Technical note PR-TN 2008/00713

Issued: 12/2008

Model-Based Segmentation of Femur and Pelvis

Şükriye Güzel
Harald S. Heese
Sebastian P. M. Dries
Philips Research, Hamburg

Unclassified
© Koninklijke Philips Electronics N.V. 2008
Concerns: End Report

Period of Work: April 2008 - November 2008

Notebooks: None

Authors’ address: H. S. Heese harald.heese@philips.com
S. P. M. Dries sebastian.dries@philips.com
Title: Model-Based Segmentation of Femur and Pelvis

Author(s): Sukriye Guzel; Harald S. Heese; Sebastian P. M. Dries

Reviewer(s):

Technical Note: PR-TN 2008/00713

Additional Numbers:

Subcategory: Project: Smart MR workflow: dynamic recognition for magnetic resonance imaging (2007-127)

Customer:

Abstract: The document consists of a diploma thesis, which describes a completely automated segmentation chain for the bones of the human hip joint from diagnostic MR images including the model-building process for the corresponding anatomical structures. Mainly relying on the well-established model-based segmentation framework, the approach discusses strategies such as the Hough Transform for pre-positioning the involved surface models in the image to enhance robustness of the model-based framework. Furthermore, simple strategies for optimal choice of parameters for the model-based framework are investigated. The proposed methods have been tested on a set of nine MR images of female patients, all suffering from hip dysplasia.

Conclusions:
MODEL-BASED SEGMENTATION OF FEMUR AND PELVIS
DIPLOMA THESIS

BY
ŞÜKRIYE GÜZEL

NOVEMBER 18, 2008

Supervisors:
Prof. Dr. habil. Ph.D. Michael M. Morlock
Prof. Dr.-Ing. Rolf-Rainer Grigat
Dr. med. Sebastian Dries
Dr. rer. nat. Harald Heese
Contents

List of Figures v
List of Tables vii

1 Introduction 1
  1.1 Segmentation 1
  1.2 Hip Dysplasia 2
  1.3 Outline of Methods 3
  1.4 Overview 6

2 State of the Art: Image Segmentation 9
  2.1 Model-based Segmentation of Images 10
    2.1.1 Snakes and Deformable Models 11
    2.1.2 Active Shape Models 12
    2.1.3 Shape Constrained Deformable Models 12

3 Anatomical Basics 13

4 Mean Model Generation and Principal Component Analysis 17
  4.1 Principal Component Analysis 18
  4.2 Model Geometry and Configuration 22
  4.3 Coordinate System 23

5 Image Preprocessing 25
  5.1 Neighbourhood of a Pixel or a Voxel 25
  5.2 Convolution 26
  5.3 Gaussian Filter 27
  5.4 Edge Detection with Sobel Filter 28
  5.5 Thresholded Grey Level Image Conversion 30

6 Object Detection Using the Hough Transform 33
  6.1 Hough Transform 34
  6.2 Classical Hough Transform for Circles 35
  6.3 Bresenham Algorithm for Circles 39
  6.4 Using the Bresenham Algorithm for Circles to Generate Spheres 48
  6.5 Hough Transform for Spheres 52

© Koninklijke Philips Electronics N.V. 2008
7 Applying Point-based Registration by Computing Subspaces with Singular Value Decomposition and Computing the Rotation of Subspaces 57

7.1 Point-based Registration .............................. 58

7.2 Orthogonal Procrustes Problem and Computing the Rotation of Subspaces using Singular Value Decomposition ............ 59

8 Adaptation of Deformable Models to Anatomical Structures 63

8.1 Mesh Adaptation to the Anatomical Structure ............... 63

8.1.1 External Energy .................................. 64

8.1.2 Internal Energy .................................. 66

8.2 Feature Parameter Settings ................................ 67

8.3 Simulated Annealing ................................ 68

8.4 Refining of Surfaces .................................. 72

9 Work Flow 75

9.1 Prepositioning ........................................... 75

9.2 Refinement of Model Topology ......................... 76

9.3 Adaptation, Evaluation and Generation of Mean-models .... 78

9.4 Summary ................................................. 78

10 Results 79

10.1 Evaluation ............................................. 80

10.2 Conclusion ........................................... 87

11 Outlook 91
## List of Figures

1.1 Initial models and adapted models are illustrated to show the deformation of the models. Simultaneously, the segmented models are shown. The initial models are coloured grey, and the adapted models are given in blue. These models are surface rendered models of the right and the left pelvis and femur, the cartilage of the femoral head and the cartilage in the acetabulum.

1.2 An example for application of point-based registration to preposition the models at the location of the true anatomical structure in the image of the patient

2.1 Voxel grid; every "sphere" represents a voxel

2.2 A grey shaded image overlaid with the edges of the object

3.1 Proximal right femur and pelvis in front view

3.2 Surface rendering of the right pelvis, in lateral view

3.3 Movements of the hip joint with defined axes

3.4 Right and left pelvis, femur, femur’s cartilage and cartilage in the acetabulum

4.1 Triangulated surfaces for the right and left pelvis, femur and cartilages, front view

4.2 Illustration of the principal component analysis on a set of 2D data

4.3 Modes of the mesh of the pelvis

4.4 Wireframe of a femur

5.1 Connectivity

5.2 Convolution

5.3 Gaussian Template

5.4 Sobel filter

5.5 Sobel filter kernels

5.6 Sobel filtered image

6.1 Duality between \((x, y)\) and \((x_0, y_0)\)

6.2 An example for mapping a binary image into the accumulator space

6.3 A binary image is mapped into the accumulator space

© Koninklijke Philips Electronics N.V. 2008
6.4 The blue contour points are generated with 
\[ x = x_0 + r \cos(t_i), y = y_0 + r \sin(t_i), \]
with \( x_0 = 0, y_0 = 0 \) and \( 0 \leq t_0 \leq t_1 \leq \ldots \leq t_n \leq 2\pi \). Further, the four quadrants of a circle in Cartesian coordinates are illustrated; \((x_{\text{start}}, y_{\text{start}}) = (0, r)\) indicates the starting point and the red arrow shows the workflow direction.

6.5 Movement possibilities (grey points) in each subfigure c, d and e for each successor of a point. The black bordered circle in subfigures c,d and e indicates the next point which is going to be chosen. Each subfigure c, d and e demonstrates a possible direction of movement.

6.6 The five possibilities for the circle’s intersection with the coordinate lines \( x_i + 1 \) and \( y_i - 1 \) are shown.

6.7 Bresenham circle overlaid with a circle calculated with the trigonometrical formula. The inputs are the circle’s parameters, i.e. the circle centre and the radius.

6.8 Different perspective views of the surface rendered right femur. The figures demonstrate the anatomical femoral head and its approximately spherical structure. The approximated discrete sphere is given in figure (b).

6.9 Generating a sphere using the BCA.

6.10 Completing the sphere using the symmetry to the \( x, y \) plane through its centre point.

6.11 Grey level image in the axial plane and its Hough-transformed image.

7.1 Rigid transformation.

7.2 Point-based registration of the left pelvis. \( T \) is calculated between the initial femur model in grey and the adapted femur model in blue.

