization of anatomical landmarks on medical images need to be elaborated.

Recent publications addressing this problem use regression models. Despite convenient results the main shortcomings of most of these models are superfluous time and memory consuming computations. Inspired by the memory efficient random regression fern model, the aim of this thesis is to develop a novel regression model that allows to obtain accurate results with memory efficient computations.

Based on an exhaustive literature research on existing methods, regression based state of the art methods are analyzed to enable the development of a novel approach. The accuracy of this new approach is evaluated using K-fold cross validation on CT head scans, MRI T1 weighted head scans and CT whole body scans.

The proposed method achieves a mean deviation of $13.05mm$ on CT whole body scans in less than a minute.

The contribution of this thesis to the improvement of methods for the efficient automatic localization of anatomical landmarks on medical images is three-fold: (1) Two novel feature descriptors tailored to medical images are designed. One of the introduced image features (cuboidalBRIEF) outperforms all other tested feature descriptors. (2) A robust boosted regression model inspired random regression ferns is developed. The model stands out through its significantly higher accuracy as well as time and memory efficient computations. (3) A generalized multi-phase landmark location system allowing is presented. While the second phase results turn out to be less accurate than anticipated the first phase results of the system are highly satisfying.

Kurzfassung

Die Anzahl der täglich aufgenommen digitaler medizinischen Bilder ist durch die Weiterentwicklung bildgebender Verfahren exponentiell gestiegen. Effiziente Verfahren zur vollautomatischen Lokalisierung anatomischer Landmarken auf medizinischen Modalitäten gewinnen daher immer mehr an Bedeutung.

In akktuellen Publikationen werden zu diesem Zweck Methoden auf Basis von Regressionsmodellen verwendet. Trotz zufriedenstellender Resultate dieser Ansätze sind die angewandten Modelle meist zu speicher- und rechenaufwendig um einen hohen Durchsatz zu ermöglichen. Inspiriert durch das speichereffizientes random regression fern Verfahren wurde in dieser Diplomarbeit ein innovatives Regressionsmodell entwickelt, dass Speichereffizienz und exakte Lokalisierungsresultate kombiniert. Dieser neuartige Ansatze wurde auf Basis einer ausgiebigen Literaturrecherche über existierende Verfahren ausgearbeitet. Die abschließende Evaluierung erfolgte mittels K-fold cross validation auf CT und MRI T1 gewichteten Scans des Kopfes sowie auf CT Scans des ganzen Körpers.

Der Beitrag der vorliegenden Arbeit zu den aktuellen Forschungsbemühungen gliedert sich in drei Teile: (1) Es wurden zwei neue Merkmalsextraktoren für medizinische Modalitäten entworfen. Die Resultate aller getesteten Merkmalsextraktoren werden durch die Resultate eines der neuen Merkmalsextraktoren (cuboidalBRIEF) übertroffen. (2) Aufbauend auf das random regression fern Model wurde ein robustes Regressionsmodell entwickelt, welches sich sowohl durch Speicher- und Zeiteffizienz als auch durch gute Lokalisierungsresultate auszeichnet. (3) Im Zuge dieser Arbeit wird ein allgemeines Mehrphasenkonzept zur automatischen Landmarklokalisierung vorgestellt. Diese liefert drotz der geringen Verbesserung der Resultate durch die zweite Phase bereits in der ersten Phase sehr zufriedenstellende Resultate liefert.

Contents

CHAPTER 1

Introduction

In this chapter motivates the topic of this thesis and gives an overview of the state of the art. Subsequent the contribution of this thesis is discussed. Furthermore, the mathematical notation used in this thesis is introduced.

1.1 Motivation

Medical image acquisition methods continues to undergo rapid innovation and improvement of its use in medical diagnosis e.g., examination of the human body evolved from conventional film radiography imaging developed by Wilhelm Conrad Röntgen in 1895 to digital state of the art high throughput image production methods [5]. There has been an exponential increase of image data produced in the medical field during the last decade [55]. As an example the increase of acquired images at the Radiology Department of the Geneva University hospitals is shown in Figure 1.1.

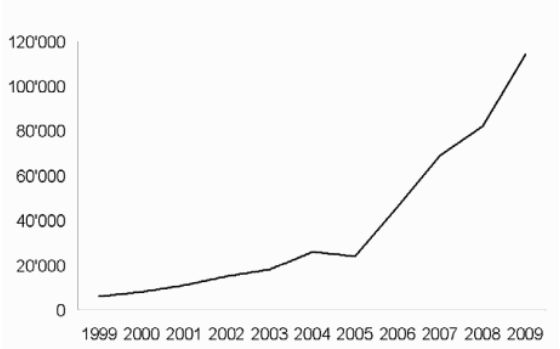


Figure 1.1: Daily image production in the Radiology Department of the Geneva University hospitals, by Müller et al. [55].

The integrated delivery system consortium Kaiser Permanente acquired about 700 TB of data by early 2009 and the University Hospital of Vienna reported in 2011, to produce $\approx$ 100 GB new images per day [20]. In 2010 the European Commission announced that 30% of the world storage capacity is occupied by medical images [78]. Deserno [20] states that the amount of images is expected to increase and therefore a pressing need to support the radiologists by automatically image analysis exists. According to Müller et al. [56], Content Based Image Retrieval (CBIR) on medical images to support the clinical decision-making were proposed by several articles. Rehman et al. [66] give a comprehensive survey on CBIR including CBIR in the medical domain. To approached an automatic retrieval of similar cases the localization of anatomical landmarks is a crucial aspect, according to Donner et al. [21].

1.2 Problem Statement

The aim of this thesis is to provide a reasonable fast and accurate anatomical landmark prediction system for Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) images. In this thesis anatomical landmarks are any kind of predefined, not necessarily anatomically important locations in an image. Even though the landmark prediction system is only evaluated on CT and MRI T1[1] weighted images, the method itself is a generic approach for landmark localization on large data using regression analysis. An introduction to CT and MRI images is given in Section 2.4.

The estimated anatomical landmark positions can act as a basis for morphometric measurements e.g., knee alignment angles [3, 27, 39, 81] or assessment of osteoporosis fractures [31, 34]. Moreover, segmentation approaches such as Active Shape and Appearance Models [13, 14], Graph Cuts [4] or Random Walks [30] require coarse initial positions. The landmark detection also serves as basis for landmark and image based registration methods [12, 44, 71].

1.3 State of the Art

This section gives an overview on literature in the field of landmark localization in medical images. The related work to this thesis is grouped into four classes: optimization based approaches, classification based approaches, marginal space learning based approaches and regression based approaches.

Optimization based approaches can be separated into heuristic optimization methods and discrete optimization methods. Cardillo et al. [10] and Inness et al. [38] propose genetic programming based approaches for landmark localization on medical images. However, there is not any further development of genetic programming methods for landmark localization. Until 2013, several publications on landmark detection approaches using discrete optimization techniques are published e.g., [21, 24, 25, 52, 70]. Several of those approaches e.g., Major et al. [52], Donner et al. [21], use a mixture of discrete optimization and machine learning techniques such as classification or regression to acquire accurate results in an reasonable amount of time.

[1] To capture different anatomical structures, MRI is usually generates T1 or T2 weighted images.

Classification using machine learning can be approached by discriminative and generative models (cf. [57]). Iglesias et al. [37] propose a combination of generative and discriminative classification models for automatic segmentation of anatomic structures on CT Scans. A random forest classification approach for lung detection and segmentation is proposed by Montillo et al. [54].

Until 2013, several publications on landmark localization methods using regression models for estimating displacement vectors to the landmark positions have been released. One of the first efficient approaches using random forest regression on CT volumes were published by Criminisi et al. [15]. Later on the proposed approach were generalized for organ segmentation on CT series [16] and several related methods for medical image data are published e.g., [11, 18, 21]. Furthermore, Pauly et al. [64] propose a memory efficient alternative method on MR dixon sequences by using random fern regressors. Donner et al. [21] use a hough forests based two step approach for accurate landmark localization on CT images.

In addition to the classification and regression approaches there exists work on marginal space learning for localization of anatomical landmarks by Zheng et al. [83, 85]. The proposed method were patented by Siemens Corporate Research in 2012 [84].

1.4 Contribution

This work presents a generalized approach for anatomical landmark prediction on medical images. Moreover, an evaluation of the current state of the art binary feature descriptors tailored to medical images is given. Two novel binary feature descriptors are presented. The recently published approaches on landmark localization use regression analysis. Therefore, a robust boosted extension of the random fern regression [64] combined with a multi-step procedure allowing fast and accurate landmark detection in large data is presented. Finally, this work gives an evaluation of the method described on CT and MRI T1 weighted images.

1.5 Structure of the Thesis

The remainder of this thesis is structured as follows. Chapter 2 introduces basic concepts that are used in this thesis. Chapter 3 presents the novel feature representations which have been developed in the context of this thesis and Chapter 4 introduces the novel regression model used for the landmark predictions.

A detailed discussion of the evaluation measurements, methods and an explanation of the results are given in Chapter 6. Chapter 7 gives a discussion of the results and explains possible future work. Additional figures and the source code of the feature descriptors developed for this thesis can be found in the Appendix.

1.6 Mathematical Notation

The following mathematical notation is used in this thesis.

$\mathbb{B}$	The set $\{0, 1\}$.
$\mathbb{R}$	The set of real numbers.
$\mathcal{N}(\mu, \sigma^2)$	A normal distribution with mean μ and variance σ^2.
pdf	The probability density function of a probability distribution.
$\mathrm{E}[x]$	The expectation value of a random variable x.
$P(\mathbf{x})$	The probability of random event $\mathbf{x}$.
$P(\mathbf{x}\|\mathbf{y})$	The conditional probability of random event $\mathbf{x}$ if $\mathbf{y}$ occurred.
$\mathbf{I}$	The identity matrix.
$\mathbf{0}$	A vector of zeros.
$h(\mathbf{x})$	A hypothesis function.
$L(h(\mathbf{x}))$	Model loss of a hypothesis function.
ϵ	Random noise or error value.
$\mathbf{x}^i$	The ith vector of independent input variables or a multivariate feature response.
x^i_j	The jth component of the ith vector $\mathbf{x}^i$.
$\mathscr{X}$	The space of independent input variables.
$\mathbf{y}^i$	The ith vector of conditional output variables.
$\hat{\mathbf{y}}$	The prediction result of conditional output variables.
y^i_j	The jth component of the ith vector $\mathbf{y}^i$.
$\mathscr{Y}$	The space of conditional output variables.
$\mathscr{D}$	A training / testing data set.
I	A volumetric image.
$\mathbf{p}$	A position vector on I.
$\langle \mathbf{x}, \boldsymbol{\beta} \rangle$	The inner product of vector $\mathbf{x}$ and vector $\boldsymbol{\beta}$.
$\mathbf{x} \circ \boldsymbol{\beta}$	The component wise product of vector $\mathbf{x}$ and vector $\boldsymbol{\beta}$.
$N_{\mathbf{p}}$	The neighborhood of a position vector.
$\#N_{\mathbf{p}}$	The number of elements in $N_{\mathbf{p}}$.
$I(\mathbf{p})$	The intensity function applied on I.
P	A partition of a space.
C	A cell of a partition.
F	A fern of an ensemble of random ferns.
IG	Information gain.
H	Information entropy.
β	Regression model parameters.
ω, γ	Weighting parameters.
ν	Confidence value of hypothesis function.
η, θ	Threshold parameters.

CHAPTER 2

Preliminaries

This chapter reviews the basic concepts used in this thesis. General concepts on regression analysis are given in Section 2.1 and 2.2. Furthermore, basic concepts that are related to discrete cross-sectional images and medical image modalities are given in Section 2.3 and 2.4.

2.1 Linear Regression Analysis

Regression analysis is a statistical technique for estimating the relationship between independent and conditional variables. Linear regression analysis assumes that the relationship is based on a linear oracle function $o(\mathbf{x})$ and tries to model this relationship based on sampled training data. Formally, the independent variables are denoted by $\mathbf{x}$ and the conditional variable is denoted by y. Moreover, we want to estimate the hypothesis function $h^*(\mathbf{x})$ which is the optimal approximation of $o(\mathbf{x})$.

Let a scalar conditional variable y considered to be linear in $\mathbf{x} = [x_1, \ldots, x_{D^x}]^\top$. Moreover, the differences of y around the expectation value $\mathrm{E}[y|x_1, \ldots, x_{D^x}]$ are assumed to be additive and Gaussian distributed. According to Hastie et al. [33] the linear regression model can therefore be formulated by

$$h(\mathbf{x}) = \beta_0 + \sum_{j=1}^{D^x} x_j \beta_j + \epsilon \tag{2.1}$$

where $\boldsymbol{\beta}$ is a the vector of unknown model coefficients and ϵ is additive and zero-mean Gaussian distributed noise.

Furthermore, suppose that the conditional vector $\mathbf{y} = [y_1, \ldots, y_{D^y}]$ depends on $\mathbf{x}$ and a linear model for each y_i can be assumed. Equation 2.1 can then be extended to the multiple multivariate linear regression model:

$$y_i = \beta_{0,i} + \sum_{j=1}^{D^x} x_j \beta_{j,i} + \epsilon \tag{2.2}$$

The linear regression model can be rewritten in matrix notation as:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E} \tag{2.3}$$

The model coefficients of a multiple multivariate linear regression model can be found by minimizing the least squares error

$$\beta = \underset{\beta \in \mathbb{R}}{\operatorname{argmin}} \sum_{i=1}^{N} \sum_{j=1}^{D^y} (y_j^i - \boldsymbol{\beta}_j \mathbf{x}^i)^2. \tag{2.4}$$

The final model can then be used to predict the corresponding conditional variables on unseen independent variables $\mathbf{x}$.

2.2 Generalized Regression Analysis

However, in real world problems the relation between $\mathbf{y}$ and $\mathbf{x}$ is typically not linear. In the case of non-linear regression models we augment the input vector $\mathbf{x}$ with $1, \ldots, M$ transformations of $\mathbf{x}$. We can thereby model the relationship between $\mathbf{y}$ and $\mathbf{x}$ as a linear basis expansion in $\mathbf{x}$.

$$y = \sum_{m=1}^{M} t_m(\mathbf{x}) \boldsymbol{\beta}_m \tag{2.5}$$

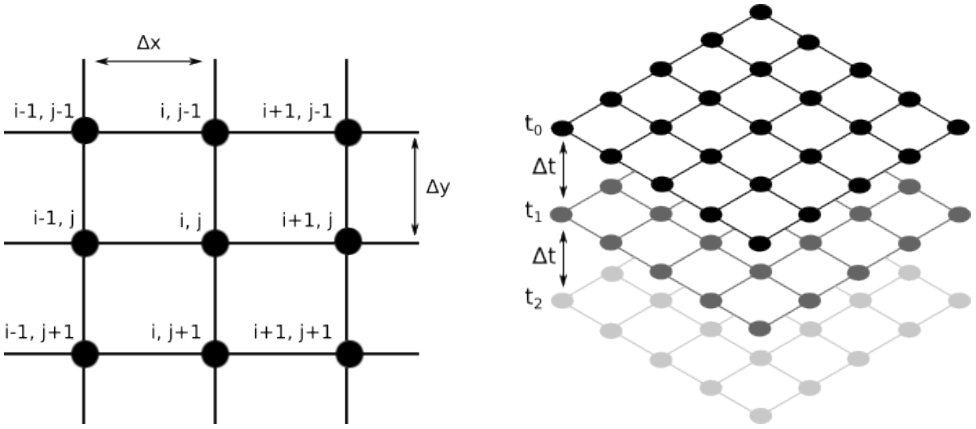
The transformations $t_m(\mathbf{x})$ can be linear, polynomial, non-linear transformations of single or multiple inputs or indicator functions for piecewise linear models. In the following, the transformations will be similar to indicator functions for piecewise linear models.

Note that the estimation of the model coefficients for non-linear regression models is not trivial and the underlying basis functions are unknown.

2.3 Cross-Sectional Volumetric Images

A three-dimensional cross-sectional image I is made up of a matrix of size $\mathrm{I}_X \times \mathrm{I}_Y \times \mathrm{I}_Z$ where $I(\mathbf{p})$ denotes the intensity value on I at position $\mathbf{p}$. Each cross-sectional image is represented by a stack of parallel two-dimensional cross sections (planes) through the patient tissue. Therefore, each plane can be considered to be a rectangular image represented by a matrix of size $\mathrm{I}_X \times \mathrm{I}_Y$ with equal horizontal Δx and vertical picture element (pixel) spacings Δy. Figure 2.1a illustrates a two-dimensional image in terms of a uniform grid with origin on the left upper corner. Each pixel holding an intensity values in the discrete image is illustrated by a grid point on the lattice.

However, the planes of a cross-sectional image do not necessarily have to be equidistant and the distances between the planes can differ from the horizontal and vertical pixel spacing within the planes. Therefore, each volumetric element (voxel) $\mathrm{I}_{x,y,z}$ additionally holds the properties width, height and depth to the underlying intensity values. Figure 2.1b illustrates a cross-sectional image as a stack of two-dimensional slices, each voxel is visualized by a grid point on the three-dimensional lattice. Inspired by the general notation of three-dimensional data, the



(a) Illustration of 2D uniform grid.

(b) Illustration of cross-sectional grid.

Figure 2.1: Illustration of image data on a discrete grid in the x, y and t domain. Figures created by author.

third-dimension is illustrated in the time domain with slice spacing Δt. Please note that Figure 2.1b shows a equidistant stack which is not necessarily the case for general cross-sectional images.

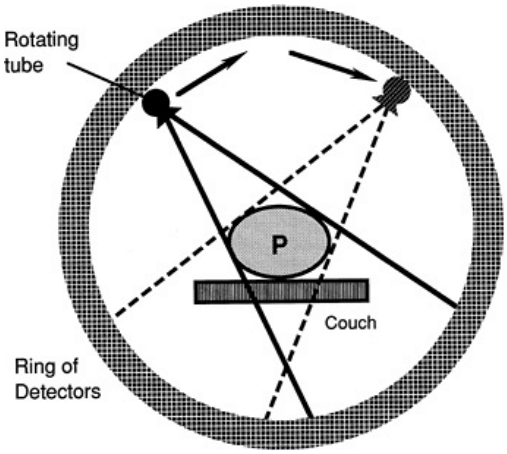
2.4 Image Modalities

This section briefly explains two relevant cross-sectional imaging techniques. Both methods produce three-dimensional volumetric images of living patients in the form of multiple two-dimensional slices. The resulting volumetric image consists of voxels, each representing the patient tissue at a given position.

2.4.1 Computed Tomography

CT is a medical imaging technique that uses slice-wise measurements of X-ray transmissions through a patient to reconstruct a cross-sectional image of axial planes. Measurements of narrow X-ray beams from different directions will be used to construct the image. An illustration of a CT scanner is shown in Figure 2.2a.

An algorithm assigns intensity values in Hounsfield unit scale to the captured measurements of the transmitted X-rays. The Hounsfield unit scale ranges from $-1024HU$, corresponding to air, up to $3072HU$ for very dense bone. Therefore, CT voxel intensity values are well-defined and directly correspond to the underlying tissue type. However, the actual intensity values of the tissues vary from one CT scanner to another and the intensities of the images are therefore not directly comparable. An exemplary whole body CT scan of a patient is shown in Figure 2.2b.



(a) CT scanner with patient P placed on the examination table, by Brand and Helms [5].

(b) Exemplary visualization of a CT scan in the coronal plane, taken from the Whole Body Morphometry Project [58]. Figure created by author.

Figure 2.2: Illustration of a CT scanner (a) and an exemplary visualization of CT scan (b).

2.4.2 Magnetic Resonance Imaging

In contrast to CT, which only evaluates X-ray absorption, MRI allows to evaluate multiple tissue characteristics. Brant and Helms [5] give a detailed discussion on the tissue characteristic that can be captured by MRI. To evaluate the landmark localization on MRI images the MRI T1 weighted scans acquired by Stephan Wolfsberger from the Medical University of Vienna, in the context of the development of an intra-operative virtual endoscopy system by Schulze et al. [72], have been used. However, only MRI T1 weighted and magnetic resonance angiography (MRA) data has been acquired during the studies. Therefore, only T1 weighted MRI images will be discussed in this thesis. T1 is a measurement of how quickly a tissue can get magnetized. In brain images T1 relaxation times are useful to distinguish between white matter and grey matter. White matter is mostly used to transmit signals from one region to another whilst grey matter contains neural cell bodies which are not contained in white matter. Figure 2.3 illustrates the visual difference between white and grey matter on MRI image of the human brain. Areas marked with (1) and (2) are regions of grey matter and the area marked with (3) is a region of white matter.

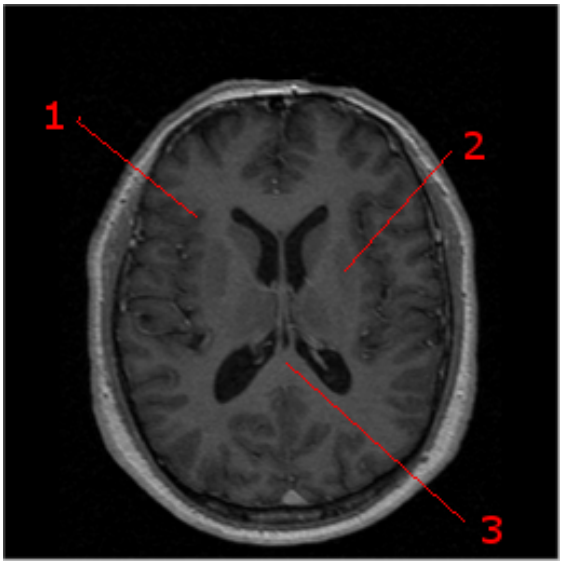


Figure 2.3: Illustration of a MRI T1 weighted scan in the axial plane, Figure created by author. Areas (1) and (2) illustrate regions of grey matter whilst area (3) illustrates regions of white matter.

Another advantage of MRI is that the images can be obtained in any arbitrary plane. However, in contrast to CT the intensity values vary greatly from image to image and therefore need to be normalized afterwards if a correspondence between images based on the intensity values needs to be established. The MRI images of the human brain used for this thesis are affine registered to a common image but not intensity normalized. However, the feature descriptors presented in Chapter 3 do not require pre-normalized images.

2.5 Summary

One possibility to estimate the relationship between independent and conditional variables is linear regression. The optimal parameters of a linear model are found by minimizing the least squares error of the model (cf. Equation 2.4). By augmenting the independent variables with transformations of them self, the regression model can be generalized for non-linear problems.

This thesis concerns about medical imaging techniques that reconstruct cross-sectional images. A three-dimensional cross-sectional images is defined by a grid of intensity values. The spacings between the grid elements are defined by horizontal, vertical and slice spacings. In the medical domain those cross-sectional images are not necessarily equidistant.

To obtain cross-sectional images from patient tissue this thesis focuses on the imaging techniques CT and MRI. CT images contain intensity values ranging from $0HU$ to $3000HU$ and have a well defined correspondence to the underlying tissue category. In MRI the correspondence is not well defined and the images produced depend on the weighting of the scans. This this focuses on T1 weighted scans of the human brain which allow the examination of grey matter, white matter and tumor tissue.

CHAPTER 3

Feature Representation

In order to allow the statistical analysis of image data the intensity values for the voxels of the volumetric image have to be transformed into feature responses. In image processing an image feature response denotes information about specific structures of the underlying image. This structures can range from simple features like edges or corners to complex features such as curvature, structure tensor [43], texture information or binary tests between voxels or image regions. Jähne [40] gives a comprehensive discussion of different types of feature representation for digital image processing.

In contrast to high level features like structure tensor, Gabor filter [19] or Tamura [74] in this thesis two less computationally expensive approaches tailored to medical images are used and evaluated. The first approach is based on the combination of Local Binary Patterns (LBP) on 3D asymmetric cuboidal regions [64] and a Multivariate Gaussian LBP feature by Donner et al. [23]. The second approach combines the idea of LBP on cuboidal regions with the Gaussian distributed variant of the Binary Robust Independent Elementary Feature (BRIEF) [9].

In the following sections LBP, LBP on 3D asymmetric cuboidal regions and BRIEF are explained. Furthermore, the two novel feature descriptors used in this thesis are introduced in Section 3.1 and 3.2. Implementation details on the proposed feature descriptors are given in Section refsec:featureImplementation. An evaluation of the feature descriptors is given in Chapter 7.

3.1 Gaussian Distributed Binary Tests on Cuboidal Regions

The LBP are effective descriptors of the textural neighborhood of pixels in an image by using binary derivations of the intensity values [49]. Moreover, LBP benefit from their computational simplicity and discriminative power [65]. The following subsections give an introduction to the previous work and introduce the novel feature GaussLBP.

3.1.1 Local Binary Pattern (LBP)

The textural feature LBP was first described by Ojala et al. [59] as an extension of the textural units published by Wang and He [80]. In case of the LBP, the 3×3 neighborhood of a pixel location $\mathbf{p}$ is thresholded by the intensity value pixel $\mathbf{p}$ and each resulting neighbor response $n = 1, \ldots, 8$ is multiplied by a exponential weight $\omega_n = 2^{n-1}$. Figure 3.1 illustrates the approach of Ojala et al. [59] using the 3×3 neighborhood around the center pixel with intensity value six. Finally, the sum of the multiplied values is obtained.

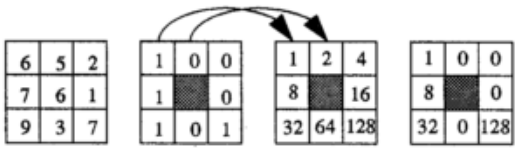


Figure 3.1: Illustration of local binary patterns, by Ojala et al. [59].

