Thomas Ebner, BSc

Localization of Hand Bones from 3D Magnetic Resonance Images for Bone Age Estimation

Master's Thesis

to achieve the university degree of
Diplom-Ingenieur

Master's degree programme: Telematics

submitted to

Graz University of Technology

Supervisors

Univ.-Prof. DI. Dr. Horst Bischof
Institute for Computer Graphics and Vision, Graz University of Technology

DI. Dr. Martin Urschler
Ludwig Boltzmann Institute for Clinical Forensic Imaging, Graz

Graz, Austria, January 2015

Deutsche Fassung:
Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008
Genehmigung des Senates am 1.12.2008

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am …… ……..
(Unterschrift)

Englische Fassung:

STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

…… ……..
date (signature)

Abstract

Localization of anatomical structures is a very important step in many medical image analysis algorithms. However, our interest in localization algorithms is related to an ongoing research study, which investigates the possibility of replacing X-ray imaging with radiation-free MRI based imaging for the purpose of Bone Age Estimation (BAE). BAE is an important topic in both forensic and clinical medicine, at the moment, mainly performed by trained radiologists manually inspecting the hand bones within X-ray images. They suffer from a high inter-observer variability and are very time consuming and tedious. Therefore, development of a fully automated BAE method based on MRI images of the hand, allowing an objective estimation of the bone age without the need for harmful ionizing radiation, is considered of high practical impact. To extract features for BAE, localization of the joints between hand bones is a crucial first step in a fully automated BAE pipeline.

We propose a landmark localization algorithm using multiple Random Regression Forest localization stages at different scales together with a weighting scheme, that lets local structures have a higher contribution to the position estimation, thus following the idea that coarse localization of landmarks is supported by global information from all over the image, while closer structures provide more information to increase the precision of landmark localization. We are able to clearly outperform related approaches on our dataset of 60 T1-weighted MR images, achieving a mean landmark localization error of 1.4±1.5mm, while having only 0.25% outliers with an error greater than 10mm.

Keywords: Anatomical Landmark Localization, Hand Bones, Magnetic Resonance Imaging, Bone Age Estimation, Fully Automated, Random Regression Forests

Kurzfassung

Die Lokalisierung anatomischer Strukturen ist ein wichtiger Bestandteil vieler medizinischer Bildanalysealgorithmen. Anlass für unser Interesse an Lokalisierungsalgorithmen gibt eine aktuelle Forschungsstudie des LBI zur Knochenaltersschätzung, welche versucht Röntgen basierte Bildgebungsverfahren durch strahlungsfreie Magnetresonanztomographie (MRI) zu ersetzten. Anwendungsbereiche findet die Knochenaltersschätzung beispielsweise in der Medizin und in der Forensik. Derzeitige Verfahren zur Knochenaltersschätzung sind zeitaufwendig, da sie auf einer manuellen Untersuchung von Röntgenbildern der linken Hand durch geschulte Radiologen basieren. Zudem unterliegt die Untersuchung durch unterschiedliche Interpretationsmöglichkeiten der Röntgenbilder einer breiten Streuung. Ein objektives vollautomatisiertes Verfahren zur Knochenaltersschätzung gestützt auf MRT Bildgebung wäre somit von großer praktischer Bedeutung. Um jedoch Knochenaltersschätzung zu automatisieren, ist die Lokalisierung der Knochen innerhalb der MRT Bilder ein wichtiger Schritt.

Wir stellen einen Lokalisierungsalgorithmus vor, welcher auf mehreren Random Regression Forest Lokalisierungsschritten sowie einem Gewichtungsschema beruht. Mit einem Lokalisierungsfehler von nur 1.4±1.5mm auf unserer Datenbank, bestehend aus 60 T1-gewichteten MRT Bildern, erreichen wir ein deutlich besseres Ergebnis als vergleichbare Algorithmen.

Schlüsselwörter: Lokalisierung, Handknochen, Magnetresonanztomographie, Knochenaltersschätzung, Vollautomatisiert, Random Regression Forests

Acknowledgments

At this point, I would like to express my gratitude to Prof. Horst Bischof for supervising my thesis. I would like to give special appreciation and thanks to my advisor Dr. Martin Urschler; you have been a tremendous mentor for me. I would like to thank you for encouraging my research and supporting me in all matters.

Furthermore, I would like to thank my loving parents, who gave me the opportunity to study and have supported me throughout my life.

Last but not least, I would like to thank my beloved girlfriend Iris, who gave me support and encouragement when I needed it most.

Contents

List of Figures

Chapter 1

Introduction

Contents

1.1 Motivation

Localization of anatomical structures is a very important step in many medical image analysis algorithms [46] to provide a coarse initialization for subsequent image analysis steps, e.g. segmentation algorithms such as Active Shape Models [10] and Active Appearance Models [9], registration algorithms [29], or to provide an initialization for feature extraction [46] and classification or regression, especially in the field of computer-aided diagnosis [17]. Localization is also important in applications where the position of anatomical landmarks is directly assessed for morphometric measurements, for example to measure angles in the knee, thus identifying varus-valgus misalignment [33].

Localization of organs in Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) scans is needed for intelligent navigation and visualization tools [14] or to retrieve selected parts of patients scans from radiological database systems [38], thus reducing the amount of data transferred from the database.

Localization can be achieved by placing landmarks manually [7, 43], which is very time consuming, is prone to high inter-observer variability and is difficult, since 3D images are often visualized by presenting 2D slices, where the visual appearance of anatomical structures depends strongly on the slice orientation [42]. To overcome these limitations,

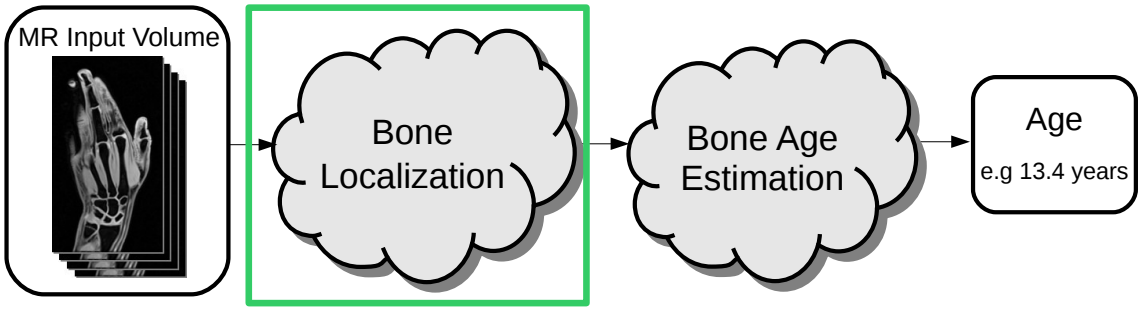


Figure 1.1: Fully automated Bone Age Estimation pipeline, which aims to estimate the age of a subject from an MRI input volume. The localization of the hand bones is a crucial first step in that pipeline.

localization algorithms can be used, with fully automatic image processing pipelines often even requiring automated localization algorithms.

There has been many work on anatomical landmark localization via low-level interest-point detectors such as Harris Corners [19] or by fitting 3D intensity models [56]. However, due to the progress in the field of machine learning, powerful algorithms such as Random Forests (RFs) [6], which learn how to localize anatomical landmarks from training data, have emerged and have become a widely used technique for localization of anatomical structures.

Our interest in localization algorithms is related to an ongoing research study at the LBI-CFI* in Graz, which investigates the possibility of replacing X-ray imaging with radiation-free MRI for the purpose of age estimation of young unaccompanied asylum seekers without identification document, who are currently routinely scanned for the Austrian government. To automate age estimation, localization of the anatomical structures, where age relevant features can be found, is a crucial first step in an age estimation pipeline as shown in Fig. 1.1.

1.1.1 Bone Age Estimation

During maturation of children and young adolescents, changes in anatomy and physiology define biological age. The biological age of subjects can be estimated by a radiological examination of the skeletal development, which is referred to as Bone Age Estimation (BAE).

*The Ludwig Boltzmann Institute for Clinical Forensic Imaging (LBI-CFI) dedicates its research efforts to Forensic Radiology. The scientific goal of the interdisciplinary research team is developing basic parameters for the clinical-forensic use of MR and CT imaging.

Estimating the age of children and young adolescents has many applications in clinical and legal medicine. In clinical medicine the chronological age of subjects is usually known and compared to the biological age. Deviations of the biological age from the chronological age are used to diagnose endocrinological diseases [37] or to make growth predictions [50]. Another clinical application of BAE is to plan the time-point for orthopedic surgical interventions in pediatric cases of leg length discrepancy [36] or scoliosis [55].

However, our main focus lies on applications for legal medicine, where age estimation is an important procedure when proper identification documents are missing. The chronological age is unknown in such cases and is approximated using BAE techniques. Applications are legal majority age determination of young unaccompanied asylum seekers [44], to determine whether the juvenile or adult law should be applied in criminal prosecutions, to prevent age manipulations in age-related tournaments in sports [22] or to perform victim identification after disasters [3, 16].

A widely established approach for BAE is to follow the ossification process of the bones within the hand or the clavicle. Up to an age of around 19 years BAE can be done by examining bones within the hand. After that, the ossification process is finished within the hand and no more changes are visible. Therefore, other bones such as the clavicle are used for estimating the age for ages up to around 23 years.

At the moment, BAE is mainly performed by trained radiologists manually inspecting the hand bones within X-ray images according to methods proposed by Greulich-Pyle (GP) [27] or Tanner-Whitehouse (TW) [49]. Age estimation as proposed by GP is done by comparing a hand X-ray to an atlas consisting of reference images from subjects of different ages. By finding the most similar reference image, the age is estimated by taking the age of the reference image. According to TW, the skeletal development stage of specific parts of the hand is determined independently. Age is estimated by fusing the independent ratings of individual bones according to the estimated development stages.

Such a manual inspection is very time consuming, tedious and a large intra- and inter-observer variability due to different human interpretations can be observed. Recently, fully automatic approaches, e.g. the BoneXpert method [51], were proposed to overcome limitations of manual inspections. However, the exposure of subjects to harmful ionizing radiation is still a limitation, especially when applied to healthy subjects. Since this exposure is even prohibited in many countries for non diagnostic reasons, i.e. in legal medicine, BAE based on MRI has recently gained in importance.

Clinically established methods suffer from high inter-observer variability due to man-

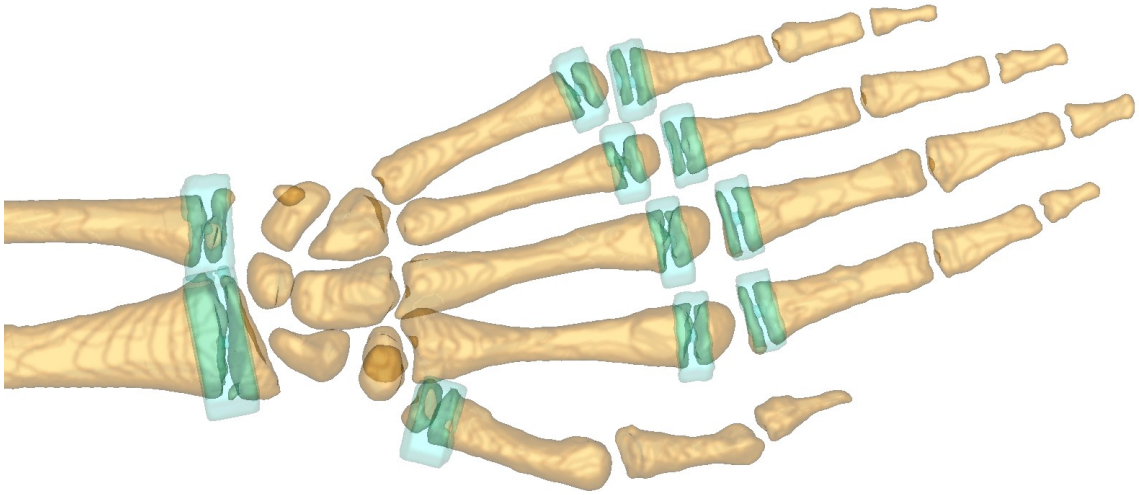


Figure 1.2: Bone skeleton of the hand with regions where epiphyseal plate fusion can be seen (green).

ual inspection or involve harmful ionizing X-ray radiation. Therefore, development of a fully automated BAE method based on MRI images of the hand, allowing an objective estimation of the bone age without the need for harmful ionizing radiation, is considered of high practical impact.

Features relevant for describing the age of a person can be found in the regions around the epiphyseal plates within the hand as can be seen in Fig. 1.2. To extract features from these regions for BAE, localization is a crucial first step in a fully automated BAE pipeline. Localization of these regions could be achieved by a segmentation of the epiphyseal plates followed by an extraction of features from the appearance of the segmented areas. However, such a segmentation task is very challenging and any mis-segmentations would directly influence the age estimation result. Extraction of age relevant features is possible without requiring a segmentation, but instead using the predicted location of the epiphyseal plates. Therefore, we use localization and avoid a much more challenging segmentation task. Since the epiphyseal plates are located inside the bones' metaphysis, their position is constrained by the location of the bones. Therefore, we propose a localization of anatomical landmarks, located in the joints between the bones as shown in Fig. 1.4, thus defining the position of the bones and the epiphyseal gaps. Localization is a crucial and mandatory first step of an age estimation pipeline as can be seen in Fig. 1.1.

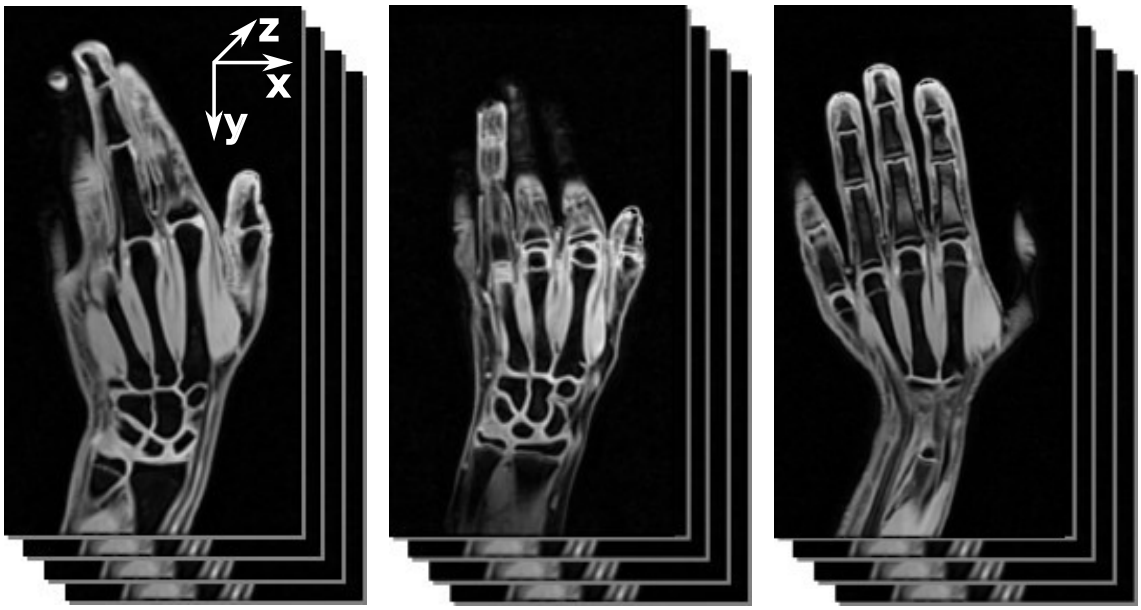


Figure 1.3: Example 2D slices of our database of T1-weighted MR images showing variations related to different poses and different ages.

1.1.2 Goal of this thesis

The goals of this thesis are:

(1), to give an overview over state of the art algorithms for landmark localization, while focusing on those well suited for a BAE pipeline based on 3D MRI images of the hand.

(2) The development of a fully automated landmark localization algorithm, meeting the requirements of robustness and good localization accuracy and precision, since it is the first part of the BAE pipeline.

(3) The developed algorithm requires thorough evaluation and comparison to related state of the art algorithms.

1.2 Data

In the course of an ongoing research study at the LBI-CFI in Graz with the goal of developing a BAE method based on MRI, scans are taken from volunteering children and young adolescents. We were provided with a dataset consisting of 60 MR images from Caucasian male subjects in an age range of 13 up to 23 years. The 3D images were obtained

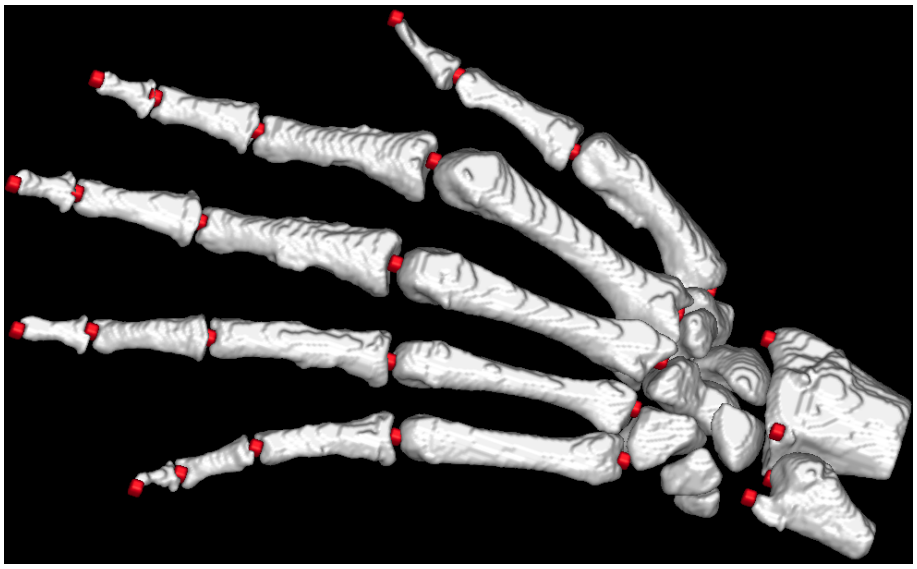


Figure 1.4: Hand bones with our annotated anatomical landmarks (red points).

using a T1-weighted gradient echo sequence with a voxel size of $0.45x0.45x0.9mm^3$ and an average volume dimension of $294x512x72$ voxels. The hands are located roughly in the center of the images with a rotation about the z-axis varying in the range of $\pm 15°$. During the scanning procedure subjects are told to put their hand into the head and neck coil of the scanner and a sand-sack is put on top of their hand to keep its position fixed and reduce finger pose variations. However, since we are dealing with children and juveniles, they are trying to find a comfortable positions for their hands and therefore the scans still include finger pose variations. Fig. 1.3 shows example images taken from the dataset illustrating the different finger poses and variations due to the age range that we are dealing with. Fully automated landmark localization is challenging due to

- the presence of a lot of repeating structures in the hand, which may lead to ambiguous localization results,
- finger pose variations,
- anatomical variations related to the age range of the investigated subjects.

Annotation To enable the use of machine learning algorithms requiring ground-truth landmarks during training and to allow an evaluation, a manual annotation of the location of anatomical landmarks in the hand as shown in Fig. 1.4 is required.

1.3 Contribution and Outline

In Chapter 2 we give an overview over related work on localization of anatomical structures ranging from low-level approaches and application-specific solutions for hand-bone localization up to powerful machine learning algorithms.

In the following, we propose a novel landmark localization algorithm following the idea that the location of anatomical landmarks is constrained by all of their surrounding structures. Coarse localization of landmarks is supported by global information from all over the image, while closer structures provide more information to increase the precision of landmark localization.

We realize this idea using a multi-scale setup of multiple Random Regression Forest (RRF) [12] localization stages together with a weighting scheme, that lets local structures have a higher contribution to the position estimation. The RRF framework together with the proposed weighting scheme is introduced in Chapter 3. In Chapter 4 we connect several RRF stages in a multi-scale setup to finally derive our proposed localization algorithm.

To highlight the benefit of our method, we provide a comprehensive evaluation of our algorithm and a comparison to related approaches in Chapter 5. Chapter 6 summaries and concludes this thesis.

Chapter 2

Localization of Anatomical Structures

Contents

This chapter is dedicated to give a brief overview over existing methods on the localization of anatomical structures. In Section 2.1 we provide a categorization of localization according to the semantic representation of the position and distinguish between approaches making a prediction based on local and global context. In the remaining sections, different approaches from the literature, ranging from low-level and atlas-based to powerful machine learning algorithms, are discussed and concluded in Section 2.7.

2.1 Categorization of Localization Algorithms

2.1.1 Semantic Representation

Anatomical structure localization approaches can be categorized according to their semantic representation. They aim to estimate either

- Bounding Boxes (BBs) around structures,
- the position of anatomical landmarks,
- or yield voxel-wise labels for an anatomical structure, i.e. a segmentation.

Performing segmentation is a very hard task, especially when using supervised machine learning algorithms requiring training images, because it requires a precise ground-truth segmentation of all 3D training images, which is very time-consuming and cumbersome to obtain.

Detecting BBs has the advantage to get both, the location and extent of a structure. A disadvantage of BBs is that their faces are defined by the largest extent of an anatomical structure, which is not necessarily of interest. Especially axis-aligned BB designs lack in precision in the presence of rotations. Therefore, anatomical landmarks are used when a more precise localization is needed.