8.1 Simulated Annealing results.

8.2 Proximal right femur and pelvis in front view.

9.1 Hough Workflow.

10.1 Distance between two vertices. The line represents the distance from a reference vertex (left side) to an adapted mesh vertex (right side). The mean distance is given subsequently.

10.2 Parameter search using simulated Annealing.

10.3 Segmentation example.

10.4 Segmentation result with an the PBR method.

10.5 Prepositioning using HT.
List of Tables

4.1 Numbers of vertices and triangles for different meshes . . . . . . . . 22
10.1 Initial models without prepositioning: average distance . . . . . . . 82
10.2 Initial models without prepositioning: standard deviation . . . . . . 82
10.3 Manual prepositioning: average distance . . . . . . . . . . . . . . 83
10.4 Manual prepositioning: standard deviation . . . . . . . . . . . . . . 83
10.5 Mean models: average distance . . . . . . . . . . . . . . . . . . . . 84
10.6 Mean models: standard deviation . . . . . . . . . . . . . . . . . . . 84
10.7 Point based registration of models: average distance . . . . . . . . 85
10.8 Point based registration of models: standard deviation . . . . . . . 85
10.9 2 Level segmentation: average distance . . . . . . . . . . . . . . . 85
10.10 2 Level segmentation: standard deviation . . . . . . . . . . . . . . 85
10.11 Segmentation using HT: average distance . . . . . . . . . . . . . . 85
10.12 Segmentation using HT: standard deviation . . . . . . . . . . . . . 86
10.13 Landmark distances to the segmented pelvis: average distance . . 87
10.14 Landmark distances to the segmented pelvis: standard deviation . 87
10.15 Processing time of the HT . . . . . . . . . . . . . . . . . . . . . 87
10.16 Visual evaluation of the segmentation results using manual preposi-
tioning, translation and Rotation . . . . . . . . . . . . . . . . . . 88
10.17 Visual evaluation of the segmentation results using HT prepositioning 88
10.18 Visual evaluation of the segmentation results using PBR preposition-
ing . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 89
10.19 Visual evaluation of the segmentation results using 2Level preposi-
tioning with mean models . . . . . . . . . . . . . . . . . . . . . . 89
# List of Algorithms

<table>
<thead>
<tr>
<th></th>
<th>Algorithm</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BresenhamCircle($x_0, y_0, \text{radius}$)</td>
<td>49</td>
</tr>
<tr>
<td>2</td>
<td>SBCA($\text{radius}$)</td>
<td>53</td>
</tr>
<tr>
<td>3</td>
<td>SimulatedAnnealing($T, T_{stop}, \alpha$)</td>
<td>72</td>
</tr>
</tbody>
</table>
Abstract

In the last half century the role of medical imaging has expanded beyond the simple visualisation and inspection of anatomic structures. Nowadays surgical planning and simulation, intra-operative navigation and radiotherapy planning can be prepared with medical imaging tools. In the context of computer aided surgery, the exact knowledge about the shape of the anatomical structure of the patient, its geometry and its alignment in the body is of great importance. Therefore the model-based approach is applied to segment the anatomical structures of interest.

The main subject of this thesis is the detection of particular shapes for the prepositioning of the initial deformable models in the images of the patients, and the segmentation of the anatomical structures from these images. To this end the model-based approach is used to segment the right and left pelvis and femur.

Three methods are used to achieve a good prepositioning of the anatomical structures in the image of the patient. First, the Hough transform (HT) of particular shapes is used for aligning the initial models correctly in the image of the patient, and thereby achieving good segmentation results. The HT is a well known technique to localise particular shapes in images.

The second approach for the prepositioning of the anatomical structures is to use a point-based registration. One of the deformable models is used as the initial model to initiate the segmentation and subsequently serve as the reference structure for prepositioning of the other models. After employing the methods of point-based registration to the reference mesh (initial mesh) and the adapted mesh, useful information about the rotation, translation and scaling of the adapted mesh can be extracted. The obtained knowledge can then be applied to the other meshes. Through this an improvement of the model-based approach is also obtained.

Third prepositioning method is more dexterous employment of the model-based segmentation parameters by attempting to segment anatomical strutures in two adaptations. In the first segmentation a parameter set is chosen that allows the models to move within the image during adaptation. In the second segmentation the parameter set is chosen so that the models adapt themselves locally to the anatomical structure.

This study delivers a segmentation of the anatomical structures that best approximates them. The whole segmentation chain is automated by using the detection method, the Hough transform (HT) of particular shapes. Moreover, the study benefits from combining many algorithms which are used to solve different problems in the digital image processing. Known algorithms, such as simulated an-
nealing, the Hough transform for particular shapes, the Bresenham algorithm, the principal component analysis, the point based registration and the model-based approach are suitable for improving the segmentation result. Furthermore, many other tools support the whole work flow to reach the best possible result.
Declaration

I hereby confirm that I have authored this thesis independently and without use of others than the indicated resources. All passages taken out of publications or other sources are marked as such.

Hamburg, November 18, 2008

City, Date                          Sign
Acknowledgements

I sincerely would like to thank Dr. habil. Ph.D. Michael M. Morlock for providing me with an interesting and challenging topic of research. Special thanks to Dr. med. Sebastian Dries giving me the opportunity to perform this work in the digital imaging department at Philips Research. Special thanks also to Dr. rer. nat. Harald Heese for his support and motivation. I would also like to thank Prof. Dr. Rahul Peter Das and Dr. Clas Abels for proof-reading parts of this thesis and their support. My deep thanks go to Manuel Das for understanding and motivation. I also thank Maide Güzel, Dipl. Ing. Grigoriy Quiring, Zahide Tarim and friends for their support during the making of this thesis and the whole study.
Chapter 1

Introduction

In recent years there has been increased interest in automatic methods for segmentation of the pelvis and the femur in tomographic images. The tomographic imaging systems change their quality from day to day, leading to the availability of an ever increasing amount of high-resolution image data. Such high resolution data allow for high quality reconstruction of anatomical structures and are the foundation for advanced therapy and surgery planning systems. Medicine is being revolutionised by the rapid development and proliferation of medical imaging technologies. Medical imaging allows scientists and physicians to extract potentially life-saving information by peering noninvasively into the human body. In the last half century the role of medical imaging has expanded beyond the simple visualisation and inspection of anatomic structures. Nowadays surgical planning and simulation, intra-operative navigation and radiotherapy planning can be prepared with medical imaging tools. In the context of computer aided surgery, the exact knowledge about the shape of the anatomical structure of the patient, its geometry and its alignment in the body is of great importance. Extracting the relevant information from images is hardly feasible without applying a priori knowledge, because the anatomical structures have very complex shapes. Therefore the automated image analysis intensively uses anatomical models including shape knowledge.

1.1 Segmentation

The main subject of this thesis is the detection of particular shapes for the prepositioning of the initial deformable models in the images of the patients, and the segmentation of the anatomical structures from these images. To this end a model-based approach is applied to segment these anatomical structures. Within the segmentation the models of the anatomical structures have almost exactly the form of the patient’s anatomical structures, and contain the information about their geometry, form and anatomical alignment, as well as the alignment of anatomical structures to each other.