Moreover, Ojala et al. [60] presente an arbitrary circular derivation for the LBP. Therefore, the symmetric circular neighborhood is defined by $p > 0$ neighbors with equally spread pixel spacings and a radius $r > 0$. The displacement coordinates of the neighboring pixel n are given by:

$$[x, y] := \left[r \cos \frac{2\pi n}{p}, -r \sin \frac{2\pi n}{p} \right] \tag{3.1}$$

The symmetric circular neighborhood is then captured with the same thresholding and reduction steps as described above. Because of the implicit dimension reduction the LBP with a 3×3 neighborhood can be efficiently stored using an unsigned 8-bit integer [49]. Therefore, Fehr and Burkhardt [28] present a variant of the LBP for volumetric data that works on voxels instead of pixels. Moreover, 3D LBP is shown to be an effective image feature on medical image data for search and retrieval tasks [8, 65].

3.1.2 LBP on Asymmetric Cuboidal Regions

Pauly et al. [64] propose to use a textural rich adaption of the 3D LBP by considering asymmetric cuboidal regions. Instead of a single center voxel on voxel location $\mathbf{p} \in \mathbb{R}^3$ a 3D region $R^s_{\mathbf{p}}$ at scale s centered on $\mathbf{p}$ is used. In order to capture the neighborhood of $\mathbf{p}$ the K neighbors $N^s_{\mathbf{p}} := \{R^s_{\mathbf{q_1}}, \ldots, R^s_{\mathbf{q_K}}\}$ at voxel positions $\mathbf{q}_i \in \mathbb{R}^3$ with 3D regions of different sizes, orientations, and offsets are extracted. The different sizes, orientations and offsets are chosen randomly. Moreover, let the function BV be defined as:

$$BV(a, b) = \begin{cases} 1, & \text{if } a < b \\ 0, & \text{else} \end{cases} \tag{3.2}$$

And the mean intensity over any region $R^s_{\mathbf{p}}$ on a scale s from any voxel location $\mathbf{p}$ is defined as

$$IM(\mathbf{p}) = \frac{1}{\#R^s_{\mathbf{p}}} \sum_{\mathbf{q} \in R^s_{\mathbf{p}}} I(\mathbf{q}) \tag{3.3}$$

where $I(\mathbf{q})$ is the intensity value on voxel location $\mathbf{q}$. Therefore, the LBP on 3D asymmetric cuboidal regions for K neighbors is defined as:

$$LBP(\mathbf{p}) := \sum_{i=1}^{K} 2^{i-1} BV\left(IM(\mathbf{q}_i), IM(\mathbf{p})\right) \tag{3.4}$$

Figure 3.2 shows a 2D example with six neighbors (yellow rectangles) of the voxel location $\mathbf{p}$ and the rectangle $R^s_{\mathbf{p}}$ (red rectangle).

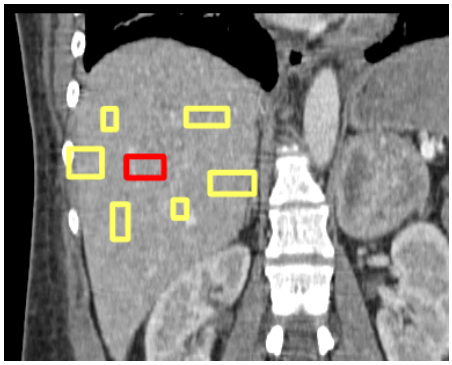


Figure 3.2: LBP on 3D Asymmetric Cuboidal Regions (2D example). Figure created by author.

The proposed adaption benefits from the increased number of degrees of freedom i.e., nine times K, combined with random sampling. Using regions instead of single voxels also results in a less noise sensitive feature description and emphasizes textural information to be captured. In order to reduce the dimensionality of the resulting binary vector Pauly et al. [64] propose to use the same weighting scheme as used for LBP. As described in [64] this feature descriptor turns out to be efficient for MRI Dixon sequences. In contrast to common MRI, Dixon published an imaging technique for water and fat separation. Ma [50] gives further details on MRI Dixon sequences. To benefit from the separation, the feature response will be computed over the water and fat channel of the image.

3.1.3 Extending LBP on Gaussian Distributed Cuboidal Regions

Inspired by the multivariate Gaussian distributed binary test feature proposed by Donner et al. [21] and the good results achieved with the LBP on asymmetric cuboidal regions a combination of both features is presented is this thesis.

In contrast to the common LBP, Donner et al. [21] propose to compute a binary vector of gray value differences between the voxel location and K randomly chosen neighbors. In order to reinforce the local intensity changes a multivariate Gaussian distribution is used to generate the neighborhood offsets. Combining the feature representation described with the idea of cuboidal

regions leads to the novel feature descriptor Gaussian distributed binary tests on cuboidal regions (GaussLBP).

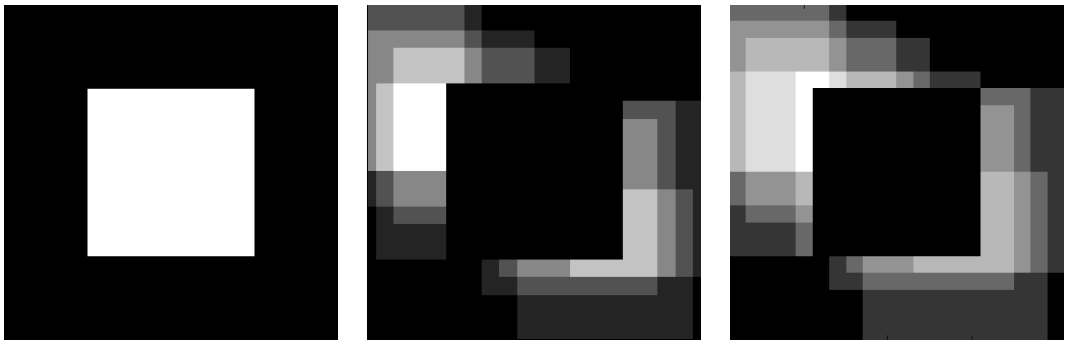
Similar to Equation 3.4 let us consider voxel location $\mathbf{p} \in \mathbb{R}^3$ and K neighbors $N^s_{\mathbf{p}} := \{R^s_{\mathbf{q_1}}, \ldots, R^s_{\mathbf{q_K}}\}$ located at voxel locations $\mathbf{q}_i \in \mathbb{R}^3$ with their rectangular 3D regions with different sizes, orientations and offsets. The proposed K dimensional feature vector is described by:

$$GaussLBP(\mathbf{p}) := \begin{pmatrix} BV(IM(\mathbf{q}_1), I(\mathbf{p})) \\ \ldots \\ BV(IM(\mathbf{q}_K), I(\mathbf{p})) \end{pmatrix} \in \mathbb{B}^K \tag{3.5}$$

Using this description allows to combine the benefit of capturing local intensity changes with the benefit of capturing textural information by using a larger number of degrees of freedom. As can be seen in comparison to Equation 3.4 for the Gaussian distributed variant only the intensity value of the voxel location $\mathbf{p}$ is used instead of the mean intensity over a randomly defined cuboidal region around $\mathbf{p}$. By using this simplification the degrees of freedom are reduced but the property of capturing the surrounding textural information will be kept.

In contrast to the LBP based approach of Pauly et al. [64] reducing the binary vector by using their decimal representation is not used for this approach. Instead, the feature is represented by the computed binary vector which is similar to the approach proposed by Donner et al. where ($K = 100$) neighbors are used to describe the texture. Furthermore, the representation of 100-dimensional binary vectors as decimal values is not feasible using Matlab as the largest positive value that can be represented for 64-bit unsigned integers is $2^{64} - 1$.

A visualization of exemplary feature responses of the feature descriptor proposed by Donner et al. [21] and the modified feature descriptor using 3D cuboidal regions on a synthetic 3D image and a real CT full body scan are shown on Figure 3.3 and Figure 3.4. The visualizations show one slice of the volumetric datum and in case of the feature responses the voxel intensities correspondent to the sum of the binary vector computed in voxel location.



(a) (b) (c)

Figure 3.3: Middle XY-Slice of: synthetic 3D image (a), feature response by Donner et al. [21] (b), feature response using GaussLBP (c). Figures created by author.

(a) (b) (c)

Figure 3.4: 215th axial plane of a downsampled CT full body scan (a), taken from the Whole Body Morphometry Project [58]. Feature response by Donner et al. [21] (b) and feature response using GaussLBP (c). Figures created by author.

It can be seen that the proposed descriptor gives a smoother and more global response over the image than the feature descriptor by Donner et al. [21] which results in a better partitioning of the feature space, see Chapter 7. On the other hand GaussLBP suppresses small local differences which might lead to an unintentional loss of information. Figure 3.4 shows an exemplary case of GaussLBP with $K = 15$ neighbors sampled from a multivariate Gaussian distribution $\mathcal{N}(\mathbf{p}, \Sigma)$ with $\Sigma = \mathrm{I} * 10$ and I is the 3×3 identity matrix. The regions are sampled from a uniform distribution with $\mu = 0$ and $\sigma^2 = 5$. It can be seen that GaussLBP is with this configuration too rough to capture small structures for the $127 \times 127 \times 430$ image. It is important to note that for simplicity the implementation of GaussLBP given in Appendix A.1 computes the sum of intensity values inside a region bidirectional using the same offset. Therefore, the cuboidal region dimensions are twice the sampled values which leads in the case of Figure 3.4 to region dimensions of up to $10 \times 10 \times 10$.

3.2 Binary Tests on Cuboidal Region Pairs (cuboidalBRIEF)

An alternative to the common LBP for 2D images was proposed by Calonder et al. [9] in 2010. As described by Heinly et al. [35] who evaluated the performance of several binary features in terms of robustness against each other, BRIEF outperforms other binary feature descriptors for keypoint recognition. The following subsections discuss BRIEF in detail and introduce the novel feature cuboidalBRIEF which extends BRIEF by using cuboidal regions.

3.2.1 Binary Robust Independent Elementary Features (BRIEF)

BRIEF follows the assumption that image patches can be classified using 512 pairwise binary tests inside a region [35]. In contrast to LBP which computes point wise descriptions of the neighboring texture the BRIEF descriptor computes a decimal representation for a region located at center pixel $\mathbf{p} \in \mathbb{R}^3$ of size $S \times S$ by evaluating pair wise differences of tuples $(\mathbf{q}', \mathbf{q}'')$ with

$\mathbf{q}', \mathbf{q}'' \in \mathbb{R}^2$ inside the region. BRIEF is therefore defined as:

$$BRIEF(\mathbf{p}) := \sum_{i=1}^{K} 2^{i-1} BV(I(\mathbf{q}_i'), I(\mathbf{q}_i'')) \tag{3.6}$$

Moreover, Calonder et al. [9] evaluated different neighborhood functions to select the test locations $(\mathbf{q}_k', \mathbf{q}_k'')$. Calonder et al. [9] propose to choose the test locations in BRIEF using $(\mathbf{Q}', \mathbf{Q}'') \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$. The tests are sampled from an multivariate Gaussian distribution with mean on the region center $\boldsymbol{\mu} = \mathbf{p}$ and the covariance $\Sigma = \mathrm{I} * \frac{1}{25} S^2$ where I is the identity matrix. Moreover, Calonder et al. [9] propose to use 128, 256 or 512 sampled tests for the BRIEF descriptor.

3.2.2 Extending BRIEF with cuboidal regions

Influenced by the BRIEF descriptor and the proposed GaussLBP an extension of BRIEF using cuboidal region pairs is implemented. In contrast to BRIEF each pair wise binary test $(\mathbf{q}', \mathbf{q}'')$ is augmented by the intensity mean (cf. Equation 3.3) calculated over the regions around $\mathbf{q}'$ and $\mathbf{q}''$. The feature response of the binary test feature on cuboidal region pairs (cuboidalBRIEF) for a region located on voxel location $\mathbf{p} \in \mathbb{R}^3$ of size $S \times S \times S$ with K pair wise tests is defied as:

$$cuboidalBRIEF(\mathbf{p}) := \sum_{i=1}^{K} 2^{i-1} BV(IM(\mathbf{q}'), IM(\mathbf{q}'')) \tag{3.7}$$

Similar to LBP on asymmetric cuboidal regions by Pauly et al. [64] the dimensions of the cuboidal regions are sampled from an uniform distribution. Moreover, the proposed extension implicitly solves the noise-sensitivity issue of BRIEF described by Calonder et al. [9] and additionally captures the textural information around each test pair. In contrast to the 128, 256 or 512 binary test used for BRIEF in this thesis only a small amount of binary tests have been used. This is done to reduce the amount of required memory and to keep the computation of the decimal representation feasible.

3.3 Implementation Details

The feature descriptors proposed, GaussLBP and cuboidalBRIEF, depend on the computation of summed cuboidal regions. In order to efficiently compute those sums a summed-area table also known as integral image is used.

3.3.1 Summed-Area Tables

The summed-area table has first been published in 1984 by Crow [17] and became prominent in 2001 through Viola and Jones [79]. The basic idea is to replace the intensity values on an image by a value that represents the sum of the intensities of all pixels above and to the left of the pixels. More formally the value at pixel location (x, y) is defined as:

$$T(x,y) = \sum_{x'=x_0}^{x} \sum_{y'=y_0}^{y} I(x',y') \tag{3.8}$$

Representing an image by the summed-area table allows to compute the sum of intensities over an arbitrary region in constant time by evaluating

$$S(x,y,w,h) = T(x,y) + T(x-w,y-h) - T(x,y-h) - T(x-w,y) \tag{3.9}$$

where (x,y) denotes the lower right corner of a rectangle with width w and height h. In 2005 Ke et al. [41] generalized the summed-area table for volumetric data. The summed-area table for volumes will be computed by:

$$T(x,y,z) = \sum_{x'=x_0}^{x} \sum_{y'=y_0}^{y} \sum_{z'=z_0}^{z} I(x',y',z') \tag{3.10}$$

In order to retrieve the sum of intensities over an arbitrary cuboidal region the eight region corners have to be evaluated. Tapia [75] generalizes the computation of a sum of intensities for a region given a n-dimensional integral image with the equation

$$S := \sum_{\mathbf{b} \in \mathbb{B}^n} (-1)^{n-|\mathbf{b}|_1} T(\mathbf{p}^{\mathbf{b}}) \tag{3.11}$$

where $\mathbf{b}$ is a binary vector from the set of binary values $\mathbb{B}$ and $\mathbf{p}^{\mathbf{b}}$ corresponds to the region corners e.g., $\mathbf{p}^{\{0,0,0\}}$ corresponds to corner $I(x_0, y_0, t_0)$ in Figure 3.5 and $\mathbf{p}^{\{0,1,0\}}$ corresponds to corner $I(x_0, y_1, t_0)$.

In the implementation of summed-area tables an additional padding is necessary such that single voxels on the edges of the volume can also be evaluate using Equation 3.3.1. Moreover, due to the commutative computation of the summed-area table the numerical error accumulates over the table. In this thesis the summed-area tables are therefore always computed using double precision values.

The pseudo codes of GaussLBP and cuboidalBRIEF are depicted in Appendix A.1 and A.2. Both implementations use summed area tables to efficiently estimate the intensity mean inside a region.

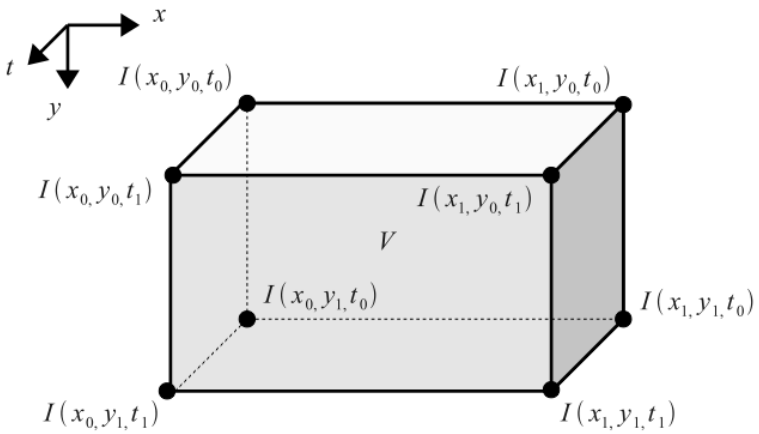


Figure 3.5: Illustration of a region in a 3D summed-area table, by Tapia [75].

3.4 Summary

Inspired by textural rich adaption of the 3D LBP by Pauly et al. [64] and the feature descriptor by Donner et al. [21] this thesis derives the feature descriptor GaussLBP. In the notion of LBP GaussLBP uses binary derivations of the sum of intensity values of randomly sampled cuboidal regions around center voxel. The response is computed by applying Equation 3.5. In contrast to LBP the binary vector obtained using GaussLBP is not represented by its decimal value, as the number of binary tests are too many, $K = 100$.

In addition to GaussLBP, the BRIEF feature descriptor is extended in a similar manner. As BRIEF describes image patches using binary tests, the extension cuboidalBRIEF describes sub volumes using binary tests between cuboidal regions using Equation 3.7. In order to efficiently compare the sum of intensity values of two regions the n-dimensional generalization by Tapia [75] of the summed-area table is used for GaussLBP and cuboidalBRIEF.

CHAPTER 4

Regression Model

Several articles present methods to localize anatomical landmarks on medical images by applying machine learning techniques on extracted image features. Especially the use of multiple multivariate regression analysis to predict landmark position is proposed by several publications until 2013 e.g., [11, 15, 16, 18, 21, 23, 64]. Inspired by the, in comparison to [16, 21], less accurate but memory efficient random regression ferns by Pauly et al. [64] a boosted variant of the random regression fern approach is introduced.

Section 4.1.1 introduces the basic concepts of random ferns for classification problems. In Section 4.1.3 the random regression ferns approach is described. The first derivation of boosted random regression ferns are given in Section 4.2. In the last section the extension of robust boosted random regression ferns, which are used for the anatomical landmark prediction, are introduced.

4.1 Random Regression Ferns

The random fern classifier is a memory-efficient alternative to random forest and were first published by Özuysal et al. [62]. The idea of using random ferns for key point recognition is based on the observation that image patches can be recognized on the basis of randomly chosen binary tests. Lepetit et al. [46] use simple, randomly chosen binary tests in combination with decision trees (random trees). However, as decision trees are known to overfit the training set and therefore result in a high generalization error, pruned trees or ensembles of trees are used, cf. Rokach [67]. Therefore, Lepetit et al. [46] propose to use an ensemble of random trees and to average the votes of the individual random trees.

4.1.1 Random Ferns for Classification

Random ferns classify test observations based on the binary features in a naïve Bayesian manner. The aim of classification is to assign an unknown observation $\mathbf{x}$ to a class of a predefined set of classes. The relation between input observations and classes will be learned during the training

of the model. Formally, the ith feature consisting of D low-level features obtained using a feature descriptor e.g., first eigenvalue of an eigenvalue decomposition of the Hessian matrix, is defined by $\mathbf{x}^i \in \mathscr{X}^D$ where $\mathscr{X}^D \subset \mathbb{R}$. Furthermore, $y^i \in \mathscr{Y}$ denotes the ith class. Therefore, the training dataset $\mathscr{D}_{train}$ for the supervised learning problem over N training observations can be formulated as follows:

$$\mathscr{D}_{train} := \{(\mathbf{x}^1, y^1), \ldots, (\mathbf{x}^N, y^N)\} \subset \mathscr{X}^D \times \mathscr{Y} \tag{4.1}$$

Model Derivation. Given an unseen observation with feature $\mathbf{x}$. In order to classify $\mathbf{x}$ the maximum a posterior probability is evaluated as follows:

$$\hat{y} := \underset{y \in \mathscr{Y}}{\operatorname{argmax}} P(y|\mathbf{x}) \tag{4.2}$$

According to Bayes' theorem $P(y|\mathbf{x})$ can be rewritten as follows:

$$P(y|\mathbf{x}) = \frac{P(\mathbf{x}|y)P(y)}{P(\mathbf{x})} \tag{4.3}$$

By assuming the prior to be positive uniform and since the denominator is a scaling factor that is independent from the class, the problem reduces to the following equation:

$$\hat{y} := \underset{y \in \mathscr{Y}}{\operatorname{argmax}} P(\mathbf{x}|y) \tag{4.4}$$

Similar to Lepetit et al. [46], Özuysal et al. [62] use weak binary features depending on the pixel intensity values. In order to strengthen the discriminative power Özuysal et al. [62] use $\mathbf{x}^i := [x^i_1, \ldots, x^i_D]^\top$ with $D \approx 300$. As computing the joint probability (cf. Equation 4.4) for a large number of binary features is not feasible, Özuysal et al. [62] assume complete independence between the features and therefore reduce the problem. In order to keep the correlation between features but still have a traceable problem Özuysal et al. [62] partition the features into M groups of size $S = \frac{D}{M}$ called *ferns*. Please note that the groups itself are modeled to be independent from each other, which leads to a naïve Bayesian formulation of the model. Therefore, the conditional joint probability over all *ferns* can be computed as follows

$$P(\mathbf{x}|y) = \prod_{k=1}^{M} P(F^k|y) \tag{4.5}$$

where $F^k := \{x_{\sigma(k,1)}, x_{\sigma(k,2)}, \ldots, x_{\sigma(k,S)}\}$ represents the kth fern and $\sigma(k, s)$ is a random permutation function with range $1, \ldots, D$. It can be seen that the problem complexity is reduced from 2^D, which is not traceable for large D, to $M \times 2^S$. This reduction makes the problem traceable while some of the dependencies between the features are still modeled.

4.1.2 Comparison to Random Forest

Random forest is an ensemble of independent trained random decision trees and was first introduced by Breiman [7]. For classification problems, the training set $\mathscr{D}_{train}$ is separated into individual subsets using bagging to improve the generalization and robustness.

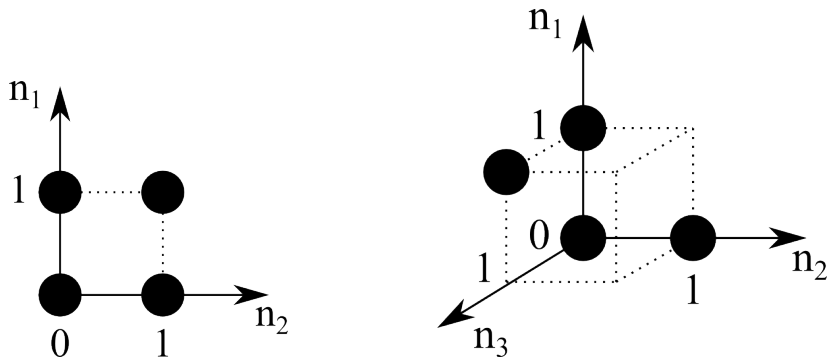
Bagging, also called bootstrap aggregating, was first presented by Breiman [6] in the context of random forests. According to Hastie et al. [33], bagging is a general machine learning technique to improve unstable methods and prevent overfitting. Given a training set $\mathscr{D}_{train}$, bagging generates M new training sets by sampling from $\mathscr{D}_{train}$ uniform and with replacement.

After the generation of individual training sets, each of the M decision trees is trained using the corresponding training set $\mathscr{D}^m_{train}$. The basic building block of decision tree training are iterative binary separation of the training data and class distribution estimation at each tree leaf node. To estimate which input variable should be used for optimal separation of the training data, the reduction in uncertainty, also known as information gain, is computed. Information gain for decision forests is defined as

$$IG(\mathscr{D}_{train}) = H(\mathscr{D}_{train}) - \sum_{i \in \{L,R\}} \frac{|\,\mathscr{D}^i_{train}\,|}{|\,\mathscr{D}_{train}\,|} H(\mathscr{D}^i_{train}) \tag{4.6}$$

where D^L_{train} and D^R_{train} denote the training subsets propagated to the left and right branch and $H()$ denotes the information entropy which is a measure of uncertainty. If the optimal separation is found and the information gain is sufficiently large, the training set is separated into D^L_{train} and D^R_{train} to compute the optimal separations for the next tree depth. According to Breiman [7], this iterative splitting is continued until the tree is fully grown. Hastie et al. [33] describes several approaches for pruning trees to reduce the effect of overfitting.

Random forest differs from random ferns by the following aspects: In contrast to random ferns, where the class distributions at each node are computed using all observations, random forest estimates the class distributions at each leaf node only over a small subset of training observations. This is due to the hierarchical structure of random forests which is different to the flat structure of random ferns. Moreover, in random ferns the probabilities are multiplied in a naïve Bayesian manner whilst the probabilities for random forests are averaged over the trees. Figure 4.1a and 4.1b illustrate the feature spaces of random ferns and random forest which can be evaluated using two nodes or a tree-depth of two. The theoretic expressive power of both methods is similar even though the feature space of trees seems at first glance to be much higher dimensional. Özuysal et al. [61] give a detailed discussion on the differences of random ferns and random forests and conclude that random ferns are competitive classifiers for keypoint recognition.



(a) Feature space for ferns with two nodes.

(b) Feature space for trees with tree-depth two.

Figure 4.1: Illustration of the feature spaces that can be evaluated by the different methods. Figure adapted by author from Özuysal et al. [61].

4.1.3 Random Ferns for Regression

Random regression ferns based multiple organ detection on MR Dixon sequences were first published by Pauly et al. [64] in 2011. In order to localize anatomical landmarks on MR Dixon sequences, Pauly et al. [64] propose to use the same problem formulation as earlier published by Criminisi et al. [15] on CT images.