In case of our BAE pipeline, we are interested in the position of the epiphyseal plates, which are located inside the metaphysis part of the bones. Their position can be derived more accurately from anatomical landmarks in the joints between bones than from BBs around the bones.

2.1.2 On the context

From a different perspective, one can make a distinction between localization algorithms estimating based on local- or global context. Both approaches are dealing differently with the presence of repeating structures, e.g. a lot of landmarks within the hand share a similar local appearance.

Approaches based on local context model only the local appearance around the landmarks. Due to repeating structures, this requires to use a high-level model to distinguish between the different landmarks. Such a high-level model usually selects landmark locations based on some kind of geometric relationship between the different landmarks.

Another approach to handle repeating structures is to locate the landmarks based on global context. This means that predictions are made by using appearance information from all over the image. Approaches based on global context typically do not require a high-level model such as a geometric model.

2.2 Low-Level Approaches

The localization of anatomical structures as an initialization for further processing is often achieved by very application-specific tailored low-level approaches. A domain specific algorithm for locating the epiphyseal regions from X-ray images was presented in [40]. They locate finger tips by thresholding and parsing the input image in horizontal lines as shown in Fig. 2.1.

Figure 2.1: Locating fingers from X-ray images via horizontal parsing. (Source: [40])

Once the finger tips are found by looking for the top most response, they follow the centerline of each finger and locate the epiphyseal gaps by interpreting gradient information. However, this algorithm locates only the phalanges, an extension to 3D MRI images is not straightforward and the algorithm is not robust to typical variations in clinical images.

In [56], they locate anatomical landmarks by fitting 3D parametric intensity models of the landmark's local appearance to the image intensities. To handle repeating structures within the hand, a subsequent high-level prediction step including geometric constraints would be required for practical use in landmark localization.

An approach using global optimization to select landmark locations from candidates obtained via interest point detectors, was presented in [19]. They propose to use a new interest point detector based on symmetry properties of the local appearance in combination with Harris corners. After interest points are found, they perform global optimization using a Markov Random Field (MRF), thus modeling geometric relationships between landmarks. While this algorithm can handle repeating structures, the problem of not being very robust to local appearance changes in the presence of anatomical variations remains.

2.3 Approaches based on Atlas Registration

In the past, atlas-based approaches have been widely used in the field of anatomical structure localization [24, 45, 57].

An image, with known ground-truth labels of the anatomical structures of interest, is taken as a reference image or also called atlas. Localization is performed by aligning a new unseen image to the reference image using registration techniques. Labels from the atlas are propagated to the input image, thus obtaining the location of anatomical structures. Such approaches are suitable for locating landmarks, BBs or even rough segmentations. However, due to the flexibility of a lot of parts of the human body, simple rigid registration will introduce errors. To overcome this limitation, non-rigid registration algorithms may be used. They are able to handle slight variations in shape, but due to higher model complexities, computational requirements are highly increased.

Multi-atlas approaches have been developed to handle the anatomical variations between subjects or the use of different modalities in medical images. An input image is registered to multiple reference images. The labels from the best reference image, according to the overall registration cost, are selected. Another possibility is to fuse the labels of the different reference images and propagate them to the input image [32]. Label propagation is performed via a spatially varying fusion process, depending on an assessment of the local registration cost. In general it can be said that registration based approaches often need a careful initialization to guarantee convergence.

2.4 Statistical Shape Models

An approach for incorporating shape prior information when localizing anatomical landmarks are Active Shape Models (ASMs) [9, 10]. They consist of a Statistical Shape Model (SSM) to model the geometric relationships between the landmarks. Based on a set of training images, the variations in shape of the landmark positions are learned. When locating landmarks in an unseen image, landmark positions obtained based on image intensities are regularized using the learned Statistical Shape Model (SSM) to allow only shapes similar to the shapes in the training images. ASMs are typically implemented using an iterative scheme with the following steps:

1. Start with a coarse initialization $\boldsymbol{l}_{init}$ of the landmark positions $\boldsymbol{l} = \boldsymbol{l}_{init}$.

2. Obtain new landmark locations $\boldsymbol{l}$ based on image intensities in a small region around

current landmark locations $\boldsymbol{l}$.

3. Regularize the obtained landmark locations $\boldsymbol{l}$ using an SSM.

4. Repeat steps 2-3 until convergence.

Limitations of ASMs are that they require a careful initialization to guarantee convergence and typically require a large number of training data to prevent over-constrained models [30].

As an example approach using SSMs, we describe an algorithm for anatomical landmark localization from hand CT images based on SSM in the following. Another approach based on SSMs is presented in Section 2.6.1.2.

2.4.1 Top-Down Image Patch Regression

Top Down Image Patch Regression (TDPR) as proposed in [20], is a very fast method for localizing anatomical landmarks in 2D and 3D data. TDPR maintains an appearance codebook, containing an efficient representation of patches from all over the image at different scales. Each patch is associated with a displacement vector, modeling the distance to the landmark location.

Landmark locations are predicted in a multi-scale search, starting from the coarsest level, using the appearance codebook. An SSM is used at each scale to model the spatial distribution of the landmarks, thus constraining the landmark locations and allowing to handle the presence of repeating structures within the hand.

In the following, the training and testing stages of TDPR are described.

2.4.1.1 Training

Training of TDPR consists of learning a local appearance model and an SSM.

Appearance Model During training of TDPR, multi-scale regression codebooks for each of the L landmarks $x \in 1, ..., L$ and S different scales $s \in 1, ..., S$ are built, resulting in S x L codebooks, as can be seen in Fig. 2.2. Each codebook consists of patches $\mathbf{P}^p$ with index p around the landmarks with varying offsets and scaling, where for each patch $\mathbf{P}^p$, relative landmark position offsets $\mathbf{L}^p$ of all visible landmarks on the patch are stored.

Since the memory required for storing the large amount of patches is very high, and a comparison of patches at testing involves a lot of computations, an efficient representation is necessary. Each patch $\mathbf{P}^p$ is compressed using Principal Component Analysis

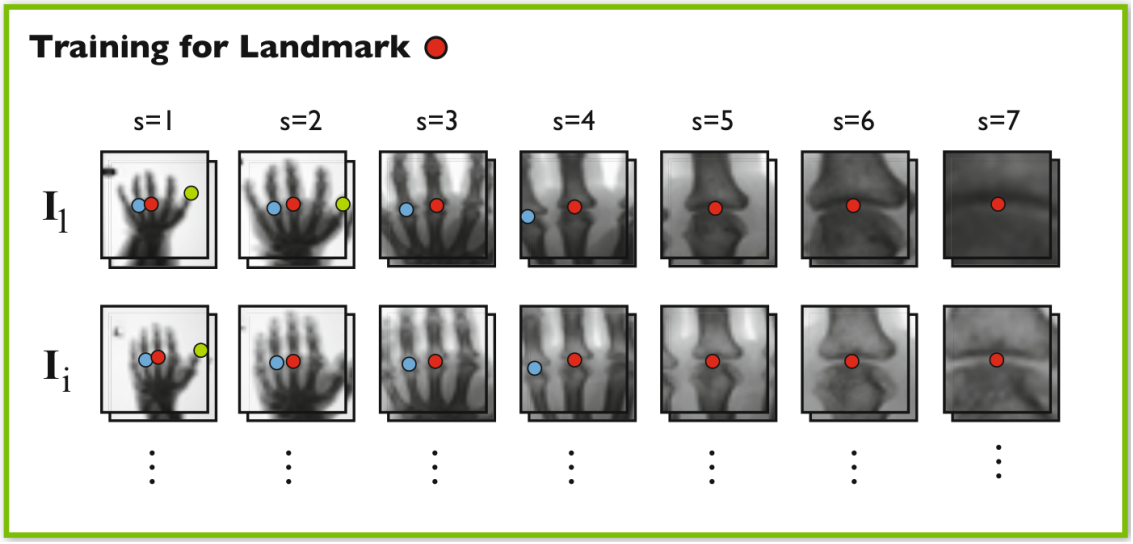


Figure 2.2: TDPR Training: Construction of the patch codebook over several scales for each landmark. (Source: [20])

(PCA) [4], resulting in PCA coefficients $\mathbf{P}_{PCA}$. The final codebooks are consisting of the tuples $< \mathbf{P}^p_{PCA}, \mathbf{L}^p >$.

Statistical Shape Model To model the spatial distribution of the landmark positions $\boldsymbol{s} = \langle \boldsymbol{x}^i_1, ..., \boldsymbol{x}^i_L \rangle$, a PCA based shape model $S = \{\overline{\boldsymbol{s}}, \boldsymbol{S}\}$ is learned. The shape model consists of the mean shape $\overline{\boldsymbol{s}}$ and the Eigen-decomposition $\boldsymbol{S}$ of the covariance matrix of the landmark positions. This is a generative model, allowing to construct shapes using a parameter vector $\boldsymbol{b}$ in the following way:

$$\boldsymbol{s} = \overline{\boldsymbol{s}} + \boldsymbol{S}\boldsymbol{b} \tag{2.1}$$

This model allows to generate linear combinations of all shapes occurring in the training set.

2.4.1.2 Testing

As can be seen in Fig. 2.3, multi-scale landmark localization starts at the largest scale $s = 1$. The matrix $\boldsymbol{L}^*_{s=1}$, holding all landmark position estimates, is initialized with the center of the test volume. For each landmark x, a patch $\boldsymbol{P}^x$, centered at $\boldsymbol{L}^*_s$, is extracted and projected on to the PCA space. The patch in PCA-space $\boldsymbol{P}^x_{PCA}$ is used for a nearest

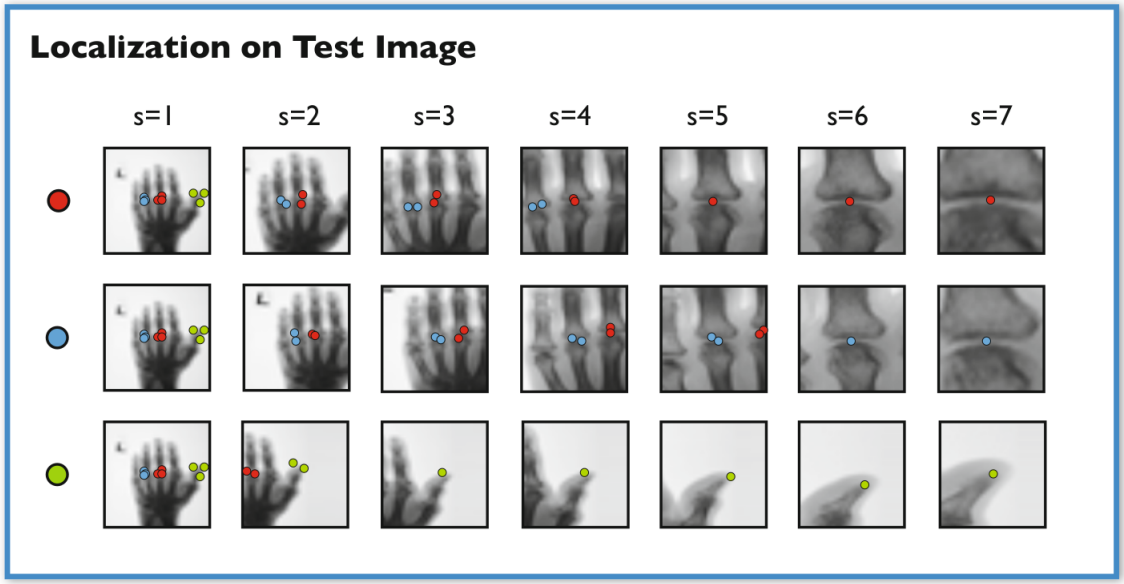


Figure 2.3: TDPR Testing: Multi-scale localization scheme, starting at the largest scale $s = 1$. For each landmark a patch is extracted and compared to the patch codebook, yielding multiple predictions for each landmark. (Source: [20])

neighbor search in the codebook according to the Euclidean distance, yielding the prediction $\boldsymbol{L}_p^{x*}$. Since a landmark may be visible on multiple patches coming from different landmarks, multiple predictions are available. The final landmark positions are estimated by taking the median over all predictions.

Statistical Shape Model For all scales $s \leq S - 3$, the position estimates $\boldsymbol{L}_s^*$ are regularized according to the PCA-based shape model S by projecting $\boldsymbol{L}_s^*$ onto the shape space $\boldsymbol{S}$, and reconstructing it back again. Thus, the landmark positions are restricted to linear combinations of shapes observed in the training set. This step actually prevents landmark localizations wandering off wrongly to neighboring landmarks with a similar local appearance. At the smallest scales $s \geq S - 3$, this regularization step is omitted and the prediction is made solely based on local appearance, thus achieving a good precision.

2.5 Marginal Space Learning

Recently, an object localization technique called Marginal Space Learning (MSL) [58], received a lot of attention. This work aims to estimate the pose parameters position,

orientation and size of an anatomical structure, which can be seen as estimating an oriented BB around an object. The authors showed the capability of MSL for localizing the heart-chambers in 3D cardiac CT volumes and this technique was also successfully applied to other problems, like the localization of wrist bones from CT [35], or the detection of axillary lymph nodes from CT [2].

In 3D object localization by BBs there are 9 unknown parameters to estimate, namely three for translation, scales in three directions and three rotation angles. The number of hypotheses increases exponentially with the number of parameters, which makes an exhaustive search over all parameters infeasible, especially in 3D.

2.5.1 Idea

The idea behind MSL is to avoid an expensive exhaustive search over all parameters, but to split the problem into smaller sub-problems by first estimating a limited set of parameters, while keeping the remaining parameters fixed. In subsequent steps, more parameters are included into optimization, but the parameter space is restricted based on the estimations of the previous parameters.

2.5.2 Toy Example

The toy example in Fig. 2.4 shows the reduction of the parameter space, when finding the maximum of a joint probability P(X,Y) with two parameters X,Y. First, the marginal distribution P(Y) with only one parameter Y is estimated. The number of possible hypotheses for the parameter space spanned by Y is much smaller than the space spanned by X and Y together. As a result, we obtain a few candidates for the parameter Y. In any subsequent steps, the search space of parameter Y is reduced according to the obtained candidates. A second classifier is applied on a restricted space to estimate the joint distribution P(X,Y).

2.5.3 Application to Object Localization

For object detection, the authors of MSL propose to estimate parameters in the following sequence:

1. translation

2. translation and orientation

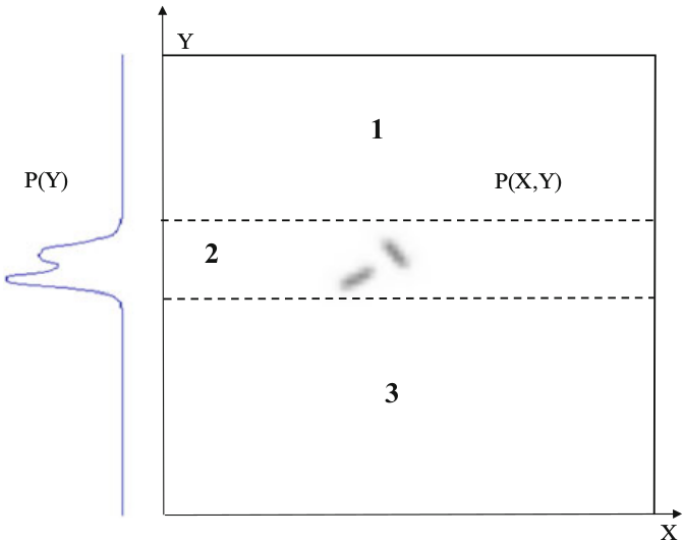


Figure 2.4: Toy example showing the parameter space reduction using MSL. The joint probability P(X,Y) is estimated within the restricted space by first estimating the marginal distribution P(Y) to reduce the search space. (Source: [58])

3. translation, orientation and scale

In each of the three steps a classifier based on the Probabilistic Boosting Tree (PBT) [52] is used to select the best candidates for the set of parameters to estimate. In the subsequent step, those candidates are augmented with additional parameters, used in the next step, and again a classifier is choosing the best candidates.

A common approach when estimating the orientation of an object is to use one classifier, which is trained at one fixed orientation. At testing, the image is rotated in discrete steps and the classifier is evaluated on each rotated version of the image [18]. To overcome the limitation of expensive image rotations, steerable features are used for training an efficient classifier for object orientation and scaling. The idea behind steerable features is, that the locations of the features are depending on the hypotheses for orientation and scaling. Local features are sampled using a sampling pattern. This pattern is rotated and scaled according to the orientation and scaling of the hypothesis as shown in Fig. 2.5. Thus, the classifiers for all orientations and scales can be trained and during testing hypotheses with different orientations and scales can be tested by transforming the sampling pattern instead of the input image, which makes the algorithm much faster.

Since structures with shape variations are not handled very well when estimating only the rigid transformation parameters, Nonrigid MSL was proposed in [59]. Nonrigid MSL aims to additionally estimate shape parameters of a PCA based SSM.

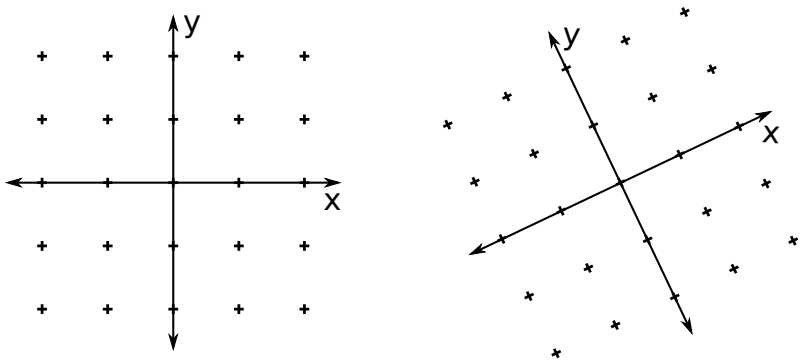


Figure 2.5: Rotation and scaling of the feature sampling pattern according to a hypothesis of object orientation and scaling. Each '+' within the image corresponds to a location, where local appearance is sampled. (Source: [58])

2.5.4 Conclusions

MSL is an object localization technique, well suited for detecting BBs around anatomical structures in 3D images due to its efficiency. In [35] it was shown that MSL can also be applied to detect wrist bones from T1 and T2 weighted MRI images. However, for our goal, the localization of hand bones, further effort is necessary, since MSL does not handle the presence of repeating structures. One approach to overcome this limitation is to use an iterative scheme as proposed in [34], which significantly adds to the computational cost.

2.6 Localization using Random Forests

Due to the progress in the field of machine learning, powerful algorithms such as RFs [6] have emerged and have become a widely used technique for localization of anatomical structures. Therefore, we give a small overview over existing related work.

An RF is a powerful machine learning algorithm, able to perform classification and regression tasks by taking as an input a set of features. The objective of classification is to predict discrete class label, e.g. perform a voxel-wise classification. The goal of regression is to predict a continuous label, e.g. the relative distance from a patch to a landmark.

An RF is an ensemble [5] of multiple decision trees, where each tree performs hierarchically arranged feature tests, thus ending up in leaf nodes where predictions are stored. Like in every ensemble method, the predictions of all the trees are combined, e.g. by averaging [6]. By introducing randomness during training of the decision trees, e.g. using bagging [5], the individual decision trees are decorrelated, thus making the ensemble a very powerful classifier. A more detailed survey about RFs can be found in Chapter 3.

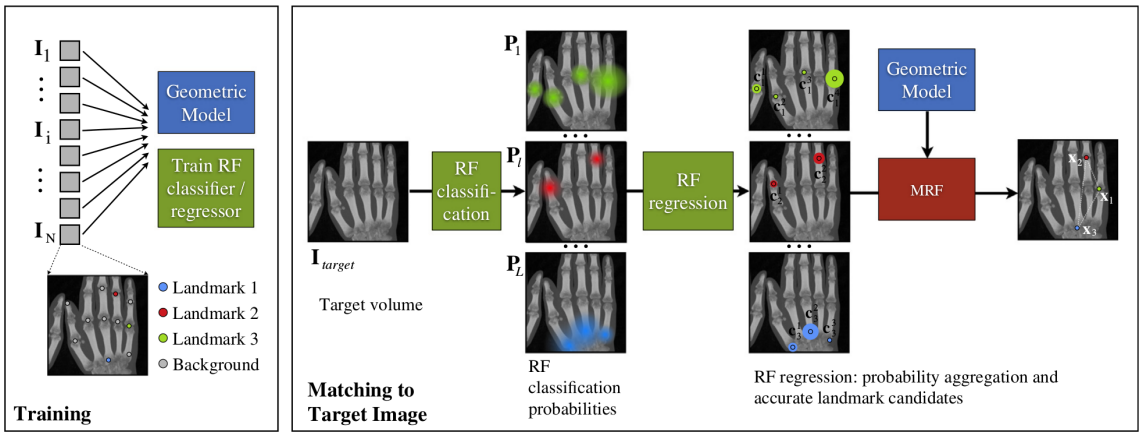


Figure 2.6: Pre-filtered Hough Forests: Overview of the localization pipeline. (Source: [21])

In the following, we divide the literature on detection algorithms using RFs into localization based on local and global context.

2.6.1 Local Context

This section is about localization algorithms, where RFs are employed to capture the local context, thus a subsequent step, performing reasoning on the high-level semantics of the RF output is needed to handle the presence of repeating structures.