The research topic of this thesis is "Model-Based Segmentation of Femur and Pelvis". It emerged from the cooperation of Klinikum Eilbek, the Hamburg Uni-
versity of Technology and the Philips Research Group of Hamburg, and is a very important part of the main project "Triple/MR-Analyse der Beckendreifachosteotomie". The head of the Department of Orthopedics and Trauma at the Klinikum Eilbek is Prof. Dr.med. Ekkehard Hille. His team provides a full range of elective and emergency specialist procedures as well as minimally-invasive treatment of conditions affecting the musculoskeletal system of patients in every age group. Prof. Dr.habil. Ph.D. Michael M. Morlock leads the Department of Biomechanics at the Hamburg University of Technology and has supervised this thesis. The programs of this thesis were developed at the Philips Research Group of Hamburg in the Department of Digital Imaging under the leadership of Dr.med. Dries. They serve to correctly segment the right and left pelvis, the femur and its cartilage and the cartilage in the acetabulum, and improve the result of the segmentation (see figure 3.4). Within this project, computer aided surgery methods for triple pelvic osteotomy are the final objective. Currently, a conventionally treated cohost is observed prospectively over the time.

The clinical study "Triple/MR-Analyse der Beckendreifachosteotomie" aims at establishing a new technique for the exact prediction of essential corrections in cases of triple pelvic osteotomy with the magnetic resonance imaging (MRI) measuring method, and comparing it with the current standard of conventional x-raying. It examines the possible utility of computer aided surgery. To this end it observes the surgical process by means of a navigation system prototype application. The summarised hip factor (Hüftwert)\textsuperscript{1} is determined by x-ray overviews of the pelvis, and the MRI images used in this thesis deliver more accurate 3D volume information on the anatomical structures after segmentation.

\subsection*{1.2 Hip Dysplasia}

Residual dysplasia is a major cause of osteoarthritis and should be corrected surgically at an early age if severe. Figure 1.2 illustrates the normal hip joint in the left image, the hip dysplasia in the middle image, and the hip dislocation in the right image. The hip joint forms the attachment of the leg to the body, and is a ball and socket joint; this is explained in section 3. The ball portion is the head of the femur while the socket, the acetabulum, is part of the pelvis.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{hip_joints.png}
\caption{The normal hip joint (left), hip dysplasia (middle), and hip dislocation (right).}
\end{figure}

In a normal joint the ball rotates freely within the socket and has six degrees of freedom, as is shown in figure 3.3. Developmental dysplasia or even dislocation

\footnote{For a definition of this term see section 1.2}
of the hip (DDH) is an abnormal formation of the hip joint in which the femoral head is not sufficiently contained in the socket. In some instances, the capsule of the hip joint may be loose and stretched. The hip dysplasia or the dislocation varies. In some children, the femoral head is simply loose in the socket at birth. In other children, the femoral head is completely out of the socket. In still other children, the looseness worsens as the child grows and becomes more active. When the condition is detected at birth, it can usually be corrected by conservative treatment. But if the hip is not dislocated at birth, the condition may not be noticed until the child begins walking. At this time, treatment is more complicated and uncertain, but triple pelvic osteotomy (TPO) is a possible treatment. It is observed that DDH tends to run in families. It can be present in either hip and in any individual, though it usually affects the left hip and is predominant in girls and first-born children. For children older than two years treatment tends to be difficult. Deformities may worsen, making open surgery necessary to realign the hip. Based on the evaluation of 817 radiograms of hips including the clinical records, the term "summarised hip factor"(Hüftwert) is introduced in connection with the diagnosis of hip joint disorders in the paper of Busse et al. [7]. Its value depends upon well-known radiological criteria. The summarised hip factor permits the summarised description of hip deformities and, in addition, offers the possibility of statements concerning the prognosis.

1.3 Outline of Methods

This study delivers a segmentation of the anatomical structures that best approximates them. The whole segmentation chain is automated by using the detection method, the Hough transform (HT) of particular shapes, for aligning the initial models correctly in the image of the patient, and thereby achieving good segmentation results. Several approaches help to further improve the segmentation. Moreover, the study benefits from combining many algorithms which are used to solve different problems in the digital image processing. Known algorithms, such as simulated annealing, the Hough transform for particular shapes, the Bresenham algorithm, the principal component analysis, the point based registration and the model-based approach are suitable for improving the segmentation result. Furthermore, many other tools support the whole work flow to reach the best possible result.

First, the patients belonging to the project have been screened and the tomographic image obtained for further processing. The tomographic image is isotropic in two directions. The resolution in the $x-$ and $y-$directions is 0.625 mm, and in the $z$-direction it is 0.8 mm. A slice is 0.8 mm thick. The model generation aims to extract useful information about anatomical structures from MRI images and to represent them geometrically with the assistance of computers. The anatomical structures are manually segmented from the image of the patient and represented as deformable models: the right and left pelvis, femur and corresponding cartilage, and cartilage in the acetabulum. These anatomical structures are illustrated
in section 3. The models represent the learning data. The aim is to automate
the segmentation of the anatomical structures. In this work a triangulated sur-
face represents the models. The first step now is to adapt these models to the
anatomical structures in the image of the patient while the anatomical structures
are simultaneously segmented.

The model-based segmentation has the advantage of already being in possession
of a large amount of key knowledge concerning the objects to extract. The model
captures the geometrical information and the shape of the anatomical structures.
By the mesh adaptation the energy function $E(\text{I}_{\text{grey}}, \text{v}) = E_{\text{ext}}(\text{I}_{\text{grey}}, \text{v}) + \alpha E_{\text{int}}(\text{v})$
is minimised. The $E_{\text{ext}}(\text{I}_{\text{grey}}, \text{v})$, which is called external energy, attracts the initial
model to the object boundaries, thereby the initial model adapts to the anatomical
object. The $E_{\text{int}}(\text{v})$, called internal energy, penalises the deviation of the adapted
model from the initial model, whereas $\alpha$ regulates the interaction of these two
energies. The section 8 describes the external and the internal energy in detail.
The vector $\text{v}$ represents the vertices of the models, and $\text{I}_{\text{grey}}$ represents the input
image. The model-based approach is explained in section 8.

The model-based segmentation needs as its input parameters the image of the
patient, the mesh of the anatomical structure and some parameters which drive
the segmentation. These parameters play an important role in the external and
internal energies which are used in the model-based approach. Different choices
of parameter sets lead to different results of the segmentation. Especially in the
case of the model-based approach, the multitude of different parameters influenc-
ing the adaptation with deformable models makes a general parameter choice very
difficult. This problem is increased by the fact that an optimal setting for any
one parameter depends on the setting of the others. Furthermore, the choice is
complicated by the dependency on the image data, the anatomical structure to be
segmented, and the accuracy of the model prepositioning. Therefore, parameters
need to be chosen for every given problem individually. It is thus obvious that it
is necessary to analyse the parameter settings controlling the result of the model-
based approach in dependency on the image of the patient. The search for the
parameter settings is challenging due to the large amount of combination possi-
bilities. There is more than one solution. Intuitively, one would probably at first
tend to use the exhausting search, which would take large computation time. A
more intelligent search algorithm is the simulated annealing algorithm which pro-
vides good parameter sets for the model-based approach. The section 8 explains
this algorithm in detail.