The multiple multivariate regression model estimates the relationship between features extracted on the images (independent variables) $\mathbf{x}^i \in \mathscr{X}^D$ with $\mathscr{X}^D \subset \mathbb{R}^D$ and relative displacement vectors to K landmark positions (dependent variables) $\mathbf{y}^i \in \mathscr{Y}^{3K}$ with $\mathscr{Y}^{3K} \subset \mathbb{R}^{3K}$. Each feature $\mathbf{x}^i$ consist of D concatenated low-level features. Moreover, Each $\mathbf{y}^i$ is denoted by concatenating the relative displacement vectors to all K landmarks. Therefore, $\mathbf{x}$ is the input of the regression model and the aim is to regress the corresponding relative displacement vector $\hat{\mathbf{y}}$.

In order to estimate the relationship between features and there corresponding relative displacement vectors, Pauly et al. [64] propose to use an ensemble of random regression ferns denoted by $\mathrm{F} = \{F^1, \ldots, F^Z\}$. Similar to the previous definition of random ferns each fern consists of a set of L of nodes. Furthermore, each node is associated with a randomly chosen vector $\boldsymbol{\beta}_l^z \in \mathbb{R}^D$ with $l = 1, \ldots, L$ and a randomly chosen threshold $\theta_l^z \in \mathbb{R}$. Please note that the feature vector $\mathbf{x}$ and the vector $\boldsymbol{\beta}_l^z$ are of the same dimensionality and therefore the inner product $\langle \mathbf{x}, \boldsymbol{\beta}_l^z \rangle$ is defined. After successfully training the random regression ferns each fern F^z is associated with a partition $\mathrm{P}^z = \{\mathrm{C}_1^z, \ldots, \mathrm{C}_{2^L-1}^z\}$ of the feature space and simple hypothesis functions as well as multivariate Gaussian distributions for each cell C_j whose parameters are obtained during the training.

4.1.4 Training

The training set computed over N training observations obtained from all training images is formulated as:

$$\mathscr{D}_{train} := \{(\mathbf{x}^1, \mathbf{y}^1), \ldots, (\mathbf{x}^N, \mathbf{y}^N)\} \subset \mathscr{X}^D \times \mathscr{Y}^{3K} \tag{4.7}$$

As a first step of the training of random regression ferns, for each node of a fern the vector $\boldsymbol{\beta}_l^z$ and threshold θ_l^z will be randomly chosen and fixed. Pauly et al. [64] propose to sample $\boldsymbol{\beta}_l^z$ from a multivariate Gaussian distribution with zero mean and the $D \times D$ identity matrix as covariance. Moreover, Pauly et al. [64] propose to select θ_l^z from a Gaussian distribution as well. However, this does not guaranty to result in expedient separations of the feature space. Therefore, in this thesis θ_l^z will be randomly selected for each node from a uniform distribution. The interval of the uniform distribution is defined by:

$$\delta_{min} := \min_{\mathbf{x} \in \mathscr{D}_{train}} \langle \mathbf{x}, \boldsymbol{\beta}_l^z \rangle \tag{4.8}$$

$$\delta_{max} := \max_{\mathbf{x} \in \mathscr{D}_{train}} \langle \mathbf{x}, \boldsymbol{\beta}_l^z \rangle \tag{4.9}$$

Therefore, the probability density function is defined as:

$$g(\theta) = \begin{cases} \frac{1}{\delta_{max} - \delta_{min}}, & \text{if } \delta_{min} \leq \theta \leq \delta_{max} \\ 0, & \text{else} \end{cases} \tag{4.10}$$

Moreover, let the function $BV(c, d)$ be defined by:

$$BV(c, d) = \begin{cases} 1, & \text{if } c \leq d \\ 0, & \text{else} \end{cases} \tag{4.11}$$

During training the random regression fern model obtains a binary vector $\mathbf{b}^i \in \mathbb{B}^L$ of each feature $\mathbf{x}^i$ by concatenating the binary values b_l^i obtained using:

$$b_l^i := BV(\langle \mathbf{x}^i, \boldsymbol{\beta}_l^z \rangle, \theta_l^z) \tag{4.12}$$

Each binary vector $\mathbf{b}^i$ implicitly encodes the index of the cell C_j^z of partition P^z in which the feature vector falls. Therefore, each binary vector is converted into its decimal representation using:

$$DEC(\mathbf{b}) = \sum_{l=1}^{L} 2^{l-1} b_l \tag{4.13}$$

Moreover, let the indexing function $IDX : \mathbb{B}^L \to \mathrm{P}^z$ be defined by:

$$IDX(\mathbf{b}) := \mathrm{C}_{DEC(\mathbf{b})}^z \tag{4.14}$$

To simplify the modeling of the posterior probabilities $P(\mathbf{y}|\mathbf{x})$ of the high dimensional data Pauly et al. [64] propose to model the posterior probabilities based on a mixture of posterior distributions. Moreover, Pauly et al. [64] propose to use for each cell a multivariate Gaussian

distribution whose parameters are estimated over the feature vectors that fall into the cell. In order to estimate the parameters each cell is associated with

$$D_j^z := \{\mathbf{x}^i \in \mathscr{X}^D \,|IDX(\mathbf{b}^i) = \mathrm{C}_j^z\} \tag{4.15}$$

where the components of binary vector $\mathbf{b}^i$ are chosen according to Equation 4.12. Therefore, the mean and the covariance of the multivariate Gaussian distribution $\mathscr{N}(\boldsymbol{\mu}_j^z, \Sigma_j^z)$ for cell C_j^z are estimated by calculating the mean and the covariance over D_j^z. Therefore, the posterior probability in each cell C_t^z is modeled by:

$$P(\mathbf{y}|\mathbf{x} \in D_j^z, \mathrm{P}^z) \sim \mathscr{N}(\boldsymbol{\mu}_j^z, \Sigma_j^z) \tag{4.16}$$

In order to model the relationship between the feature vectors $\mathbf{x}^i$ and there corresponding relative displacement vectors $\mathbf{y}^i$, simple hypothesis function $h_t(\mathbf{x})$ e.g., a linear function, a robust linear function or a constant function, are estimated for each cell based on the corresponding D_j^z. Finally, each fern is associated with a partition of the feature space and the corresponding $2^L - 1$ multivariate Gaussian distributions and $2^L - 1$ simple hypothesis function. Figure 4.2 illustrates the approach on one-dimensional toy data.

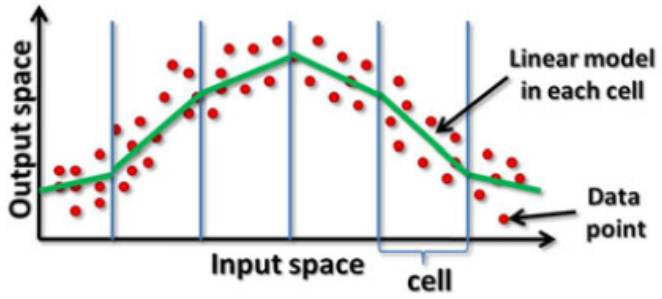


Figure 4.2: Illustration of one-dimensional toy example using linear functions for each cell, by Pauly et al. [64].

4.1.5 Prediction

In order to predict the relative displacement vector $\mathbf{y} \in \mathscr{Y}^{3K}$ for a training observation $\mathbf{x} \in \mathscr{X}^D$ the weighted predictions of all ferns are considered. Therefore, the conditional output variable $\hat{\mathbf{y}} \in \mathscr{Y}^{3K}$ is estimated using all fern models and posterior estimates from the different partitions are then combined using averaging. According to Pauly et al. [64] the probability distribution of $\mathbf{y}$ over the full feature space according to partition P^z is defined as:

$$P(\mathbf{y}, \mathrm{P}^z) = \sum_{\mathbf{j=1}}^{2^L-1} P(\mathbf{y}|D_j^z, \mathrm{P}^z) P(D_j^z) \tag{4.17}$$

Formally the weight $\psi \in \mathbb{R}$ of the random regression ferns for $\mathbf{y}$ is defined by

$$\psi = \sum_{\mathbf{z=1}}^{Z} P(\mathbf{y}|\, \mathrm{P}^z) \tag{4.18}$$

and the confidence $\nu \in \mathbb{R}$ with $\sum_{i=1}^{N} \nu^i = 1$ of for the prediction $\mathbf{y}$ is denoted by:

$$\nu = \frac{\psi}{\sum\limits_{\mathbf{i=1}}^{N} \psi^n} \tag{4.19}$$

Therefore, $\hat{\mathbf{y}}$ is calculated as

$$\hat{\mathbf{y}} = \frac{1}{\psi} \sum_{\mathbf{z=1}}^{Z} h(\mathbf{x})^z \; P(h(\mathbf{x})^z|\, \mathrm{P}^z) \tag{4.20}$$

where $h(\mathbf{x})^z$ defines the hypothesis function for partition P^z of the cell where the feature vector $\mathbf{x}$ falls into according to Equation 4.14 with components of the binary vector $\mathbf{b}$ chosen according to Equation 4.12. After estimating the displacement vectors the landmark locations are computed by shifting each voxel location used for feature extraction by their corresponding relative displacements. Moreover, Pauly et al. [64] propose to discard all predictions with low confidence value to average over all estimated locations with high confidence e.g., $\nu \geq 0.5$.

4.2 Boosted Random Regression Ferns

In machine learning, a prominent classifier using boosting is called AdaBoost by Freund and Schapire [29]. In contrast to randomized techniques like random forests or random ferns, AdaBoost refers to a machine learning technique that combines the output of many *weak learners* to an effective *committee* [33]. The main idea is to iteratively train *weak learners* (simple models) and update a distribution of weights over the trainings set in each iteration. The weight on the nth training observation at iteration t is denoted by $\gamma_t^i \in \mathbb{R}^+$. Initially all training observations are equally weighted and on each round the weights will be updated according to the model loss of the *weak learner*. For binary classification Freund and Schapire [29] propose to use the following update rule for the distribution of weights where $y^i \in \{-1, 1\}$ and $\mathbf{x}^i \in \mathscr{X}^D$:

$$\gamma_{t+1}^i := \frac{\gamma_t^i \exp(-\nu_t y^i h_t(\mathbf{x}^i))}{Z_t} \tag{4.21}$$

Please note that $\nu_t \in \mathbb{R}^+$ is the confidence of the *weak learner* $h_t(\mathbf{x})$ and the denominator Z_t is a positive normalization factor ensuring that $\sum_{i=1}^{N} \gamma_{t+1}^i = 1$.

In the context of regression analysis using AdaBoost, an update rule similar to Equation 4.21 can be formulated. In 1997 Bertoni et al. [1] introduced AdaBoost-RΔ and propose to use Equation 4.24 as an update rule for regression problems. Please note that $y^i \in \mathscr{Y}$ with $\mathscr{Y} \subset \mathbb{R}$ and $\mathbf{x}^i \in \mathscr{X}^D$. Moreover, please note that the name of AdaBoost-RΔ reveals that this approach allows some Δ-deviation to the actual data. The indicator function HS is denote by:

$$HS(a) = \begin{cases} 1, & \text{if } a \geq 0 \\ 0, & \text{else} \end{cases} \tag{4.22}$$

Moreover, the training error is defined as:

$$\epsilon_t := \sum_{i=1}^{N} \nu_t \; HS(||h_t(\mathbf{x}^i) - y^i|| - \Delta) \tag{4.23}$$

where $\nu_t \in \mathbb{R}^+$ is the confidence weight of the *weak learner* $h_t(\mathbf{x})$ and $\Delta \in \mathbb{R}^+$ defines the allowed deviation. Therefore, $\gamma_{t+1}^i \in \mathbb{R}^+$ is defined as:

$$\gamma_{t+1}^i := \gamma_t^i \left[\frac{\epsilon_t}{1 - \epsilon_t} \right]^{1 - HS(||h_t(\mathbf{x}^i) - y^i|| - \Delta)} \tag{4.24}$$

It has to be mentioned that Bertoni et al. [1] do not use the computed weights γ_t^i for the *weak learners* directly but rather assess the training data using $\omega_t^i := \frac{\gamma_t^i}{\sum_{j=1}^{N} \gamma_t^j}$.

Inspired by AdaBoost-RΔ and the performance of AdaBoost in the field of face recognition e.g., Viola-Jones face detector by Viola and Jones [79], a boosted variant of the random regression ferns is developed. The boosted random regression ferns effectively combine the intuition of AdaBoost-RΔ with the stochastic approach of random regression ferns.

4.2.1 Training

Besides the combination of boosting with random regression ferns the partitioning procedure of random regression ferns is modified. In the case of boosted random regression ferns each node N_l^z of a fern F^z consists of a randomly chosen vector $\boldsymbol{\beta}_l^z \in \mathbb{R}^D$ sampled from a multivariate Gaussian distribution with zero mean and the $D \times D$ identity matrix as covariance and two threshold values $\eta_l^z \in \mathbb{R}$ and $\theta_l^z \in \mathbb{R}$ with $\eta_l^z < \theta_l^z$. Evaluating an observation $\mathbf{x}^i$ on node N_l^z is done by deciding if $\eta_l^z \leq \langle \mathbf{x}^\mathbf{i}, \beta_l^z \rangle \leq \theta_l^z$. Therefore, the previously defined Equation 4.11 is substituted by:

$$BV(c, d, e) = \begin{cases} 1, & \text{if } c \leq d \leq e \\ 0, & \text{else} \end{cases} \tag{4.25}$$

Moreover, please note that therefore each component of the binary vector $\mathbf{b}$ is defined by Equation 4.26 instead of using Equation 4.12. Equation 4.26 is defined by:

$$b_l^i := BV(\langle \mathbf{x}^i, \boldsymbol{\beta}_l^z \rangle, \theta_l^z) \tag{4.26}$$

Therefore, the proposed extension allows to augment random regression ferns by an adaptive boosting procedure. Therefore the Equation 4.24 used as update rule for AdaBoost-RΔ is substituted by:

$$\gamma_{t+1}^i := \gamma_t^i \left[\frac{\epsilon_t}{1 - \epsilon_t} \right]^{1 - HS(|h_t(\mathbf{x}^i) - \mathbf{y}^i|_1 - \Delta)} \tag{4.27}$$

where $\mathbf{x}^i \in \mathscr{X}^D$ and $\mathbf{y}^i \in \mathscr{Y}^{3K}$ and $|\mathbf{x}|_1$ defines the L_1 norm of vector $\mathbf{x}$. Algorithm 4.1 illustrates the training of boosted random regression ferns.

input : A set of trainings data $\mathscr{D}_{train}$ containing N observations.
output: An ensemble of boosted random regression ferns.

// Initialize weight distribution
$\gamma \leftarrow \mathbf{1}$;
for $i \leftarrow 1$ **to** Z **do**
  // set weight distribution
  for $n \leftarrow 1$ **to** N **do**
    $\omega_n = \frac{\gamma_n}{\sum_{j=1}^{N} \gamma_j}$;
  end
  // create nodes for fern i
  for $j \leftarrow 1$ **to** L **do**
    $\boldsymbol{\beta}_j^i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$;
    $\mathscr{D}_{boost} \leftarrow$ `RandSampleWithReplacement` $(\mathscr{D}_{train}, \boldsymbol{\omega})$;
    $\delta_{min} \leftarrow \min_{\mathbf{x} \in \mathscr{D}_{boost}} \langle \mathbf{x}, \boldsymbol{\beta}_j^i \rangle$;
    $\delta_{max} \leftarrow \max_{\mathbf{x} \in \mathscr{D}_{boost}} \langle \mathbf{x}, \boldsymbol{\beta}_j^i \rangle$;
    // ensure that $\eta_j^i < \theta_j^i$
    $\eta_j^i \sim$ uniform distribution using Equation 4.10;
    $\delta_{min} \leftarrow \eta_j^i + \epsilon$;
    // $\epsilon \in \mathbb{R}^+$ is small and ensures that $\eta_l^z < \theta_l^z$
    $\theta_j^i \sim$ uniform distribution using Equation 4.10;
  end
  // estimate distributions
  for $(x, y) \in \mathscr{D}_{train}$ **do**
    $\mathbf{b} \leftarrow$ components are computed using Equation 4.26 ;
    // calculate cell id based on binary vector
    $k \leftarrow$ `Decimal` $(\mathbf{b})$;
    update estimations for $P(\mathbf{y}|\mathbf{x} \in D_k^i, \mathrm{P}^i)$;
    update estimations for $h_k^i(\mathbf{x})$;
  end
  update $\boldsymbol{\gamma}$ using Equation 4.27;
end

Algorithm 4.1: Training: Boosted Random Ferns

Please note that the boosting weights only effect the selection of thresholds η_l^z and θ_l^z. Formally, the thresholds are sampled from a uniform distribution using the probability density function described in Equation 4.10. However, instead of using the minimum and maximum values of all projected training data for $\delta_{min} \in \mathbb{R}$ and $\delta_{max} \in \mathbb{R}$ a subset $\mathscr{D}_{boost} \subseteq \mathscr{D}_{train}$ consisting of with replacement randomly sampled training observations are used. The probability of a training observation $\mathbf{x}^i$ to be selected for $\mathscr{D}_{boost}$ in the tth iteration is related to the boosting weight ω_t^i.

4.2.2 Prediction

The procedure for predicting $\hat{\mathbf{y}} \in \mathscr{Y}^{3K}$ of a unseen observation $\mathbf{x} \in \mathscr{X}^{D}$ using boosted random regression ferns differs to those of random regression ferns only by the calculation of the cell id using thresholds $\eta_l^z \in \mathbb{R}$ and $\theta_l^z \in \mathbb{R}$ (cf. Algorithm 4.2). Even though the algorithm described only processes one observation at a time, the prediction algorithm is implemented in a way that allows a matrix of unseen observations to be processed. Similar to Pauly et al. the result can be refined by removing predictions with a low confidence.

input : A test observation $\mathbf{x}$.
output: A predicted output value $\hat{\mathbf{y}}$.

$\hat{\mathbf{y}} \leftarrow 0$;
$\omega \leftarrow 0$;
for $i \leftarrow 1$ **to** Z **do**
  $\mathbf{b} \leftarrow$ components are computed using Equation 4.26 ;
  `// calculate cell id based on binary vector`
  $k \leftarrow$ `Decimal` $(\mathbf{b})$;
  `// add estimate of fern`
  $\hat{\mathbf{y}} \leftarrow \hat{\mathbf{y}} + h_k^i(\mathbf{x}) P(h_k^i(\mathbf{x}) | \, \mathrm{P}^i)$;
  $\psi \leftarrow \psi + P(h_k^i(\mathbf{x}) | \, \mathrm{P}^i)$;
end
$\hat{\mathbf{y}} \leftarrow \frac{\hat{\mathbf{y}}}{\psi}$

Algorithm 4.2: Prediction: Boosted Random Ferns

4.2.3 Conclusion

As shown in Algorithm 4.1, in the context of boosted random regression ferns each *weak learner* refers to a randomly chosen regression fern. The boosting weights only influence the separation of the feature space whilst the coefficients of the models are still chosen in a random manner. As illustrated in Figure 4.3 choosing a upper $\theta \in \mathbb{R}$ and lower threshold $\eta \in \mathbb{R}$ for each fern node instead of only one threshold allows the boosted random regression ferns to build a specific partitioning of the feature space. Therefore, the prediction loss is significantly lower for boundary values, cf. corresponding error values to $x \in [6, 6.5]$ in Figure 4.3.

The effect of extending random regression ferns by using a upper and lower bound is illustrated in Figure 4.3. The Residual Error (RE) values used in Figure 4.3 are computed according to

$$RE(\hat{\mathbf{y}}, \mathbf{y}) := \sum_{j=1}^{3K} (\hat{y}_j^i - y_j^i)^2 \tag{4.28}$$

which is equivalent to computing the Residual Sum of Squares (RSS) without summation over all $i = 1, \ldots, N$ predictions. It has to be noted that RE is computed based on the test data, which is different to the common use of the RSS. Moreover, the training set is produced

by applying a one-dimensional synthetic oracle function with additional noise. In order to get a representative estimate of the error for the different models, the mean RE values over 100 independent training and prediction runs has been used. In the case of Figure 4.3 both model types have been trained using an ensemble of 20 ferns each with 8 nodes. It should be noted that the effect of lower error values for the last input values carries through all synthetic tests independent of the signal-to-noise ratio and the model quality.

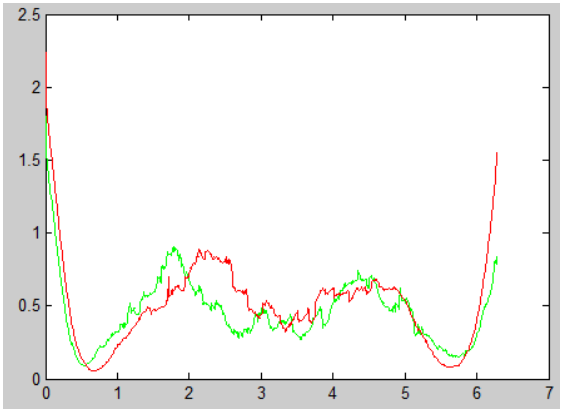


Figure 4.3: Illustration of the mean RE over a synthetic oracle function. The ordinate axis shows the RE-values whilst the abscissa axis shows the one-dimensional input space. Green = model with lower and upper threshold, red = model with only one threshold. Figure created by author.

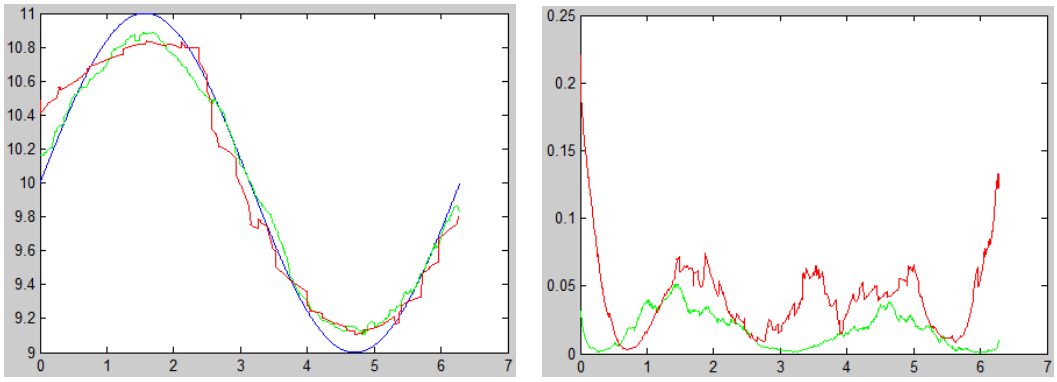
By applying the proposed boosting approach and using the promising modification of lower and upper thresholds for feature space separation the error values can be dramatically reduced. Figure 4.4 illustrates how the proposed boosting variant outperforms random regression ferns on a one-dimensional synthetic oracle function defined by

$$f(x) = \sin(x) + 10 + \epsilon \tag{4.29}$$

where ϵ is noise sampled from a uniform distribution with values between $-0.4 \leq \epsilon \leq 0.4$ and $x \in \mathbb{R}$.

The plots in Figure 4.4 are generated by evaluating 10 individual training and prediction runs using the boosted random regression ferns using lower and upper thresholds and the random regression ferns proposed by Pauly et al. [64]. Moreover, for each fern ensemble a number of 20 ferns and 8 nodes are used. The regularization term $\Delta \in \mathbb{R}^+$ is set to $\Delta = 2$ for all runs.

As can be seen in Figure 4.4a even the boosted model with the highest error value outperforms the model of Pauly et al. [64]. The corresponding mean RE values computed over the 10 runs are visualized in Figure 4.4b. The boosted random regression ferns model is only an intermediate model. However, an exhaustive evaluation of the overall model performance is beyond the scope of this thesis.



(a) Estimations of models with highest testing error.

(b) Mean RE with ordinate and abscissa similar to Figure 4.3.

Figure 4.4: Comparison of boosted random regression ferns with random regression ferns on an one-dimensional toy example. Green = boosted random regression ferns, red = ordinary random regression ferns, and blue = ground truth. Figures created by author.

4.3 Robust Boosted Random Regression Ferns (RobustBRRFerns)

Despite the promising results of the boosted random regression ferns it is possible that the model overfits the training data and underperforms on edge cases. Therefore, a robust modification which prevents overfitting on the training data has been developed.

4.3.1 Overfitting and Underfitting

The statistical field of overfitting and underfitting deals with the trade-off between model complexity and prediction error. The aim is to select a model such that the prediction error over the training set and over the test set will be minimized. Overfitting occurs if a statistic model describes the random noise contained in the training set. On the contrary, underfitting describes the effect of using a statistical model with too low expressive power. Figure 4.5 illustrates over and underfitting of statistical models.

Models with complexity lower than the optimal model h^* will underfit whilst models with higher complexity will overfit the data, cf. Rockach [67]. As can be seen in Figure 4.5, estimating the model performance only based on the training error does not guarantee the selection of h^*. Therefore the model performance needs to be estimated using a different error estimation. The generalization error of a statistical model is defined as an estimate of the testing error by:

$$L(h)_{\mathscr{D}_{test}} = \sum_{i=1}^{N} RE(h(\mathbf{x}^i), \mathbf{y^i}) \tag{4.30}$$

For simplification purposes, it is assumed that all test observations are drawn from a uniform distribution we do not need to weight the prediction errors according to the density of $\mathbf{x}$. As described by Bishop [2] one way to estimate the generalization error is to use *cross-validation*.

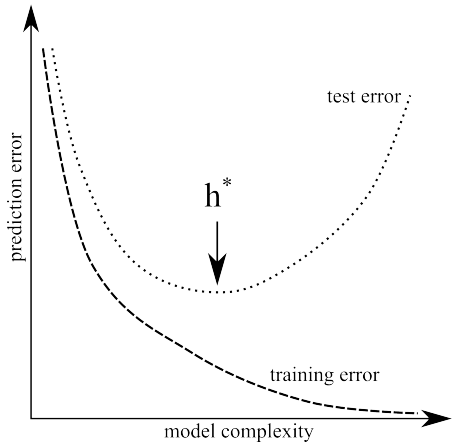


Figure 4.5: Trade-off between prediction error and model complexity. Figure adapted by author from Rockach [67].