2.6.1.1 Pre-filtered Hough Forests and Discrete Optimization

The work proposed in [21] aims to localize anatomical landmarks in 2D and 3D medical images, with a special focus to handle the presence of repeating structures, thus making the algorithm perfectly suitable for localization of landmarks within the hand.

Localization consists of three steps, as shown in Fig. 2.6. At the beginning, multiple candidate positions are obtained for each landmark via voxel-wise RF classification. In the next step, candidate positions are refined using RF regression. Among those refined candidates, the one resulting in the highest probability according to a geometric model, is selected using a Markov Random Field (MRF). The geometric relationship between landmarks is automatically learned from training data. In the following, we will discuss each step in more detail.

RF Classification The objective of the first step is to obtain candidates for the landmark positions, which is achieved using RF classification. An RF is used to predict for

each voxel within an image a discrete class label corresponding to background or one of the L landmarks, resulting in a classification task with $L + 1$ discrete class labels.

When training the RF, positive samples for each landmark are chosen by taking all voxels within a small radius around each landmark, while background samples are drawn randomly from areas of the images outside a small radius around all landmarks. The RF learns to distinguish between the classes by local appearance around each voxel. Local appearance is described by a feature vector, consisting of gray value differences between the voxel and voxels at random offsets around the voxel.

During testing, each voxel within the image, is classified using the previously trained RF, resulting in probability maps $\boldsymbol{P}_c$ for each landmark c.

RF Regression The information obtained in the previous step is refined using an RRF, or also called Hough Forest [25], to get precise candidates for the landmark positions.

For each landmark, a Hough Forest is trained to predict relative distance vectors $\boldsymbol{d}_c = \boldsymbol{l}_c - \boldsymbol{v} = \{d_x, d_y, d_z\}$ from a reference voxel $\boldsymbol{v}$ to a landmark $\boldsymbol{l}_c$, according to feature tests around the reference voxel. Training voxels are selected by taking all voxels in a small region around the ground-truth landmark positions.

During testing, the probability maps $\boldsymbol{P}_c$ obtained from RF classification, are thresholded with $\beta = 0.5 \cdot max(P_c)$. The remaining voxels are pushed through the Hough Forests of each landmark, thus getting for each voxel a relative prediction vector $\boldsymbol{d}_c$. The voxels of the probability map are shifted by $\boldsymbol{d}_c$, resulting in a highly accurate probability map. The probability $p(\boldsymbol{c}_l)$ of a candidate $\boldsymbol{c}_l$ at a certain location, is calculated as the sum over all probabilities from the classification RF shifted to the candidate position. The number of candidates is reduced using non-maxima suppression. Further, only the D candidates with the highest probabilities are used, thus resulting in accurate candidate positions for each landmark.

Candidate Selection using a Geometric Model The task of this final step is to select from all the candidate positions for each landmark the best candidate using an MRF. An MRF is an undirected graph, with each node in the graph corresponding to one landmark. The edges e, connecting the nodes in the graph, are modeling geometric relationships between the landmarks.

Not all landmark positions can be used to reliably predict another landmark's position. Therefore, the topology of the graph is automatically learned by connecting each node only to the nodes with a strong geometric relationship, according to a differential entropy

measure.

The goal of solving the MRF is to assign to each node in the graph one of the candidates, such that the confidence

$$Conf(M) = \sum_{c=1}^{L} \boldsymbol{U}(c, M(c)) + \sum_{e=1}^{E} \boldsymbol{B}(e, M(e)) \tag{2.2}$$

of the landmark configuration M, consisting of unary terms $\boldsymbol{U}$ and binary terms $\boldsymbol{B}$, is maximized. The unary terms, representing likelihoods of candidates, are set to the normalized probabilities of the candidates $p(\boldsymbol{c}_l)$. The binary terms are modeling the confidence between two landmarks connected by an edge e according to their geometric relationship. The confidence value of an edge between two landmarks s,t located at $\boldsymbol{l}_s$ and $\boldsymbol{l}_t$ is a function $k_{s,t}(\boldsymbol{l}_s - \boldsymbol{l}_t)$ depending on the relative offset between the two landmarks. This function $k_{s,t}$ is constructed during training by summing up normal distributions at all landmark offsets occurring in the training set.

Having calculated all the confidence values for the unary and binary terms, the MRF is solved using loopy belief propagation [39].

2.6.1.2 Shape Model Fitting using Random Forest Regression Voting

In work described in Section 2.6.1.1, a shape model was incorporated by performing discrete optimization to select the landmark positions among a limited set of candidate positions. In [8] a more generic way of including a shape model was proposed, skipping this discretization into candidate positions and directly fitting a shape model to the response of a previous feature detector as can be seen in Fig. 2.7. They employ a Hough Forest as a feature detector, very similar to the one used in the previous section, which generates response images based on local appearance. Hough forests are applied to randomly sampled patches of an input image, and landmark positions relative to the patches are estimated. The resulting position estimates are accumulated in a response image, which can be seen as a probability map of a landmark position. Finally, an SSM is fitted directly to the obtained response image using the Constrained Local Model (CLM) framework [15] as discussed in the following.

Constrained Local Models The objective of CLMs [15] is to fit a statistical shape model of the landmarks c to a probability map $P_c(\boldsymbol{l})$ of the landmark positions $\boldsymbol{l}$. The

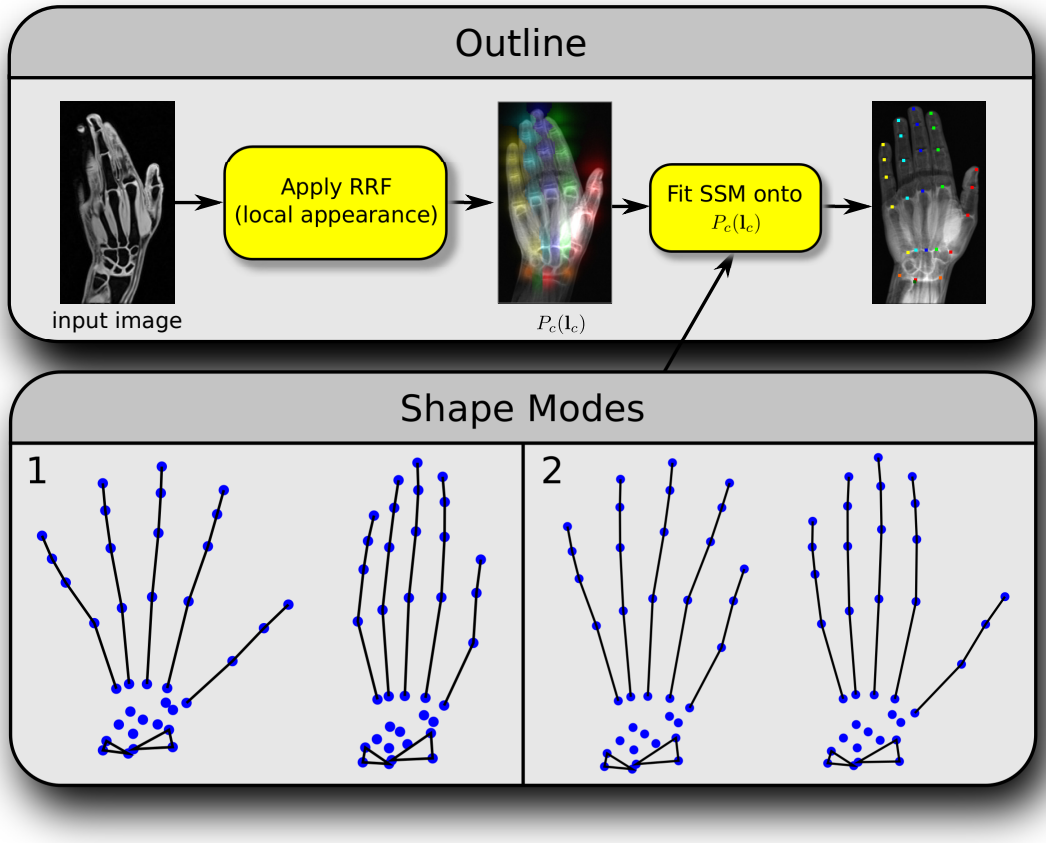


Figure 2.7: Outline of the shape model fitting approach using Random Forest Regression voting to obtain a probability map $P_c(\boldsymbol{l}_c)$. The image below shows the first two shape modes of the SSM, where captured shape variations can be seen. (Source: [8])

probability map is reformulated to a log-likelihood function

$$C_c(\boldsymbol{l}) = -\log(\max(P_c(\boldsymbol{l}), p_0)), \tag{2.3}$$

where $p_0 > 0$ increases the likelihood of landmarks being at locations with a probability of 0, thus increasing the robustness in the presence of occlusions.

The shape model consists of a similarity transform T with parameters $\boldsymbol{t}$, combined with a PCA-based SSM with parameter $\boldsymbol{b}$, learned from training data. The resulting generative model is of the form

$$\boldsymbol{l}_c = T(\bar{\boldsymbol{l}}_c + \boldsymbol{P}_c\boldsymbol{b}; \boldsymbol{t}), \tag{2.4}$$

where $\bar{\boldsymbol{l}}_c$ holds the mean landmark positions and $\boldsymbol{P}_c$ the most important eigenvectors of

the covariance matrix.

Fitting of the SSM is formulated as an optimization task, with the goal of maximizing the probabilities of the landmark positions according to the log-likelihood $C_c(\boldsymbol{l})$, such that the landmarks are restricted to a hyper-ellipsoid in shape space. Mathematically, this fitting can be formulated as finding parameters $\{\boldsymbol{b}, \boldsymbol{t}\}$, minimizing an energy

$$Q(\boldsymbol{b}, \boldsymbol{t}) = \sum_{c=1}^{L} C_c(\boldsymbol{l}_c) \qquad \text{subject to} \quad \boldsymbol{b}^T \boldsymbol{S}_b^{-1} \boldsymbol{b} \leq M_t, \tag{2.5}$$

where $\boldsymbol{S}_b$ is the covariance matrix and M_t a threshold on the Mahalanobis distance. This energy minimization aims to maximize the landmark probabilities and limits the shapes to the hyper-ellipsoid $\boldsymbol{b}^T \boldsymbol{S}_b^{-1} \boldsymbol{b} = M_t$. This optimization is implemented using an iterative scheme with the following steps:

1. Initialize search radius: $r \longleftarrow r_{max}$

   and the initial landmark location l with the mean landmark location l_c

2. Find best landmark locations

   $$\boldsymbol{l}_i' = \arg\max_{\boldsymbol{l}} C_c(\boldsymbol{l}) \tag{2.6}$$

   in a certain radius r around the previous obtained landmark locations $\boldsymbol{l}_i$ according to the log-likelihood. In the first iteration the algorithm starts searching around the mean location $\bar{\boldsymbol{l}}_c$.

3. Estimate parameters $\{\boldsymbol{b}, \boldsymbol{t}\}$ from $\boldsymbol{l}_i'$. The similarity transform parameter $\boldsymbol{t}$ is estimated using Procrustes analysis [28], the shape parameters b by projection of $\boldsymbol{l}_i'$ onto the shape space.

4. Move $\boldsymbol{b}$ to the nearest point on the hyper-ellipsoid $\boldsymbol{b}^T \boldsymbol{S}_b^{-1} \boldsymbol{b} = M_t$.

5. Calculate new landmark locations $\boldsymbol{l}_i$ using the generative model in Equation (2.4).

6. As long as search radius $r > r_{min}$, reduce search radius r and repeat from step 2.

2.6.2 Global Context

Instead of modeling local appearance in combination with an SSM, there is also related work in anatomical landmark localization on making a prediction based on global context

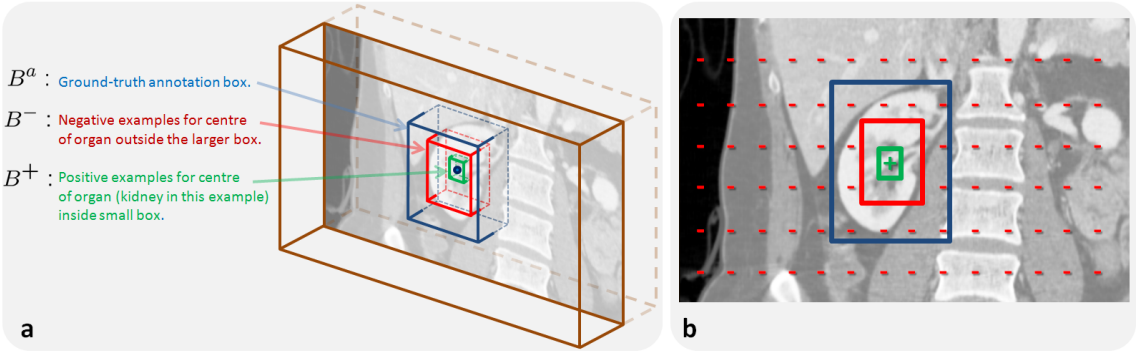


Figure 2.8: Positive and negative sample selection for training a classification RF. (Source: [13])

to handle repeating structures within medical images. This concept is realized by using context-rich features, describing not just local structures, but taking also information from all the surrounding anatomy into account.

2.6.2.1 Organ Localization using Random Classification Forests

Detecting and locating organs, such as kidneys, liver, heart, lung, etc., within CT-scans in the form of BBs was the goal of the work presented in [13]. Their proposed algorithm is capable of handling arbitrary field of view scans, which makes it necessary to tell whether an organ is present in the volume or not in addition to localization. They formulate this problem as a classification task, with the goal of assigning class labels to each voxel within an image. The class labels are corresponding either to one of the organs, or to the background. This classification task is performed using an RF, which is very similar to the one already described in Section 2.6.1.1, but instead of simple pixel comparisons they use context-rich features, thus capturing the global context.

Classification Labels for Training Their labels of the ground truth-database are consisting of bounding boxes around organs. When training a classification RF, one has to think about which voxels to use as positive, and which as negative training examples. As shown in Fig. 2.8, positive samples for an organ are selected from a small box in the center of the BB around an organ. All voxels outside a larger box, 50% of the dimensions of the BB, are considered as background class labels. Since they do not consider all voxels within the BB as positive samples, they train a classifier which produces only a small response on a testing image, thus increasing the localization precision.

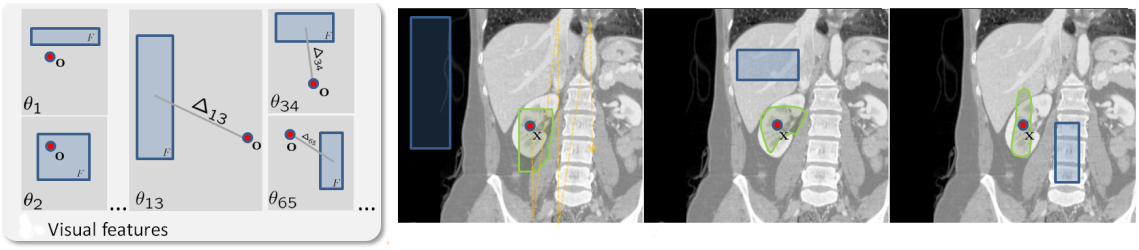


Figure 2.9: Image on the left is showing examples of the context-rich box-features, relatively defined to the reference voxel. Image on the right is illustrating, how the RF is able to capture the global context by performing feature tests, describing all surrounding structures. (Source: [13])

Context-rich features The goal of context-rich features is to describe the appearance of all surrounding structures, thus capturing the global context. The features are based on cuboids of arbitrary size and position in each dimension, with the parameters of the cuboids being denoted as $\boldsymbol{\theta}$. The cuboid positions are defined relative to the voxel position, as can be better understood with the help of the examples shown in Fig. 2.9. The feature response $f(\boldsymbol{x}, \boldsymbol{\theta})$ at position $\boldsymbol{x}$, using two cuboids F_1 and F_2 within an image I, is calculated according to

$$f(\boldsymbol{x}, \boldsymbol{\theta}) = \frac{1}{|F_1|} \sum_{\boldsymbol{q} \in F_1} I(\boldsymbol{x} + \boldsymbol{q}) - b \frac{1}{|F_1|} \sum_{\boldsymbol{q} \in F_2} I(\boldsymbol{x} + \boldsymbol{q}). \tag{2.7}$$

The parameter $b \in \{0, 1\}$ defines, whether to use the mean intensity within F_1 or the difference between the intensities of F_1 and F_2. The feature response can be calculated very efficiently using integral images [54].

Detection and Localization A previously trained RF is applied to all voxels within an image, to obtain for each class c probability maps $P_c(\boldsymbol{x})$. A class is considered present within an image, when the maximum probability

$$P_c = max\ P_c(\boldsymbol{x}) \tag{2.8}$$

exceeds a certain threshold $\beta = 0.5$. In case a class is present, the location is determined as the mean

$$\boldsymbol{x}_c = \sum_{\boldsymbol{x}} \boldsymbol{x} P_c(\boldsymbol{x}) \tag{2.9}$$

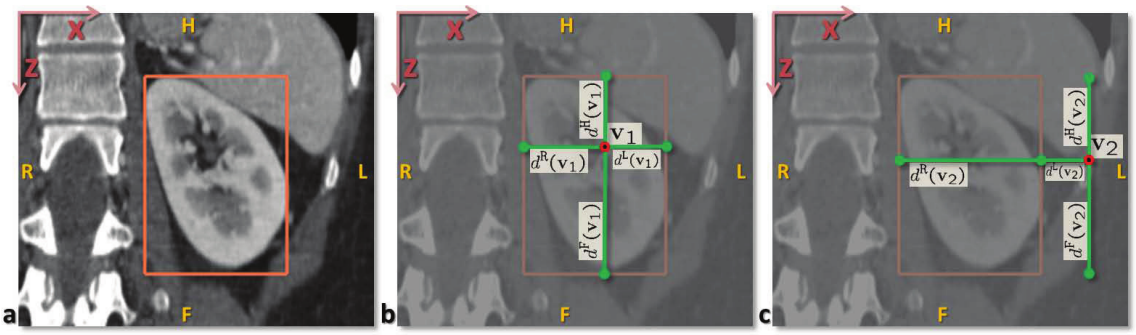


Figure 2.10: Locating BBs using RRFs: An RRF models the distances $d(\boldsymbol{v})$ from all voxels $\boldsymbol{v}$ (inside and outside the BB) within an image to all faces of the BBs. (Source: [11])

over the whole volume.

2.6.2.2 Organ Localization using Random Regression Forests

In the previous section an algorithm was presented, which achieves the goal of localization by performing voxel-wise classification and inferring the location from the classification labels in a subsequent step. As this indirect localization using classification is not optimal, a more direct approach was presented in [11], which aims to predict directly the faces of the BB around an organ. They skip the classification part, and formulate the problem as a regression task using an RRF. The regression outputs are the distances from voxels within an image to the BB faces in all three dimensions $d = \{x, y, z\}$, as shown in Fig. 2.10. The RRF makes the distance estimates using the same context-rich features as described in the previous section. When testing an image the distance estimated from voxels all over the image are accumulated in a voting space.

In comparison with the classification based technique, this approach provides in addition to the location of an object, information about the extent by means of a BB around an object. However, the same principles may also be used for localizing landmarks [26]. Another advantage is that localization is performed directly, without having to choose positive and negative training examples. Further, they could show that the error achieved using this regression scheme is less than half the error achieved using voxel-wise classification.

2.7 Conclusions

In this chapter we have provided an overview over different approaches for landmark localization. Low-level approaches such as landmark localization by local feature descriptors do not meet the requirements in terms of robustness to variations in clinical images.

MSL seems to be an effective tool for estimating multiple parameters such as location, rotation and scale of an object. A limitation is that MSL fails to handle repeating structures, therefore, iterative schemes trying to overcome these limitations were proposed. However, MSL is suitable for localizing BBs, but we prefer a localization of anatomical landmarks, due to a higher expected precision.

In the past, atlas-based approaches have enjoyed a lot of popularity, but they have a few drawbacks, like handling variations in shape and anatomy. Due to the progress and development of new algorithms in the field of machine learning, such state of the art approaches are able to clearly outperform atlas-based approaches in terms of precision, robustness and runtime.

To handle repeating structures, SSMs have been widely used in the past. As an example approach using SSMs, we have presented a method called TDPR, which is able to allow a fast localization of landmarks using a patch codebook and is also able to handle repeating structures by including an SSM.

Due to the popularity of RFs for landmark localization, we compared different approaches based on RFs. We divided RF localization algorithms into those making a prediction based on local and global context. Using local appearance, i.e. capturing the local context, requires a subsequent geometric model to handle repeating structures. This step can be omitted when using context-rich features, thus capturing the global context.

Localization using RFs can be formulated as a voxel-wise classification problem using classification RFs. However, formulating localization of anatomical structures as a regression task using RRFs seems more natural and allows better results compared to classification approaches.

For our goal of localizing landmarks within hand MRI images, we propose a novel algorithm and evaluate its localization accuracy by comparing to TDPR and RRFs in Chapter 5.

Chapter 3

Random Regression Forest Framework

Contents

In this chapter we introduce the basic concepts of Random Forests (RFs) as well as our Random Regression Forest (RRF) framework for localizing landmarks, inspired by the work presented in [11]. We extend this RRF framework by introducing a novel weighting scheme in 3.4.3, which lets local structures have a higher contribution to the position estimation. In Chapter 4, multiple instances of this framework are combined to get our final proposed landmark localization algorithm.