The quality of the models has also to be considered for obtaining better results.
For amalgamating the properties of all adapted models a mean model is generated.
The results of the model-based segmentation is improved considerably if the mean
model is used.

The model-based segmentation is based on a large amount of previous knowledge.
Ultimately, the quality of the model-based segmentation relies on both the algo-
rithm and on the a priori knowledge encoded in the model.

Figure (1.1) illustrates the surface rendered mesh used for segmentation. The
initial surface rendered meshes are grey coloured, the adapted meshes are in blue.
The adapted models are also shown in figure (1.1), and from different perspectives. The grey meshes indicate the initial positioned meshes in the image, and the blue meshes are the adapted ones.

![Initial models and adapted models](image)

Figure 1.1: Initial models and adapted models are illustrated to show the deformation of the models. Simultaneously, the segmented models are shown. The initial models are coloured grey, and the adapted models are given in blue. These models are surface rendered models of the right and the left pelvis and femur, the cartilage of the femoral head and the cartilage in the acetabulum.

Obviously, the initial models are prepositioned at the location of the true anatomical structures (see figure (1.1)). The first approach to prepositioning in the image of the patient is achieved by the Hough transform for spheres. The Hough transform for spheres is sensitive only for spheres in the image, and transforms the whole image into a so-called accumulator space. Each coordinate of this space defines a parameter set for the sphere, and its integer value expresses the availability of the sphere in the image. The larger the value, the higher the probability of the presence of the sphere in the image. The main idea is to find the femoral head in the image of the patient. Thereby the parameters, the centre point \((x_0, y_0, z_0)\) and the radius \(r\) of the anatomical femoral head are calculated. With respect to these parameters the deformable model of the femur is aligned to the anatomical femoral head in the image of the patient; all other models can be aligned with respect to this too.

The second approach for the prepositioning of the anatomical structures is to use a point-based registration. One of the deformable models is used as the initial model to initiate the segmentation and subsequently serve as the reference structure for prepositioning of the other models. Usually, point-based registration is used between images to fit the first image into a second one by minimising the error between the landmarks. But in this case the point based registration formulates the relationship between the initial model and the adapted model. This relationship, or transformation, is then used to preposition the other structures with reference to the mean model. The point-based registration is discussed in section 7.

In other words, point based registration will be employed in this work for the prepo-
Figure 1.2: An example for application of point-based registration to preposition the models at the location of the true anatomical structure in the image of the patient.

sitioning of the models in the image by exploiting the information obtained about the adaptation. The adapted mesh contains information about the true location and extent of the patient’s anatomical structure. For example, after employing the methods of point-based registration discussed in section 7 to the reference mesh (initial mesh) and the adapted mesh, useful information about the rotation, translation and scaling of the adapted mesh can be extracted. The obtained knowledge can then be applied to the other meshes. The aim is to achieve a good prepositioning of the models.

Through this an improvement of the model-based approach is also obtained. Figure (1.2(a)) illustrates the initial model of the femur in grey and the adapted model in blue. First, the point-based registration is calculated between the initial femur model and the adapted femur model; see figure (1.2(a)). The obtained matrix $T$ is now applied to the grey model of the pelvis. The registered model is illustrated in red in figure (1.2(b)). To the positioned pelvis model the model-based approach is applied. The blue pelvis model is obtained as the result of the model-based segmentation, as figure (1.2) shows. Different perspectives are shown for a better understanding of the adaptation.

1.4 Overview

This work is organised as follows:

**Chapter 2:** This chapter introduces the basic concepts of image segmentation. Starting with the definition of segmentation, the usage and the aim of image segmentation is explained. It is used to partition the image into regions. For a computer the input image is a field of numbers, each representing a picture element called a pixel or voxel. Certain measures for grouping these numbers are applied to the computer to make it connect the numbers, thus creating the regions. The usage of computer aided planning is growing in accordance with the growth
of the field of medical imaging technologies, to which model-based segmentation, introduced in section 2.1, belongs. But segmenting structures from medical images, and the reconstruction of these, is difficult, because the same structures vary significantly between individuals. This inter-individual variability is successfully captured by some methods which use active contouring and deformable models. Section 2.1.1 discusses the advantages of such deformable models.

Chapter 3: In this chapter anatomical basics are briefly explained. These are useful for comprehending the complex structure of the right and left pelvis and the femur.

Chapter 4: Model and mean model generation also pose a challenge. The initial models are generated by manual segmentation of a patient’s image. With the help of the marching cube algorithm triangulated meshes are generated. This chapter describes the model generation. These models are then applied to other patients and adapted individually, giving new meshes. Section 4.1 discusses the principal component analysis (PCA) describing shape variation. This PCA provides eigenvalues and eigenvectors; with these, new models, approximating the individual models, can be generated. These model variations are turned over to the algorithm, so as to be better able to handle individual differences. The model geometry is explained in section 4.2. Furthermore, the coordinate system describing the data is introduced in section 4.3.

Chapter 5: Before segmentation the image is prepared to emphasise the anatomical structures of interest as contrasted to the surrounding structures. The parameters used for this are determined experimentally. Then the neighbours of a pixel or voxel are defined (section 5.1), and the convolution explained mathematically (section 5.2). The Gaussian filter smoothing the image before edge detection is explained in section 5.3; edges are detected in the image using a $3 \times 3 \times 3$ Sobel filter explained in detail in section 5.4. All edges with magnitude lower than an experimentally predefined threshold are pruned. The thresholding suppresses the less important anatomical structures and is described in section 5.5.

Chapter 6: The localisation of the anatomical structures in the image of the patient, their adaptation and the automation of the workflow are the main subject of this study, and are described in this chapter. The localisation (prepositioning) is crucial and is the first step of the segmentation chain; it is done with the Hough transform for spheres. But it is challenging due to shape variability. The pelvis and the femur of different patients differ in alignments too; usually the rotation of pelvis and femur vary strongly from patient to patient. In this chapter, first of all the anatomical structure which can be most easily found in the image of the patient is determined; it is the femoral head because this is suitable for a parametric definition. For detecting the femoral head, a sphere has to be localised. For this, the Hough transform is employed; a classical Hough Transform is explained in detail in section 6.1. The geometry of a sphere can be interpreted in several ways. One of them is to interpret the contours and plane sections of the sphere as circles. Therefore the Hough transform for circles is described in section 6.2; this allows a straightforward generalisation to the Hough transform for spheres. But the explicit formula for circles of spheres uses trigonometric terms, which increases
computation time. To decrease this by using simple arithmetic, i.e. addition or subtraction, the contour points of a sphere can be approximated by the Bresenham algorithm for circles. This algorithm approximates the true circle arc on a voxel grid; it is discussed for circles in depth in the section 6.3, while the straightforward generalisation to spheres is given in section 6.4.