Therefore, the available data will be split into a training and an evaluation set and the model quality will be measured based on the evaluation set. A more detailed description is given in Chapter 6.

As the proposed boosted random regression ferns iteratively adapt according to the training error the model tends to overfit greatly. Vezhnevets and Barinova [77] discuss the problem of overfitting in boosting and propose to avoid overfitting by removing confusing observations that build border cases between two classes. In the case of regression problems it is not possible to compute probability values for strict classes, as there are not classes. In boosting regression approaches overfitting can be reduced by using a learning rate to lower the effect of confusing observations [67]. However, as the main idea of Vezhnevets and Barinova [77] is closely related to *cross-validation* a similar approach is developed. More precisely Vezhnevets and Barinova [77] propose to separate the training set $\mathscr{D}_{train}$ into three strict subsets $\mathscr{D}^1_{train} \cup \mathscr{D}^2_{train} \cup \mathscr{D}^3_{train} = \mathscr{D}_{train}$ to detect and remove confusing observations. Please recall that $\mathscr{D}_{train}$ is defined according to Equation 4.7. In contrast to use all training observations for estimating the model parameters and computing the boosting weights Vezhnevets and Barinova [77] propose to use the subset $\mathscr{D}^1_{train}$ to estimate the model parameters, $\mathscr{D}^2_{train}$ to compute the model probabilities and $\mathscr{D}^3_{train}$ to update the boosting weights. In this thesis the separation approach is used for the whole training phase to provide a robust estimation of the training error.

4.3.2 Robust Training

In order to prevent the proposed model to overfit the training data the robust boosted random regression ferns (RobustBRRFerns) are developed. Analogously to *cross-validation* the training set is separated into strict subsets. As the probabilities of the *weak learners* are implicitly given by the model itself the training set is only split into two strict sets $\mathscr{D}^1_{train} \cup \mathscr{D}^2_{train} = \mathscr{D}_{train}$. The model parameters are estimated using $\mathscr{D}^1_{train}$ whilst the boosting weights are computed on $\mathscr{D}^2_{train}$. Algorithm 4.3 illustrates the training of robustBRRFerns by extending Algorithm 4.1

with $\mathscr{D}^1_{train}$ and $\mathscr{D}^2_{train}$. Moreover the training is extended by introducing an additional learning rate $\delta \in \mathbb{R}^+$.

In addition to splitting the training set the *weak learners* are replaced by *robust weak learners*. In this thesis a *robust weak learner* is denoted by a robust hypothesis function and a confidence value. Formally, a *robust weak learner* is defined by the tuple $(h(\mathbf{x}), \boldsymbol{\alpha})$ where $\boldsymbol{\alpha} \in \mathbb{R}^{3K}$ and every component $0 \leq \alpha_i \leq 1$ with $i = 1, \ldots, 3K$. The robust hypothesis function can be any kind of robust estimator. Furthermore, a simple estimation based on a constant function is used. The definition of *robust weak learners* with a constant function as hypothesis function is defined by

$$(\mathrm{med}(\mathbf{X}), \exp(-0.5\frac{\mathrm{iqr}(\mathbf{X})}{\kappa}) \tag{4.31}$$

where med is the median vector and iqr is the vector of interquartile ranges of the matrix $\mathbf{X} \in \mathscr{X}^{D \times M}$ containing M features $\mathbf{x}^i \in \mathscr{X}^D$. Moreover, $\kappa \in \mathbb{R}^+$ is a positive weighting factor. According to Huber [36] the median is a robust measure of scale whilst the interquartile range is a robust measure of statistical dispersion.

input : A set of trainings data $\mathscr{D}_{train}$ containing N observations.
output: An ensemble of robustBRRFerns.

`// Initialize weight distribution`
$\gamma \leftarrow \mathbf{1}$;
for $i \leftarrow 1$ **to** Z **do**
  `// randomly separate training set`
  $\mathscr{D}^1_{train} \cup \mathscr{D}^2_{train} = \mathscr{D}_{train}$
  `// set weight distribution`
  for $n \leftarrow 1$ **to** N **do**
    $\omega_n = \delta \frac{\gamma_n}{\sum_{j=1}^{N} \gamma_j}$;
  end
  `// create nodes for fern` i
  `CreateFern` $(\mathscr{D}^1_{train}, \boldsymbol{\omega})$
  `// estimate distributions`
  for $(x, y) \in \mathscr{D}^1_{train}$ **do**
    $\mathbf{b} \leftarrow$ components are computed using Equation 4.26 ;
    `// calculate cell id based on binary vector`
    $k \leftarrow$ `Decimal` $(\mathbf{b})$;
    update estimations for $P(\mathbf{y}|\mathbf{x} \in D^i_k, \mathrm{P}^i)$;
    update estimations for $(h^i_k(\mathbf{x}), \boldsymbol{\alpha}^i_k)$;
  end
  update $\boldsymbol{\gamma}$ based on $\mathscr{D}^2_{train}$ using Equation 4.27;
end

Algorithm 4.3: Training: RobustBRRFerns

input : Training set $\mathscr{D}_{train}$ and weights $\boldsymbol{\omega}$.
output: A set of nodes.

for $j \leftarrow 1$ **to** L **do**
  $\beta_j^i \sim \mathscr{N}(\mathbf{0}, \mathbf{I})$;
  $\mathscr{D}_{boost} \leftarrow$ `RandSampleWithReplacement` ($\mathscr{D}_{train}$, $\boldsymbol{\omega}$);
  $\delta_{min} \leftarrow \min_{\mathbf{x} \in \mathscr{D}_{boost}} \langle \mathbf{x}, \boldsymbol{\beta}_j^i \rangle$;
  $\delta_{max} \leftarrow \max_{\mathbf{x} \in \mathscr{D}_{boost}} \langle \mathbf{x}, \boldsymbol{\beta}_j^i \rangle$;
  // ensure that $\eta_j^i < \theta_j^i$
  $\eta_j^i \sim$ uniform distribution using equation 4.10;
  $\delta_{min} \leftarrow \eta_j^i + \epsilon$;
  // $\epsilon \in \mathbb{R}^+$ is small and ensures that $\eta_l^z < \theta_l^z$
  $\theta_j^i \sim$ uniform distribution using equation 4.10;
end

Algorithm 4.4: CreateFern

4.3.3 Robust Prediction

The prediction phase of RobustBRRFerns directly benefits from the confidence values of the *robust weak learners* introduced. Algorithm 4.5 illustrates how the new confidence values are used.

input : A test observation $\mathbf{x}$.
output: A predicted output value $\hat{\mathbf{y}}$.

$\hat{\mathbf{y}} \leftarrow 0$;
$\omega \leftarrow 0$;
for $i \leftarrow 1$ **to** Z **do**
  $\mathbf{b} \leftarrow$ components are computed using Equation 4.26 ;
  // calculate cell id based on binary vector
  $k \leftarrow$ `Decimal` ($\mathbf{b}$);
  // add estimate of fern, please note that $\circ$ defines the component wise multiplication of vectors.
  $\hat{\mathbf{y}} \leftarrow \hat{\mathbf{y}} + h_k^i(\mathbf{x})\ P(h_k^i(\mathbf{x}) | \mathrm{P}^i)\ \circ \boldsymbol{\alpha}_k^i$;
  $\psi \leftarrow \psi + P(h_k^i(\mathbf{x}) | \mathrm{P}^i)\ mean(\boldsymbol{\alpha}_k^i)$;
end
$\hat{\mathbf{y}} \leftarrow \frac{\hat{\mathbf{y}}}{\psi}$

Algorithm 4.5: Prediction: RobustBRRFerns

The idea of augmenting each hypothesis function with a confidence value is based on the observation that some regions in CT images may contain homogenous textures. Especially if features with a low spread (high locality) are used the feature descriptor can have the same

response on voxel locations spread over the whole image. In CT images this mainly occurs if the feature describes voxels which are located in homogenous regions e.g., lung or air surrounding the patient, where the mean intensity values of the cuboidal regions are the same. Because RobustBRRFerns build a partitioning in the feature space those feature responses are summarized into one cell of the partition. However, as the posterior probabilities are estimated using Equation 4.16 and predictions are computed based on constant functions the prediction for feature observations $\mathbf{x}$ describing air act like noise with high influence. The illustration of a two-dimension toy example in Figure 4.6 explains the problem of homogenous regions. The red circle visualizes the landmark location whilst the yellow stars are positions in the image that share the same feature response.

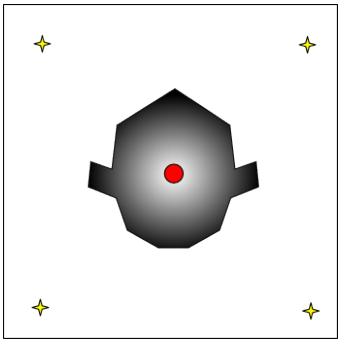


Figure 4.6: Two-dimensional toy example containing homogenous regions. Yellow stars illustrate critical pixels with similar feature patterns, the red circle illustrates the landmark position. Figure created by author.

Assuming that random regression ferns construct a constant function for the feature response visualized as yellow stars by averaging their displacement vectors during training phase. The resulting constant function is therefore ≈ 0 for all dimensions. Moreover, the computed posterior probability for the cell containing the feature responses of the yellow stars is close to one. During the prediction phase random regression ferns weight the displacement vector $(0, 0)$ with high probability and therefore falsify the estimated landmark position. It has to be noted that it is not possible to reject this votes by simply constraining the posterior probability.

In the case of constant functions, introducing additional confidence values allows to suppress cell predictions which contain depending variables with high variation. If linear regression functions are used instead of constant functions the confidence values allow to suppress predictions which are based on poor approximations.

4.3.4 Advantages

The proposed RobustBRRFerns effectively combine the benefits of random regression ferns and boosting while preventing overfitting and reducing noise sensitivity. Table 4.1 shows a comparison of random regression ferns, random forests[1] and RobustBRRFerns for different signal-

[1] implementation by Liaw and Wiener [47]

to-noise ratio cases on a synthetic one-dimensional sinus function. The signal-to-noise ratio is measured in dB calculated by $SNR := 20 \log_{10} \frac{Asignal}{Anoise}$ where $A \in \mathbb{R}$ is the amplitude of the signal. In this thesis the amplitude is calculated using Root Mean Squares over all observations. Therefore, if the SNR value is high, the amount of noise in the signal is low. To get representative error estimations the mean RSS values used for the comparisons are computed over 100 independent training and prediction runs with equal configurations for both random ferns models. For the random forest the default settings has been used.

Regression Model	11.50 dB	11.14 dB	10.37 dB
Random Regression Ferns	46.40	49.55	62.73
Random Forest	8.26	23.01	99.33
RobustBRRFerns	6.55	14.23	26.49

Table 4.1: Mean RSS values for random regression ferns and RobustBRRFerns computed based on oracle Function 4.29 with different signal-to-noise ratios.

It can be seen that the testing error of all models increases by decreasing signal-to-noise ratio whilst the RobustBRRFerns still outperforms the random regression ferns and the random forest on very low signal-to-noise ratios. As shown in Table 4.2 this observation also holds if the dimensionality of the independent and the conditional variables are increased. To get representative error estimations the mean RSS values over 50 independent runs (cf. Table 4.2) are computed for the multivariate unnormalized $sinc(\mathbf{x})$ function and the norm of the independent variables $\mathbf{x} \in \mathbb{R}^2$ computed with:

$$y_1 = \frac{\sin(||\mathbf{x}||)}{||\mathbf{x}||} + 10 + \epsilon \tag{4.32}$$

$$y_2 = ||\mathbf{x}|| + \epsilon \tag{4.33}$$

Please note that the ϵ-noise is sampled from a univariate Gaussian distribution with zero-mean. The RobustBRRFerns have only been tested against the random regression ferns because the random forest implementation by Liaw and Wiener [47] allows only single multivariate regression. As the problem is more complex than the estimation of the one-dimensional synthetic function a larger ensemble size, i.e. $Z = 25$, and a higher amount of nodes, i.e. $L = 10$, for both models are used. Please note that the selected parameters are found empirically. One interesting effect is the decreasing RSS values for both models. This is based on the fact that the frequency of the $sinc(\mathbf{x})$ function on low amplitudes is to high to correctly estimate the underlying function if a high amount of noise is visible. Therefore, both models heavily adapt to the Gaussian noise but luckily are still able to estimate values with low amplitude.

Regression Model	11.50 dB	11.14 dB	10.37 dB
Random Regression Ferns	107.34	101.88	103.73
RobustBRRFerns	30.56	29.41	29.31

Table 4.2: Mean RSS values for random regression ferns and RobustBRRFerns computed based on oracle function 4.32 with different signal-to-noise ratios.

In view of the promising results on synthetic data the proposed RobustBRRFerns are used to automatically predict anatomical landmarks based on the feature descriptors explained in Chapter 3. As a global view on the image gives a good but rough estimations of the landmark locations whilst a local view gives precise estimations of the landmark locations but can only act on a local scale the next logical step is a multi-step prediction procedure.

4.4 Summary

Özuysal et al. [62] published in 2007 a memory-efficient alternative to random ferns for keypoint recognition: random ferns. In contrast to random forest the random fern classifier evaluates the independent variable using lists of nodes (ferns) instead of tree hierarchies. Subsequent Pauly et al. [64] published in 2011 a multivariate regression model based on random ferns to localize anatomical landmarks in MR Dixon sequences. Random regression ferns partition the feature space by evaluating the features against random test. Those random test are defined by applying random linear functions on the features and test the results against a randomly selected threshold.

By considering the previous work on boosted regression models e.g., AdaBoost-RΔ by Bertoni et al. [1], a boosted variant of random regression ferns is derived. In contrast to random ferns the boosted variant evaluates binary test on the features using intervals instead of single thresholds. As illustrated in Figure 4.3 this allows boosted random regression ferns to build specific clusters in the feature space and therefore reduces the error loss on the boundaries. To prevent the model to overfit on the data RobustBRRFerns are derived from the boosted random regression ferns. In contrast to the boosted variant of random regression ferns the training set will be separated into two strict sub set at each individual fern training. The separation of the training set and an additional learning rate guarantee that RobustBRRFerns does not adapt to noise during training. As shown in Table 4.32 and 4.2 the RobustBRRFerns outperform random regression ferns on synthetic data in all test cases and random forests on one dimensional synthetic data.

CHAPTER 5

Multi-Pass Landmark Prediction

According to Suykens [73], two-pass or multi-pass methods are successful approaches to develop robust and fast global illumination systems. Those approaches distribute the computation of light transport in a scene over multiple passes [73]. In the context of image processing, image pyramids are used to detect stable regions in the image. However, in contrast to multi-pass algorithms image pyramids do not tackle the problem of computational complexity but improve the robustness of detection a pattern of interest only.

The proposed multi-pass method roughly estimates the landmark locations in the first step and computes a refined result based on local textural information. Moreover, the multi-pass method allows to estimate the landmark locations fast and memory-efficient. The proposed method therefore combines robustness, similar to image pyramids, with efficiently distributed computations.

5.1 Global Localization

The first pass produces global predictions on downsampled volumes by evaluating sparse sampled feature descriptor responses on the image. In this thesis images have been downsampled by using convolution of the image, with a Gaussian kernel. Formally, the new image I^1 will be calculated by

$$\mathrm{I}^1 = \mathrm{I}^0 * G(x, y, z) \tag{5.1}$$

where $G(x, y, z)$ denotes the Gaussian kernel. Afterwards every even-numbered row and column will be removed to reduce the image resolution. Based on the reduced volumes a sparsely sampled training set $\mathscr{D}_{train}$ is composed. Similar to the approach of Pauly et al. [64] for the global landmark localization only at every fourth voxel a feature response is computed. It has to be mentioned that Pauly et al. [64] do not downsample the image and therefore compute significantly more features per image. Moreover, the RobustBRRFerns are trained to predict all landmark locations in a single prediction step.

The depending variables $\mathbf{y} := (\mathbf{d}^1, \ldots, \mathbf{d}^K)^T \in \mathscr{Y}^{3K}$ of $\mathscr{D}_{train}$ consist of the displacement vectors $\mathbf{d}^k \in \mathbb{R}^3$ with $k = 1, \ldots, K$ for each of the K landmarks.

In order to predict the locations of anatomical landmarks on an unseen image, the image is downsampled and a sparse representation are extracted. The RobustBRRFerns are used to compute the estimated positions on the downsampled image. To obtain a sparse representation, similar to Pauly et al. [64] the features for the prediction procedure are extracted on each fourth voxel. Note that the estimated locations are computed on the downsampled image and therefore have to be transformed back into the original image space I^0. A simple way to achieve this is to multiply the predicted locations with the downsampling rate used for the Gaussian kernel.

5.2 Local Refinement

The rough estimations will be refined by using the second pass of the multi-pass algorithm. This is achieved by training individual RobustBRRFern models for each landmark on the feature response of the adjacent voxel in the original images I^0. Defining the neighborhood for the training is crucial, therefore the neighboring voxels of the landmark location $\mathbf{p}^k \in \mathbb{R}^3$ are defined by:

$$N_{\mathbf{p}^k} \sim \mathscr{N}(\mathbf{p}^k, \Sigma) \tag{5.2}$$

where $\Sigma = \mathrm{I} * 2\,\mathrm{E}[L_k(h(\mathbf{x}))]$ and I denotes the 3×3 identity matrix. Furthermore, $\mathrm{E}[L_k(h(\mathbf{x}))]$ denotes the expectation value of the generalization error of the kth landmark location for the first pass. By training an ensemble of RobustBRRFerns each estimated landmark location can be refined by the corresponding regressor. The regressor ensemble for K landmarks is denoted by $H = \{h^1(\mathbf{x}), \ldots, h^K(\mathbf{x})\}$. In order to predict the exact landmark positions, features in the vicinity of the rough estimates are extracted. The neighborhood for the feature extraction is defined by random samples $\mathbf{q} \in \mathbb{R}^3$ drawn from a uniform distribution with offset values between $-\,\mathrm{E}[L_k(h(\mathbf{x}))] \leq q_i \leq \mathrm{E}[L_k(h(\mathbf{x}))]$. The refined locations will then be estimated by each RobustBRRFerns regressor individually. Please note that in contrast to the approach of Pauly et al. [64] only a small set of randomly sampled feature responses is extracted on the original image. To acquire higher accuracy, analogously to Donner et al. [21] the local refinement is iteratively repeated multiple times on robust estimations of the new predicted landmark locations. Alternatively, the iterative convergence to the landmark positions using a regression model can also be replaced by voxels-wise classification in the vicinity of the rough estimates. To enhance the accuracy and robustness of the refinement, in this thesis multivariate outlier detection is used to identify and remove prediction that falsify the estimated landmark location.

5.2.1 Outlier Removal

The robust Mahalanobis distance is used in several statistical methods for multivariate outlier detection over a set $X^D \subset \mathbb{R}^D$ is defined by

$$MD^i := \sqrt{(\mathbf{x}^i - T(X))^\top C(X)^{-1}(\mathbf{x}^i - T(X))} \tag{5.3}$$

where $T(X)$ is the robust dimension-wise mean estimate and $C(X)$ is the corresponding robust covariance matrix. Because the Mahalanobis distance MD^i is chi-square distributed with respective degrees of freedom, the result can be used to measure if observations are outliers. It can be seen in Equation 5.3 that the Mahalanobis distance measures the span of an observation to the estimated mean with respect to the covariance of the observations.

Based on this observation and the previously published Minimum Covariance Determinant (MCD) method of Rousseeuw [68] in 1999 Rousseeuw and Driessen [69] present a fast MCD estimator (FAST-MCD) for outlier removal. The basic procedure of FAST-MCD is described in Algorithm 5.1 where a random subset $Z_{old}^D \subset X^D$ with cardinality of z is given parameter. The output after L iterations is the new subset Z_{new}^D with minimum covariance determinant.

input: Initial random subset Z_{old}.
$T \leftarrow$ estimate robust mean of subset;
$C \leftarrow$ estimate robust covariance of subset;
for $i \leftarrow 1$ **to** L **do**
    compute distances using T, C with Equation 5.3 ;
    sort observations ascending in distance ;
    $Z_{new} \leftarrow$ first z observations ;
    $T \leftarrow$ estimate robust mean of subset ;
    $C \leftarrow$ estimate robust covariance of subset ;
end

Algorithm 5.1: FAST-MCD

For simplicity reasons, in this thesis the FAST-MCD implementation included in the Library for Robust Analysis (LIBRA) by Verboven and Hubert [76] is used to remove outlying predictions. Figure 5.1 visualizes landmark predictions for the entry point of the right optical nerve computed on a CT image of the human head. Predictions colored in grey are outliers detected by FAST-MCD whilst blue dots are acceptable predictions. The correct position of the right optical nerve is colored in red. Please note that Figure 5.1 shows a two-dimensional projection of three-dimensional prediction. Therefore, grey colored predictions, which are visually close to the landmark position, might actually lie behind or in front of the landmark.

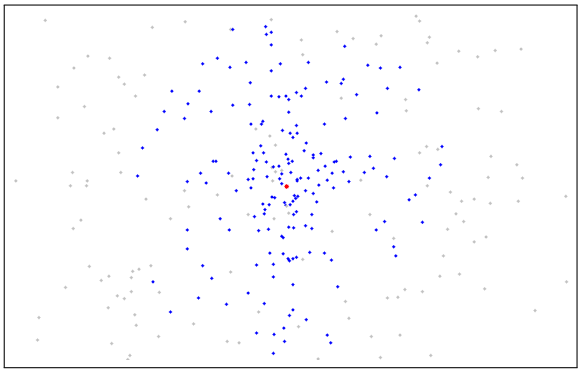


Figure 5.1: Two-dimensional projection of three-dimensional landmark location predictions (blue) with detected outliers (grey). The ground truth landmark location is colored in red. Figure created by author.

5.3 Multi-Pass Model

The complete multi-pass landmark localization pipeline for medical images is illustrated in Figure 5.2. The approach is a generic model tailored to landmark detection of large medical images. The hypothesis functions shown in Figure 5.2 can be any arbitrary regressor or classifier.

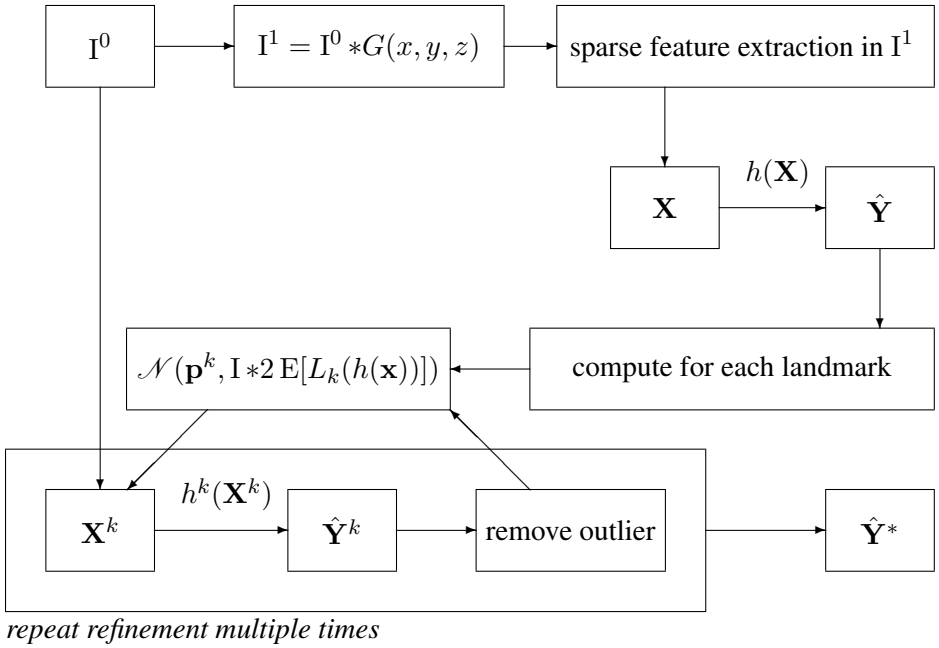


Figure 5.2: Multi-pass model for automatically medical landmark localization. Figure created by author.

As shown in Figure 5.2 the initial volume I^0 represents the entry point of the multi-pass pipeline. After convolving the initial volume with a Gaussian kernel and reducing the resolution of the resulting image every 64th voxel will be described by a feature response. Applying the global prediction phase directly results in coarse guesses of all landmark locations. The following steps are individually applied for each landmark. For each new refinement the estimations of the last refinement step will be used to sample the features for the next iteration. Please note that the number of repetitions of the refinement step can be either defined by a fixed number of iterations or depend on a stopping criterion e.g., the difference between the old and the new estimates. Due to the performance of RobustBRRFerns in synthetic test cases, in this thesis RobustBRRFerns are used as a hypothesis functions for all passes.

5.4 Summary

To provide a fast and robust prediction system, the landmark localization approach is separated into different passes. The global localization initially estimates all landmark locations at once. To allow a fast estimation, the volume is downsampled and only a sparse set of feature responses is extracted and used for the localization.

The predicted positions are then used as initial guesses for the refinement passes. The refinement will be computed for each landmark individually. Each refinement phase extracts feature responses in the vicinity of the initial guess on the original volume. The initial location is updated according to the output of the hypothesis function on the extracted feature responses. Please note that a robust sub set of predictions, estimated using FAST-MCD, is used instead of all predictions. The multi-pass model is outlined in Figure 5.2.