An RF is a machine learning algorithm based on decision trees that can be used for many different kinds of problems, e.g. classification and regression tasks. The goal in classification problems is to assign a discrete class label c to a generic object called datapoint, while in regression problems a continuous label y is predicted from a given datapoint.

As discussed in Chapter 2, the task of landmark localization can be formulated either as a classification or a regression task. However, localization via regression is more intuitive and allows better precision. Therefore, we will focus on the application of RFs for regression

problems, i.e. RRFs. A broader overview over other applications of RFs can be found in [12].

3.1 Decision Trees

A decision tree is a model allowing to solve complex problems by performing hierarchically arranged feature tests. It is a directed graph consisting of split nodes and leaf nodes connected with edges in a tree structure. Each split node has edges to two child nodes and stores a binary decision, which guides datapoints according to a feature test to the left or the right child node. Starting at the root node, which is the first split node in a tree, feature tests are applied recursively at each split node until a leaf node is reached. At each leaf node a final answer (prediction) is stored. The basic structure of a tree, as well as a toy example of a decision tree is shown in Fig. 3.1.

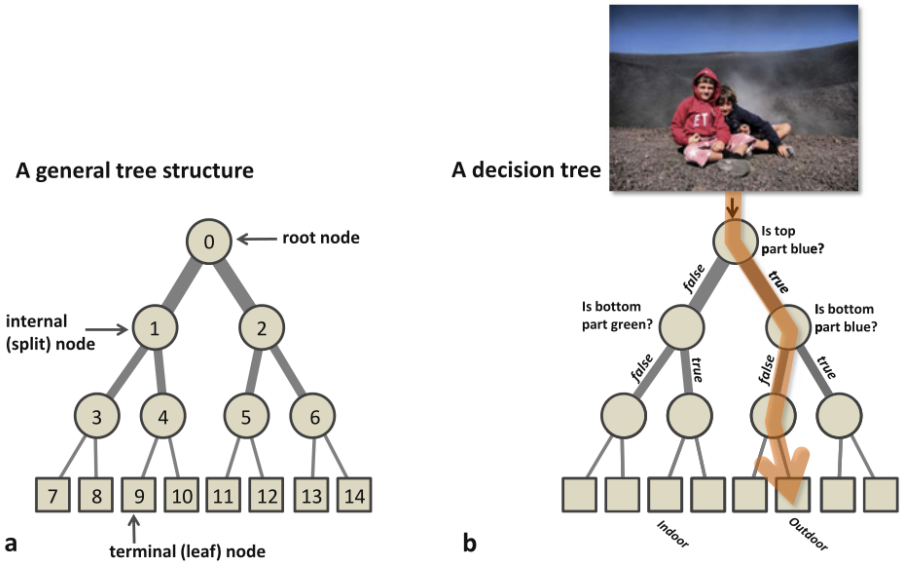


Figure 3.1: **a)** Basic structure of a decision tree consisting of split nodes (circles) and leaf nodes (rectangles). **b)** Toy example of a decision tree for classifying the given image. (Source: [12])

An alternative interpretation of a decision tree is, that a complex problem that needs to be solved is divided hierarchically into smaller sub-problems, which are easier to solve.

One can think of creating a decision tree by manually arranging feature tests. However, for more complex tasks, such as we are dealing with, decision trees are learned automatically from training data. This task is referred to as training of trees, while testing of trees

means to apply a previously trained decision tree to unseen data. Training and testing of decision trees for the goal of landmark localization will be the topic of the following chapters.

3.2 Random Decision Forests

A Random Decision Forest, or also called RF, is a machine learning algorithm originally proposed in [6]. An RF is an ensemble of multiple decision trees, learned automatically from training data. During training, the decision trees are built independently and randomness is injected to ensure that the different trees are uncorrelated, thus learning different aspects of the data. When combining all trees to an ensemble of decision trees, the outcome is a powerful machine learning algorithm. The big advantage of an RF over a single fully optimized decision tree is the increased generalization capabilities. Here, generalization describes how well an algorithm performs on previously unseen testing data. The opposite of good generalization is often referred to as overfitting, which means that the algorithm performs well on training images, but not very well on testing data.

To avoid overfitting and achieve a good generalization with RFs, it is very important that the individual trees are decorrelated. Thus, each single tree is sub-optimal and usually not as good as a fully optimized decision tree, but the ensemble of decorrelated sub-optimal trees performs well. Decorrelating the trees is achieved by injecting randomness during training. Usually, the two following methods are commonly used in practice:

- Training set sampling: When training the different trees, each tree is trained on a randomly chosen subset of the training data. This principle is called *Bagging Predictors*. [5, 6]

- Random node optimization: During construction of the trees, randomly chosen subsets of the feature vectors are used. [1, 31]

During testing of an RF, all decision trees are applied to the data, thus getting as much predictions as the number of trees in the forest. There are different ways of combining those predictions to a final prediction, but the most commonly used one is to average over all predictions coming from the trees.

3.3 Random Regression Forests

An RRF is an RF, which predicts one or multiple continuous labels $\boldsymbol{y}$ based on an arbitrary number of feature dimensions $\boldsymbol{x}$, i.e. the goal is to learn a function $\boldsymbol{y}(\boldsymbol{x})$.

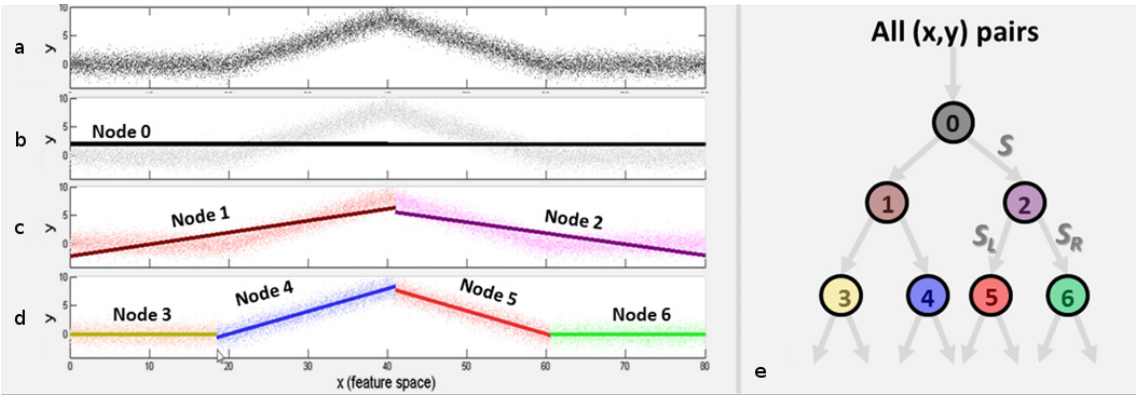


Figure 3.2: Toy example, showing how a regression tree is able to perform non-linear regression on the samples shown in **a**. Image **e** shows a regression tree, splitting the input feature space, thus partitioning the regression problem into smaller sub-problems. Images **b-c** show the leaf prediction models of the individual nodes of the tree.(Source: [11])

As the example in Fig. 3.2 illustrates, a non-linear regression problem can be approximated by splitting the function into smaller parts and estimating those smaller parts using simpler models, e.g. linear, or constant models.

This idea is picked up by RRFs, which split the feature space using hierarchically arranged feature tests within a tree structure. Due to splitting, only a small subspace of the whole feature-space reaches each leaf node, where a prediction model is stored.

When combining the regression trees to an ensemble by averaging over the predictions, the outcome is a regression function, which generalizes well as can be seen in Fig. 3.3 by looking at the predicted values further away from training samples. Single trees may have a non-smooth transition between the training-samples due to simple models in the leaf nodes, while the whole forest produces a smooth transition between training samples, thus generalizing well.

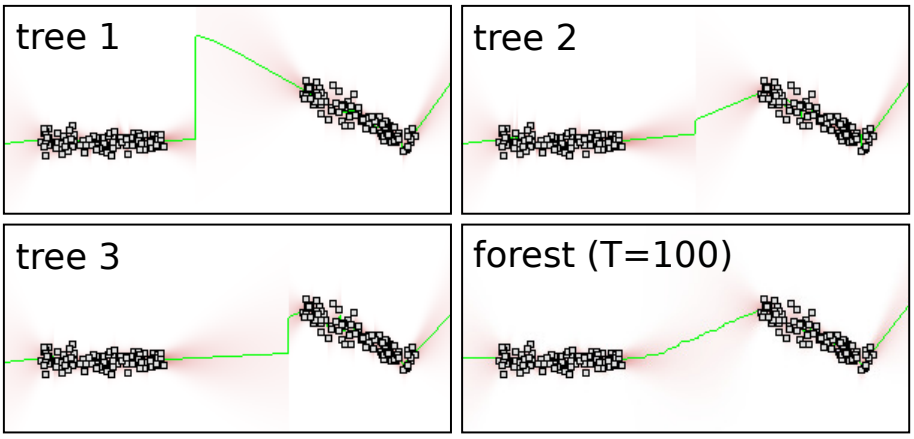


Figure 3.3: Visualization of the achieved effect, when combining multiple random regression trees to a forest, on the basis of a 1D regression problem. The x-axis within each image represents the input feature dimension, while the y-axis is the regression output dimension. Training samples are represented by gray dots in the images and the green line shows the learned function, obtained by applying each tree/RRF to all values on the x-axis. Single trees may overfit to the data, while the forest consisting of 100 trees produces a smooth transition between training samples.

3.4 Random Regression Forests for Landmark Localization

In the case of landmark localization, RRFs are used to predict the relative distances

$$\boldsymbol{d}_c(\boldsymbol{v}) = \boldsymbol{l}_c - \boldsymbol{v} \tag{3.1}$$

from the voxel positions $\boldsymbol{v}$ in an image to multiple landmark positions $\boldsymbol{l}_c$ in x, y and z direction. The features (input dimensions) of the regression problem are derived from the appearance around the voxel $\boldsymbol{v}$. An RRF is trained based on a set of input images with labeled ground-truth landmark positions $\boldsymbol{l}_c$. The datapoints used for training the RRF are all voxels within the training images. Feature values are calculated for all training voxels and the non-linear regression problem is learned to predict the displacements $\boldsymbol{d}_c(\boldsymbol{v})$ to the landmark positions.

During testing, this previously trained RRF is used to predict displacements to the landmark positions $\boldsymbol{d}_c(\boldsymbol{v})$ from multiple voxels within an image. The predicted displacements coming from different voxels can be seen as voting vectors which vote relative to the absolute voxel position $\boldsymbol{v} = x, y, z$. All obtained votes are accumulated in a voting space. This voting space can be seen as a probability map for the landmark location.

In the following, we describe how RFs are trained and applied to the task of landmark

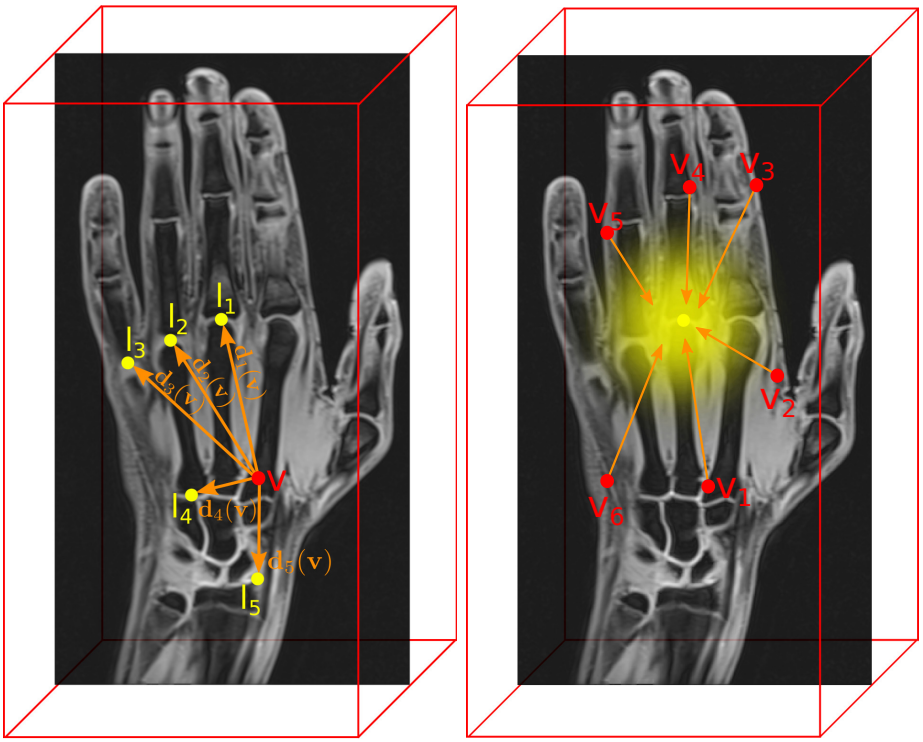


Figure 3.4: Left Figure shows the principle of RRFs predicting multiple landmark positions simultaneously relative to the voxel position $\boldsymbol{v}$. Right Figure shows that predictions $\boldsymbol{d}_c(\boldsymbol{v})$ coming from multiple voxels within the image are accumulated in a voting space. For each landmark a separate voting space is used.

localization.

3.4.1 Training

During training of an RF, the goal is to build decision trees by finding good splitting functions in the split nodes and to store predictions within the leaf nodes. We will define what is meant with good splitting functions later.

Forest training is achieved by constructing each of the T trees independently. The trees are trained in a greedy optimization manner, thus optimizing each of the nodes independently by selecting node splitting functions. Training starts at the root node, which is the first split node, with all voxels from the training images as datapoints. After finding the first node splitting function, voxels are sent either to the left or right child node. Training continues recursively on both child nodes and stops when the maximum tree depth D is reached or the number of voxels arriving at a node is less than a certain threshold $N = 20$ to avoid overfitting. At the leaf nodes, predictions are stored as distance

histograms according to the voxels reaching that node as illustrated in Fig. 3.5.

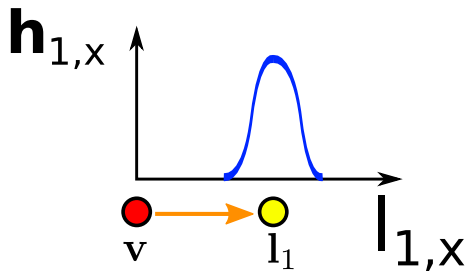


Figure 3.5: Illustration of a distance histogram for one dimension (x) representing the distribution of the distances from voxels in a leaf node to landmark positions in x-direction.

In the following, we will have a look at how the splitting functions are defined and how nodes are optimized.

3.4.1.1 Node Optimization

Node splitting functions split the set of voxels (S) reaching the node into voxels reaching the left (S_L) and the right child node (S_R), according to binary tests $h(\boldsymbol{v}, \boldsymbol{\theta}, \tau)$, with parameters $\boldsymbol{\theta}$ and τ, computed for a voxel $\boldsymbol{v}$. Binary tests are defined as

$$h(\boldsymbol{v}, \boldsymbol{\theta}, \tau) = f(\boldsymbol{v}, \boldsymbol{\theta}) > \tau, \tag{3.2}$$

where a feature response $f(\boldsymbol{v}, \boldsymbol{\theta})$ is thresholded with τ. The calculation of the feature response is described in the following.

Features Features are responsible for providing a good description of the area around the voxels. We use a generalization of the Haar-like features as proposed in [54], since they are able to provide a context-rich description.

Our feature response calculated for each voxel is the difference between the mean of the intensity values within two cuboids F_1, F_2, which can be written as

$$f(\boldsymbol{v}, \boldsymbol{\theta}) = \frac{1}{|F_1|} \sum_{\boldsymbol{q} \in F_1} I(\boldsymbol{v} + \boldsymbol{q}) - \frac{1}{|F_2|} \sum_{\boldsymbol{q} \in F_2} I(\boldsymbol{v} + \boldsymbol{q}), \tag{3.3}$$

where $I(\boldsymbol{v})$ is the intensity value at location $\boldsymbol{v}$ and $\boldsymbol{q}$ are the positions of all voxels within the cuboids. Using integral images they can be computed very efficiently. The cuboid positions are defined relative to the voxel position $\boldsymbol{v}$. The position in each of the 3 dimensions can be arbitrary within a certain range $[-fr_{max}, fr_{max}]$, where fr_{max} is the

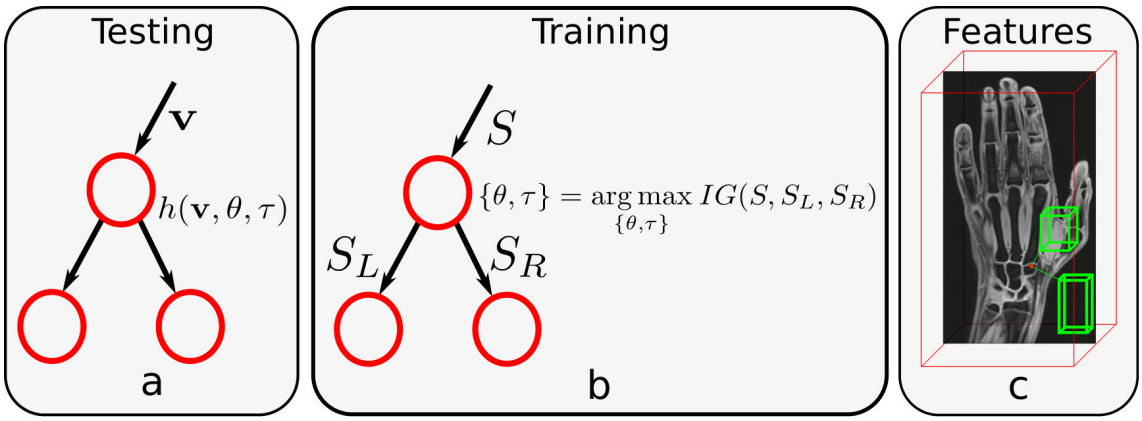


Figure 3.6: **a)** Shows how a single voxel is sent either to the left or right child node, according to the binary test $h(\boldsymbol{v}, \boldsymbol{\theta}, \tau)$. **b)** Shows how training data arriving at a node S are split into S_L and S_R to maximize an information gain criterion. **c)** Shows the principle of feature boxes used to calculate the feature response for a voxel.

maximum feature range. The size of the feature boxes is limited to fs_{max} in each dimension. The parameters describing the size and position of the feature boxes are summarized within the variable $\boldsymbol{\theta}$.

During node optimization, the goal is to select good features, which will be discussed in the following.

Feature selection To find good features, an information gain criterion IG is maximized at each node split. Since the parameter space of the features and thresholds is high dimensional, it is computationally expensive to find the best possible combination of feature and threshold at each node. Since we are interested in injecting some randomness in the training process, we can use *random node optimization*, thus considering only a small randomly chosen subset of the feature space.

At each node, F random feature parameters $\boldsymbol{\theta}$ and for each of them, L random thresholds τ_i are generated. The node splitting function $h(\boldsymbol{v}, \boldsymbol{\theta}, \tau)$ splits for all combinations of features and thresholds the set of voxels S arriving at the node into S_L and S_R. The best feature and threshold, defined as the one which results in the largest information gain IG according to

$$IG(S, S_L, S_R) = H(S) - \sum_{i \in \{L,R\}} \frac{|S_i|}{|S|} H(S_i), \tag{3.4}$$

is selected and stored in the split node. The maximization of the information gain aims to minimize the entropies $H(S_{\{L,R\}})$, reflecting the uncertainties $\Lambda_c(S_{\{L,R\}})$ of the voting vectors coming from the voxels in left and right child node. With entropy $H(S)$ and uncertainty $\Lambda_c(S_{\{L,R\}})$ defined as following:

$$H(S) = \sum_c p(c;S) \cdot log|\Lambda_c(S)| \tag{3.5}$$

$$\Lambda_c(S) = \frac{1}{|S|} \sum_{i \in S} ||\mathbf{d}_c(\boldsymbol{v}_i) - \frac{1}{|S|} \sum_{j \in S} \mathbf{d}_c(\boldsymbol{v}_j)||^2, \tag{3.6}$$

The variable $p(c;S)$ is the ratio between the number of voxels that vote for landmark c and the total number of voxels within the set S. In the case of all voxels voting for all the landmarks simultaneously, the variable $p(c;S)$ is equal for all classes and can therefore be ignored.

3.4.1.2 Leaf Node Statistics

At each leaf node, we compute for the x, y and z components of $\boldsymbol{d_c}(\boldsymbol{v})$ a 1D histogram of all the voxels reaching the node. The histograms are denoted as $\boldsymbol{h}_{d,c}(l_t(\boldsymbol{v}))$, where d is one of the three dimensions $d = x, y, z$. Those histograms are stored at the leaf node, to be available when testing an image.

3.4.2 Testing

After having described the procedure for training an RRF, we will now show how to use it to perform landmark localization.