**Chapter 7:** Point based registration, explained in detail in section 7.1, is employed for the prepositioning of the models in the image by exploiting the information obtained about the adaptation from the initial mesh and the adapted mesh. The image registration problem can be reduced to the orthogonal Procrustes problem by subtracting its mean value from the data set, the data set being the vector which describes the meshes. The vector has the coordinates of the points, and has a notation for combining the points to triangles. The orthogonal Procrustes problem is solved by a singular value decomposition, and is explained in section 7.2; it solves the problem if the matrix $V$ can be rotated into the matrix $\hat{V}$.

**Chapter 8:** This chapter mathematically explains the model-based approach, which adapts the models to the anatomical structures. The aim of this adaptation is a precise segmentation of the anatomical object, which is required for automated image analysis. The model-based adaptation is performed accurately matching the boundaries of the patient’s anatomy [12]. But the choice of the appropriate parameters is very difficult because the number of the parameters influencing this approach in particular is huge. The dependency of the parameters on each other compounds the difficulty. Furthermore, the choice of the parameters also depends on the image values, the anatomical structure which has to be segmented, and the accurate prepositioning of the models. All these influence segmenting considerably. Therefore, a promising search algorithm is applied to find the best parameter set. This algorithm, simulated annealing, is explained in section 8.3. By starting with a good parameter set, it obtains the next globally best parameter set. To improve the models on crucial regions such as circular areas the number of the triangles defining the curve is refined. The refining function is described in section 8.4.

**Chapter 9:** This chapter shows the application of the algorithms to the image of the patient.

**Chapter 10** In this chapter all the experiments and tests are gathered and the conclusion is given.

**Chapter 11:** In this chapter the future outlook is given.
Chapter 2

State of the Art: Image Segmentation

In this chapter image segmentation is briefly explained, and the models described using the model-based approach. The aim of image segmentation is to partition the image into regions, wherein the regions are defined simply by a set of pixels or voxels. The word "voxel" is a combination of "volume" and "pixel". It describes a volumetric location of a grey value (Figure 2.1) [8].

![Figure 2.1: Voxel grid; every "sphere" represents a voxel](image)

In image segmentation the representation of the object is changed to describe the object in a more meaningful way so as to facilitate the analysis. As a result of image segmentation, regions are obtained which can be described by pixels similar to some characteristics, e.g. mean grey value. [3] In this work the segmentation is performed for medical images; the principal aim is to improve medical diagnosis through segmented images. The medical image segmentation pursues, among other things, the following aims:

- computer guided surgery
• tumor localisation

• early diagnosis and treatment

The variety of fields in which segmentation of medical images can be applied for the improvement of human medicine is huge. Several general-purpose algorithms and techniques have been developed for image segmentation. Note that the image segmentation has to be adapted to every problem individually; usually the combination of several methods for a single problem leads to results of high accuracy. In many scientific disciplines a model-based segmentation with deformable shape models is used to improve and to automate the image analysis. Therefore image segmentation is a fundamental issue in the biomedical imaging sphere. A brief overview about the algorithm methods and techniques is given in section 2.1.

2.1 Model-based Segmentation of Images

Segmenting structures from medical images and the reconstruction of these is difficult, because the structures vary significantly between individuals. This inter-individual variability is successfully captured by some methods which use active contour methods and deformable models.

In this work a triangulated surface is adapted to the respective anatomical structure by model-based adaptation. To overcome the misadaptation of the rigid models to anatomical structures deformable models are used. These are able to accommodate the varying anatomical structures of individual patients [42]. Model-based segmentation is divided into two parts, the first consisting of region based segmentation, and the second of boundary based segmentation.

Region based segmentation evaluates the whole region, measuring the similarity of the pixels or voxels in the region. The homogeneity of a region is of interest for segmentation in general. Region-based segmentation methods attempt to partition or group regions according to common image properties. These image properties consist of

• intensity values from original images, or computed values based on an image operator

• textures or patterns that are unique to each type of region

• spectral profiles that provide multidimensional image data

Boundary based segmentation is often used to look for explicit or implicit image boundaries between two regions. The most commonly used boundary method is known as edge detection (Section 5.4).

Figure 2.2 illustrates the object’s borders in an image.
2.1.1 Snakes and Deformable Models

Deformable Models are physically described models which can be deformed according to the law of Newton mechanics, in particular by the theory of elasticity expressed in Lagrange dynamics. They are able to capture the variability of biological structures in different individuals. The potency of these models stems from their ability to segment anatomical structures by exploiting conditions derived from the image data together with a priori knowledge about the shape of these structures [39]. Furthermore, deformable models support highly intuitive interaction mechanisms that allow medical scientists to bring their expertise to bear on model-based segmentation.

The dynamic equation for the motion of a deformable model can be written as a second order differential equation

\[ M \frac{d^2 \mathbf{P}(r, t)}{dt^2} + C \frac{d \mathbf{P}(r, t)}{dt} + K \mathbf{P}(r, t) = \mathbf{F}(r, t) \]

where \( M \) is the mass matrix, \( C \) the damping factor, \( K \) a stiffness matrix of the mesh structure and \( \mathbf{P} \) a vector of all vertices. \( \mathbf{F} \) is a vector of external forces (et al. Zhukov [25]).

These models (snakes, balloons, etc.) are active models. The first 2D segmentations were done by Kass et al. in 1988 [26]. Snakes are spline curves which are embedded in the image. Parametric snake models consist basically of an elastic curve (or surface) which can dynamically conform to object shapes in response to internal energy (elastic forces) and external energy (image and constraint forces). These energies result from global minimisation based on local information, e.g., the intensity of the gradient.

The internal energy is minimal if the snake is similar to the model shape. The external energy is minimal if the snake adapts itself to fit the object boundary in the image. The adaptation occurs in response to both forces, wherein the snakes dynamically adapt to the position and the shape of the object in the image. Hereby a prepositioning of the models is preferred. Otherwise the approach leads to misadaptation, as the models are not customised for a specific kind of shape.
2.1.2 Active Shape Models

Active shape models are statistical models of the shape. These models can only be deformed in a way that is consistent with a training set [11] [42]. To obtain statistical information, a statistical model can be built from the training set if the size of the data set is high enough. The results of the statistical analysis are some eigenmodes and a mean model. The eigenmodes can be embedded in the energy terms, as an expression of the variability of the shapes. As many variations of the image shape as possible are captured with the training set. However, huge training sets are needed for a good generalisation of the features.

2.1.3 Shape Constrained Deformable Models

The advantages of both methods are combined in a new approach, namely the shape constrained deformable method. The so-called shape constrained deformable model combines the information on anatomical structures captured by the deformable model with the results of the principal component analysis that calculates the variability of these models in different individuals. During this process a new model is obtained.
Chapter 3

Anatomical Basics

This chapter introduces the relevant anatomical structures. All the terms which describe the different parts of the structure of the hip, of the femur and its cartilage as well as the cartilage in the acetabulum will be briefly explained [29] [36]. Note that it is especially these structures in the female which are of interest here.

The hip is one of the largest joints in the body and assists walking, running and squatting. It consists of two bones, the pelvis and the femur. The femur (thighbone) is the largest and strongest bone in the body and accounts for one fourth of a person’s height. The top part articulates with the pelvis bone and the bottom section forms the top of the knee. The femur bone is divided into femoral head, neck and shaft 3.1. The femoral head is the uppermost end of the femur. It has approximately the form of a sphere. The neck anastomoses the femoral head with the femur. The top of the shaft has two bony protrusions, which are called greater trochanter and lesser trochanter.