CHAPTER 6

Evaluation and Results

This chapter introduces the evaluation methods used in this thesis. The results achieved by state of the art methods are given. Subsequent the results achieved by the proposed landmark prediction system are presented.

6.1 Evaluation Methods

In machine learning evaluating models aims to determine how well a model computed by a specific algorithm fits the data. If a certain loss function used for the evaluation produces a high value, the model can be considered to provide a bad representation of the data. Section 6.1.2 gives an overview of the loss functions used to estimate the model error. A second topic in the field of machine learning is feature analysis, i.e. to evaluate how discriminative a certain feature is. Methods used to evaluate the proposed features are described in the following section.

6.1.1 Feature Evaluation

Machine learning techniques follow the *garbage in – garbage out* principle (cf. Lidwell et al. [48]) and therefore the prediction accuracy of the learning system depends on the discernibility and the signal-to-noise ratio of the independent variables [53]. Therefore, a discriminative and low-noise representation of the image space improves the model accuracy. Whilst for classification problems the feature class correlation is a criterion for feature selection (cf. Hall [32]), for regression problems a strong correlation between the independent and the conditional variables is desired. In the context of this thesis, it is desirable that the feature descriptor responses are correlated their relative displacement vectors. Therefore, statistical estimates of the relative displacement vectors depending on the features are used to evaluate the effectiveness of the feature descriptors.

Maximum Dependent Variable Variance. Computing the Maximum Dependent Variable Variance (MDVV) where the variances of each dimension of the dependent variables D over all feature vectors N are evaluated gives a insight into the performance of the features. The MDVV is defined by:

$$MDVV := \max_{i=1}^{N} \max_{j=1}^{D} \sigma_{i,j}^2 \tag{6.1}$$

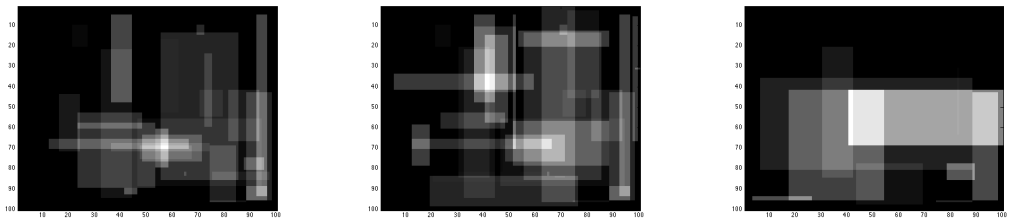
Please note that low MDVVV value indicates that compact distributions of displacement vectors are related to the feature.

Maximum Kernel Density. Furthermore, the maximum density of a kernel density estimation is used to measure the performance of the feature. One possibility to estimate a non-parametric distribution is the univariate Parzen window estimator by E. Parzen [63] for kernel density estimation which is defined for $x^i \in \mathbb{R}$ iid samples with $i = 1, \ldots, N$ drawn from a unknown density function by:

$$f(x) = \frac{1}{N\omega} \sum_{i=1}^{N} K\left(\frac{x - x^i}{\omega}\right) \tag{6.2}$$

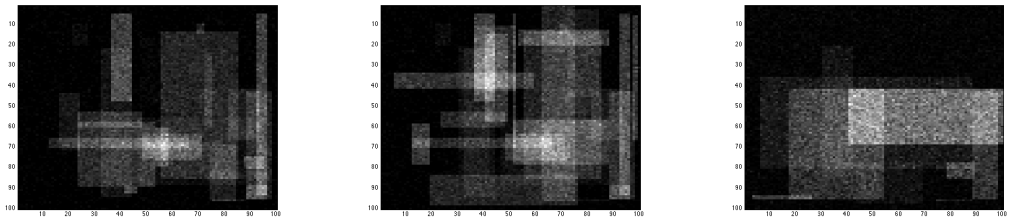
where $\omega \in \mathbb{R}^+$ denotes a smoothing factor and $K(x)$ is a given kernel function e.g., Gaussian kernel. The maximum density of a kernel density estimation is therefore defined by the maximum of $f(x)$. If the maximum density value for the jth dimension with $j = 1, \ldots, 3K$ is large, close to or larger than 0.5, a compact distribution over the displacement vectors in dimension j is assumed.

Noise Sensitivity. To ensure stable correlations between features and the output space it is crucial that features are noise-insensitive. Especially in the field of medical image analysis it is important to allow robustness against specific types of noise. In MRI scans noise is usually modeled by observations sampled from a Rice distribution [51] whilst noise in CT images can be modeled using the Poisson distribution. To test the robustness of the different features a synthetic image affected by different types of noise are used. The two test cases used for the evaluation of stability are robustness against Rice distributed noise and robustness against Poisson distributed noise in the synthetic test image. Figure 6.1 shows three slices of the synthetic test image used for the evaluation. The noisy volumes are shown in Figure 6.2 and 6.3. Please note that dynamic noise models are used and therefore the amount of noise depends on the underlying intensity values of the original test volume.



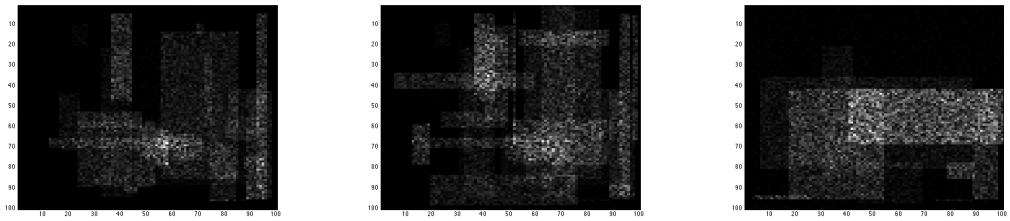
(a) Slice 33 of 100. (b) Slice 66 of 100. (c) Slice 99 of 100.

Figure 6.1: Illustration of the synthetic test volume used for evaluating the noise sensitivity. Figures created by author.



(a) Slice 33 of 100. (b) Slice 66 of 100. (c) Slice 99 of 100.

Figure 6.2: Illustration of the Poisson noisy volume used for evaluating the noise sensitivity. Figures created by author.



(a) Slice 33 of 100 (b) Slice 66 of 100. (c) Slice 99 of 100.

Figure 6.3: Illustration of the Rice noisy test volume used for evaluating the noise sensitivity. Figures created by author.

6.1.2 Model Evaluation

As described in Chapter 4 due to the effect of overfitting the model performance can not be evaluated by using the training error. In this thesis the model loss is therefore defined by computing the generalization error over multiple folds of cross validation.

K-fold Cross Validation

Hastie et al. [33] formally describe K-fold cross validation by an indexing function

$$\kappa : \{1, \ldots, N\} \rightarrow \{1, \ldots, K\} \tag{6.3}$$

that indicates the random partitioning of all N samples from $\mathscr{D}_{train}$ into K distinct parts. The generalization error is estimated over K runs or folds. For each fold all parts except the kth part are used to train the model $h^k(\mathbf{x})$ and the model loss function $L(h^k(\mathbf{x}))$ is evaluated only using the kth part. Figure 6.4 illustrates the idea of cross-validation with five folds.

Figure 6.4: Schematic illustration of 5-fold cross-validation at the $k = 4$ fold. Figure adapted by author from Hastie et al. [33].

After computing the error values of each fold using $L(h^k(\mathbf{x}))$, the error values are averaged to obtain an estimate of the generalization error for $h(\mathbf{x})$ over $\mathscr{D}$. Hastie et al. [33] describe that common selections for K are $K = 5$ and $K = 10$. Because of the low number of observations per dataset in this thesis either 4-fold cross validation or *leave one out validation* is used. In the case of *leave one out validation* all observations except one observation are used for training and the excluded observation will be used to evaluate the model performance. Therefore, *leave one out validation* is only feasible if the number of observations in the training set is low such that $K = N$ folds can be realized.

Loss Functions

The following loss functions are used in this thesis to evaluate the goodness of the fitted regression model. Please note that those measures are used due to their simplicity and expressive power. The Least Absolute Deviation (LAD) provides a robust estimate for the loss and is defined by:

$$LAD := \sum_{i=1}^{N} |\mathbf{y}^i - h(\mathbf{x}^i)|_1 \tag{6.4}$$

The Residual Sum of Squares (RSS), also known as Sum of Squares Error (SSE), provides more expressive estimation of the model loss then the LAD but is considered to be less robust. The RSS is defined by:

$$RSS := \sum_{i=1}^{N} \sum_{j=1}^{D} (y_j^i - h(\mathbf{x}^i)_j)^2 \tag{6.5}$$

Please also note the the R^2-loss function has not been used in this thesis because this measurement assumes a generalized linear regression model. The following sections present the results of the evaluation.

6.2 Datasets

The following three datasets are used for the evaluation. Please note that dataset 3 is not used to evaluate the feature descriptors because the images from dataset 1 and 3 are from the same image modality and therefore the evaluation of the feature descriptors would lead to similar results.

6.2.1 Dataset 1: Head CTs

The CT head dataset consists of six different scans taken from patients with different stages of brain tumor and terms of quality and resolution. The images were acquired in the course of the development of an intra-operative virtual endoscopy system, published by Schulze et al. [72]. The resolution of the scans ranges from $512 \times 512 \times 94$ to $512 \times 512 \times 208$ voxels per volume with voxel sizes of approximately $0.45mm \times 0.45mm \times 1mm$. This dataset contains five anatomical landmarks for each scan. All landmarks were annotated by a medical expert. Furthermore, the annotations label anatomical features that are visible in CT and MRI T1 images. More specifically, the following anatomic structures are annotated: right optical nerve exit point, left optical nerve exit point, cerebellum, caput mandibulae right and caput mandibulae left. Figure 6.5 illustrates the landmarks colored green, red, blue, magenta yellow with the corresponding maximum intensity projection of the CT scan from the first patient. The maximum intensity projections of all patients is shown in Appendix C.1.

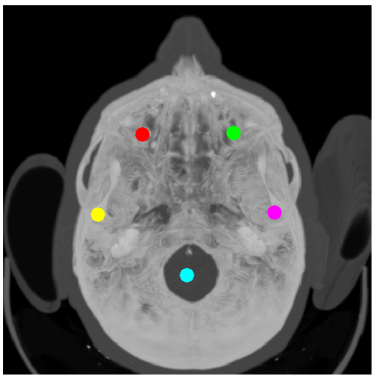


Figure 6.5: Maximum intensity projection of the first datum of dataset 1 with corresponding landmarks. The skullcap has been removed to see the interior of the brain. Figure created by author.

6.2.2 Dataset 2: Head T1 weighted MRIs

The T1 weighted MRI dataset contains the corresponding MRI T1 scans from the patients used in dataset 1. Because MRI T1 weighted scans help to see the soft tissue of the brain their variation in texture and quality are expected to be high. In contrast to the CT images, the scans have a lower resolution which ranges from $256 \times 256 \times 80$ to $256 \times 256 \times 120$ voxels. The average voxel size for the MRI scans is $0.89mm \times 0.89mm \times 2mm$. The anatomical landmarks correspond to those used for dataset 1 and were also annotated by a medical expert. Figure 6.6 illustrates the landmark positions on the maximum intensity projection of the first patient. Appendix C.2 shows maximum intensity projections of the whole dataset.

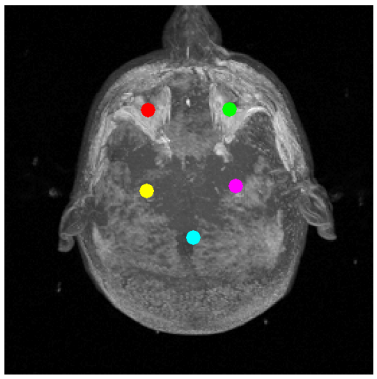


Figure 6.6: Maximum intensity projection of the first datum of dataset 2 with corresponding landmarks. Figure created by author.

6.2.3 Dataset 3: Whole-Body CTs

The whole-body CT dataset is the largest dataset containing 20 different scans with 57 annotated landmarks. The dataset is taken from the Whole Body Morphometry Project [58]. The CT scans have an average size of $512 \times 512 \times 1900$ voxels with voxel sizes of $1.3mm \times 1.3mm \times 1mm$. The manually annotated landmarks were provided by Rene Donner from the Computational Imaging Research Lab of the Medical University of Vienna. To localize all major body parts the landmarks have been distributed throughout the whole body. Those annotations are also used by Donner et al. [21] for evaluation purposes on the whole body scans. To ensure comparability with the results of Donner et al. [21] the whole body scans are also downsampled by the factor two. Figure 6.7 illustrates the landmarks on one of the CT scans. The complete dataset illustrated using maximum intensity projections can be found in Appendix C.3.

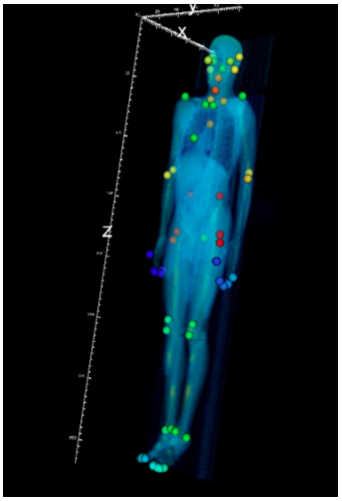


Figure 6.7: Exemplary volumetric visualization of dataset 3, Donner et al. [21].

6.3 Evaluation of Feature Descriptors

This subsections presents the results achieved by the feature descriptors described in Chapter 3. Subsequent to the experimental setup used for the evaluation, the statistical evaluation and a discussion about the robustness of the different approaches is given.

6.3.1 Experiment Setup

In order to evaluate the feature descriptors the following configurations are used. Please note that the setups are either empirically chosen or selected according to the related publication. In case of the empirically chosen setups, no parameter optimization like grid search is used.

Donner et al. [21] According to Donner et al. [21] the multivariate Gaussian distributed binary test feature uses $K = 100$ randomly chosen neighbors sampled from the zero-mean multivariate Gaussian Distribution $\mathcal{N}(\mathbf{0}, \Sigma)$ where the covariance is defined by $\Sigma = \mathrm{I} * 10mm$ where I is the 3×3 identity matrix.

BRIEF. In contrast to Calonder et al. [9] the feature vector consists of 10 independently chosen responses with $K = 8$, 10 independently chosen responses with $K = 16$, and 10 independently chosen responses with $K = 32$ pairwise binary test to keep the evaluation feasible. Therefore, each image patch is described by a 30 dimensional feature vector. The mean is defined by $\boldsymbol{\mu} = (0, 0, 0)^T$ and the covariance of the multivariate uniform distribution is set to $\Sigma = \mathrm{I} * 5mm$ where I is the 3×3 identity matrix for all responses.

gaussLBP. Similar to the feature of Donner et al. [21], $K = 100$ randomly chosen neighbors from $\mathcal{N}(\mathbf{0}, \Sigma)$ are used. The mean is defined by $\boldsymbol{\mu} = (0, 0, 0)^T$ and the covariance $\Sigma =$

$\mathrm{I} * 10mm$ of the multivariate Gaussian distribution where I is the 3×3 identity matrix is used for the offsets and the covariance matrix $\Sigma = \mathrm{I} * 3mm$ where I is the 3×3 identity matrix used for the cuboidal dimensions.

cuboidalBRIEF. For cuboidalBRIEF a setup similar to the configuration of BRIEF is used. The covariance matrix of the multivariate Gaussian distribution used for the offsets of each of the 30 responses is defined by $\Sigma = \mathrm{I} * 5mm$ where I is the 3×3 identity matrix whilst the cuboidal dimensions are sampled from a multivariate uniform distribution with the covariance matrix $\Sigma = \mathrm{I} * 3mm$ where I is again the 3×3 identity matrix. The mean is defined by $\boldsymbol{\mu} = (0, 0, 0)^T$ for both cases.

6.3.2 Statistic Evaluation Results

To keep the statistic evaluation using MDVV and maximum density of the feature descriptors feasible the evaluation measurements are computed over a random subset of all feature vectors of a complete training set. More detailed, 10% of all vectors over all images on dataset 1 and 2 are used. The size of the random subset are empirically chosen. To get univariately distributed responses over the complete image space, the samples are sampled using a univariate distribution. To get a better understanding of the evaluation results and to prevent results to be falsified by the dimension sizes, the measurements are performed according to all X-displacements, all Y-displacements and all Z-displacements separately.

MDVV Results

In order to evaluate the MDVV of the different approaches, the median over the MDVV values are calculated. In contrast to the mean the median is insensitive against outliers to some extend as explained by Huber [36]. Table 6.1 shows the results achieved by the individual feature descriptors in terms of the MDVV. Please recall that small values for the MDVV are desirable. Numbers written in bold illustrate the best result per column. Please note that because of the heterogeneous feature descriptor responses of BRIEF it is not possible to estimate proper MDVV values for BRIEF on dataset 1.

It can be seen that the proposed approaches, GaussLBP and cuboidalBRIEF, slightly outperform the approach of Donner et al. [21] and BRIEF on dataset 2. Please note that cuboidalBRIEF generally outperforms BRIEF and the approach by Donner et al. [21] on dataset 2 whilst GaussLBP outperforms the other approaches only in the X and Y directions. In the case of dataset 1 GaussLBP outperforms all other evaluated feature descriptors.

Table 6.1: Median values for maximum dependent variable variance, groups with variance $= 0$ are ignored.

Method	Dataset 1			Dataset 2		
	X	Y	Z	X	Y	Z
Donner et al. [21]	9383.56	9334.82	76.20	2090.92	1699.17	38.84
BRIEF	-	-	-	2100.96	1699.17	43.37
GaussLBP	**8075.78**	**7498.09**	**10.15**	**1883.81**	**1549.52**	43.37
cuboidalBRIEF	81142.73	7666.04	94.67	2044.33	1660.59	**30.98**

Maximum Density Results

To evaluate the maximum density of fitted kernel density estimations, similar to the evaluation of MDVV, the median over the maximum densities are calculated. To estimate the non-parametric distributions a Gaussian kernel is used for the Parzen window estimator. More specifically the Matlab function `ksdensity` with the default configuration is used. Table 6.2 shows the median values of the maximum densities for the different approaches. Please recall that a high maximum density value reflects narrow density estimations which is a desirable property. Similar to the evaluation using MDVV it is not possible to estimate maximum density values for BRIEF responses on dataset 1.

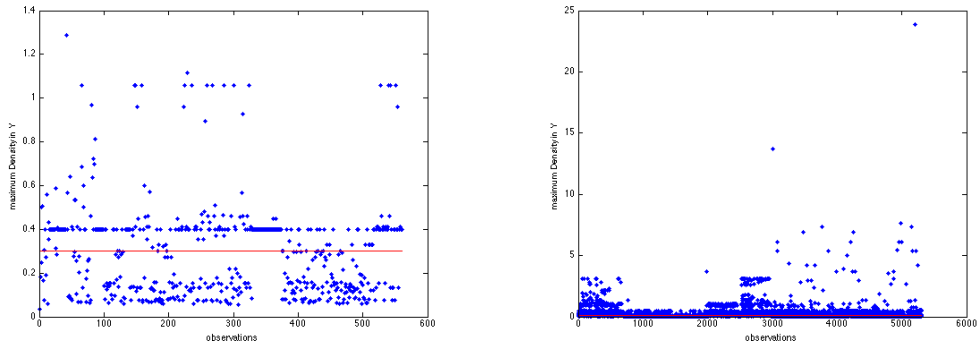
Table 6.2: Median values for maximum density estimations calculated using kernel density estimation with Gaussian kernel, groups with maximum density $= 0$ are ignored.

Method	Dataset 1			Dataset 2		
	X	Y	Z	X	Y	Z
Donner et al. [21]	48.18	63.78	48.91	20.41	118.83	29.77
BRIEF	-	-	-	6.10	**328.81**	19.72
GaussLBP	**55.81**	**74.00**	**86.39**	**48.22**	63.78	44.65
cuboidalBRIEF	28.08	34.22	58.62	14.65	76.76	**47.62**

All numbers are in the notation of $\times 10^{-3}$.

It is interesting to note that for dataset 2 gaussLBP outperforms BRIEF and the approach by Donner et al. [21] in all displacement dimensions except the Y-dimension. In general the novel descriptors perform less good in the Y-dimension for this dataset. Figure 6.8 visualizes the distributions of maximum density values for BRIEF and cuboidalBRIEF. It can be seen that the samples size of BRIEF is much smaller then of cuboidalBRIEF, this is due to the fact that the maximum density can only be estimated if a non-parametric distribution can be estimated over

the output space of a specific feature vector. However, BRIEF results in many distinct vectors which results in a small number of possible distribution estimations. Which is the reason for the large maximum density value in the Y-dimension of BRIEF. Furthermore, it can be seen in Table 6.2 that GaussLBP outperforms all other evaluated feature descriptors on the first dataset.



(a) (b)

Figure 6.8: Visualization of maximum density distributions in the Y-dimension for BRIEF (a) and cuboidalBRIEF (b). The red line indicates the median value. The ordinate shows the observations of feature vector categories and the abscissa shows the maximum density values of the feature vector categories. Figures created by author.

6.3.3 Noise Sensitivity Results

The robustness of the different feature descriptors are evaluated based on the L_1 difference between the feature vectors on the original image and a noisy volume. The L_1 difference values are obtained by computing the L_1 distance between the corresponding feature vectors at each voxel location. In order to obtain noisy volumes in the first test case a Poisson distribution is used and in the second test case a Rice distribution is used. Because the ranges of the feature vectors for LBP based approaches and BRIEF based approaches are different two independent result tables are used.

Table 6.3 shows the comparison of the approach proposed by Donner et al. [21] and GaussLBP. It can be seen that the extension using cuboidal regions gives additional stability for both noise classes. Moreover, GaussLBP captures the main features of the image more robust for Rice noisy images than the approach of Donner et al. [21]. It should be noted that GaussLBP captures the neighboring texture relative to the intensity of a single voxel but still achieves an improvement of stability.

As shown in Table 6.4 the extension of BRIEF with cuboidal regions improves the noise robustness significantly. In contrast to the LBP based approaches that benefit from the large number of neighbors and the direct use of the binary vector BRIEF is highly sensitive to noise. Due to the dimension reduction step in the case of the test configuration the response can change

Table 6.3: L_1 comparisons of LBP based features of Poisson noisy volume and Rice noisy volume to original volume.

Method	Poisson Noisy			Rice Noisy		
	Mean	Median	Variance	Mean	Median	Variance
Donner et al. [21]	0.1514	0.00	0.1710	0.2033	0.00	0.2454
GaussLBP	**0.0994**	0.00	**0.1222**	**0.0725**	0.00	**0.0960**

by 2^{32}. The response of cuboidalBRIEF benefits from the intensity averaging which dramatically reduces the effect of Poisson distributed noise.

Table 6.4: L_1 comparisons of BRIEF based features of Poisson noisy volume and Rice noisy volume to original volume.

Method	Poisson Noise			Rice Noise		
	Mean	Median	Variance	Mean	Median	Variance
BRIEF	16.04	0.00	3288.66	30.57	8.00	4467.47
cuboidalBRIEF	**0.24**	0.00	**4.08**	**15.72**	**2**	**7.58**

6.3.4 Conclusion

Based on the statistical measurements presented it can be concluded that the proposed features, especially GaussLBP, outperform the approach by Donner et al. [21] and BRIEF. However, it should be mentioned that this does not guarantee a better performance if used in combination with a regression model. In the case of BRIEF it can be seen that the cuboidal extension (cuboidalBRIEF) almost always outperforms the original feature descriptor. Because medical images are always affected by noise and the scanned tissues can strongly vary in terms of shape, location and intensity it is important that the feature descriptor is robust against noise and image variation. CuboidalBRIEF outperforms BRIEF regarding noise insensitivity and therefore represents a reasonable candidate as medical image feature descriptor.

6.4 Evaluation of the Landmark Prediction Approach

This subsections stepwise presents the results of the anatomical landmark prediction system. At first the results of state of the are methods are introduced and the following subsection discusses the results of the parameter optimization used to analyze the effect of the learning rate and the

Δ-deviation on the RobustBRRFerns. Subsequently the results on the global prediction model using the four feature descriptors discussed are given. The last subsections present the results achieved by the local refinement step and discuss the run times of the presented approach. Please note that for the local refinement only the most promising feature descriptor according to the global prediction results is used.

6.4.1 State of the Art Results

This subsection introduces the results achieved by the approach of Pauly et al. [64], Cuingnet et al. [18], and Donner et al. [21]. To compare the multi-pass anatomical landmark prediction approach with the state of the art methods the technical details and results of the different approaches are briefly discussed.

Pauly et al. [64]

The approach by Pauly et al. [64] introduces random regression ferns for multiple organ detection and localization. Pauly et al. [64] present a novel regression based method for automatic prediction of positions and bounding boxes of multiple organs on MR Dixon channels. Pauly et al. [64] compute features over two channels and combine the feature vectors for each voxel to one feature vector. Instead of evaluating every voxel only every fourth voxel will be described using the 3D LBP computed over 26 cuboidal regions on three different scales. Moreover, the results are obtained using 14 individual random regression ferns each with six nodes. The approach of Pauly et al. [64] achieve a mean localization error for the organs head, left lung, right lung, liver and heart of $14.95mm$ with a standard deviation of $11.33mm$. Donner et al. applied the approach by Pauly et al. [64] on dataset 3 using the same annotations used in this thesis. Therefore, the prediction error values on dataset 3, published by Donner et al. [21], can be directly compared to those of this thesis and are shown in Table 6.5. The approach by Pauly et al. [64] takes in average $0.7\,\mathrm{sec}$ for an average volume with a size of 192×433.