During testing, voxels are pushed through all of the T trained trees. Starting at the root node, voxels are passed recursively to the left or right child node, according to binary feature tests stored at the split nodes, until a leaf node $l_t(\boldsymbol{v})$ is reached. We apply the distance estimates given by the histograms at the leaf nodes $\boldsymbol{h}_{d,c}(l_t(\boldsymbol{v}))$ relative to the voxel positions $\boldsymbol{v}$ and sum them up with a weight $w_c(\boldsymbol{v})$, according to (3.7), to get for each landmark three histograms $\boldsymbol{h}_{d,c}$, representing the probabilities of a landmark being located at a certain position separately for x, y, and z.

$$\boldsymbol{h}_{d,c} = \frac{1}{T \cdot \sum_v w_c(\boldsymbol{v})} \sum_{t=1}^{T} \sum_{\boldsymbol{v}} w_c(\boldsymbol{v}) \boldsymbol{h}_{d,c}(l_t(\boldsymbol{v})) \tag{3.7}$$

The final probability estimate $p(\boldsymbol{l}_c)$ for each class is obtained by the product of the three histograms for x, y and z as follows

$$p(\boldsymbol{l}_c = (v_x, v_y, v_z)) = \boldsymbol{h}_{x,c}(v_x) \cdot \boldsymbol{h}_{y,c}(v_y) \cdot \boldsymbol{h}_{z,c}(v_z) \tag{3.8}$$

and the final landmark estimates for the landmark positions $\hat{\mathbf{l}}_c$ by the maximum of $p(\boldsymbol{l}_c)$ according to

$$\hat{\mathbf{l}}_c = \arg\max_{\mathbf{l}_c} p(\mathbf{l}_c). \tag{3.9}$$

3.4.2.1 Alternative Voting Schemes

Above we have presented a method to obtain a probability for a landmark being at a certain position in each dimension $\boldsymbol{h}_{d,c}$ by summing up the leaf histograms $\boldsymbol{h}_{d,c}(l_t(\boldsymbol{v}))$. This means that each voxel is voting with distance histograms for the landmark positions. This approach is referred to as histogram voting scheme in the following. Summing up of all the leaf histograms is computationally expensive and the memory requirements are high, due to histograms stored in the leaf nodes of the trees. Therefore, we will investigate alternative voting schemes to overcome those drawbacks. Inspired by the voting schemes proposed in [8], we compare following schemes using

1. **histogram voting:** multiple votes from the training samples, i.e. voting with histograms,
2. **single voting(mean):** a single vote at the mean offset,
3. **single voting(max):** a single vote at the maximum of the histogram calculated for the offsets,
4. **weighted single voting:** a single weighted vote at the mean offset, using a weight $|\Lambda_c|^{-0.5}$ to put less emphasis on votes with a high uncertainty,
5. **Gaussian voting:** a Gaussian spread of votes, using a Gaussian with covariance matrix Λ_c.

According to the results of [8], single voting seems to be a good alternative to histogram voting, gives even better results and is also much faster than histogram voting. A disadvantage of single voting is that the uncertainties of the leaf nodes are not propagated to the final probability distribution, e.g. the final probability distribution may have a low

uncertainty, even when leaf histograms have a high uncertainty. This has the consequence that the uncertainties of the probability distributions are less meaningful. Weighting of the single votes gives no additional benefit in terms of precision. Casting Gaussian votes gives similar results as the single voting scheme, but is significantly slower.

In Chapter 5, we will compare only the results of single voting (mean), single voting (max) to histogram voting, since the other approaches show no benefits according to [8].

3.4.3 A Novel Scheme for Weighting of Votes

One of our main contributions is the introduced weighting factor $w_c(\boldsymbol{v})$ in (3.7). When applying an RRF to an image, votes coming from different parts of the image are accumulated, but of course not all of the votes can provide a precise prediction. However, coarse localization of landmarks is supported by global information from all over the image, while closer structures provide more information to increase the precision of landmark localization. We realize this idea by introducing our weighting function, which increases the contribution of local structures by decreasing the weight of the voting vectors according to their length $||\boldsymbol{d_c}||$. This weight is computed as

$$w_c(\boldsymbol{v}) = e^{-||\boldsymbol{d_c}|| \cdot \alpha}, \tag{3.10}$$

where α is a parameter allowing to adjust the steepness of the weighting function. For example using a large value for α would decrease the weights very fast for increasing voting vector lengths, thus only very local information is used to predict the landmark positions. The voting vector $\boldsymbol{d_c}$ has to be estimated from the distance histograms in x, y and z dimension, which is done by calculating the mean of the histograms $\boldsymbol{h}_{d,c}(l_t(\boldsymbol{v}))$ in the leaf nodes.

3.5 Conclusions

In this chapter we have presented our localization framework based on RRFs, inspired by the ideas presented in [11]. Given an MRI input image, and a previously trained RRF, the algorithm is able to localize landmarks by estimating a probability distribution of the landmark positions. Our main contribution in this framework is the introduced weighting scheme, which lets local structures around the landmark have a higher contribution to the final landmark positions. Thus, we can increase the localization accuracy in terms of mean and standard deviation, as can be seen from the Experiments in Chapter 5.

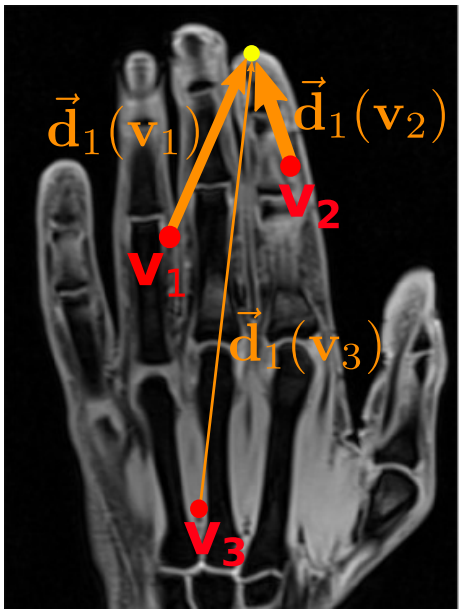


Figure 3.7: Illustration, showing weighting of votes according to the voting vector length. Larger weights are indicated as bolder vectors.

Chapter 4

Multiscale Random Regression Forests for Landmark Localization

Contents

In this chapter we propose a novel multi-scale setup of multiple RRFs based on the framework introduced in Chapter 3 and our novel idea of the weighting factor from Section 3.4.3.

In Section 4.1, we start with the motivation for using a multi-scale setup of RRFs, Section 4.2 will explain the concept based on an example with two consecutive RRF stages. The limitations of this two-step approach such as localization results wandering off wrongly to neighboring landmarks are topic of Section 4.3. To overcome those limitations, we extend this two-step approach to have an arbitrary number of consecutive RRF stages in Section 4.4 and add an implicit model of the landmark configuration to the RRF framework in Section 4.5.

4.1 Motivation

The location of anatomical landmarks is constrained by all of their surrounding structures. However, global information from all over the image supports to distinguish between the repeating structures, while closer structures provide the information for a precise landmark localization. We realize this concept by using multiple localization steps in combination with the weighting scheme introduced in Section 3.4.3. The first steps aim to predict landmarks based on the global shape of the hand, while the latter steps refine this prediction based on local appearance around the landmarks. The weighting scheme lets local structures have a higher contribution to the estimation of landmark positions.

To implement this idea, the RRF framework [11], as described in Chapter 3, is perfectly suitable, since it selects proper image structures that vote for landmark distances in a probabilistic fashion, where position estimates can be weighted by the distance to the estimate. Additional information about the landmark position can subsequently be obtained by connecting multiple estimation steps, where the output of individual steps restricts the area for estimating landmarks in the following step. This connection is made by using several RRF stages, that gradually decrease the areas around landmarks, where structural information is taken from. Together with the weighting scheme, we regard this idea as our main contribution compared to related work [11, 21].

4.2 Two Step RRF Localization

For our application of landmark detection from hand MR images, we propose using two RRF steps as shown in Fig. 4.1. In the following, we describe the two landmark detection steps, each using an RRF based on the framework introduced in Chapter 3. We refer to the combination of first and second detection step as our Gradually Improving Random Regression Forest (GIRRF) localization method.

4.2.1 CRRF - Coarse Random Regression Forest

The first RRF, referred to as Coarse Random Regression Forest (CRRF), coarsely locates the landmarks using appearance information from all over the image, i.e. CRRF is capturing the global context. Appearance information is modeled using long-range context-rich features. This strategy allows to distinguish between the different repeating structures within the hand, without explicitly modeling geometric relationships between the landmarks.

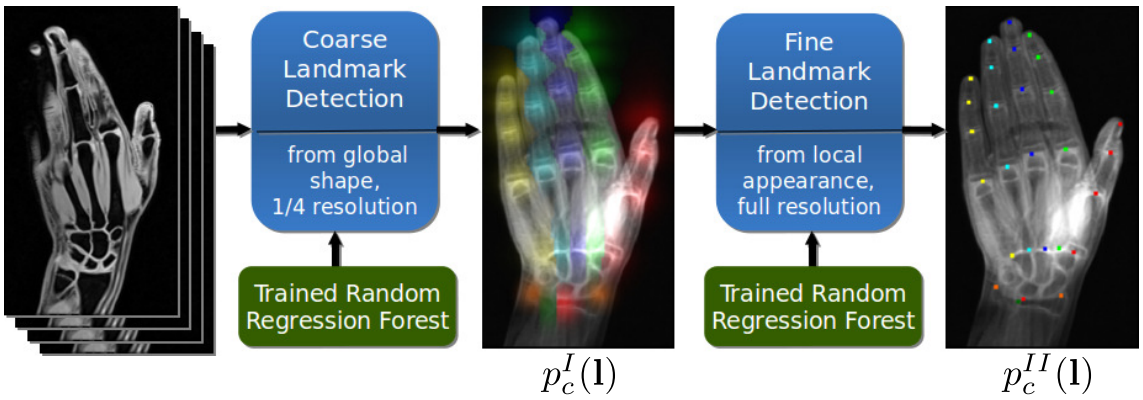


Figure 4.1: Overview of the proposed localization method using RRFs at different scales. The probability distributions $p_c^I(\boldsymbol{l})$ and $p_c^{II}(\boldsymbol{l})$ are presented in 2D images, where the the 3D MRI image was projected to 2D by summing up all intensity values along the z-dimension.

4.2.1.1 Training

We train an RRF according to Section 3, using all voxels within the training images as datapoints. Input images are resampled to a quarter of the original resolution, since this first step only requires a coarse localization, and experiments on full resolution did not show any additional benefit in terms of localization accuracy and precision.

All voxels used for training vote for all landmark positions simultaneously. This means that all landmarks are considered when calculating Information Gain (IG) and at the leaf nodes, histograms for the displacements of all landmarks are stored.

To learn a good description of the shape of the hand, we allow the feature boxes to be large and have large distances to the voxels. The parameters we used can be found in Section 5.1.

4.2.1.2 Testing

When testing an unseen image, all voxels of the image are used. Same as during training of the RRF, input images are resampled to a quarter of the original resolution. Voxels are pushed through the trees of the forests by applying feature tests, thus ending up in leaf nodes. The histograms for all landmark positions, which are stored in the leaf nodes, are summed up using our weighting scheme. Thus, each voxel contributes to the position of all landmarks.

4.2.2 Final Prediction using Local Appearance

The second RRF learns from restricted areas around landmarks, given by the first step, thus improving localization precision. One crucial step is choosing the size of those restricted areas properly for both, training and testing. For a proper localization the predicted position from the first localization step has to be within those restricted areas. Defining that size is a trade-off between reliability and precision of the landmark localization algorithm. Choosing large areas results in a reliable detector, because it is very likely that the first localization step ends up within this area. On the other hand, using larger areas will include anatomical structures further away from the actual landmark position, thus resulting in a worse precision. We use only one single forest for all the landmarks, which makes effective use of feature sharing, since a lot of landmarks share similar local appearance, an idea that was presented in [41].

Since the goal of this step is a good localization precision, training and testing is performed at the full image resolution.

4.2.2.1 Training

When training the second RRF, only voxels close to the landmark positions $\boldsymbol{l}_c$ are used. Those voxels are selected according to the precision and accuracy of the first localization step. In case of a poor precision of the first step, a larger region and in case of a good precision, a smaller region around the landmarks is considered for training the RRF. Therefore, before training the second step, we need to evaluate the performance of the first detection step.

CRRF Performance Estimation For each of the N training images with index $j = 1, .., N$, we apply the first localization step, to get a probability $p^I_{j,c}(\boldsymbol{l})$ of the landmark c being at position $\boldsymbol{l}$, as well as a separate probability $p^I_{j,c,d}(l_d)$ for each dimension $d = \{x, y, z\}$. As can be seen in Fig. 4.1, we observed that these probability distributions have a Gaussian-like shape with the maximum being in general not at the ground-truth landmark position $\boldsymbol{l}_{j,c}$. This probability distribution tells us some information about the precision and accuracy of the first localization step. We can use the variance of $p^I_{j,c}(\boldsymbol{l})$ as well as the deviation of the mean of $p^I_{j,c}(\boldsymbol{l})$ from the ground-truth position to select voxels for training.

For each of the images in the training set, we fit a one dimensional Gaussian function

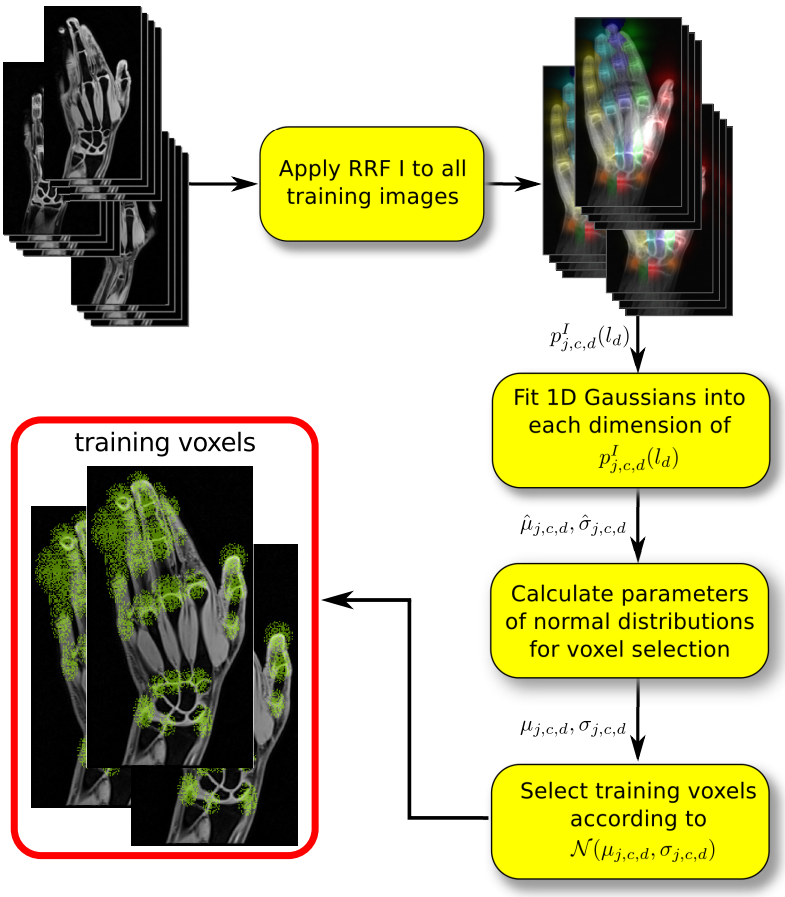


Figure 4.2: Schematic overview of voxel selection for training the second RRF.

$$f(l) = \frac{1}{\hat{\sigma}\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{l-\hat{\mu}}{\hat{\sigma}}\right)^2} \tag{4.1}$$

into the probabilities $p^I_{j,c,d}(l_d)$. The estimated parameters of the function are the standard deviation $\hat{\sigma}$ and the mean $\hat{\mu}$. By fitting that Gaussian function into $p^I_{j,c,d}(l_d)$ for each landmark c, each training image j and each dimension d, we obtain the standard deviations $\hat{\sigma}_{j,c,d}$ and mean values $\hat{\mu}_{j,c,d}$.

Voxel Selection We select voxels for training according to normal distributions. For each training image and each landmark, we use a separate normal distribution with mean values $\mu_{j,c,d}$ and standard deviations $\sigma_{j,c,d}$, thus focusing on local structures. The maximum of the probability distribution should be located at the landmark position. Therefore, we use the x, y and z components of the ground truth landmark position $\boldsymbol{l}_{j,c}$ of image j as

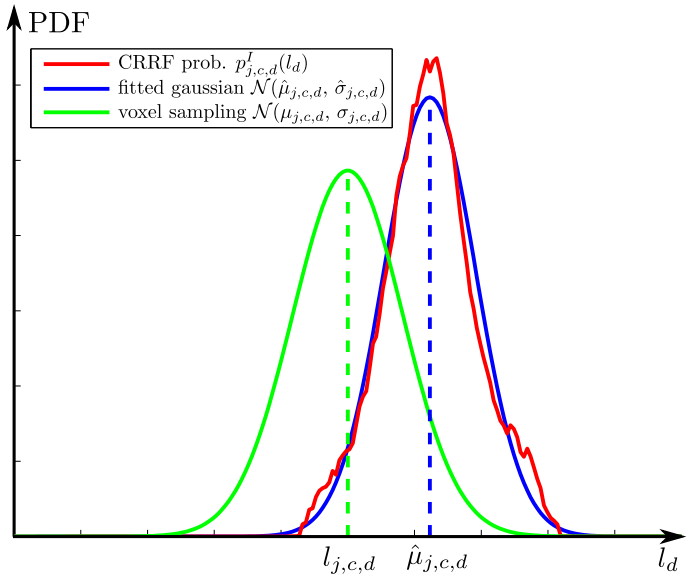


Figure 4.3: Estimating the precision of the first RRF for voxel selection. Example showing PDFs of a landmark position in one dimension (e.g. x dimension). Red Curve shows a probability distribution $p^I_{j,c,d}(l_d)$ estimated using CRRF, blue is the PDF of a Gaussian with parameters $\hat{\mu}_{j,c,d}$, $\hat{\sigma}_{j,c,d}$ fitted into $p^I_{j,c,d}(l_d)$. Green curve shows a normal distribution around the ground-truth landmark position for voxel selection.

mean values of the normal distributions $\hat{\mu}_{j,c,d}$ as following:

$$\mu_{j,c,d} = l_{j,c,d} \tag{4.2}$$

The standard deviations $\sigma_{j,c,d}$ control the size of the region used for training. We derive $\sigma_{j,c,d}$ from the estimated standard deviations $\hat{\sigma}_{j,c,d}$ and the distance between the estimated mean and the ground-truth landmark location according to

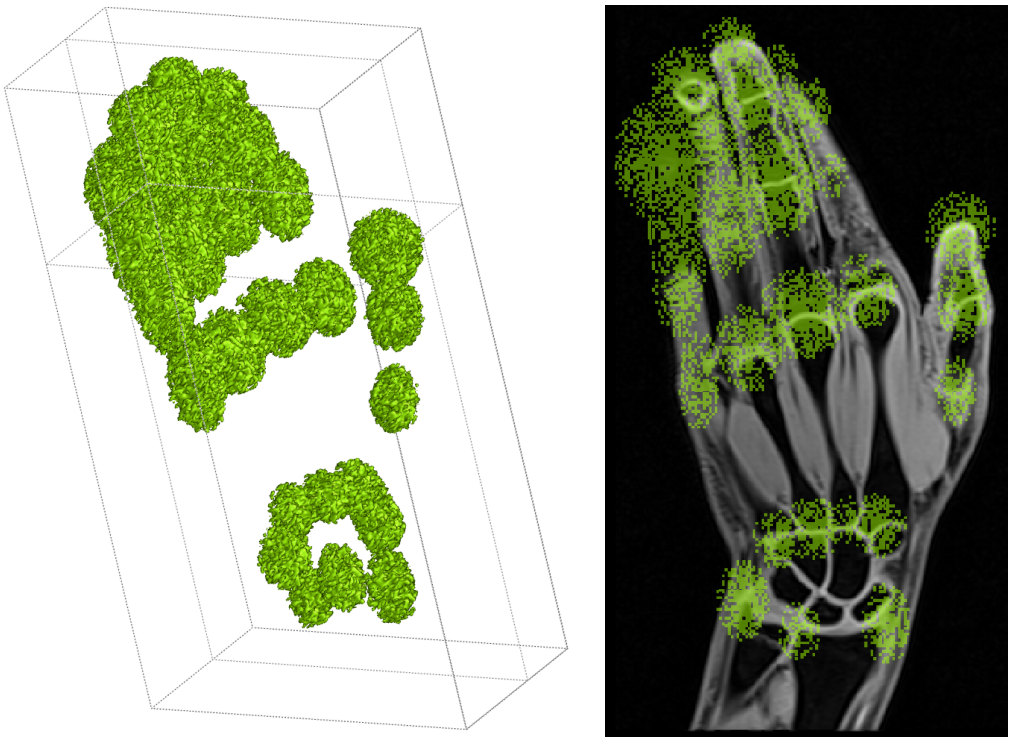
$$\sigma_{j,c,d} = \sqrt{\frac{1}{N}\sum_{j=1}^{N}\hat{\sigma}^2_{j,c,d}} + \sqrt{\frac{1}{N}\sum_{j=1}^{N}(\hat{\mu}_{j,c,d} - l_{j,c,d})^2}. \tag{4.3}$$

Additionally we apply a threshold ρ to the probability of the normal distribution, to eliminate voxels with a low probability.

Each tree is trained with a different random subset of all voxels within the training images. The number of voxels in the subsets is a certain fraction $\lambda = 0.1$ of the overall number of voxels. Experiments showed that the value of λ has no significant impact on the localization accuracy and precision. For each training image, the voxels in the random subsets are drawn without replacement from all voxels within the image, according to

normal distributions for each landmark with the parameters defined above. An example of the selected voxels of a training image can be seen in Fig. 4.4.