![Figure 3.1: Proximal right femur and pelvis in front view](image)

The pelvis is the bony structure located at the base of the spine 3.1. It separates the trunk from the lower extremities. Each os coxae (hipbone) consists of three bones, the ilium, ischium, and the pubis; see figure 3. The ilium is the largest
and uppermost, the ischium the posterior-inferior (back-lower), and the pubis the anterior (front) part of the hipbone. Two pelvis bones form the pubis anteriorly at the symphysis and the pelvis posteriorly to the sacrum.

![Surface rendering of the right pelvis, in lateral view](image)

Figure 3.2: Surface rendering of the right pelvis, in lateral view

The acetabulum forms a hollow cup on the outside of the coxal bone. The inside of the acetabulum and the femoral head are covered by the cartilage. When the femoral head is fitted into the acetabulum the hip joint is completed.

**Movements of the Hip Joint**

Designed as a ball and socket joint, the hip joint has three rotation axes. All of these axes go through the center of the femoral head and are perpendicular to each other. Thereby three degrees of freedom are obtained:

1. Transverse axis: anteverision and retroversion
2. Sagittal axis: abduction and adduction
3. Longitudinal axis: interior and external rotation

The whole hip and the femur are illustrated in the figure 3.4.
Figure 3.3: Movements of the hip joint with defined axes

Figure 3.4: Right and left pelvis, femur, femur’s cartilage and cartilage in the acetabulum
Chapter 4

Mean Model Generation and Principal Component Analysis

The rapid development and proliferation of medical imaging technologies is revolutionising medical science. Medical imaging allows scientists and physicians to extract potentially life-saving information by peering non-invasively into the human body [39]. The model generation aims to extract, with the assistance of computers, clinically useful information about anatomical structures imaged through MRI and their geometric representation. Deformable models are used for the representation of these anatomical structures, capturing their geometry according to their segmentation from medical images. The deformable models are able to segment, match and track images of anatomical structures by exploiting constraints derived from the image data together with a priori knowledge about the location, size, and shape of these structures. The process of creating these models is performed with a "marching cubes" algorithm that creates triangulated models of constant density surfaces from 3D medical data. The detail in images produced from the generated surface models is the result of maintaining the inter-slice connectivity, surface data, and gradient information present in the original 3D data [41]. The triangulated surface model is also called mesh.

In this work the segmentation of the pelvis, the femur and its cartilage, as well as the cartilage in the acetabulum, is performed using these models. All these models are generated as mentioned above. Figure 4.1 illustrates the meshes for these bones.

The anatomical structure of the pelvis is one of the complexer structures of the human body in section 3. This bone is irregularly shaped and has a hole, the obturator foramen. It is created by the ischium, (Figure 3.2) and pubis bones of the pelvis to let nerves and muscles pass through [36].

The first part of this chapter, section 4.1, describes the statistical method known as principal component analysis (PCA), and all the terms which are used for the calculations. Besides, a brief mathematical description of the variables and the terms is also given. After applying the PCA, the principal components of the meshes can be calculated, namely eigenvectors and eigenmodes. These are used to generate variable shapes mathematically. Thereby the statistical model can be
PCA highlights the similarities and differences of all individual meshes by some statistical evaluations of their vertex deviation (a vertex is a corner element of a polygon). Thereby PCA decreases the number of dimensions by retaining the same information in the vertex set, thus reducing a complex data set to lower dimensions for easier analysis. Furthermore, the PCA extracts relevant information from the models and improves their information content about the anatomical structure and geometry. Usually this technique is applied on image compression. In several publications a mean model is used to improve the segmentation results of individualised meshes. The mean mesh is generated when PCA is applied to the models’ vertices.

The second part introduces the model geometry in section 4.2. The model configuration is explained in detail. Furthermore, how to extract the triangulated surface from the volumetric data is briefly explained. To describe the relation between the vertices, edges and triangles two formulae are given 4.6. A Cartesian coordinate system is used in section 4.3 to describe positions in the volumetric data. The models are also expressed in this common coordinate system.

4.1 Principal Component Analysis

A mean mesh model includes the most important characteristics of all adapted models [36]. The PCA is used to obtain it. Within the PCA the principal components (e.g. the eigenvalues) of the models can be derived by a single value...
decomposition. To begin with, the mathematical terms and formulae used here are described. This introduction establishes the standard deviation, covariance, eigenvectors and eigenvalues.

The standard deviation is defined as the deviation of the data values from the mean value $x$ as given in the formula:

$$
\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_i - \overline{x})^2}
$$

$$
\overline{x} = \frac{1}{N} \sum_{i=1}^{N} x_i
$$

(4.1)

The measure of spread of data in a data set is the variance $\sigma^2 = \frac{1}{N} \sum_{i=1}^{N} (x_i - \overline{x})^2$. These two formulae are given in one dimension. However, many data sets have more than one dimension, allowing covariances between dimension pairs to be given. The covariance describes the connection between two dimensions. A covariance with the value zero means that the observed data sets have no connection. The formula is given below:

$$
\text{var}(x) = \frac{1}{N} \sum_{i=1}^{N} (x_i - \overline{x})(x_i - \overline{x})
$$

(4.2)

$$
\text{cov}(x, y) = \frac{1}{N} \sum_{i=1}^{N} (x_i - \overline{x})(y_i - \overline{y})
$$

(4.3)

Note that the covariance of $x$ and $y$ can be expressed in vector notation as $\sigma_{xy}^2 = \langle(x_i - \overline{x}), (y_i - \overline{y})\rangle$, wherein $\langle . \rangle$ denotes the average of values indexed by $i$. A large value means high redundancy, and a small value means low redundancy [19]. The covariance is $\sigma_{xy}^2 \geq 0$.

Eigenvalues are a special set of scalars associated with a linear system of equations (i.e., a matrix equation) that are sometimes also known as characteristic roots, characteristic values [23], proper values, or latent roots [27].

A value $\lambda \in \mathbb{R}/\mathbb{C}$ is an eigenvalue of the matrix $A$, if a vector $x \in \mathbb{R}/\mathbb{C}$ exists with

$$
Ax = \lambda x \quad \text{and} \quad x \neq 0
$$

In this case $x$ is called the eigenvector belonging to the eigenvalue $\lambda$ [1].

The aim of the PCA is to reduce the multidimensional data to lower dimensions for an easier analysis. To apply the PCA, first the input sample vectors have to be defined. In this work the coordinates of the vertices are interpreted as a random/sample vector.

Every model consists of the vertices which are combined to triangles. These vertices are described in a Cartesian coordinate system, and their coordinates are available in the work flow. The reduction of the dimension is effected by transforming the data set basis in such a way that the new coordinate axes point in the direction of the variance of the data. These axes, also called principal components, provide information about the patterns in the data.
The data set is analysed by means of the PCA. Through this, the main characteristics can be described by means of only a few parameters (the mean vector $\mu$, the eigenvectors $\phi_i$ and the eigenvalues $\lambda_i$).