Table 6.5: Prediction results of four-fold cross-evaluation on dataset 3 published by Donner et al. [21] (in mm).

Method	Mean	Median	Std
Random Regression Ferns	54.80	43.37	86.98

Cuingnet et al. [18]

Cuingnet et al. [18] published an automatic segmentation approach for kidneys shown on CT images. The approach is based on a multi-hypothesis approach using random forest for localizing the kidneys. The approach uses random forest for regressing the coarse positions of the kidneys and for refining the regions of interest by fitting an ellipsoid. The approach is evaluated using

multiple forests, each consisting of seven trees with a depth of 15. Cuingnet et al. [18] achieve in the landmark detection phase a mean deviation of the coarse kidney location of $23mm$ for the left kidney and $26mm$ for the right kidney. The refined locations deviate by $11mm$ for the left and $10mm$ for the right kidney.

Donner et al. [21]

The method of Donner et al. [21] proposes a global localization of anatomical structures using Hough forests [82] and Markov random fields [42] for model based estimation of the landmark positions. The approach uses three steps to estimate the landmark positions. In the first step random forest classifiers obtain regions of interest which are rough estimates of the landmark locations. The second step uses Hough forest for regressing the landmark locations based on the local neighborhood. A Markov random field is used to obtain the accurate locations of the landmarks. The random forest classifier and the Hough forest regressor are trained using 32 extremely randomized trees. The tree depth has not been limited. To efficiently predict the landmark locations the approach uses downsampled volumes instead of the full resolution scans. The results achieved by Donner et al. [21] on dataset 3 are shown in Table 6.6. The average run time for a volume from dataset 3 using the approach of Donner et al. [21] is $120\,\mathrm{sec}$.

Table 6.6: Prediction results of four-fold cross-evaluation on dataset 3 published by Donner et al. [21] (in mm).

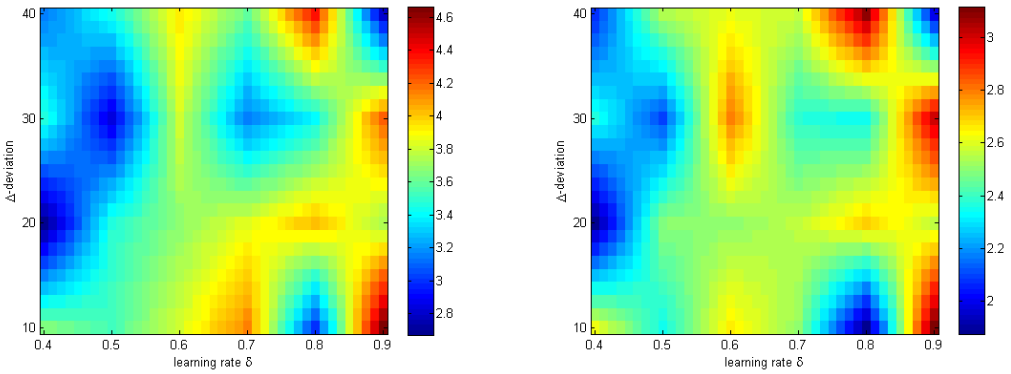
Method	Mean	Median	Std
Donner et al. [21]	5.25	2.71	15.08

6.4.2 Parameter Estimation

To asses effective values for the learning rate δ and the allowed Δ-deviation the parameters are optimized using grid search evaluated on dataset 1 and 2. This thesis does not provide an exhaustive parameter space evaluation of the whole body dataset because the parameter space is assumed to be similar to the one of the first dataset. Therefore, in this thesis the optimal configuration for dataset 1 is also used for tests on dataset 3. Lengyel et al. [45] first introduced an approach for optimizing the configuration space for robot motion planning which is comparable to grid search for parameter optimization. The effect of the learning rate is evaluated for values between $\delta = 0.4$ and $\delta = 0.9$ with a step size of 0.1 and the Δ-deviation is evaluated between $\Delta = 10$ and $\Delta = 40$ with a step size of 10.

Figure 6.9 illustrates the parameter space with the corresponding loss values colored from dark blue (small loss) to red (high loss) evaluated on dataset 1 using LAD and RSS as loss functions. Please note that the figures shown are interpolated versions of the original grids. It can be seen that the region around the configuration of $\delta = 0.5$ and $\Delta = 30$ is a stable area with

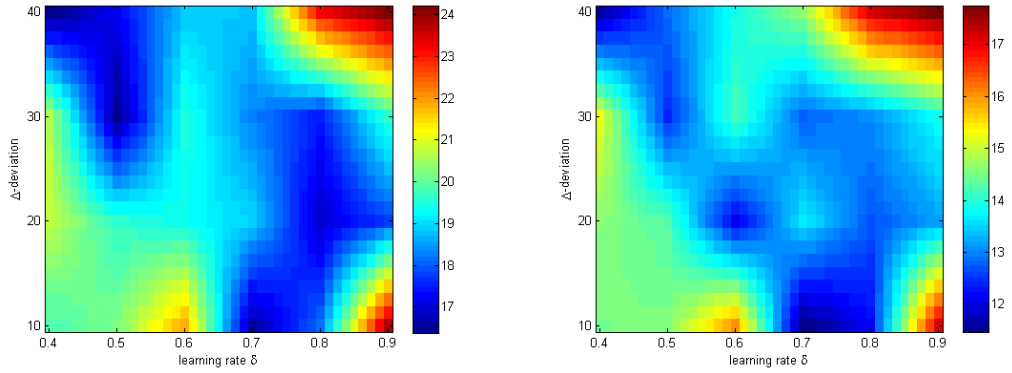
small loss. Therefore, this configuration is used for further evaluations. Moreover, it can be seen that increasing the learning rate up to a value of 0.9 has a negative effect on the performance if the Δ-deviation values lie above or underneath $\Delta = 30$.



(a) (b)

Figure 6.9: Visualization of the parameter space on dataset 1 with LAD as loss function (a) and with RSS as loss function (b). Small loss values are mapped dark blue color whilst high loss values are mapped to red colors. Figures created by author.

Figure 6.10 illustrates the evaluation of the parameter space on dataset 2 using LAD and RSS as loss functions. In contrast to the configuration space evaluated on dataset 1 a learning rate of $\delta = 0.7$ with $\Delta = 10$ outperforms other configurations on dataset 2. Please note that the parameter space for dataset 1 also shows low loss values around $\delta = 0.7$ if $\Delta = 30$. Because the region around $\delta = 0.8$ and $\Delta = 20$ is the most stable region with low loss for further evaluations on dataset 2 the learning rate is set to $\delta = 0.8$ and the Δ-deviation of $\Delta = 20$ is used.



(a) (b)

Figure 6.10: Visualization of the parameter space on dataset 2 with LAD as loss function (a) and with RSS as loss function (b). Small loss values are mapped dark blue color whilst high loss values are mapped to red colors. Figures created by author.

Outlier Removal Configuration

The configuration for the FAST-MCD were selected empirically without exhaustive parameter optimization. For the second dataset the size of the quantile of the observations used to minimize their covariance determinant is set to $h = 20$ which is similar to the example configuration given in the documentation.

6.4.3 Global Prediction Results

Results of the global prediction phase are computed using the following setups. According to the definition of the multi-pass landmark prediction model in Chapter 5 all prediction are estimated using sparse sampled features on downsampled volumes. For the purpose of this evaluation, all volumes are by the factor of two. Moreover, the features are only computed at every fourth voxel in the X-dimension, every fourth voxel in the Y-dimension and every fourth voxel in the Z-dimension which makes every 64th voxel in the downsampled volume. All tests have been performed using the loss functions introduced in Subsection 6.1.2. Please note that for evaluation purposes the mean and the median of the model loss values are used. To get a better understanding of the depicted deviations the standard deviation values and the median absolute deviation values are given respectively in the tables. Please note that the loss values used in the following comparison tables denote the deviations of the predictions after transformation to the original volumes. The transformed predictions are obtained by multiplying their positions by two.

Experiment Setup

In the case of dataset 1 the optimal model parameters $\delta = 0.5$ and $\Delta = 30$ are used for the *leave one out evaluation*. For dataset 2 the stable model parameters $\delta = 0.8$ and $\Delta = 20$ are used for each of the six folds of the *leave one out evaluation*. In the case of dataset 3 the same parameter configuration as for dataset 1 is used in each fold of the four-fold cross-validation. To obtain the results for the different test the number of ferns is set to $Z = 14$ and the number of nodes is set to $L = 6$ which is similar to the configuration proposed by Pauly et al. [64]. Please note that this setup is used for random regression ferns and RobustBRRFerns. Moreover, the a weighting factor of $\kappa = 100$ is used. Location predictions with confidence lower than 0.5 are removed.

Results: Dataset 1

Table 6.7 shows the results achieved on dataset 1 measure in mm. It can be seen that cuboidal-BRIEF outperforms all analyzed feature descriptors on this dataset. Moreover, it should be mentioned that a mean deviation of $9.34mm$ is significantly smaller than the mean localization error of $14.95mm$ published by Pauly et al. [64]. Moreover, by comparing the best results achieved by RobustBRRFerns and by random regression ferns it can be seen that the RobustBRRFerns generally outperform the random regression ferns (cf. Table 6.8). Please note that the outlier removal using FAST-MCD is applied on all test cases.

Table 6.7: Global prediction results of *leave one out evaluation* on the original volumes of dataset 1 using RobustBRRFerns measured in mm.

Feature Descriptor	LAD		RSS	
	Mean	Median	Mean	Median
Donner et al. [21]	15.06 ± 4.92	15.20 ± 4.16	11.08 ± 4.26	10.84 ± 3.72
GaussLBP	18.84 ± 7.22	18.04 ± 5.80	12.88 ± 5.54	11.84 ± 4.30
BRIEF	18.86 ± 6.88	17.70 ± 5.42	13.80 ± 6.12	12.66 ± 4.74
cuboidalBRIEF	**12.54** ± 7.02	**11.28** ± 5.28	**9.34** ± 5.96	**7.86** ± 4.28

Table 6.8: Global prediction results of *leave one out evaluation* on the original volumes of dataset 1 using random ferns measured in mm.

Feature Descriptor	LAD		RSS	
	Mean	Median	Mean	Median
Donner et al. [21]	**23.16** $\pm$ 7.38	22.50 $\pm$ 5.68	**16.28** $\pm$ 5.96	**14.22** $\pm$ 5.12
GaussLBP	24.86 $\pm$ 11.28	**21.46** $\pm$ 9.30	17.48 $\pm$ 8.82	14.58 $\pm$ 6.90
BRIEF	26.42 $\pm$ 11.70	24.18 $\pm$ 9.35	20.94 $\pm$ 11.10	18.42 $\pm$ 8.68
cuboidalBRIEF	24.94 $\pm$ 7.82	26.34 $\pm$ 5.94	18.44 $\pm$ 6.54	18.28 $\pm$ 5.40

Results: Dataset 2

The results for the global prediction phase on dataset 2 are shown in Table 6.9. The results obtained using random ferns are depicted in Table 6.10. No loss values are shown in Table 6.10 for the BRIEF response, because the random fern model is not able to successfully cluster the feature space. Please note that the location errors are significantly higher then those for dataset 1. One reason for the high deviations is the small amount of intensity variations on the MRI T1 scans, which results in large textural homogenous areas. Another reason is the smoothing of the texture due to downsampling.

Table 6.9: Global prediction results of *leave one out evaluation* on the original volumes of dataset 2 using RobustBRRFerns measured in mm.

Feature Descriptor	LAD		RSS	
	Mean	Median	Mean	Median
Donner et al. [21]	72.60 $\pm$ 63.64	49.66 $\pm$ 47.50	64.08 $\pm$ 61.06	42.18 $\pm$ 45.52
GaussLBP	93.28 $\pm$ 70.78	75.68 $\pm$ 56.44	79.16 $\pm$ 66.34	55.80 $\pm$ 53.38
BRIEF	76.18 $\pm$ 61.94	44.90 $\pm$ 45.26	57.82 $\pm$ 59.08	38.58 $\pm$ 43.22
cuboidalBRIEF	**67.64** $\pm$ 87.40	**30.84** $\pm$ 63.38	**49.60** $\pm$ 64.42	**22.18** $\pm$ 46.88

Table 6.10: Global prediction results of *leave one out evaluation* on the original volumes of dataset 2 using random ferns measured in mm.

Feature Descriptor	LAD		RSS	
	Mean	Median	Mean	Median
Donner et al. [21]	**112.00** $\pm$ 50.42	**85.70** $\pm$ 40.04	**101.52** $\pm$ 50.40	**79.76** $\pm$ 40.76
GaussLBP	117.74 $\pm$ 62.02	99.88 $\pm$ 49.38	104.92 $\pm$ 58.38	87.86 $\pm$ 46.28
BRIEF	–	–	–	–
cuboidalBRIEF	211.08 $\pm$ 126.42	135.16 $\pm$ 106.82	134.74 $\pm$ 70.14	110.84 $\pm$ 56.66

Results: Dataset 3

For the whole body dataset only the most promising feature descriptor variant is used for the cross-validation. Table 6.11 shows the results obtained on the downsampled volumes using cuboidalBRIEF as feature descriptor. Moreover, Figure 6.11 illustrates the distribution of the deviations for the 57 individual landmarks obtained using RobustBRRFerns and cuboidalBRIEF on dataset 3. Please note that the deviation values depicted in Figure 6.11 correspond to RSS values measured in $\frac{mm}{2}$. The groups shown in the boxplot can be interpreted as follows: 1-10 = left and right finger tips, 11 - 20 = left and right toe tips, 21 - 57 = landmarks spread over the whole body e.g., left knee, right knee. It can be seen that the landmarks 21 - 57 show low deviation than the landmarks 1 - 20. This effect is interesting because it is reasonable to assume that the prediction of single finger or toe tips is difficult in a global context.

Table 6.11: Global prediction results of four-fold cross-evaluation on dataset 3 using RobustBRRFerns measured in mm.

Feature	LAD		RSS	
	Mean	Median	Mean	Median
cuboidalBRIEF	25.34 $\pm$ 12.36	23.56 $\pm$ 9.58	15.72 $\pm$ 8.20	14.46 $\pm$ 6.44

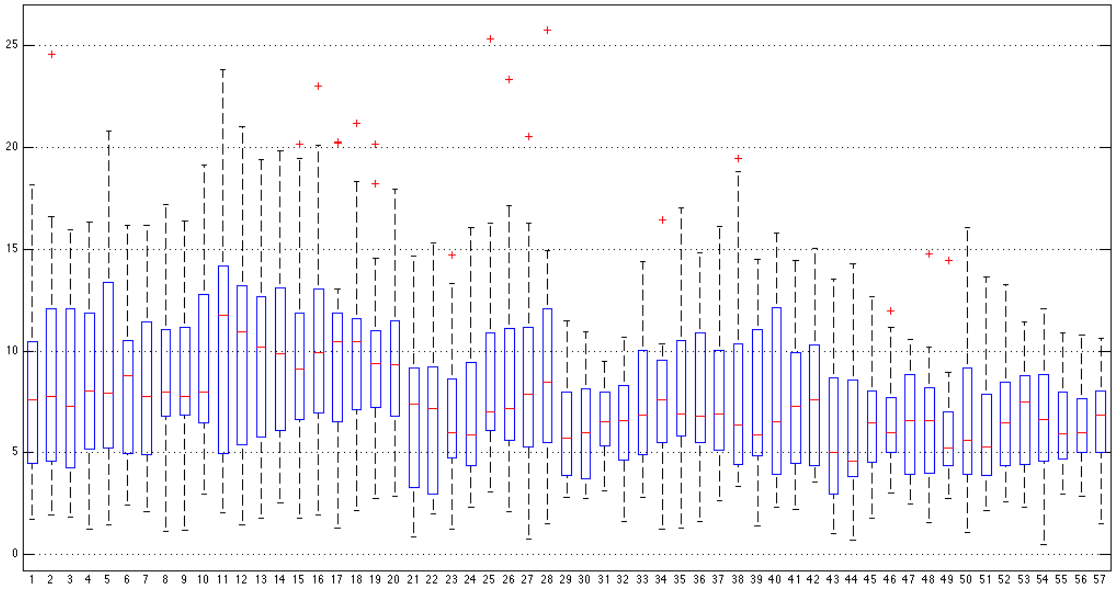


Figure 6.11: Boxplot of landmark deviations obtained using four-fold cross-evaluation on dataset 3 measured in $\frac{mm}{2}$ with RSS. Outliers are indicated using red crosses on the box plot. The ordinate shows the 57 anatomical landmarks (1-10 = left and right finger tips, 11 - 20 = left and right toe tips, 21 - 57 = landmarks spread over the whole body) and the abscissa the deviation from the ground truth data. Figure created by author.

Conclusion

The tests on the three different datasets using the global prediction stage show that cuboidalBRIEF outperforms all feature descriptors discussed in this thesis. However, the results of GausssLBP are unexpectedly bad. According to the feature evaluation using MDVV and maximum density a much lower loss was expected. The outstanding performance of cuboidalBRIEF leads to the decision that this feature descriptor is used in further evaluations. Because cuboidalBRIEF describes image patches located at each of the voxel locations this finding correlates with the results described by Donner and Bishop [22] on 2D medical images. Moreover, on average the RobustBRRFerns produce prediction errors which are half the prediction errors of the original random regression fern model. The achievements of the global prediction phase are summarized in Table B.2.

6.4.4 Local Prediction Results

This subsection presents the results achieved by the local refinement stage of the multi-step procedure for anatomical landmark localization. According to the results of the global prediction phase only the feature descriptor cuboidalBRIEF is used for the evaluation. Moreover, the local refinement is only applied and evaluated on dataset 3 and partially on dataset 1. In contrast to those datasets, the second dataset contains a very high average deviation and therefore a

local refinement will not improve the results sufficiently. For this dataset it is more reasonable to enhance the discriminativity of the feature descriptor responses instead. Dataset 1 contains landmarks that are visible in CT and MRI images. However, only two of the five landmarks contain enough expressive texture information in their vicinity. Therefore, only the landmarks for the left and right optical nerve are used. Please note that in contrast to the global phase the local refinement uses the original volumes.

Experimental Setup

For dataset 1 and 3 the same optimal model parameters $\delta = 0.5$ and $\Delta = 30$ are used. To evaluate the local refinement on dataset 2 the model parameters $\delta = 0.8$ and $\Delta = 20$ are used. Similar to the global prediction tests $Z = 14$ individual ferns with $L = 6$ nodes have been used for the RobustBRRFerns. To train the regressors, $N = 10000$ randomly sampled image patch descriptions from a multivariate Gaussian distribution with mean $\boldsymbol{\mu} = \mathbf{p}^k$ where $\mathbf{p}^k \in \mathbb{R}^3$ is the position of the kth landmark and covariance $\Sigma = \mathrm{I} * 2\,\mathrm{E}[L_k(h(\mathbf{x}))]$ where I is the 3×3 identity matrix and $\mathrm{E}[L_k(h(\mathbf{x}))]$ is the expectation value of the model loss for the kth landmark of the global prediction phase are used to build the model. The features for each image patch are computed using cuboidalBRIEF with $K = 30$ individually chosen responses of randomly sampled binary tests of different scales. More precisely, multivariate uniform distributions with $\Sigma = \mathrm{I} * 5mm$, $\Sigma = \mathrm{I} * 7.5mm$ and $\Sigma = \mathrm{I} * 10mm$ where I denotes the 3×3 identity matrix are used. The setup is chosen empirically out of different feature scales and regression model configurations. For the *robust weak learner* the weighting factor is set according to $\kappa = 10$. Similar to the approaches of Pauly et al. [64] and Donner et al. [21], predictions with confidence lower than 0.5 are removed.

Additionally, the refinement step is also evaluated using a local classification step in the vicinity of the estimates with random ferns. For the classification $Z = 60$ trees are used. In this thesis the random forest implementation of Liaw and Wiener [47] is used without further modifications. In order to train the trees, a random set of feature vectors whose corresponding voxel positions are very close, L_2-distance $< 4mm$, are used as positive observations whilst more distant responses are considered to be negative observations. More precisely, $N = 10000$ randomly sampled observations are extracted from each training volume. To obtain the observations the multivariate Gaussian distribution $\mathcal{N}(\mathbf{p}^k, \Sigma)$ with $\Sigma = \mathrm{I} * 2\,\mathrm{E}[L(h^k)]$ with I being the 3×3 identity matrix is used. The refined landmark positions are estimated by first applying particle filtering (cf. Doucet and Johansen [26]) with a sphere of radius $r = 10mm$ on the prediction volume to obtain a density volume. After applying minimum density suppression on the density volume the landmark position are estimated by calculating the mean position over the remaining 10% predictions with highest density. Using this approach allows a robust and outlier insensitive estimation of the locations. In contrast to the mean location of all positive samples, outlying samples will be suppressed by their neighborhood. Please note that the initial position is kept if the random forest classifier is not able to classify at least one sample as positive sample.

Results

Table 6.12 and 6.13 compare the local refinement results obtained using RobustBRRFerns and randomForest using cuboidalBRIEF to the global prediction errors, bold value indicate the best result per row. The results achieved using the local refinement phase are unexpectedly similar to those of the global prediction phase. Especially in the case of dataset 3 a much higher decrease of the model loss is expected.

Table 6.12: Comparison of global and local refined prediction results obtained using RobustBRRFerns, randomFerns and cuboidalBRIEF on dataset 1 measured in mm.

		Dataset 1		
		global	global & local (regression)	global & local (classification)
LAD	Mean	12.54 ± 7.02	11.56 ± 7.24	**11.24** ± 6.00
	Median	11.28 ± 5.28	10.10 ± 5.30	**9.52** ± 4.75
RSS	Mean	9.34 ± 5.96	8.61 ± 6.15	**8.37** ± 5.09
	Median	7.86 ± 4.28	7.52 ± 4.30	**7.09** ± 4.03

Table 6.13: Comparison of global and local refined prediction results obtained using RobustBRRFerns and cuboidalBRIEF on dataset 3 measured in mm.

		Dataset 3		
		global	global & local (regression)	global & local (classification)
LAD	Mean	25.34 ± 12.36	25.12 ± 13.89	**21.03** ± 17.81
	Median	23.56 ± 9.58	22.95 ± 10.92	**19.55** ± 15.44
RSS	Mean	15.72 ± 8.20	15.59 ± 8.62	**13.05** ± 11.05
	Median	14.46 ± 6.44	14.24 ± 6.78	**12.00** ± 9.58

Discussion

In the approach presented in this thesis the initial guesses for the local refinement depend on the previous predictions of the global phase. The global model allows location votes for all landmarks from every selected voxel, i.e. every 64th voxel. However, this does not guaranty that the landmark prediction lies in a texture rich voxel location. Therefore, it is possible that initial guesses are located in areas that make a local optimization difficult. Moreover, some landmarks

- especially the finger or toe tips - seem to converge in approximately 50% of the cases to a local optimum by moving to a neighboring finger or toe tip.

As shown in Table 6.12 and 6.13 the local refinement using the proposed regression model does not change the results significantly. Especially on dataset 3, only approximately 60% of all landmark position deviations decreased. The local models that deteriorated the predicted positions tend to move faster away from the actual landmark location than those that converged towards the actual positions. Therefore, the gain of the local refinement phase using the regression approach is only very small. The local refinement using a classification model however, generally improved the results. Please note that for dataset 3 the classification model is not able to detect the landmark in approximately 60% of all cases and therefore, the location is only adapted in 40% of all cases. For those landmark locations where a refined location is possible to calculate, an average decrease of the RSS loss of $16.94mm$ is achieved. Please note that in approximately 90% of the cases where all samples are classified as negative samples the initial deviation is smaller than the neighborhood used for the classification. Because the local refinement using a classification model reduced the model loss more than the refinement using the regression approach, the local refinement based on local classification is suggested. The achievements of the local refinement phase are summarized in Table B.1.

Outlook

Possible solutions to improve the results of the local refinement stage are the following. Using a multi-hypothesis global prediction approach can provide multiple initial guesses for the local optimization. In order to distinguish between guesses that converge to the landmark location and those that do not, the trajectory of the optimization can be analyzed. By iteratively applying the regression model to predict displacement vectors the mean length of the displacement vectors should decrease if the landmark prediction approaches the real landmark position. In the empirical evaluation of the local refinement behavior this assumption was true in more than 80% of the cases whilst the mean length of the displacement vector for wrong optimizations tend to increase. Alternatively, a multi-hypothesis global prediction approach could also be used to estimate multiple density volumes and computing an intersection density map that provides a more robust estimation of the global landmark positions.

Additional to the modifications of the prediction procedure the cuboidalBRIEF feature with the described setup smoothes the texture too much. In contrast to the global prediction phase, in which cuboidalBRIEF outperforms all evaluated feature descriptors, this feature is not optimal for capturing local changes of the medical images. Therefore, a more sensitive feature could increase the refinement accuracy.

6.5 Run times

The run times of the approach described in this thesis are obtained on a computer with an Intel Core i7 processor with 8 cores and 16 GB ram. The run times are determined by the prediction time for the global prediction phase and the local refinement phase. Please note that the training time for the RobustBRRFern models is significantly higher than those for random ferns. This is

mainly because of the reweighting scheme which requires a repeated computation of the depending variables and an evaluation of the model loss. Because the models need to be trained only once and the training can be carried out very efficiently on a large scale CPU cluster the training timings are been analyzed in detail. However, it is reasonable to assume that on the mentioned computer the training of each local model used for dataset 3 takes approximately $10\,\mathrm{min}$.