All selected voxels for the different landmarks are used for training one single forest. At each node split, feature tests are applied to all voxels arriving at the node, independently of the landmark they are voting for. This makes effective use of feature sharing, since a lot of landmarks share similar local appearance, an idea that was presented in [41]. When going down to deeper levels of the tree, voxels of landmarks with a different local appearance will be passed to different branches of the tree. During the IG calculation and in the voting aggregation in the leaf nodes, voxels are voting only for those landmark positions where $p_c(\boldsymbol{l}_c) \geq \rho$, i.e. they are voting only for close landmarks.



(a) (b)

Figure 4.4: Selected voxels of an exemplary image, used for training one tree of RRF II. **(a)** 3D visualization of the selected voxels **(b)** selected voxels on one single slice of the 3D MRI image.

In order to restrict RRF II to learn from local appearance, we restrict the size and the distance of the feature boxes according to parameters fs_{max} and fr_{max}, respectively.

4.2.2.2 Testing

When testing an unseen image, the first detection step is applied to the input image, to get a coarse probability distribution $p_c^I(\boldsymbol{l})$. Like in the training stage, the standard deviations $\hat{\sigma}_{c,d}$ and the mean values $\hat{\mu}_{c,d}$ of $p_c^I(\boldsymbol{l})$ are estimated. Testing voxels are sampled according to normal distributions with the estimated parameters $\hat{\sigma}_{c,d}$ and $\hat{\mu}_{c,d}$. The sampled voxels are pushed through the trees, thus ending up in leaf nodes. For the summation of the leaf histograms, we extend the weighting function by additionally weighting votes with the coarse probability according to

$$w_c(\boldsymbol{v}) = e^{-||\boldsymbol{d_c}|| \cdot \alpha} \cdot p_c^I(\boldsymbol{l}). \tag{4.4}$$

Thus, votes from regions with a high probability according to the first localization step are emphasized.

Since the localization precision increases when using votes from voxels closer to the actual landmark position, multiple iterations using the same RRF are applied to the input image. We use three iterations, where voxel selection and weighting of the votes is done by using the output of the previous iteration. In each iteration the probability distributions are refined, thus allowing a better voxel selection and weighting in the next iteration.

4.3 Limitations of the Two Step Approach

In the previous sections we have proposed a localization algorithm using two steps. The first step makes a prediction based on the global shape of the hand, while the second step is focusing on the local appearance around the landmarks. Although this approach gives good results, as shown in Chapter 5, one limitation is that the prediction from the first step has to be precise enough to be able to distinguish between the different landmarks, i.e. the result of the first step has to be closer to the actual landmark position than to any neighboring landmark positions. This is because the second step is trained only on local appearance and cannot distinguish between the joints in the hand very well. In cases where CRRF fails to capture the shape of the hand precisely, the localization results may end up in a wrong landmark position, as illustrated in Fig. 4.5. One solution might be to increase the maximum allowed distance fr_{max} of the feature boxes in the second localization step, thus obtaining a description of the area around the landmarks which is better suitable to distinguish between the different joints in the hand. However, this reduces the precision of landmark localization, because information further away from the

actual landmark position is used for position estimation. Choosing this feature range is a trade-off between achieving a good precision and having less outliers. In Sections 4.4 and 4.5 two approaches are presented to overcome this limitation.

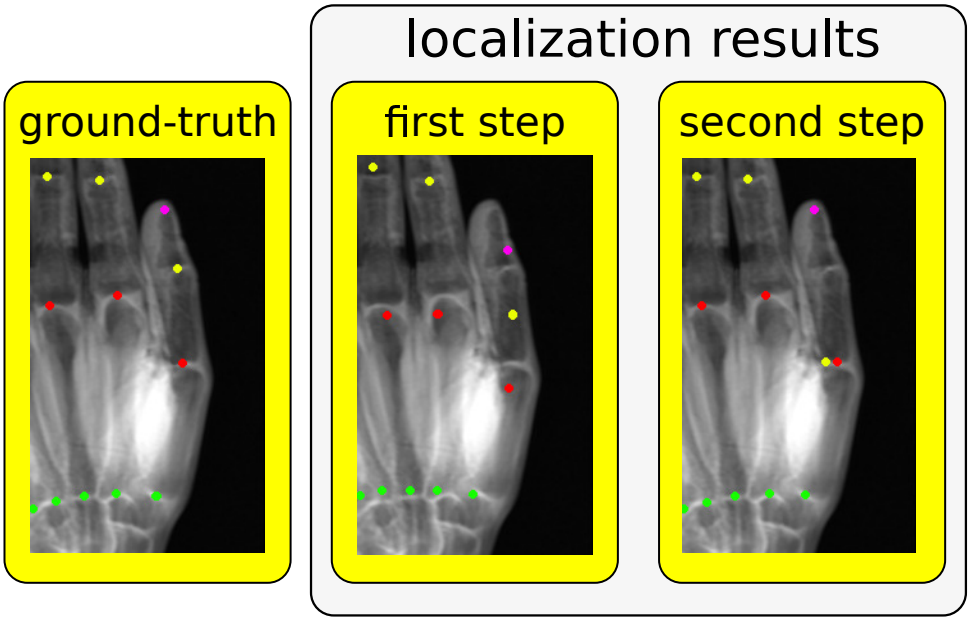


Figure 4.5: Example, which shows how the proposed two step localization approach might wander off to a wrong joint with similar local appearance. The second localization step is not able to distinguish between the red and the yellow joint on the thumb, due to their similar local appearance. In this example the first localization step is very imprecise localizing the yellow landmark located on the thumb, thus allowing the second step to wander off to the wrong joint.

4.4 Setup of Multiple Random Regression Forests

One reason for the limitation described in the previous chapter is the large jump from global shape to local appearance, when going from the first over to the second localization step. To make the transition between global shape and local appearance smoother, more RRFs can be introduced in-between. For those RRFs, the maximum size and distance of the feature boxes steps in between is linearly interpolated between the first and the last localization step. Thus, we combine the advantages of smaller and larger feature ranges and make a smoother transition between first and last localization step. The question is, how many RRFs to use, because a drawback is the increased runtime when adding more localization steps. In Chapter 5 an evaluation showing the effect of varying the number of

forests can be found.

4.5 Auto-Context - An Implicit Model of the Landmark Configuration

During the second localization step, positions of multiple landmarks are estimated independently of each other, i.e. when making a prediction of a landmark position, the position of the other landmarks are not considered. The behavior shown in Fig. 4.5 might be prevented by including a geometric model of the landmark configuration into the RRF framework.

To model geometric relations between the landmarks in the RRF, feature values can be derived from the information about the position of other landmarks. Thus, the RRF predicts landmark location based on the After applying the first RRF to an image, we obtain a probability distribution for each landmark position. This information about all landmark positions can be used in the second RRF by deriving feature values from these probability distributions for the node split functions. In the literature this is often referred to as auto-context [53].

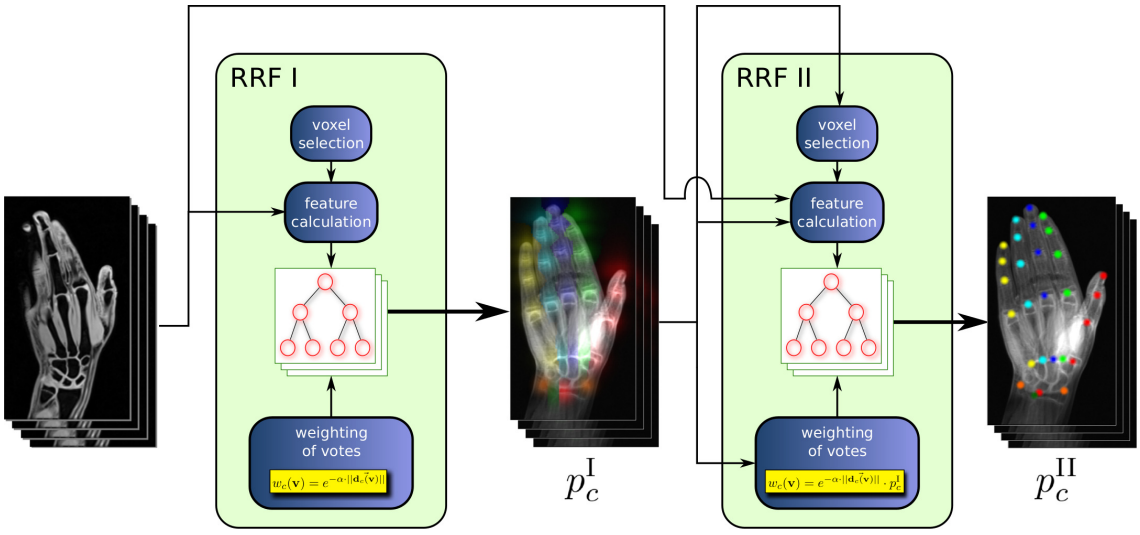


Figure 4.6: Algorithm overview, when using two RRFs with auto-context. RRF II derives auto-context features from the probability distributions p_c^I, given by RRF I.

When introducing auto-context, the RRF uses two different types of features, namely, appearance based features and auto-context features. During training, at each split node

a random decision is made, whether the split function should be based on appearance or auto-context features. The probability for using auto-context features is denoted as p_{ac}.

An auto-context feature response is derived from one of the probability distributions for the landmark positions, obtained from the previous localization step. When generating a random auto-context feature during training, first we randomly select which of the L probabilities, belonging to the different landmarks, is used to calculate an auto-context feature response. For deriving a feature value from the probability distribution, we investigate the following two auto-context feature types:

- **Type I:** Feature values are derived directly from the probability distribution using Haar-like features of the same type as used for modeling the appearance. An advantage of this feature type is, that Haar-like features are able to capture probability distributions of arbitrary shape. However, when using this feature type, the RF has to perform several feature tests to obtain a good description of the probability distribution.

- **Type II:** Feature values are derived from the position of the maximum of the probability distribution. We use the distances from the voxel to the maximum in x, y and z direction as features. The advantage of this feature type is that the RRF is able to capture the relative position of the maximum using very few feature tests. However, this feature type cannot handle probability distributions having multiple modes, thus having no well defined position of the maximum.

4.6 Conclusions

In this chapter we have shown a novel hand bone landmark detection approach based on several localization steps, each using an RRF. First of all, we have presented the concept based on two steps, where the first localization step makes a prediction based on global shape, thus handling the presence of repeating structures within the hand, while the second step predicts solely based on local appearance, thus achieving a good precision.

To overcome the limitation of landmark locations wandering off to wrong neighboring landmarks with a similar local appearance during the second localization step we have presented two approaches. In the first approach we make a smoother transition when going from global shape to local appearance by introducing more localization steps, between the first and the last step. The second approach includes an implicit model of the landmark configuration by adding auto-context features.

Chapter 5

Experiments and Results

Contents

In this chapter we compare the performance of a specific setup of our landmark localization algorithm using two localization steps with other methods from related work in Section 5.1. To get more insights into the behavior of the proposed algorithm, we provide an evaluation of some selected parameters in Section 5.2. To investigate further possible improvements regarding outlier detections, we show results when increasing the number of forests in Section 5.3 and when including an implicit geometric model of the landmark configuration using auto-context in Section 5.4. Results from all experiments are discussed in Section 5.5.

5.1 Comparison of Two-Step Landmark Localization with Related Work

We evaluated our proposed two detection steps CRRF and Gradually Improving Random Regression Forest (GIRRF) and compared it to the TDPR [20] method. Further, to show the benefit of the introduced weighting scheme, we made an experiment on the first de-

tection step with and without the use of the weighting scheme, denoted as CRRF and Standard Random Regression Forest (SRRF), respectively. Note that SRRF resembles an implementation of the method in [11], but focusing on landmark localization instead of bounding boxes, since we aim for accurate localization independent of bounding box orientation.

5.1.1 Experimental Setup

We evaluated all algorithms in a cross-validation setup with $N = 5$ rounds. In each round we randomly split the 60 available input images into 43 training and 17 testing images. The measure we used for evaluating the performance is the Euclidean distance between the ground truth and the estimated landmark position. The mean distance shows localization accuracy, while its standard deviation describes precision of localization.

The setup of the different evaluated algorithms is as following:

TDPR: We evaluated the algorithm with the parameters proposed by the authors of this work in [20].

SRRF: We trained SRRF by building $T = 8$ trees with maximum depth $D = 14$, where for each node split 100 candidate features and 10 candidate thresholds were generated. The size fs_{max} and range fr_{max} of the random feature cuboids was restricted to 50mm and 25mm in each dimension, respectively.

CRRF: The parameters of CRRF were set equally as for SRRF, but we used our weighting function $w_c(\boldsymbol{v})$, with α set to $1/mm$.

GIRRF For the first localization step of GIRRF, we used the same parameters as for CRRF. The threshold for selecting the voxels for training was chosen as $\tau = 0.4 \cdot max\{p(\boldsymbol{l}_c)\}$. During training, we built $T = 8$ trees with maximum depth $D = 15$. At each node split 20 random candidate features and 10 candidate thresholds are generated. The maximum size in each dimension and distance of the feature cuboids is 7mm.

5.1.2 Results

Figure 5.3 shows a qualitative visualization of the cross-validation results of the evaluated algorithms. For all landmarks we achieve a localization error ($\pm$ standard deviation) of 1.44$\pm$1.51mm. In x, y and z direction we achieve a mean error of $0.68mm$, $0.57mm$ and

Table 5.1: Comparison of localization errors from cross validation on hand bone landmarks, radius/ulna (R/U), carpometacarpal (CMP), metacarpal (MCP), distal and proximal interphalangeal joints (DIP,PIP), finger tips (FT).

Method	Localization Error [mm]: Mean ± Std.						
	R/U	CMC	MCP	PIP	DIP	FT	overall
TDPR [20]	2.8±2.8	2.0±1.1	2.0±2.4	2.0±3.1	1.8±3.9	2.7±4.2	2.2±3.1
SRRF [11]	7.9±5.1	7.4±5.1	6.7±3.1	6.5±3.2	6.5±3.3	8.1±5.3	7.2±4.4
CRRF	4.8±2.4	3.7±1.5	4.0±2.1	4.1±2.0	4.5±2.5	5.5±3.1	4.4±2.4
GIRRF	**1.8±1.3**	**1.5±0.7**	**1.2±0.6**	**1.3±2.2**	**1.3±2.4**	**1.5±0.8**	**1.4±1.5**

$0.84mm$, respectively. A more detailed quantitative comparison of the evaluated methods can be found in Table 5.1, as well as in Fig. 5.1, which shows a cumulative distribution of the errors. From the $5 \cdot 17 \cdot 28 = 2380$ detected landmark positions, only six outliers (0.25%) had a localization error larger than 10mm. One outlier was on the radius bone, the others occurred on the distal interphalangeal (DIP) and proximal interphalangeal (PIP) joints. The TDPR approach showed 35 (1.5%) outliers.

Runtime of our C++ algorithm, which was implemented on top of the open-source Sherwood library from Microsoft Research*, is about 400s per volume on an 8-core Intel(R) Core(TM) i7 CPU. Parallelized forest training for one round of cross validation takes 24 hours on the same PC. Runtimes for training and testing of TDPR are around 2 hours and 10s, respectively.

Memory required for training our two localization steps is around 10GB. Saving the random forests on the hard disk consumes around 2GB and 4GB for the first and second localization step, respectively. Most of the memory is used for storing leaf node histograms.

5.1.3 Discussion

As can be seen in Table 5.1 and Fig. 5.3 and 5.1, our proposed algorithm achieves superior overall and individual localization accuracy in terms of mean error and standard deviation among the compared algorithms. When comparing CRRF to SRRF, the clear improvement when introducing the weighting function can be seen.

A detailed analysis of the outliers shows that for TDPR and GIRRF they occur in hands with a finger pose that is not covered in the training set during cross validation,

*http://research.microsoft.com/en-us/downloads/52d5b9c3-a638-42a1-94a5-d549e2251728/ Accessed December 2015.

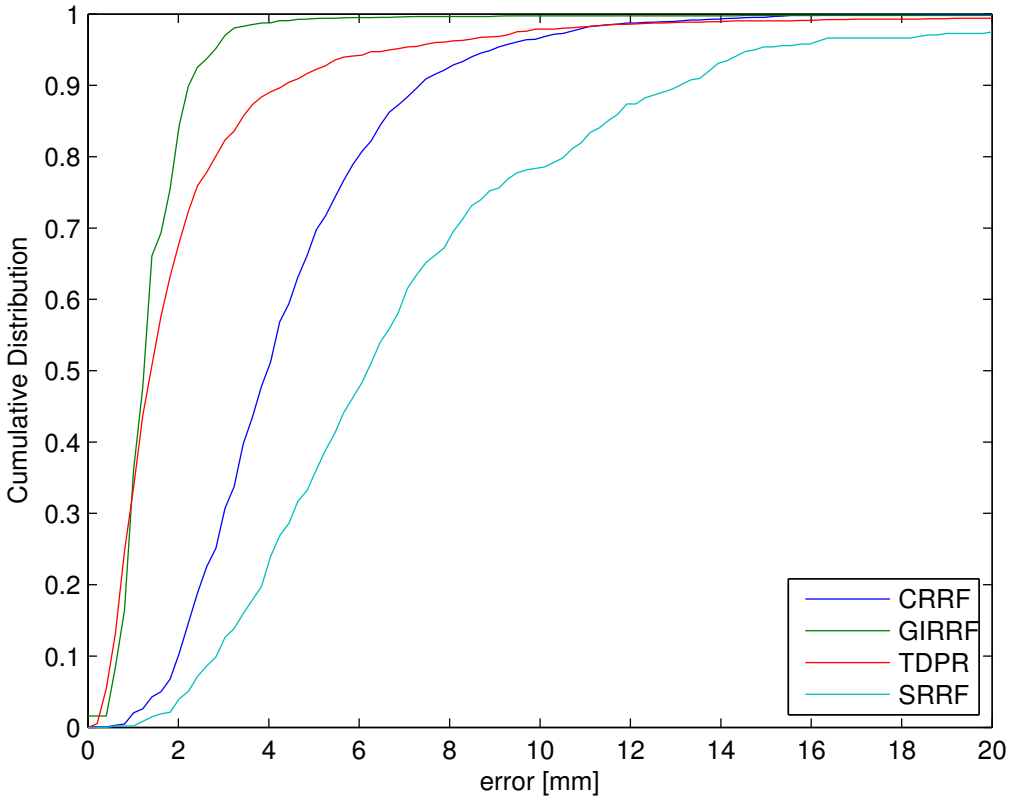


Figure 5.1: Results from cross-validation of comparison with related work. Plot shows the cumulative distribution of the localization errors.

however, more often these situations occur in the TDPR approach. In case something went wrong during the detection in the TDPR approach, almost all landmarks located on the phalanges were detected wrong in the same image. TDPR seems to be even more constrained by the variability in the training data through the explicit use of a PCA-based SSM. An experiment showed us that adding this SSM to GIRRF does not fix the remaining outliers, but rather introduces new errors on already well detected landmarks. In GIRRF, there were at most three outliers in one single image, compared to 12 for TDPR. As can be seen in Fig. 5.1, TDPR slightly outperforms GIRRF only for errors smaller than 1mm. This is because TDPR results are in sub-pixel resolution and GIRRF results only in discrete pixel locations. However, GIRRF may achieve sub-pixel resolution by using for example mean shift to estimate the final landmark positions from the probabilities.

All evaluated algorithms achieved the worst mean error on radius and ulna bone, which can be explained by the large anatomical variation especially at the ulna bone and because the landmarks had to be chosen at locations, that were hard to define in manual

annotation due to lack of proper anatomical structures near the bone. On our dataset CRRF is able to achieve a much better accuracy when including the weighting function according to (3.10), compared to a weighting equal to one as proposed in [11]. The reason for this improvement is, that local information around each landmark provides a more accurate estimation, since there is a large pose variation of the fingers in our database. This fact is exactly what has driven the development of our proposed approach. Since automatic BAE relies on a very accurate bone localization, we find that we can improve by using GIRRF compared to related work, due to its capability to extract age related features to learn an age regression model based on located bone landmarks. A drawback of our approach is higher runtime compared to e.g. TDPR. Our major bottleneck is leaf histogram summation, which could be sped up by a GPU implementation or by using an alternative voting scheme, as evaluated in Section 5.2.1.2.

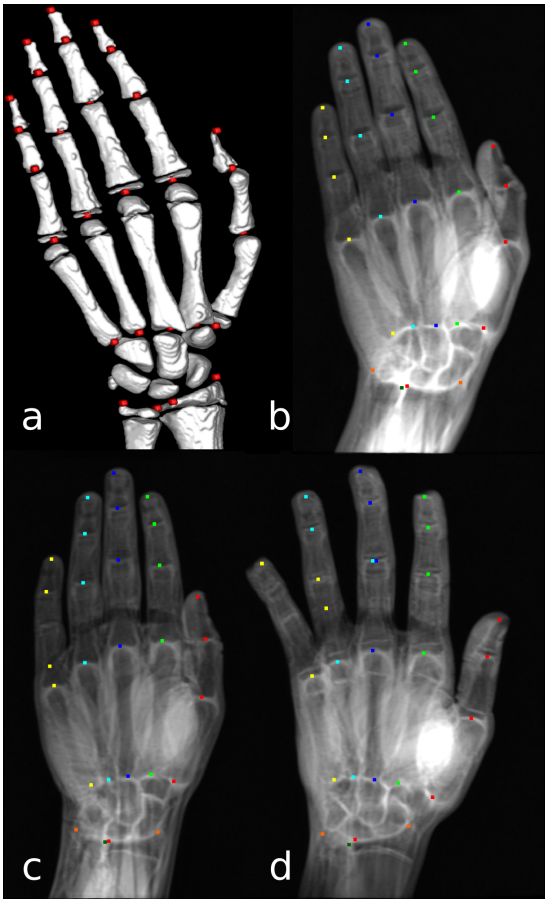


Figure 5.2: Qualitative results of GIRRF shown on a bone skeleton (a) and on 2D projections of the MRI volumes (b-d) showing usual results (a,b) and outlier detections (c,d).