The PCA computes the most meaningful basis to remap a noisy or redundant data set of $m \times n$-dimensional sample vectors where $x_i = [x_1, \ldots, x_n]$.

Let us define a new $n \times m$ matrix $X$.

$$X = \begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix} \quad (4.4)$$

With this matrix containing the sample vectors the covariance matrix is defined as:

$$C_X = \frac{1}{n-1} XX^T \quad (4.5)$$

where each row corresponds with the measurements of a particular mesh, and each column of $X$ corresponds to a set of measurements of one particular characteristic. $C_X$ is an $m \times m$ matrix, the diagonals are the variances of individual models, and the covariance between different models is in the off-diagonal. $C_X$ captures all correlations between all possible pairs of meshes. An optimised covariance matrix would have only zeros in off-diagonal terms.

The change of the basis of $X$ is given through an orthonormal transformation matrix $\Phi$ with $Y = \Phi X$. The new basis has the columns $\phi_1, \phi_2, \ldots, \phi_m$. $\Phi$ is chosen so that $C_Y = \frac{1}{m-1} YY^T$ is diagonalised, the $\phi_i$ are the principal components. This diagonalised covariance matrix is an optimised matrix.

Furthermore the $\phi_i$ are equal to the eigenvectors of the covariance matrix $C_X$ and are called the eigenmodes of the PCA. The $i^{th}$ diagonal value of $C_Y$ is the variance of $X$ along $\phi_i$.

After this calculation all sample vectors (e.g. meshes) can be approximated with

$$x \approx \mu + \Phi b,$$

wherein $\mu = \frac{1}{m}\sum_{i=1}^{m} x_i$.

The first column of the matrix $\Phi = [\phi_1, \phi_2, \ldots, \phi_t]$ corresponds to the largest eigenvalue $\lambda_1$, and all eigenvectors $\phi_i$ of the matrix $\Phi$ are sorted with respect to $\lambda_i$ in descending order. The vector $b$ is a $t$ dimensional parameter vector. By varying the parameter vector $b$ new shapes similar to those in the data set can be generated. In the relevant literature [40] a limit of $\pm 3\sqrt{\lambda_i}$ is proposed to guarantee the shape similarity in the data set.

Figure 4.2 illustrates the idea of the PCA. The principal component $\phi_1$ points in the direction of the highest variance.

Figure 4.3 illustrates the first three highest modes of the pelvis model.
Figure 4.2: Illustration of the principal component analysis on a set of 2D data

Figure 4.3: Modes of the mesh of the pelvis
4.2 Model Geometry and Configuration

All models used in this work are represented by triangulated meshes consisting of vertices connected to form $T$ triangles. The models can be expressed mathematically as unidimensional vectors with $\mathbf{v} = (v_1^T v_2^T \ldots v_n^T)^T$, wherein a vertex is defined as $v_i = (v_{i,x}, v_{i,y}, v_{i,z})^T$.

![Figure 4.4: Wireframe of a femur](image)

First the main model is generated by manual segmentation of the volumetric image of the (male) patient. Thereby the anatomical objects (Figure 4.1) of interest, the pelvis, the femur and its cartilage as well as the cartilage in the acetabulum are converted into a binary volume; a binary volume has a binary value for each volume point. By modeling the mesh, the surface boundary voxels are connected to give triangles. Through this a deformable model is obtained for each anatomical object. The following equation gives the relation between the number of edges, triangles and vertices.

\[
T = 2V - 4 \\
E = 3V - 6
\]

(4.6)

Table 4.2 shows the number of the vertices and triangles for meshes of different bones.

<table>
<thead>
<tr>
<th>Mesh</th>
<th>Number of vertices</th>
<th>Number of triangles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Right pelvis mesh</td>
<td>5032</td>
<td>10064</td>
</tr>
<tr>
<td>Left pelvis mesh</td>
<td>5039</td>
<td>10078</td>
</tr>
<tr>
<td>Right femur mesh</td>
<td>3760</td>
<td>7516</td>
</tr>
<tr>
<td>Left femur mesh</td>
<td>3756</td>
<td>7508</td>
</tr>
<tr>
<td>Both right cartilages</td>
<td>2543</td>
<td>5086</td>
</tr>
<tr>
<td>Both left cartilages</td>
<td>2501</td>
<td>4998</td>
</tr>
</tbody>
</table>

Table 4.1: Numbers of vertices and triangles for different meshes

The cartilages which cover the femoral head and the inside of the acetabulum cannot be separated by manual segmentation. Therefore both cartilages are segmented as one object according to the MRI image. Figure 4.4 illustrates the vertices connected through edges to give triangles.
The mean edge length of the pelvis meshes is 3.7 mm. For the femur meshes the mean edge length is 2.9 mm. Both the cartilages have a mean edge length about 2.0 mm.

4.3 Coordinate System

The individual data are given in a Cartesian coordinate system which is used for describing voxels. These data have three views, each having a special name in medical images.

The axial view is the $x, y$ plane, the coronal view is the $x, z$ plane and the sagittal view is the $y, z$ plane.

The $x$ axis goes from left to right, and the $y$ axis from front to back. The $z$ axis goes from head to feet. The slice thickness is 0.8 mm. The resolution in $x$ and $y$ is 0.625 mm; in the $z$ direction it is 0.8 mm.
Chapter 5

Image Preprocessing

Image preprocessing is a common name for all operations on an image that precede image analysis. The aim of preprocessing is to improve the image features. Normally for preprocessing the image intensity is used as an input parameter. The output of the methods of image preprocessing is the changed image intensity. The image features are enhanced and the noise is suppressed by image preprocessing. Most of the methods have been developed to improve the intensity or to map the intensity to another intensity interval. The changed image intensity is usually calculated with respect to the neighbouring pixel or voxel.

An example for preprocessing is histogram analysis. A histogram plots the count of each grey value in the image.

The Image preprocessing methods used here are edge detection, image smoothing and thresholding.

In this chapter, first the neighbourhood of a pixel or a voxel is described in section 5.1, followed by the convolution section 5.2. The smoothing filter (Gaussian filter) is then described in section 5.3. Edge detection with the Sobel filter is next described. Finally, the last section 5.5 describes the use of a thresholding process to improve the response of the Sobel filter.

5.1 Neighbourhood of a Pixel or a Voxel

A voxel in a $3 \times 3 \times 3$ neighbourhood has three kinds of contact (Figure 5.1(a)).

- 6 voxels share a face with the central voxel
- 8 voxels share a point with the central voxel
- 12 voxels share an edge with the central voxel

The combination of these types defines the connectivity type of the voxel.

A 6-connectivity is defined by the face neighbours (Figure 5.1(b)). Face and edge neighbours define the 18-connectivity (Figure 5.1(c)) and, finally, face, edge and point neighbours define the 26-connectivity (Figure 5.1(d)) [16].

6-connectivity corresponds with 4-connectivity in 2D and 26-connectivity corresponds with 8-connectivity in 2D.
5.2 Convolution

Convolution can be defined as a group operation. Group operations calculate the new pixel or voxel from its neighbourhood (Section 5.1). The group operation is defined as a template convolution which has weighting elements. The template is placed at the pixel or voxel of interest. Through weighting the neighbourhood the weighted value of this pixel or voxel of interest is calculated.