Dataset 1 The global prediction of all landmark locations on the first dataset takes an average running time of $3\,\mathrm{sec}$. The refinement of each of the two landmarks used in the local stage takes on average only $0.3\,\mathrm{sec}$ per landmark. This implies that the whole pipeline for a common volume from dataset 1 takes less than $4\,\mathrm{sec}$. If all five landmarks are used in the local refinement the multi stage approach takes approximately $6\,\mathrm{sec}$ for one volume.

Dataset 2 On the second dataset only the global prediction times are captured and evaluated because a local refinement will not improve the results sufficiently. The regression model estimates all landmark locations on average in less than $2\,\mathrm{sec}$.

Dataset 3 The global prediction phase using every 64th voxel on the downsampled volumes of dataset 3 takes on average $7\,\mathrm{sec}$ to estimate all landmark locations. The local refinement phase using iterative adjustment with three iterations of local predictions by applying a RobustBR-RFern ensemble takes for one landmark an average time of $0.7\,\mathrm{sec}$. This sums up to $40\,\mathrm{sec}$ for an average volume of dataset 3. Therefore, the whole multistage landmark prediction method has an approximate running time of $50\,\mathrm{sec}$.

6.6 Summary

The performance of machine learning algorithms depends on the quality of the features used for the classification or regression (cf. Mayer [53]). To estimate the quality of the feature descriptor the discernibility is evaluated using the MDVV and the maximum kernel density. The robustness of the descriptors is evaluated against Poisson and Rice distributed noise. The feature descriptor response of the GaussLBP and the cuboidalBRIEF show significant improvements in comparison to the approach used by Donner et al. [21] and BRIEF. Especially the feature descriptor GaussLBP outperforms the other feature descriptors according to the MDVV and the maximum kernel density values.

In order to obtain estimates of the model loss the performance is calculated using K-fold cross validation. The proposed approach is evaluated on three datasets. The first two dataset are CT and MRI T1 weighted scans of the human head, whilst the last dataset contains 57 annotated landmark in whole body CT scans. It can be summarized, that the global prediction phase of the proposed multi-pass model achieves a mean deviation of $9.34mm$ on the first dataset, $30.84mm$ on the second dataset and $15.72mm$ on the whole body dataset. Because of the high deviations on dataset 3 local refinement phase is only evaluated on dataset 1 and 3. However, the local refinement improved the initial results only little, from $9.34mm$ to $8.37mm$ on dataset 1 and from $15.72mm$ to $13.05mm$ on dataset 3.

CHAPTER 7

Discussion and Future Work

This chapter discusses the results achieved by the multi-pass anatomical landmark prediction system and compares the achievements to three state of the art approaches. Furthermore, a conclusion of the thesis and a perspective on future work is given in Section 7.3.

7.1 Discussion

The experimental results demonstrate the performance of the proposed generic approach for anatomical landmark prediction. A detailed discussion of the approach and a comparison to other state of the art methods is given in subsection 7.1.1. Moreover, the limitations of the proposed approach are discussed in subsection 7.1.2.

7.1.1 Comparison to State of the Art

The results and the technical details of three state of the art approaches are introduced in Chapter 6. This subsection gives discusses the achievements of the approach by Pauly et al. [64], Cuingnet et al. [18] and Donner et al. [21] to those of this thesis.

Pauly et al. [64]

Table 7.1 shows the results of Pauly et al. [64], the global prediction phase only and the multi-pass model with local refinement on dataset 3. Because Donner et al. [21] applied the approach of Pauly et al. [64] on dataset 3, the achievements can be directly compared to those of this thesis. It can be seen that the proposed approach clearly outperforms the Random Regression Ferns by Pauly et al. [64] on this dataset. Please note that the approach of his thesis downsamples the volumes for sparse feature extraction in the global prediction phase and still outperforms the approach by Pauly et al. [64].

Table 7.1: Prediction results of four-fold cross-evaluation on dataset 3 published by Donner et al. [21] (in mm).

Method	Mean	Median	Std
Random Regression Ferns	54.80	43.37	86.98
RobustBRRFerns (global)	15.72	14.46	8.20
RobustBRRFerns (local)	13.05	12.00	11.05

Cuingnet et al. [18]

The segmentation approach by Cuingnet et al. [18] is evaluated on 233 CT scans from 89 patients. Because this publication discusses an automatic localization and an automatic segmentation approach only the first part of the paper can be compared with the approach of this thesis. Considering the whole body dataset the RobustBRRFerns achieve a mean deviation of $12.00mm$ which is close to the refinement deviation of $11mm$ for the left and $10mm$ for the right kidney. It should be noted that the refinement phase is a model based approach to give a starting point for the kidney segmentation. The approach of this thesis does not induce anatomical knowledge by using e.g., a model based representation of the shape of anatomical structures or a graph based representation of the relation between the landmarks.

Donner et al. [21]

The approach of Donner et al. [21] is a complex approach to estimate proper landmark locations. As shown in table 7.2, Donner et al. [21] obtain a mean deviation of $5.25mm$ with a standard deviation of $15.08mm$ on the third dataset. Using the method described in this thesis a larger mean deviation of $13.05mm$ is achieved. Please note that Donner et al. [21] follow a similar approach including classification of promising regions and refining the locations using Hough forests. Donner et al. [21] describe that the prediction accuracy improved especially during the refinement step. However, in this thesis a significant improvement due to the refinement phase using the described setup can not be observed. Therefore, further research of the refinement phase is necessary. Even though the accuracy of the approach by Donner et al. [21] outperforms the method described in this thesis, it should be emphasized that Donner et al. [21] need $120\,\mathrm{sec}$ for an average volume of dataset 3 whilst the proposed approach estimates all landmark locations in less than a minute.

Table 7.2: Prediction results of four-fold cross-evaluation on dataset 3 published by Donner et al. [21] (in mm).

Method	Mean	Median	Std
Donner et al. [21]	5.25	2.71	15.08
RobustBRRFerns (global)	15.72	14.46	8.20
RobustBRRFerns (local)	13.05	12.00	11.05

7.1.2 Limitations

In contrast to the original random regression ferns the RobustBRRFerns adjust the clustering of the input space according to the last trained fern which results in a non-parallelizable algorithm. Therefore, it is not possible to train the individual ferns of the RobustBRRFerns ensemble at once to reduce the training run time. Moreover, the described feature descriptors are not rotation- or scale-invariant and therefore the trained model is rotation- and scale-variant which is also an implicit limitation of the approaches by Pauly et al. [64], Cuingnet et al. [18] and Donner et al. [21]. Especially the cuboidalBRIEF and the GaussLBP features suffer from the fact that the response is implicitly effected by the corners of the cuboidal regions. Due to the limitation of the integral volumes to cuboidal regions an efficient implementation of cuboidalBRIEF and GaussLBP using ellipsoids is not possible with the methods described.

7.2 Conclusion

Inspired by previous work of Pauly et al. [64] a novel robust boosted regression approach for automatic landmark localization on medical images is introduced: RobustBRRFerns. This approach is more accuracy than the model proposed by Pauly et al. [64] while keeping advantages of the random fern approach in terms of memory efficiency. On the synthetic test cases and RobustBRRFerns outperformes the random fern model. It has been shown that in general the RobustBRRFerns produce a model loss which is only half the loss of the random fern.

After a discussion of state of the art gray value difference features used in medical images has been given two novel feature descriptors tailored to medical volumetric images have been designed and evaluated.

A discussion of state of the art gray value difference feature used in the medical domain is given. Furthermore, to novel feature descriptors are introduced: cuboidalBRIEF and GaussLBP. The noise-sensibility and the suitability for regression approaches are evaluated to facilitate the effectiveness of the introduced feature descriptors. The novel feature descriptor cuboidalBRIEF turned out to outperform the other feature descriptors in the global prediction phase.

The generic system for multi-pass landmark prediction described allows memory and time efficient estimation of landmark locations on any kind of medical image modalities. The performance of the automatic landmark prediction approach is assessed by evaluating the method on

three different datasets. In order to provide direct comparability to state of the art approaches the CT Whole Body Morphometry Project [58] dataset is used as one of the datasets.

The results are compared to three state of the art approaches for automatic landmark localization. The proposed approach achieved significant better results on the CT Whole Body Morphometry Project [58] dataset than the approach by Pauly et al. [64]. The accuracy of the method published by Cuingnet et al. [18] is comparable to the accuracy of the model developed in this thesis. However, the complexity of the model by Cuingnet et al. [18] is significantly higher and therefore requires more memory and computation time. Although the approach by Donner et al. [21] outperformed the system developed in this thesis on the CT Whole Body Morphometry Project [58] dataset the computation time for the approach by Donner et al. [21] on an average dataset is more than twice the computation time of the model presented.

Despite the only modest improvements in the local refinement phase the proposed approach provides a memory and time efficient solution that achieves satisfying results in the global prediction phase.

7.3 Future Work

As mentioned in Chapter 6 further research will be done regarding the local refinement phase. A promising idea is the extension of the global prediction phase by multiple individual predictions which provide several initial guesses for the local refinement phase. Moreover, similar to the paper of Pauly et al. [64] the feature responses for MRI scans could be extracted over the T1 weighted and the T2 weighted images at once. As T1 and T2 images taken from the same patient in the same scan are implicitly co-registered to each other the responses can be concatenated. Combining the responses will lead to a significant improvement on datasets of a similar type as dataset 2. The next logical step to extend the illustrated approach will the combination with a higher order graphical model e.g., conditional random field or Markov random field. This will allow to introduce contextual knowledge on the relationship of different anatomical landmarks. An interesting extension of the regression model and the feature extraction would be the use of fuzzy regression models and fuzzy image processing. Especially the replacement of the hard input space clustering used in the fern models by fuzzy clustering approaches will improve the model's expressiveness.

Bibliography

[1] A. Bertoni, P. Campadelli, and M. Parodi. A boosting algorithm for regression. In *Artificial Neural Networks*, volume 1327, pages 343–348. Springer, 1997.

[2] C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[3] F. L. Bookstein. *Morphometric Tools for Landmark Data: Geometry and Biology*. Cambridge University Press, 1997.

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[5] W. E. Brant and C. A. Helms. *Fundamentals of Diagnostic Radiology, 3rd Edition*. Lippincott Williams & Wilkins, 2007.

[6] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[7] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[8] A. Burner, R. Donner, M. Mayerhoefer, M. Holzer, F. Kainberger, and G. Langs. Texture bags: anomaly retrieval in medical images based on local 3d-texture similarity. In *Medical Content-Based Retrieval for Clinical Decision Support*, volume 7075, pages 116–127. Springer, 2012.

[9] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. Brief: Binary robust independent elementary features. In *Computer Vision–ECCV 2010*, volume 6314, pages 778–792. Springer, 2010.

[10] J. Cardillo and M. A. Sid-Ahmed. An image processing system for locating craniofacial landmarks. *IEEE Transactions on Medical Imaging*, 13(2):275–289, 1994.

[11] C. Chen and G. Zheng. Robust proximal femur segmentation in conventional x-ray images via random forest regression on multi-resolution gradient features. In *Image Analysis and Recognition*, volume 7950, pages 442–450. Springer, 2013.

[12] A. Collignon, D. Vandermeulen, P. Suetens, and G. Marchal. Registration of 3d multimodality medical images using surfaces and point landmarks. *Pattern Recognition Letters*, 15(5):461–467, 1994.

[13] T. F. Cootes, G. J. Edwards, and C. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, 2001.

[14] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models – their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[15] A. Criminisi, J. Shotton, and S. Bucciarelli. Decision forests with long-range spatial context for organ localization in ct volumes. In *MICCAI Workshop on Probabilistic Models for Medical Image Analysis*, 2009.

[16] A. Criminisi, J. Shotton, D. Robertson, and E. Konukoglu. Regression forests for efficient anatomy detection and localization in ct studies. In *Medical Computer Vision, Recognition Techniques and Applications in Medical Imaging*, volume 6533, pages 106–117. Springer, 2011.

[17] F. C. Crow. Summed-area tables for texture mapping. *SIGGRAPH Computer Graphics*, 18(3):207–212, 1984.

[18] R. Cuingnet, R. Prevost, D. Lesage, L. Cohen, B. Mory, and R. Ardon. Automatic detection and segmentation of kidneys in 3d ct images using random forests. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2012*, volume 7512, pages 66–74. Springer, 2012.

[19] J. G. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America A, Optics and Image Science*, 2(7):1160–1169, 1985.

[20] T. Deserno. *Medical Imaging – Principles and Practices*, chapter Medical Image Search. CRC Press, 2012.

[21] R. Donner, E. Birngruber, H. Steiner, H. Bischof, and G. Langs. Localization of 3d anatomical structures using random forests and discrete optimization. In *Medical Computer Vision. Recognition Techniques and Applications in Medical Imaging*, volume 6533, pages 86–95. Springer, 2011.

[22] R. Donner and H. Bischof. One-shot learning of anatomical structure localization models. In *IEEE International Symposium on Biomedical Imaging*, pages 222–225. IEEE, 2013.

[23] R. Donner, B. Menze, H. Bischof, and G. Langs. Fast anatomical structure localization using top-down image patch regression. In *Medical Computer Vision. Recognition Techniques and Applications in Medical Imaging*, volume 7766, pages 133–141. Springer, 2013.

[24] R. Donner, B. Micušık, G. Langs, and H. Bischof. Sparse mrf appearance models for fast anatomical structure localisation. In *In British Machine Vision Conference*, pages 1–10, 2007.

[25] R. Donner, B. Micusik, G. Langs, L. Szumilas, P. Peloschek, K. Friedrich, and H. Bischof. Object localization based on markov random fields and symmetry interest points. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2007*, volume 4792, pages 460–468. Springer, 2007.

[26] A. Doucet and A. M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12:656–704, 2009.

[27] N. Fakhrai, P. Widhalm, C. Chiari, M. Weber, G. Langs, R. Donner, H. Ringl, M. Jantsch, and P. Peloschek. Automatic assessment of the knee alignment angle on full-limb radiographs. *European Journal of Radiology*, 74(1):236–240, 2010.

[28] J. Fehr and H. Burkhardt. 3d rotation invariant local binary patterns. In *International Conference on Pattern Recognition*, pages 1–4, 2008.

[29] Y. Freund and R. E. Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[30] L. Grady. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006.

[31] G. Guglielmi, F. Palmieri, M. G. Placentino, F. DiErrico, and L. P. Stoppino. Assessment of osteoporotic vertebral fractures using specialized workflow software for 6-point morphometry. *European Journal of Radiology*, 70(1):142–148, 2009.

[32] M. A. Hall. *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato, 1999.

[33] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning, Second Edition: Data Mining, Inference, and Prediction*. Springer, 2009.

[34] L. Hedlund and J. Gallagher. Vertebral morphometry in diagnosis of spinal fractures. *Bone and Mineral*, 5(1):59–67, 1988.

[35] J. Heinly, E. Dunn, and J. Frahm. Comparative evaluation of binary features. In *Computer Vision–ECCV 2012*, pages 759–773. Springer, 2012.

[36] P. J. Huber. *Robust statistics*. John Wiley and Sons, 2004.

[37] J. E. Iglesias, E. Konukoglu, A. Montillo, Z. Tu, and A. Criminisi. Combining generative and discriminative models for semantic segmentation of ct scans via active learning. In *Information Processing in Medical Imaging*, volume 22, pages 25–36, 2011.

[38] A. Innes, V. Ciesielski, J. Mamutil, S. John, and A. Harvey. Landmark detection for cephalometric radiology images using genetic programming. In *Proceedings of the 6th Australia-Japan Joint Workshop on Intellignet and Evolutionary Systems*, volume 2, pages 164–171, 2002.

[39] S. Issa, D. Dunlop, A. Chang, J. Song, P. Prasad, A. Guermazi, C. Peterfy, S. Cahue, M. Marshall, and D. Kapoor. Full-limb and knee radiography assessments of varus-valgus alignment and their relationship to osteoarthritis disease features by magnetic resonance imaging. *Arthritis Care and Research*, 57(3):398–406, 2007.

[40] B. Jähne. *Digitale Bildverarbeitung*. Springer, 2012.

[41] Y. Ke, R. Sukthankar, and M. Hebert. Efficient visual event detection using volumetric features. In *IEEE International Conference on Computer Vision*, volume 1, pages 166–173, 2005.

[42] R. Kindermann and J. Laurie Snell. *Markov random fields and their applications*, volume 1. American Mathematical Society Providence, RI, 1980.

[43] H. Knutsson. Representing local structure using tensors. In *Scandinavian Conference on Image Analysis*, volume 6688, pages 545–556, 1989.

[44] U. Kurkure, Y. Le, N. Paragios, J. Carson, T. Ju, and I. Kakadiaris. Landmark/image-based deformable registration of gene expression data. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011*, pages 1089–1096, 2011.

[45] K. Lengyel, M. Reichert, B. R. Donald, and D. P. Greenberg. Real-time robot motion planning using rasterizing computer graphics hardware. In *SIGGRAPH Computer Graphics*, volume 24, pages 327–335, 1990.

[46] V. Lepetit and P. Fua. Keypoint recognition using randomized trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9):1465–1479, 2006.

[47] A. Liaw and M. Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.

[48] W. Lidwell, K. Holden, and J. Butler. *Universal Principles of Design*. Rockport publishers, 2010.

[49] T. Lindahl. Study of local binary patterns. Master's thesis, University of Linköping, 2007.

[50] J. Ma. Dixon techniques for water and fat imaging. *Journal of Magnetic Resonance Imaging*, 28(3):543–558, 2008.

[51] A. Macovski. Noise in mri. *Magnetic Resonance in Medicine*, 36(3):494–497, 1996.

[52] D. Major, J. Hladuvka, F. Schulze, and K. Bühler. Automated landmarking and labeling of fully and partially scanned spinal columns in ct images. *Medical image analysis*, 17(8):1151–1163, 2013.

[53] R. Mayer. Machine learning - lecture 1. `http://tuwis.tuwien.ac.at/zope/_ZopeId/13719236A3jrZDEhY2Y/tpp/lv/lva_html?num=181191&sem=2009W`, October 2010. [Online; accessed 06-January-2011].

[54] A. Montillo. Context selective decision forests and their application to lung segmentation in ct images. In *MICCAI workshop on pulmonary image analysis*, pages 201–212, 2011.

[55] H. Müller, J. Kalpathy-Cramer, J.ramer, B. Caputo, and T. Syeda-Mahmood. Overview of the first workshop on medical content-based retrieval for clinical decision support. In *Medical Content-Based Retrieval for Clinical Decision Support–MICCAI 2009*, volume 5853, pages 1–17. Springer, 2010.

[56] H. Müller, N. Michoux, D. Bandon, and A. Geissbuhler. A review of content-based image retrieval systems in medical applications - clinical benefits and future directions. *International journal of medical informatics*, 73(1):1–23, 2004.

[57] A. Y. Ng and M. I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in neural information processing systems*, 14:841, 2002.

[58] Mallinckrodt Institute of Radiology Washington University School of Medicine. Whole body morphometry project. http://nrg.wustl.edu, 2013. [accessed 12-Sep-2013].

[59] T. Ojala, M. Pietikainen, and D. Harwood. Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In *International Conference on Pattern Recognition*, volume 1, pages 582–585, 1994.

[60] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[61] M. Özuysal, M. Calonder, V. Lepetit, and P. Fua. Fast keypoint recognition using random ferns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(3):448–461, 2010.

[62] M. Özuysal, P. Fua, and V. Lepetit. Fast keypoint recognition in ten lines of code. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[63] E. Parzen. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3):1065–1076, 1962.

[64] O. Pauly, B. Glocker, A. Criminisi, D. Mateus, A. Möller, S. Nekolla, and N. Navab. Fast multiple organ detection and localization in whole-body mr dixon sequences. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2011*, volume 6893, pages 239–247. Springer, 2011.

[65] Y. Qian, X. Gao, M. Loomes, R. Comley, B. Barn, R. Hui, and Z. Tian. Content-based retrieval of 3d medical images. In *International Conference on eHealth, Telemedicine, and Social Medicine*, pages 7–12, 2011.

[66] M. Rehman, M. Iqbal, M. Sharif, and M. Raza. Content based image retrieval: Survey. *World Applied Sciences Journal*, 19(3):404–412, 2012.

[67] L. Rokach. *Data mining with decision trees: theory and applications*. World Scientific Publishing, 2008.

[68] P. J. Rousseeuw. Least median of squares regression. *Journal of the American statistical association*, 79(388):871–880, 1984.

[69] P. J. Rousseeuw and K. V. Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3):212–223, 1999.

[70] S. Schmidt, J. Kappes, M. Bergtholdt, V. Pekar, S. Dries, D. Bystrov, and C. Schnörr. Spine detection and labeling using a parts-based graphical model. In *Information Processing in Medical Imaging*, volume 4584, pages 122–133, 2007.

[71] A. Schmidt-Richberg, R. Werner, J. Ehrhardt, J. Wolf, and H. Handels. Landmark-driven parameter optimization for non-linear image registration. In *SPIE Medical Imaging*, volume 7962, pages 1–8, 2011.

[72] F. Schulze, K. Bühler, A. Neubauer, A. Kanitsar, L. Holton, and S. Wolfsberger. Intra-operative virtual endoscopy for image guided endonasal transsphenoidal pituitary surgery. *International Journal of Computer Assisted Radiology and Surgery*, 5(2):143–154, 2010.

[73] F. Suykens. *On robust Monte Carlo algorithms for multi-pass global illumination*. PhD thesis, Katholieke Universiteit Leuven, 2002.

[74] H. Tamura, S. Mori, and T. Yamawaki. Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man and Cybernetics*, 8(6):460–473, 1978.

[75] E. Tapia. A note on the computation of high-dimensional integral images. *Pattern Recognition Letters*, 32(2):197–201, 2011.

[76] S. Verboven and M. Hubert. Libra: a matlab library for robust analysis. *Chemometrics and Intelligent Laboratory Systems*, 75(2):127–136, 2005.

[77] A. Vezhnevets and O. Barinova. Avoiding boosting overfitting by removing confusing samples. In *European Conference on Machine Learning*, volume 4701, pages 430–441. Springer, 2007.

[78] Vernor Vinge. Riding the wave – how europe can gain from the rising tide of scientific data. Technical report, European Commission, 2010.

[79] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 511 – 518, 2001.

[80] L. Wang and D. He. Texture classification using texture spectrum. *Pattern Recognition*, 23(8):905–910, 1990.

[81] H. Wevers, D. Siu, and T. Cooke. A quantitative method of assessing malalignment and joint space loss of the human knee. *Journal of Biomedical Engineering*, 4(4):319–324, 1982.

[82] A. Yao, J. Gall, and L. Van Gool. A hough transform-based voting framework for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2061–2068, 2010.

[83] Y. Zheng, B. Georgescu, and D. Comaniciu. Marginal space learning for efficient detection of 2d/3d anatomical structures in medical images. In *Information Processing in Medical Imaging*, volume 5636, pages 411–422, 2009.

[84] Y. Zheng, B. Georgescu, H. Ling, M. Scheuering, and D. Comaniciu. Method and system for detecting 3d anatomical structures using constrained marginal space learning, Feb 2012.

[85] Y. Zheng, B. Georgescu, H. Ling, S. K. Kevin Zhou, M. Scheuering, and D. Comaniciu. Constrained marginal space learning for efficient 3d anatomical structure detection in medical images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 194–201, 2009.

APPENDIX A

Feature Descriptors

A.1 Pseudo code of GaussLBP

input : An image I, the position vector $\mathbf{p} \in \mathbb{R}^3$, the offset matrix of the k neighbors $\mathbf{Q} \in \mathbb{R}^{3\times K}$ of $\mathbf{p}$ and the dimension Matrix of the neighbors $\mathbf{D}^{3\times K}$.
output: An binary vector $\mathbf{b} \in \mathbb{B}^K$.

// Initialize binary vector
$\mathbf{b} \leftarrow \mathbf{0}$;
// where $\mathbf{0}$ is the zero vector of dimensionality K
// create summed-area table
$T \leftarrow$ using Equation 3.10
for $i = 1, \ldots, K$ **do**
  // compute intensity mean
  $\mathbf{q} \in \mathbb{R}^3 \leftarrow \mathbf{Q}^i$;
  $\mathbf{d} \in \mathbb{R}^3 \leftarrow \mathbf{D}^i$;
  $IM \leftarrow \frac{T(q_1,q_2,q_3,d_1,d_2,d_3)}{d_1 d_2 d_3}$;
  // update binary vector
  if $IM < I(\mathbf{p})$ **then**
    $b_i \leftarrow 1$;
  end
end

Algorithm A.1: GaussLBP

A.2 Pseudo code of cuboidalBRIEF

input : An image I, the offset matrix of the k tuples $\mathbf{Q} \in \mathbb{R}^{6\times K}$ of $\mathbf{p}$ and the dimension Matrix of the tuples $\mathbf{D}^{6\times K}$.
output: A decimal value $x \in \mathbb{R}$.

```
// Initialize binary vector
```
$\mathbf{b} \leftarrow \mathbf{0}$;
```
// where 0 is the zero vector of dimensionality K
// create summed-area table
```
$T \leftarrow$ using Equation 3.10

for $i = 1, \ldots, K$ **do**
```
    // compute intensity means
```
    $\mathbf{q} \in \mathbb{R}^6 \leftarrow \mathbf{Q}^i$;
    $\mathbf{d} \in \mathbb{R}^6 \leftarrow \mathbf{D}^i$;
    $IM' \leftarrow \frac{T(q_1,q_2,q_3,d_1,d_2,d_3)}{d_1 d_2 d_3}$;
    $IM'' \leftarrow \frac{T(q_4,q_5,q_6,d_4,d_5,d_6)}{d_4 d_5 d_6}$;
```
    // update binary vector
```
    if $IM' < IM''$ **then**
        $b_i \leftarrow 1$;
    end
end
$x \leftarrow$ `Decimal(`$\mathbf{b}$`)`

Algorithm A.2: cuboidalBRIEF

APPENDIX B

Results

Table B.1: Local refinement results on all datasets in mm.