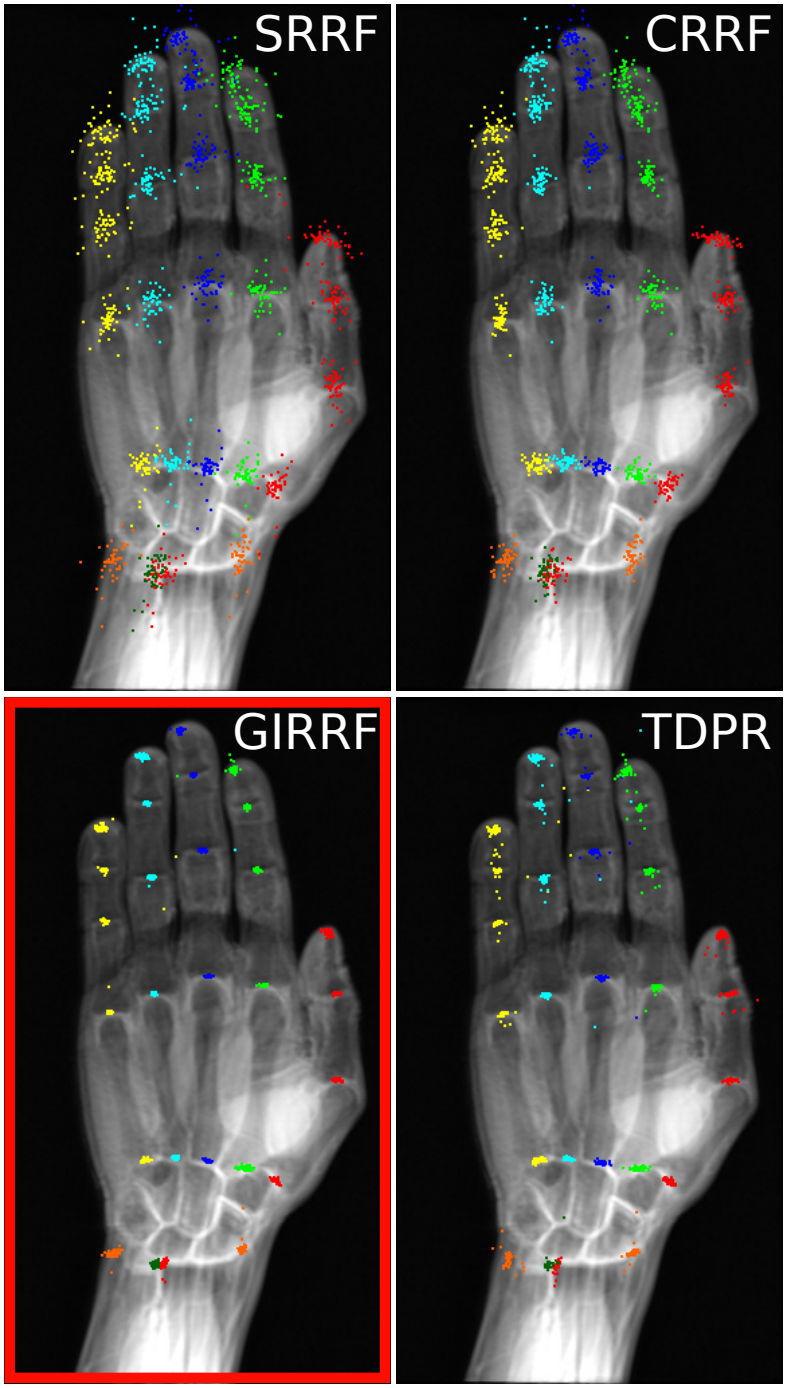


Figure 5.3: Comparison of results from cross-validation with related work within a projection of one 3D MRI image. The 3D MRI image was projected to 2D by summing up all intensity values along the z dimension. For each cross-validation test image result and each landmark, we calculated the error vector. We visualized each test result by adding the error vector to the ground truth landmark positions of the presented image and putting a point at this position. Each point within the presented image corresponds to a localization result of one single image.

5.2 Effect of Random Regression Forest Parameters

To get more insights into the behavior of the proposed landmark localization algorithm GIRRF and determine important parameters, we will show the influence of some parameters of the RRF on the localization accuracy.

Instead of the comprehensive evaluation using multiple rounds of cross validation when comparing GIRRF to related work in Section 5.1, we performed the following experiments on one random split of our data into 43 training and 17 testing data.

5.2.1 Experiments and Results

5.2.1.1 Weighting Scheme

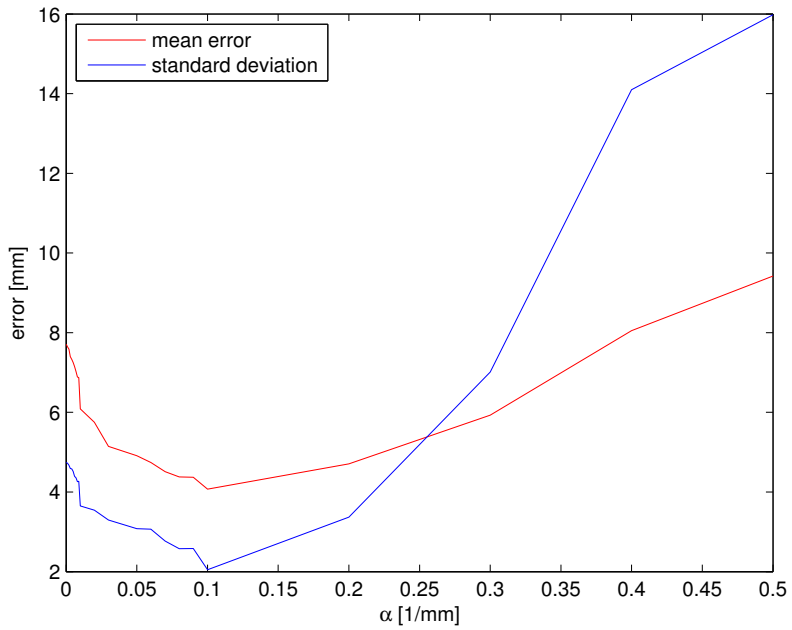
One of our main contributions is the introduced weighting scheme, which consists of an exponential function with the parameter α allowing to adjust the steepness of the weighting function. In other words α controls the range of votes that should be used for the position estimation. Setting α to zero resembles an implementation of SRRF, where all votes are weighted equally. The experiment in Section 5.1 showed improvements when using the weighting function with $\alpha = 0.1/mm$.

The question is, if we can get even more improvement by further increasing α. Therefore, we evaluated CRRF using different values for α. As can be seen in Fig. 5.4, the mean localization error gets significantly smaller when increasing α until a value of around 0.1. For $\alpha > 0.1$ the mean error starts to increase again, while the standard deviation increases even more. We observed, that the probability distributions are getting very noisy for large values for α, as can be seen from the qualitative results in Fig. 5.4(b).

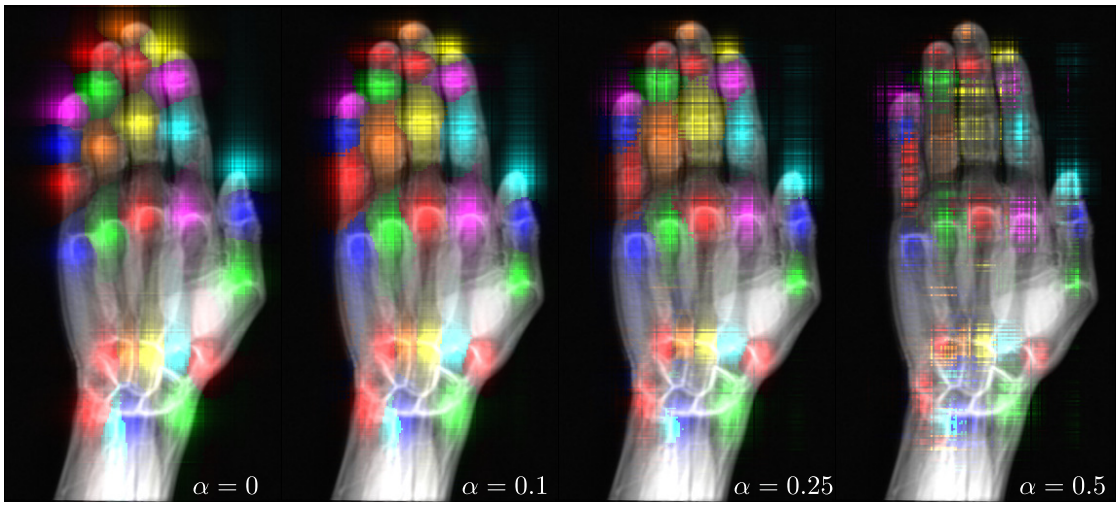
5.2.1.2 Voting Scheme

In Section 5.1 we presented the results of CRRF using the histogram voting scheme to allow a fair comparison with SRRF. Since the histograms are responsible for high memory requirements and histogram summation is the major bottleneck of the runtime, we evaluate the accuracy of two alternative voting schemes, which are faster and have lower memory requirements. We made an experiment on CRRF comparing the following three different voting schemes:

- **histogram voting**: Each voxel is voting with a histogram for the landmark position, which is same strategy as used when obtaining results in Section 5.1.



(a) Error plot over different values for α



(b) Example images with different values for α.

Figure 5.4: Experiments on the weighting scheme of CRRF using different values for α.

Table 5.2: Results in terms of mean error and standard deviation using different voting schemes for CRRF.

voting scheme	error [mm] mean	std.
histogram voting	**3.93**	**2.17**
single voting (max)	4.35	2.58
single voting (mean)	4.94	3.05

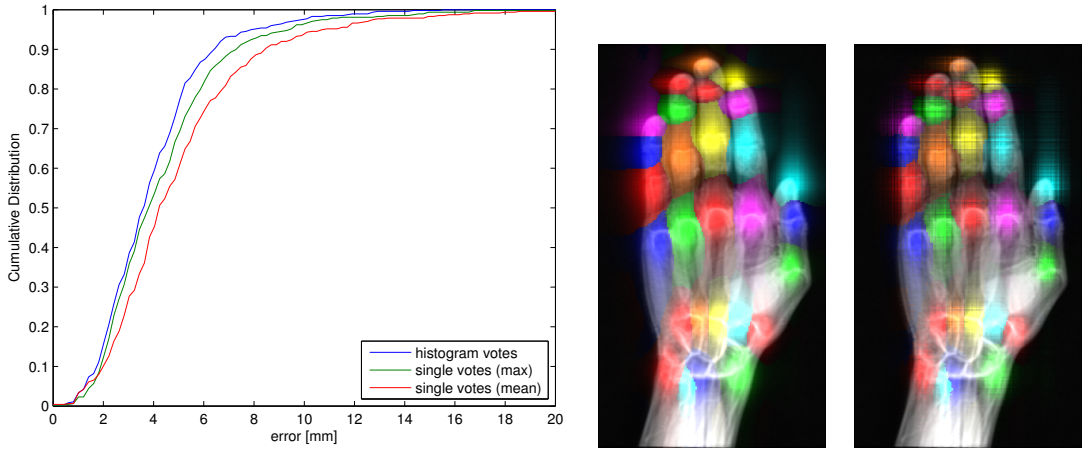
- **single voting:** Each voxel is voting for one single position, e.g. by taking the position of the mean or max in each histogram for x, y and z dimension. This approach is faster and has lower memory requirements compared to histogram voting. We distinguish between 2 different versions of singe voting using a
  - **single voting (max):** single vote at the position of the max of each histogram for x, y, z as proposed in [8].
  - **single voting (mean):** single vote at the position of the mean of each histogram for x, y, z.

Results of this comparison can be seen in Figure 5.5, which shows a cumulative distribution of the errors as well as the resulting probability distribution of one randomly chosen image. Quantitative results are shown in Table 5.2. From the cumulative distribution one can observe, that the voting scheme using histograms seems to perform best at the cost of a higher runtime and memory requirements when testing an image. The comparison between different single voting schemes showed, that the scheme using the maximum is better, compared using the mean of the histograms.

5.2.1.3 Depth and Number of Trees in the Forest

Two important parameters of the RRF are the maximum depth of trees and the number of trees in the random forest. These parameters are rather application specific. To show suitable parameters for our application, we evaluated the performance of CRRF and varied the number of trees and the maximum depth in the forest. When evaluating the number of trees, we allowed a maximum depth of 14 and when evaluating the maximum tree depth, we used eight trees.

Results of the two experiments can be seen in Fig. 5.6. The localization error gets smaller when increasing the number of trees, but after a few trees no significant improvement can be observed. The maximum depth of the trees has more influence on the



(a) Cumulative Error Distribution (b) histogram v. (c) single v.(max)

Figure 5.5: Comparison between the histogram and two different single voting schemes mean and max. The cumulative error distribution shows, for each error e on the x-axis the percentage of all localization results with an error $< e$. Images (b) and (c) are showing the resulting probability distributions of CRRF evaluated on an example image.

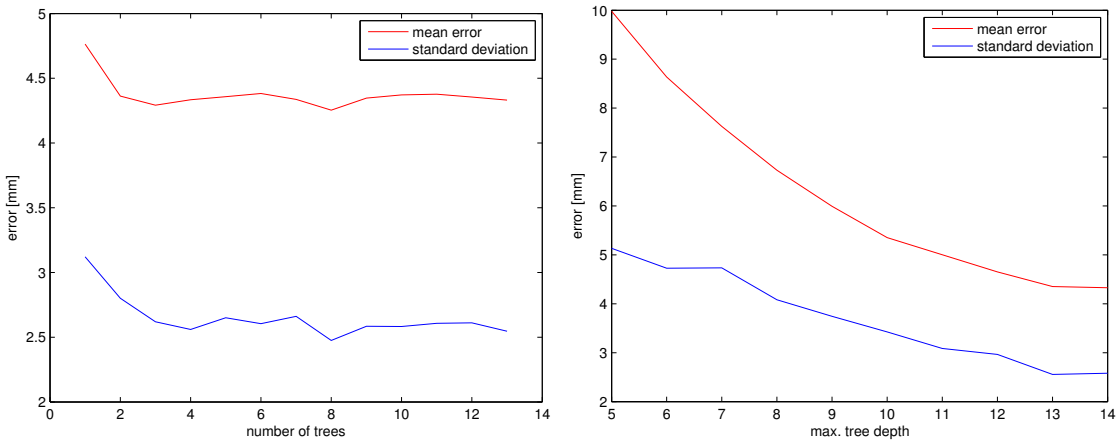


Figure 5.6: Influence of the random forest parameters number of trees and tree depth on the localization error.

localization error. Until the maximum depth used in our experiments, the error decreases significantly.

5.2.2 Discussion

Experiments on the weighting function showed, that there is an optimal value for α, controlling the steepness of the weighting function as can be seen in Fig. 5.4. When increasing α further, the probability distributions are getting noisy. This leads to a higher uncertainty in the final decision on the landmark location when taking the maximum of probability as can be seen from the increasing standard deviation for $\alpha > 0.1$.

We could show, that the number of trees in the forest is a rather uncritical parameter. While the accuracy increases almost monotonously with the number of trees, no significant improvement can be observed after a few trees. In the work presented in [11] only 4 trees are used for the RRF. A speculation about the low number of trees required is, that during this localization setup averaging is already performed when accumulating the votes from different voxels. Therefore, increasing the number of trees gives no significant improvements in terms of mean and standard deviation.

As can be seen in Fig. 5.6, the maximum tree depth can be chosen very large without observing overfitting. An extrapolation of the error curve suggests, that there might be more improvement possible, when further increasing the tree depth. However, the memory requirements scale exponentially with the tree depth, as the number of nodes in a full binary tree with depth d is $2^d - 1$. Especially the memory required for saving the leaf node histograms, prevents from further increasing the depth.

The experiment on alternative voting schemes shows, that similar localization results can be achieved using single voting (max) while requiring only a fraction of memory, because each histogram in the leaf node can be replaced by one single value. This voting scheme would allow to train deeper trees, while having the same memory requirements. Single votes at the position of the mean of the histograms gives worse results compared to single votes at the position of the maximum. The reason for that might be, because the mean of a histogram is very sensitive to outliers. Since the probability distributions look noisier for the single voting case, applying mean shift on the probability distribution might improve results compared to taking the maximum as the final landmark position.

5.3 Multiple Random Regression Forests

The results of the previous evaluations on GIRRF were obtained using two RRFs, where the first forest makes a prediction based on global shape and the second based on local appearance. Although, this two-step approach results in very few outliers, we are still

interested in further increasing the robustness, since wrongly detected bones might have an impact on the final age estimation. In Section 4.4 we claimed that by adding more RRFs in-between the first and last localization step results in a smoother transition when going from global shape to local appearance, thus reducing the number of outliers. To show the influence of the number of forests on the localization accuracy, we performed some experiments, described in the following section.

5.3.1 Experimental Setup

We evaluated our algorithm by randomly splitting data into 43 training and 17 testing images. We varied the number of forests nf, starting with two, which is the proposed setup of GIRRF, up to four forests. We set the parameters of the first and last localization step as proposed in Section 5.1. For the forests in-between we selected a feature range and size by linear interpolation between the setting of the first and last step.

Furthermore, we evaluated the algorithm with two and four localization steps in a cross-validation setup with $N = 5$ rounds, where we randomly split the 60 available input images in each round into 43 training and 17 testing images.

5.3.2 Results and Discussion

As can be seen in Fig. 5.7, increasing the number of forests in our algorithm improves the localization accuracy in terms of mean and standard deviation. The improvement when using four instead of two localization steps is about 20 percent for mean and 57 percent for standard deviation.

Increasing the number of forest improves mainly on the standard deviation, which tells us that the improvement is mainly related to localization results with a larger error, which can also be seen from the cumulative error distribution in Fig. 5.7. However, the cost of this improvement is a higher runtime, which increases approximately linear with the number of forests.

From the results of the cross-validation, as can be seen in Fig. 5.7c, we observed that the difference between using two and four localization steps is very small and that no significant improvement can be achieved.

5.4 Auto-Context

To improve on the few remaining outliers we investigated the possibility to use auto-context features to include an implicit geometric model of the landmark configuration in the RRF framework, as described in Section 4.5.

5.4.1 Experimental Setup

Since, this evaluation involved computational expensively training of many different forests, we performed all evaluation on 2D hand images. We obtained the 2D images by projecting the 3D hand MRI images along the z-dimension. We randomly split our data into 15 training and 45 testing images. We trained our algorithm with $nf = 3$ forests, each with 8 trees and with a depth of 15. We made experiments using two different auto-context feature types:

- **Type I:** Feature values are derived directly from the probability distribution using Haar-like features.
- **Type II:** The distances from the voxel to the maximum of the probability distribution in x and y direction are used as features.

In each node split we randomly decided with a probability p_{ac}, whether to use auto-context features for the node-split function. We performed evaluations on different values for the probability p_{ac}, while including zero allows a comparison the results when only appearance based features are used.

5.4.2 Results and Discussion

As can be seen in Fig. 5.8, both auto-context feature types lead to similar improvements on standard deviation, while no significant improvement on the mean error can be observed.

As can be seen from the cumulative error distributions, auto-context improves only on the localization results with an error larger than around 3mm. This can be explained by the fact, that geometric relationships between the landmarks supports only coarse localization, while the exact landmark location can only be derived from local appearance.

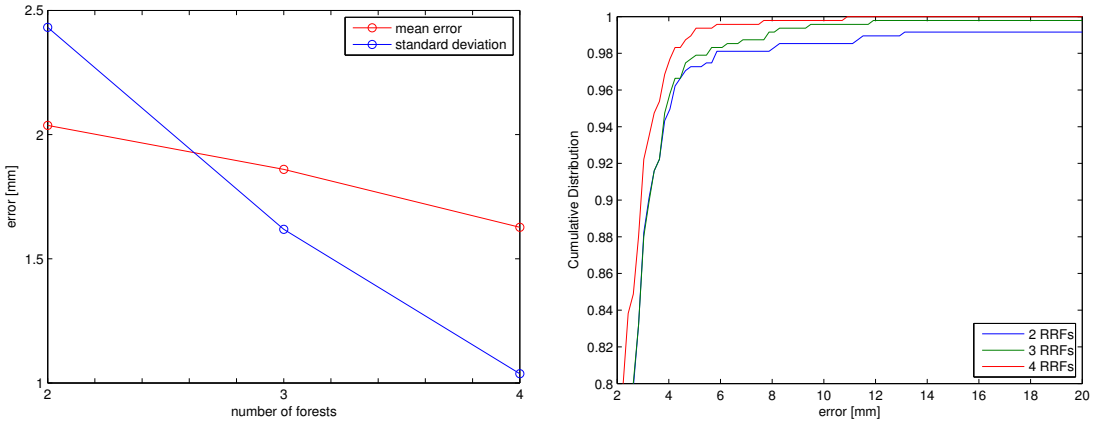
5.5 Discussion

In this chapter we provided an evaluation of our algorithm and compared one specific configuration using two localization steps to the SRRF and TDPR method in a cross-

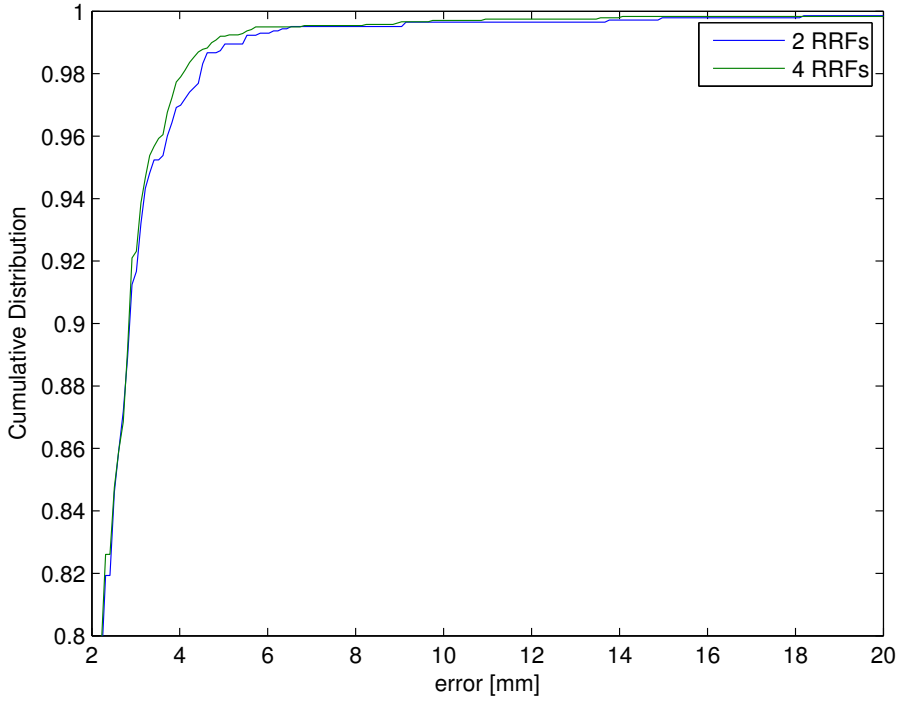
validation setup. As can be seen from the results, our proposed algorithm achieves superior overall and individual localization accuracy in terms of mean error and standard deviation among the compared algorithms.

A detailed analysis of the outliers shows that they occur in hands with a finger pose that is not covered in the training set during cross validation, however, more often these situations occur in the SRRF and TDPR method.

To improve on the remaining outliers and increase the robustness of our algorithm, we investigated the possibility to increase the number of localization steps, to achieve a smoother transition when going from global shape to local appearance. Further, we made experiments to include auto-context features, thus implicitly modeling geometric relationships between landmarks. Auto-context has no significant impact on the mean error, while very minor improvements on the standard deviation can be achieved. However, no significant improvements on the outlier localizations can be achieved by introducing more localization steps and auto-context. This may be due to our limited training set of 43 images, covering not all possible poses, thus the learned geometric model cannot improve on all remaining outliers.

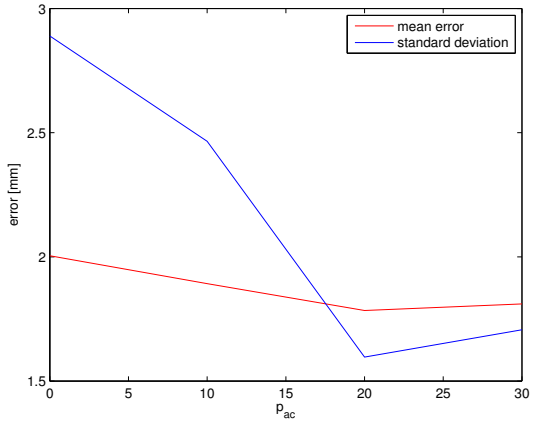


(a) (b)

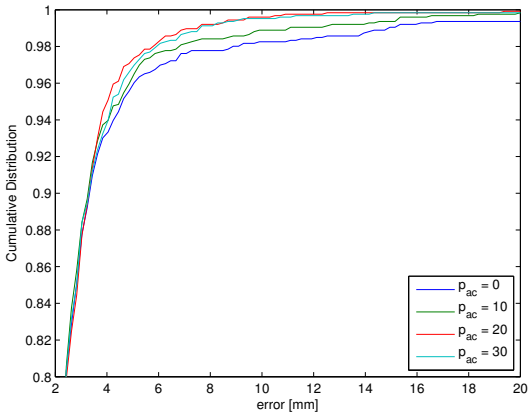


(c)

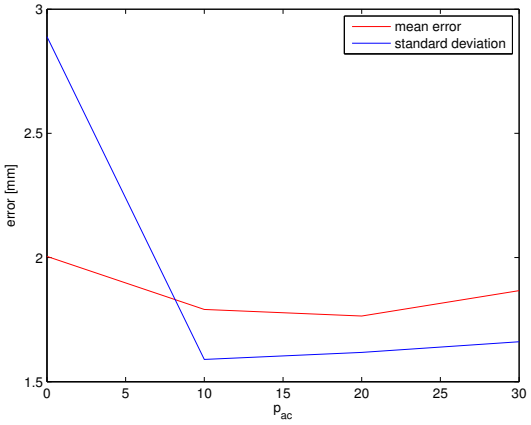
Figure 5.7: Influence of the number of forests on the localization errors. **(a)** shows the mean error and standard deviation and **(b)** the cumulative error distribution for different number of forests. **(c)** shows results from cross-validation of comparison between using two and four RRF localization steps.



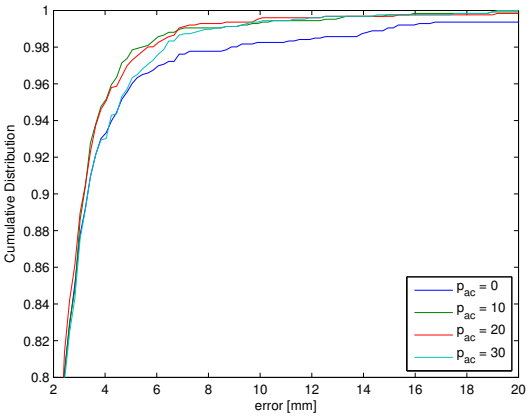
(a) Type I auto-context features



(b) Type I auto-context features



(c) Type II auto-context features



(d) Type II auto-context features

Figure 5.8: Experiments using two different types of auto-context features and various probabilities p_{ac} for auto-context features. In the case where $p_{ac} = 0$, no auto-context features are used.

Chapter 6

Conclusions and Outlook

In this thesis we showed a novel approach for localizing anatomical landmarks from left hand MRI images. Localization of anatomical structures is an important procedure in many medical applications. However, our interest is related to an ongoing research study at the LBI-CFI in Graz, which investigates the possibility of replacing X-ray imaging with radiation-free MRI for the purpose of age estimation of young unaccompanied asylum seekers without identification document. To automate age estimation, localization of the anatomical structures, where age relevant features can be found, is a crucial first step.

Our localization strategy is based upon the idea that the location of anatomical structures is constrained by all of their surrounding structures. Global information from all over the image supports coarse localization and distinguishing between the repeating structures, while closer structures provide the information for a precise landmark localization.

We realized this concept using RRFs at multiple scales. The first RRF predicts coarse landmark locations from the global shape of the hand using long range context-rich features, thus distinguishing between repeating structures within the hand. Subsequent localization steps were locally restricted by a novel weighting scheme according to the coarse localization provided by the previous RRFs. We regard this combination of multiple RRF steps together with our weighting scheme, that lets local structures have a higher contribution to the estimation of landmark positions, as our main contribution.

We showed that our approach is able to clearly outperform other methods regarding localization accuracy on our hand MRI data, achieving a mean localization error of 1.4 ± 1.5mm with only 0.25% outliers with an error greater than 10mm.

Parts of this work were presented at the MICCAI conference in 2014 [23]. Furthermore, first experiments, as presented in [47], demonstrated that the proposed algorithm is able

to initialize an automatic skeletal bone age estimation algorithm.

To further improve on the remaining outliers, we investigated including auto-context as an implicit geometric model of the landmark configuration, however, even though small improvements could be achieved, the remaining outliers could not be eliminated. We further plan to investigate including more sophisticated shape representations, such as ShapeForest [48], into our framework and we plan to investigate our proposed GIRRF approach on other data sets as well, to show its generalization capabilities.

Appendix A

List of Acronyms

ASM	Active Shape Model
BAE	Bone Age Estimation
BB	Bounding Box
CLM	Constrained Local Model
CRRF	Coarse Random Regression Forest
CT	Computed Tomography
GIRRF	Gradually Improving Random Regression Forest
GP	Greulich-Pyle
IG	Information Gain
MRF	Markov Random Field
MRI	Magnetic Resonance Imaging
MSL	Marginal Space Learning
PBT	Probabilistic Boosting Tree
PCA	Principal Component Analysis
PDF	Probability Density Function
RF	Random Forest
RRF	Random Regression Forest
SRRF	Standard Random Regression Forest
SSM	Statistical Shape Model
TDPR	Top Down Image Patch Regression
TW	Tanner-Whitehouse

Appendix B

Publications and Presentations

1. T. Ebner, D. Stern, R. Donner, H. Bischof, M. Urschler. Towards Automatic Bone Age Estimation from MRI: Localization of 3D Anatomical Landmarks. In: Proc Medical Image Computing and Computer Assisted Intervention (MICCAI) 2014; Boston, Springer LNCS 8674, p. 220-227. **(Oral Presentation)**

2. D. Stern, T. Ebner, H. Bischof, S. Grassegger, T. Ehammer, M. Urschler. Fully automatic bone age estimation from left hand MR images. In: Proc Medical Image Computing and Computer Assisted Intervention (MICCAI) 2014; Boston, Springer LNCS 8674, p. 421-428.

3. D. Stern, T. Ebner, H. Bischof, M. Urschler. Determination of legal majority age from 3D magnetic resonance images of the radius bone. In: Proc International Symposium Biomedical Imaging (ISBI), Beijing, China (May 2014). (Oral Presentation)

4. D. Stern, T. Ebner, E. Scheurer, M. Urschler. Legal Majority Age Determination from MR Images of the Radius Bone. In 22nd Annual Meeting ISMRM, May 2014, Milan, Italy.

5. K. Hammernik, T. Ebner, D. Stern, M. Urschler, T. Pock. Vertebrae Segmentation in 3D CT Images based on a Variational Framework. In: Proc MICCAI Workshop Computational Methods and Clinical Applications in Spine Imaging (CSI) 2014; Boston. **Honourable Mention Award**

Bibliography

[1] Amit, Y. and Geman, D. (1994). Randomized inquiries about shape: An application to handwritten digit recognition. Technical report, DTIC Document.

[2] Barbu, A., Suehling, M., Xu, X., Liu, D., Zhou, S. K., and Comaniciu, D. (2010). Automatic detection and segmentation of axillary lymph nodes. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2010*, pages 28–36. Springer.

[3] Bassed, R. B. (2012). Advances in forensic age estimation. *Forensic Science Medicine and Pathology*, 8(2):194–196.

[4] Bishop, C. M. et al. (2006). *Pattern recognition and machine learning*, volume 1. springer New York.

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

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

[7] Chen, G. T., Kessler, M., and Pitluck, S. (1985). Structure transfer between sets of three dimensional medical imaging data. *Computer Graphics*, 24:172–175.

[8] Cootes, T., Ionita, M., Lindner, C., and Sauer, P. (2012). Robust and accurate shape model fitting using random forest regression voting. In Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., and Schmid, C., editors, *Computer Vision, ECCV 2012*, volume 7578 of *Lecture Notes in Computer Science*, pages 278–291. Springer Berlin Heidelberg.

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

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

[11] Criminisi, A., Robertson, D., Konukoglu, E., Shotton, J., Pathak, S., White, S., and Siddiqui, K. (2013). Regression forests for efficient anatomy detection and localization in computed tomography scans. *Medical Image Analysis*, 17(8):1293 – 1303.

[12] Criminisi, A. and Shotton, J. (2013). *Decision Forests for Computer Vision and Medical Image Analysis*. Springer.

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

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

[15] Cristinacce, D. and Cootes, T. (2008). Automatic feature localisation with constrained local models. *Pattern Recognition*, 41(10):3054–3067.

[16] Cunha, E., Baccino, E., Martrille, L., Ramsthaler, F., Prieto, J., Schuliar, Y., Lynnerup, N., and Cattaneo, C. (2009). The problem of aging human remains and living individuals: A review. *Forensic Science International*, 193:1–13.

[17] Doi, K. (2007). Computer-aided diagnosis in medical imaging: historical review, current status and future potential. *Computerized Medical Imaging and Graphics*, 31(4-5):198–211.

[18] Dollar, P., Tu, Z., and Belongie, S. (2006). Supervised learning of edges and object boundaries. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1964–1971. IEEE.

[19] Donner, R., Langs, G., Mičušik, B., and Bischof, H. (2010). Generalized sparse MRF appearance models. *Image and Vision Computing*, 28(6):1031–1038.

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

[21] Donner, R., Menze, B. H., Bischof, H., and Langs, G. (2013b). Global localization of 3D anatomical structures by pre-filtered hough forests and discrete optimization. *Medical Image Analysis*, 17(8):1304–1314.

[22] Dvorak, J., George, J., Junge, A., and Hodler, J. (2007). Age determination by magnetic resonance imaging of the wrist in adolescent male football players. *British Journal of Sports Medicine*, 41(1):45–52.

[23] Ebner, T., Stern, D., Donner, R., Bischof, H., and Urschler, M. (2014). Towards automatic bone age estimation from MRI: Localization of 3D anatomical landmarks. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014*, pages 421–428. Springer.

[24] Fenchel, M., Thesen, S., and Schilling, A. (2008). Automatic labeling of anatomical structures in mr fastview images using a statistical atlas. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2008*, pages 576–584. Springer.

[25] Gall, J., Yao, A., Razavi, N., Van Gool, L., and Lempitsky, V. (2011). Hough forests for object detection, tracking, and action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(11):2188–2202.

[26] Glocker, B., Feulner, J., Criminisi, A., Haynor, D. R., and Konukoglu, E. (2012). Automatic localization and identification of vertebrae in arbitrary field-of-view CT scans. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2012*, pages 590–598. Springer.

[27] Greulich, W. W. and Pyle, S. I. (1959). *Radiographic atlas of skeletal development of the hand and wrist*. Stanford University Press, Stanford, CA, 2nd edition.

[28] Hajnal, J. and Hill, D. (2001). *Medical Image Registration*. Biomedical Engineering. CRC Press.

[29] Han, D., Gao, Y., Wu, G., Yap, P.-T., and Shen, D. (2014). Robust anatomical landmark detection for MR brain image registration. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014*, pages 186–193. Springer.

[30] Heimann, T. and Meinzer, H.-P. (2009). Statistical shape models for 3D medical image segmentation: A review. *Medical Image Analysis*, 13:543–563.

[31] Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844.

[32] Isgum, I., Staring, M., Rutten, A., Prokop, M., Viergever, M. A., and van Ginneken, B. (2009). Multi-atlas-based segmentation with local decision fusion - Application to cardiac and aortic segmentation in CT scans. *IEEE Transactions on Medical Imaging*, 28(7):1000–1010.

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

[34] Kelm, B. M., Zhou, S. K., Suehling, M., Zheng, Y., Wels, M., and Comaniciu, D. (2011). Detection of 3D spinal geometry using iterated marginal space learning. In *Medical Computer Vision. Recognition Techniques and Applications in Medical Imaging*, pages 96–105. Springer.

[35] Koch, M., Schwing, A. G., Comaniciu, D., and Pollefeys, M. (2011). Fully automatic segmentation of wrist bones for arthritis patients. In *Proceeding IEEE International Symposium Biomedical Imaging (ISBI)*, pages 636 – 640.

[36] Lee, S. C., Shim, J. S., Seo, S. W., Lim, K. S., and Ko, K. R. (2013). The accuracy of current methods in determining the timing of epiphysiodesis. *Bone & Joint Journal*, 95(7):993–1000.

[37] Martin, D. D., Wit, J. M., Hochberg, Z., Saevendahl, L., van Rijn, R. R., Fricke, O., Cameron, N., Caliebe, J., Hertel, T., Kiepe, D., Albertsson-Wikland, K., Thodberg, H. H., Binder, G., and Ranke, M. B. (2011). The use of bone age in clinical practice - Part 1. *Hormone Research in Paediatrics*, 76:1–9.

[38] Müller, H., Michoux, N., Bandon, D., and Geissbuhler, A. (2004). 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.

[39] Murphy, K. P., Weiss, Y., and Jordan, M. I. (1999). Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 467–475. Morgan Kaufmann Publishers Inc.

[40] Pietka, E., Gertych, A., Pospiech, S., Cao, F., Huang, H., and Gilsanz, V. (2001). Computer-assisted bone age assessment: Image preprocessing and epiphyseal/metaphyseal roi extraction. *IEEE Transactions on Medical Imaging*, 20(8):715–729.

[41] Razavi, N., Gall, J., and van Gool, L. (2011). Scalable multi-class object detection.

In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1505–1512.

[42] Rohr, K. (2001). *Landmark-based image analysis: using geometric and intensity models*, volume 21. Springer.

[43] Schiers, C., Tiede, U., and Höhne, K. (1989). Interactive 3D registration of image volumes from different sources. In *Computer Assisted Radiology, Proc. CAR*, volume 89, pages 666–670.

[44] Schmeling, A., Garamendi, P. M., Prieto, J. L., and Landa, M. I. (2011). Forensic age estimation in unaccompanied minors and young living adults. In Vieira, D. N., editor, *Forensic Medicine - From Old Problems to New Challenges*, chapter 5, pages 77–120. InTech.

[45] Shimizu, A., Ohno, R., Ikegami, T., Kobatake, H., Nawano, S., and Smutek, D. (2007). Segmentation of multiple organs in non-contrast 3D abdominal CT images. *International Journal of Computer Assisted Radiology and Surgery*, 2(3-4):135–142.

[46] Sonka, M. and Fitzpatrick, J. M. (2000). Handbook of medical imaging(volume 2, medical image processing and analysis). SPIE- The international society for optical engineering.

[47] Stern, D., Ebner, T., Bischof, H., Grassegger, S., Ehammer, T., and Urschler, M. (2014). Fully automatic bone age estimation from left hand mr images. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014*, pages 220–227. Springer.

[48] Swee, J. K. and Grbić, S. (2014). Advanced transcatheter aortic valve implantation (TAVI) planning from CT with shapeforest. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014*, pages 17–24. Springer.

[49] Tanner, J. M., Whitehouse, R. H., N, C., Marshall, W. A., Healy, M. J. R., and H, G. (1983). *Assessment of skeletal maturity and predicion of adult height (TW2 method)*. Academic Press, 2nd edition.

[50] Terada, Y., Kono, S., Tamada, D., Uchiumi, T., Kose, K., Miyagi, R., Yamabe, E., and Yoshioka, H. (2013). Skeletal age assessment in children using an open compact MRI system. *Magnetic Resonance in Medicine*, 69(6):1697–1702.

[51] Thodberg, H. H., Kreiborg, S., Juul, A., and Pedersen, K. D. (2009). The BoneXpert method for automated determination of skeletal maturity. *IEEE Transactions on Medical Imaging*, 28(1):52–66.

[52] Tu, Z. (2005). Probabilistic boosting-tree: Learning discriminative models for classification, recognition, and clustering. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1589–1596. IEEE.

[53] Tu, Z. (2008). Auto-context and its application to high-level vision tasks. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE.

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

[55] Wang, W. W., Xia, C. W., Zhu, F., Zhu, Z. Z., Wang, B., Wang, S. F., Yeung, B. H., Lee, S. K., Cheng, J. C., and Qiu, Y. (2009). Correlation of Risser sign, radiographs of hand and wrist with the histological grade of iliac crest apophysis in girls with adolescent idiopathic scoliosis. *Spine*, 34(17):1849–1854.

[56] Wörz, S. and Rohr, K. (2006). Localization of anatomical point landmarks in 3D medical images by fitting 3D parametric intensity models. *Medical Image Analysis*, 10(1).

[57] Yao, C., Wada, T., Shimizu, A., Kobatake, H., and Nawano, S. (2006). Simultaneous location detection of multi-organ by atlas-guided Eigen-organ method in volumetric medical image. *International Journal of Computer Assisted Radiology and Surgery*, 1:42.

[58] Zheng, Y., Barbu, A., Georgescu, B., Scheuering, M., and Comaniciu, D. (2008). Four-chamber heart modeling and automatic segmentation for 3-D cardiac CT volumes using marginal space learning and steerable features. *IEEE Transactions on Medical Imaging*, 27(11):1668–1681.

[59] Zheng, Y. and Comaniciu, D. (2014). Marginal space learning. In *Marginal Space Learning for Medical Image Analysis*, pages 25–65. Springer.