Refinement Approach	Feature	Dataset 1		Dataset 2		Dataset 3	
		Mean	Median	Mean	Median	Mean	Median
-	cuboidalBRIEF	9.34 ± 5.96	7.86 ± 4.28	**49.60** ± 64.42	**22.18** ± 46.88	15.72 ± 8.20	14.46 ± 6.44
classification	cuboidalBRIEF	**8.37** ± 5.09	**7.09** ± 4.03	-	-	13.05 ± 11.05	12.00 ± 9.58
regression	cuboidalBRIEF	8.61 ± 6.15	7.52 ± 4.30	-	-	15.95 ± 8.62	14.24 ± 6.78
Donner et al. [21]	Donner et al. [21]	-	-	-	-	**5.25** ± 15.08	**2.71** $\pm -$
Pauly et al. [64]	Pauly et al. [64]	-	-	-	-	54.80 ± 43.37	$86.98 \pm -$

Table B.2: Global prediction results on all datasets in mm.

Approach	Feature	Dataset 1		Dataset 2		Dataset 3	
		Mean	Median	Mean	Median	Mean	Median
RobustBRRFerns	Donner et al. [21]	11.08 ± 4.26	10.84 ± 3.72	64.08 ± 61.06	42.18 ± 45.52	-	-
	GaussLBP	12.88 ± 5.54	11.84 ± 4.30	79.16 ± 66.34	55.80 ± 53.38	-	-
	BRIEF	13.80 ± 6.12	12.66 ± 4.74	57.82 ± 59.08	38.58 ± 43.22	-	-
	cuboidalBRIEF	**9.34** ± 5.96	**7.86** ± 4.28	**49.60** ± 64.42	**22.18** ± 46.88	15.72 ± 8.20	14.46 ± 6.44
Donner et al. [21]	Donner et al. [21]	-	-	-	-	**5.25** ± 15.08	**2.71** $\pm -$
Pauly et al. [64]	Pauly et al. [64]	-	-	-	-	54.80 ± 43.37	$86.98 \pm -$

APPENDIX C

Datasets

C.1 MIPs of Dataset 1: CT Heads

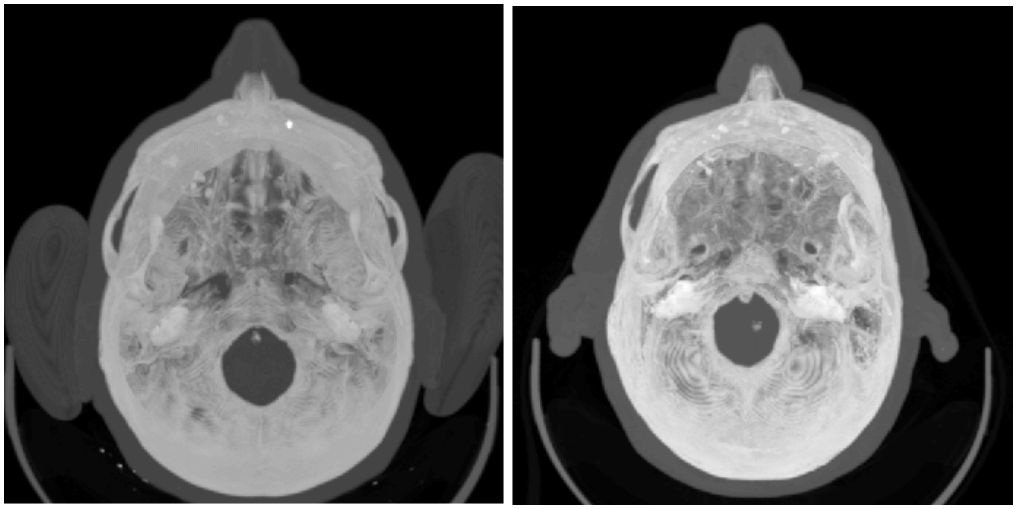


Figure C.1: Maximum intensity projection of the first and second datum of dataset 1.

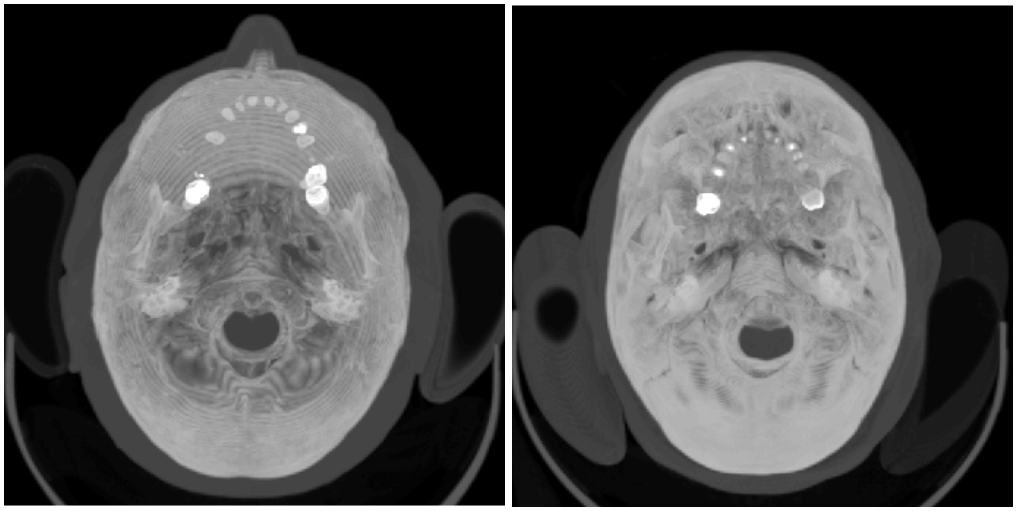


Figure C.2: Maximum intensity projection of the third and fourth datum of dataset 1.

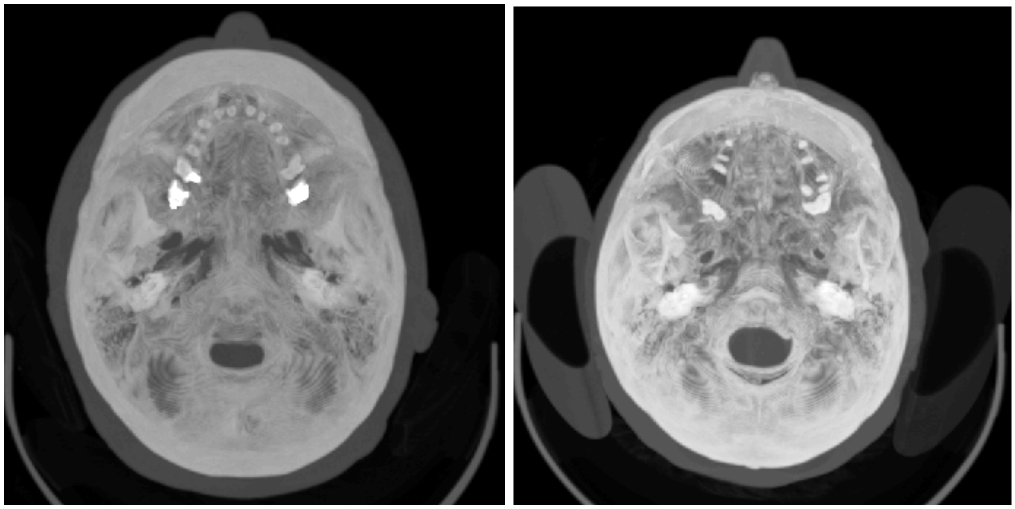


Figure C.3: Maximum intensity projection of the fifth an sixth datum of dataset 1.

C.2 MIPs of Dataset 2: MRI T1 Heads

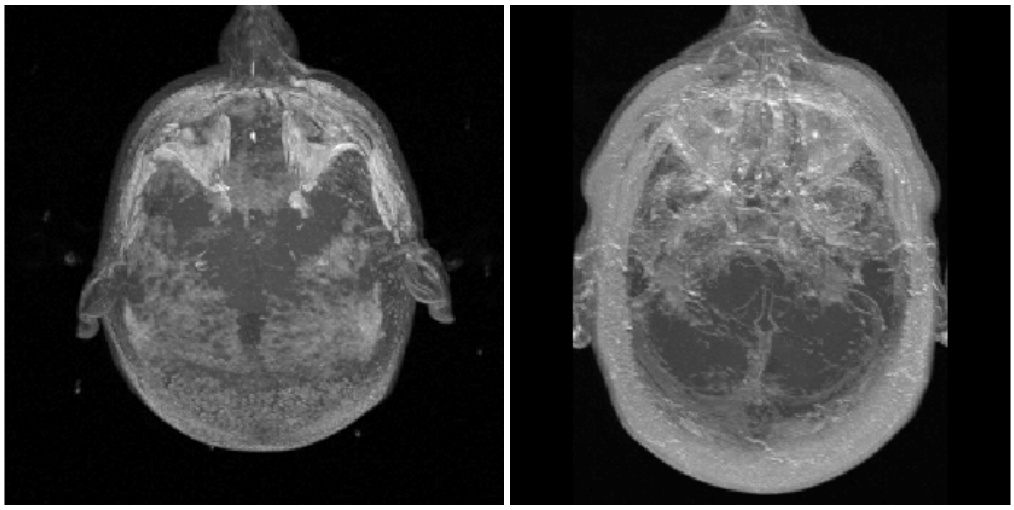


Figure C.4: Maximum intensity projection of the first and second datum of dataset 2.

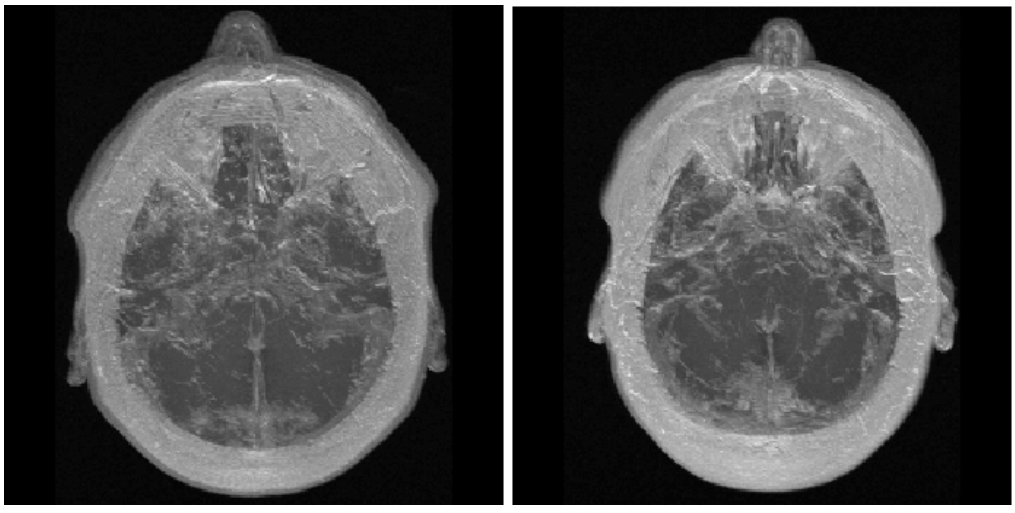


Figure C.5: Maximum intensity projection of the third and fourth datum of dataset 2.

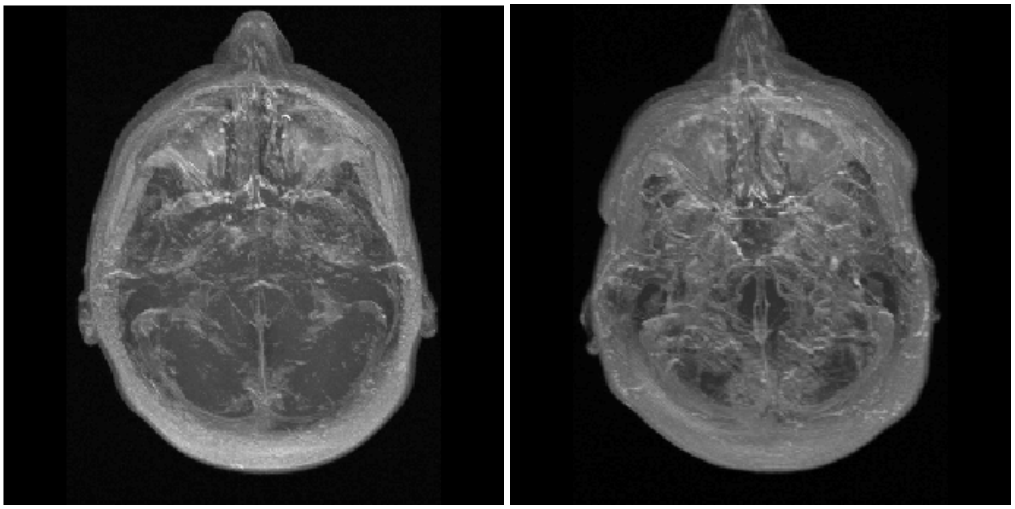


Figure C.6: Maximum intensity projection of the fifth and sixth datum of dataset 2.

C.3 MIPs of Dataset 3: CT Whole Body

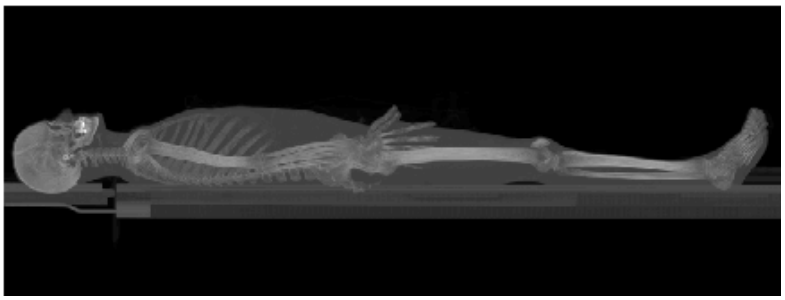


Figure C.7: Maximum intensity projection of the 1. datum of dataset 3.

Figure C.8: Maximum intensity projection of the 2. datum of dataset 3.

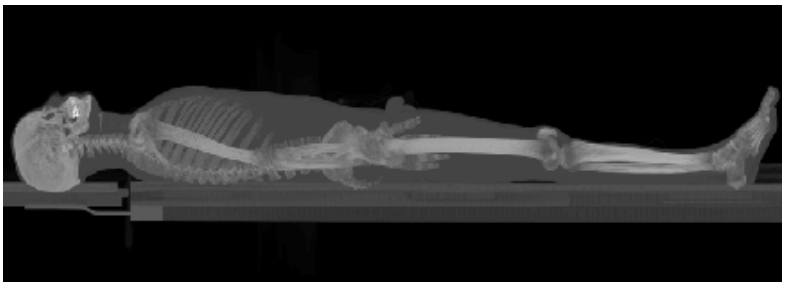


Figure C.9: Maximum intensity projection of the 3. datum of dataset 3.

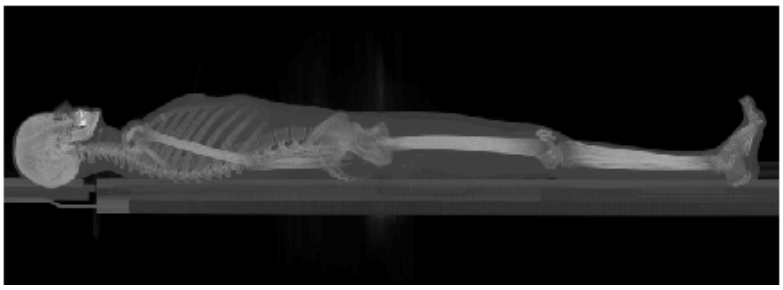


Figure C.10: Maximum intensity projection of the 4. datum of dataset 3.

Figure C.11: Maximum intensity projection of the 5. datum of dataset 3.

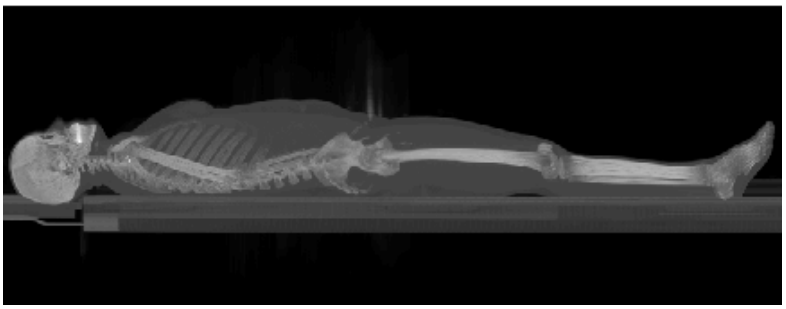


Figure C.12: Maximum intensity projection of the 6. datum of dataset 3.

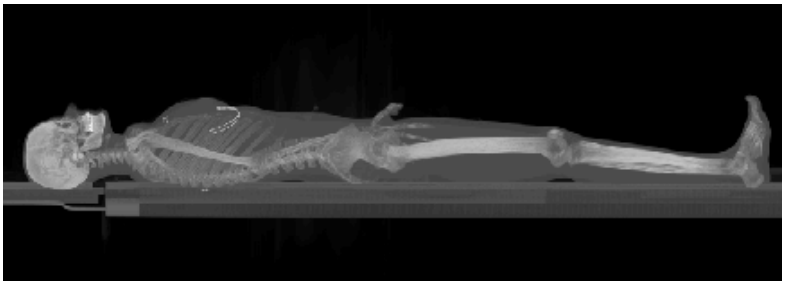


Figure C.13: Maximum intensity projection of the 7. datum of dataset 3.

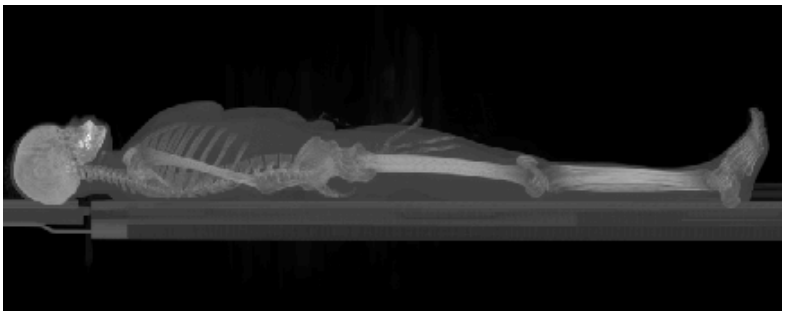


Figure C.14: Maximum intensity projection of the 8. datum of dataset 3.

Figure C.15: Maximum intensity projection of the 9. datum of dataset 3.

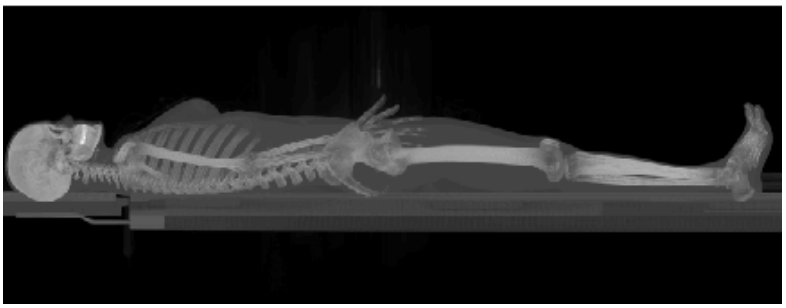


Figure C.16: Maximum intensity projection of the 10. datum of dataset 3.

Figure C.17: Maximum intensity projection of the 11. datum of dataset 3.

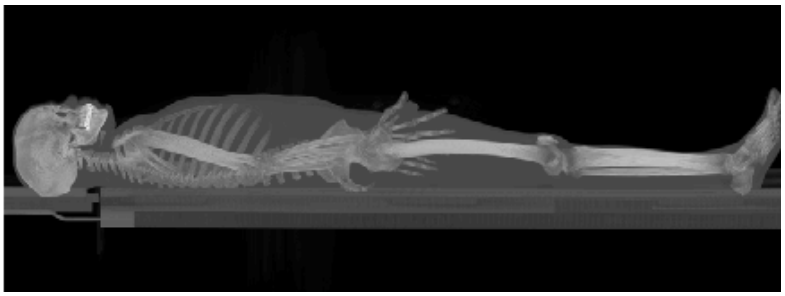


Figure C.18: Maximum intensity projection of the 12. datum of dataset 3.

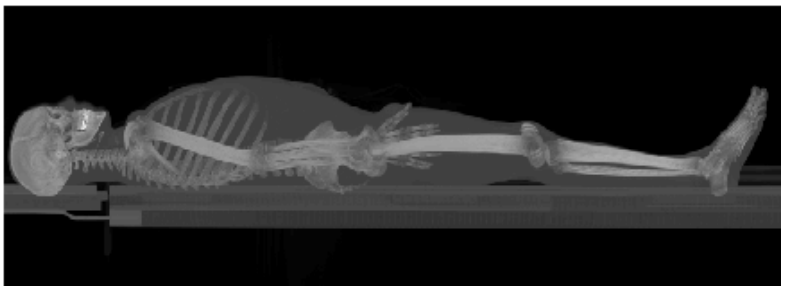


Figure C.19: Maximum intensity projection of the 13. datum of dataset 3.

Figure C.20: Maximum intensity projection of the 14. datum of dataset 3.

Figure C.21: Maximum intensity projection of the 15. datum of dataset 3.

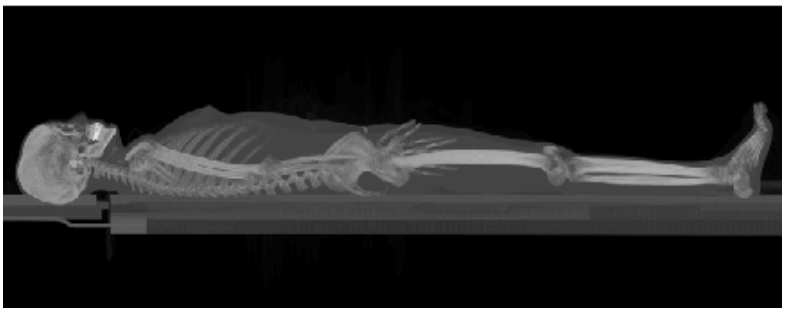


Figure C.22: Maximum intensity projection of the 16. datum of dataset 3.

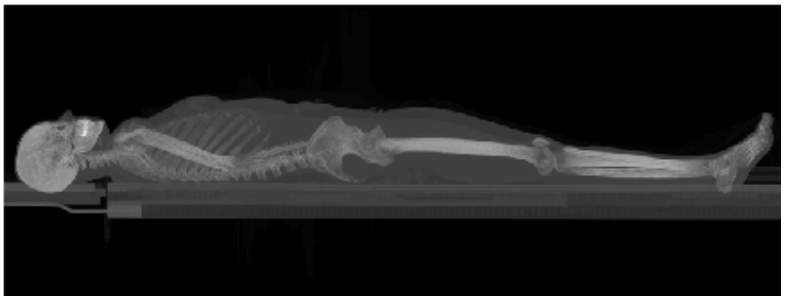


Figure C.23: Maximum intensity projection of the 17. datum of dataset 3.

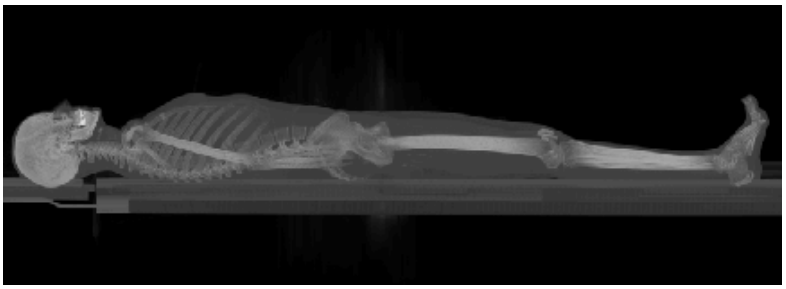


Figure C.24: Maximum intensity projection of the 18. datum of dataset 3.

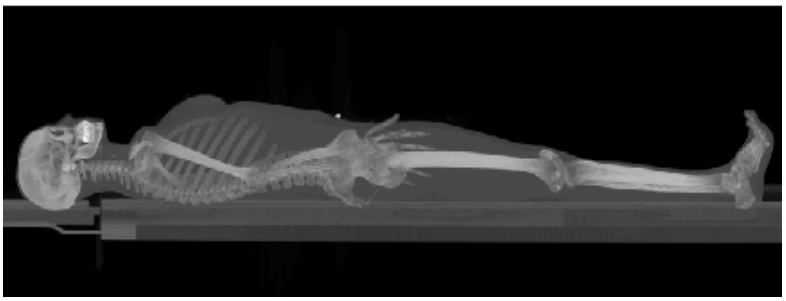


Figure C.25: Maximum intensity projection of the 19. datum of dataset 3.

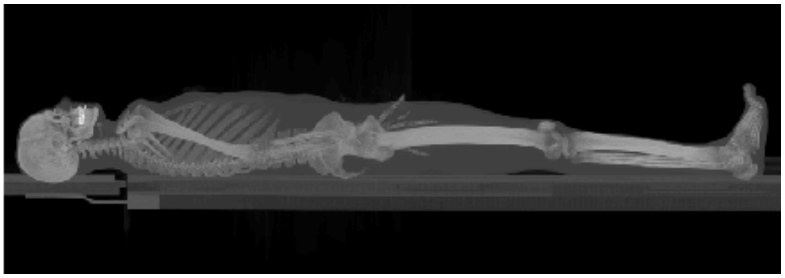


Figure C.26: Maximum intensity projection of the 20. datum of dataset 3.