Usually such kinds of templates \((3 \times 3)\) in 2D have the size of 3 pixels in width and 3 pixels in height. The template mask is square. An odd number of elements is preferred to ensure the positioning appropriately.

In 3D this template expands to a \(3 \times 3 \times 3\)-template with 3 voxels in width, 3 voxels in height and 3 voxels in depth.

A new value for the voxel of interest is calculated by multiplying the elements of the neighbourhood with the weighting coefficients of the template and summing them up.

In the beginning of the convolution process a squared filter is created. Each element in the squared filter contains a fixed number. The image consists of predefined grey values.

The filter has to be applied to the image. To calculate the filter result, each filter coefficient has to be multiplied with the grey value of the image which is covered by the filter element. By running the centre of the filter through all image elements the target image is completed.

This procedure loops until the end of the row, whereby the value of the row increases by one, usually on the \(y\)-axis. The convolution repeats itself until the end of image.

The convolution has its roots in an integral transform, decreasing several values into one value during the computation process. It is a particular kind of integral transform.

The mathematical definition of the convolution in the time domain is:
Figure 5.2: Convolution

\[(f * g)(t) \overset{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau) \cdot g(t - \tau) \, d\tau = \int_{-\infty}^{\infty} f(t - \tau) \cdot g(\tau) \, d\tau\]

where the sign (*) means the convolution.

While the convolution formula can be described as a weighted average of the function \(f(\tau)\) at the moment \(t\), the weighting function \(g(-\tau)\) is simply shifted by the amount \(t\). As \(t\) changes, the weighting function emphasises different parts of the input function.

The discrete convolution in the case of discrete signals (pixel or voxel) is defined as follows:

\[(f * g)[n] \overset{\text{def}}{=} \sum_{m=-\infty}^{\infty} f[m] \cdot g[n - m] = \sum_{m=-\infty}^{\infty} f[n - m] \cdot g[m]\]

5.3 Gaussian Filter

The neighbours of the pixel or voxel of interest are weighted equally by the convolution with an averaging filter [28]. Weighting equally can be disadvantageous with regard to smoothing the image, because the closer neighbours are more important than the peripheral neighbours; the distance of the neighbours becomes important. The direct averaging filter removes noise at the expense of losing features. [28]

The Gaussian filter belongs to the group of averaging operators. It considers the distance of the neighbours and weights them according to their difference to the pixel or voxel of interest. Through a Gaussian filter more features are retained. Therefore, the Gaussian filter can be considered to be an optimal filter for image smoothing (Figure 5.3). The formula is defined in 3D as:

\[g(x, y, z, \sigma) = \frac{1}{2\pi\sigma^2} e^{(-\frac{x^2+y^2+z^2}{2\sigma^2})}\]

The formula 5.1 calculates the coefficients for a Gaussian filter mask. Convolving the image with the Gaussian filter leads to a smoothing of the image. Usually the Gaussian filter mask is square, \(3 \times 3\) in 2D. The variable \(\sigma\) controls the variance. A small value of the variance leads to a narrow Gaussian error distribution curve.
Therefore only few neighbour pixels or neighbour voxels have an effect on the smoothing. The size of the Gaussian filter mask can be calculated depending on the variance $\sigma$. Thus, with a given value for the variance the size of the filter mask can be determined.

$$\text{filter size} = 2k + 1 \quad 2\sigma \leq k \leq 3\sigma$$  \hspace{1cm} (5.2)

Equation 5.2 gives a way to calculate the filter size. It considers the relation between the proximate and the peripheral neighbours.

For $\sigma = 1$ the filter size is $7 \times 7$ in 2D, as shown in figure 5.3(b), while $\sigma = 0.4$ leads to a filter size $3 \times 3$ in 2D as shown in figure 5.3(c).

Figure 5.3: Gaussian Template

### 5.4 Edge Detection with Sobel Filter

An edge is defined as an elongated region in an image with an abrupt change of the intensity values in the short direction and comparatively small intensity variations in the elongated direction. Edge detection belongs to the type of algorithms which try to localise and highlight contours to discriminate between objects and regions. This basic process is a low level stage in the vision process.

In many scientific disciplines edge detection is used to improve the quality of further processing stages. Mostly, the quality of the final estimation depends on the quality of the earlier estimation, which has been done with the edge detection described here.

Edge detection is a fundamental problem in image analysis. In typical images, the edges characterise object boundaries and are therefore useful for segmentation, registration, and identification of objects.

The Sobel filter is one of the filters most frequently used for edge detection in image analysis [8], [3]. It is a linear filter belonging to the group of highpass filters and has particular filters in each direction. The Sobel filter aims to give a high response on edges which runs in the direction for which the particular filter is filtering, and low response on edges in other directions or on homogeneous regions. It is better in highlighting edges in low contrast areas.
Figure 5.4: Sobel filter

Figure 5.4 illustrates the sobel filtered image and the results of all particular filters. The cross section of an edge has the shape of a ramp. An ideal edge is rooted in a discontinuity with an infinite slope. The first derivative of the image has a local maximum at an edge. For a continuous 3D image with the coordinates \( x, y, z \in \mathbb{R} \), its gradient can be determined as:

\[
\nabla f(x, y, z) = \left( \frac{\delta f}{\delta x}, \frac{\delta f}{\delta y}, \frac{\delta f}{\delta z} \right)
\]

The magnitude of the gradient is:

\[
m(x, y, z) = \sqrt{\left( \frac{\delta f}{\delta x} \right)^2 + \left( \frac{\delta f}{\delta y} \right)^2 + \left( \frac{\delta f}{\delta z} \right)^2}
\]

The Sobel filter combines edge detection filters in the \( x, y, z \) directions and simultaneously averages grey levels. The smoothing filter used by the Sobel filter is a binomial filter. This belongs to the same family of filters as the Gaussian filter. It is in the case of an \( 3 \times 3 \) kernel identical to the Gaussian filter of the same size. The coefficients of the binomial filter are equal to the binomial coefficients constructed with Pascal’s rule (Pascal’s triangle) [1].

The Sobel filter calculates the first derivative in the \( x, y, z \) directions, and smooths in the orthogonal direction. The coefficients of the kernels of the Sobel filter are
taken from the relevant literature [3]. The kernels are given in figure 5.5; they are defined as follows:

In this study, particularly these kernels are used in the work flow. Also, the Sobel filter provides the basic gradient image for the Hough transform [35] in section 6.

### 5.5 Conversion of a Grey Level Image into a Binary Image with a Thresholding Process

The result of the Sobel Filter is a grey level image. Abrupt changes of the intensities of neighbouring pixels lead to high gradient value at this point. Using the a priori knowledge about the image, the less important anatomical structures can be removed, as the intensity levels of an anatomical structure have characteristic grey level intervals. Edge reduction is used to suppress unwanted anatomical structures. It can be considered as a thresholding process [4] [3] [28]. A typical threshold value can be learned from the data set. Through the thresholding process a binary image is created in which only the high value intensities remain. A binary image is an image which has only the intensities 0 or 1. In most cases the 1 represents white and the 0 represents black, but this definition is not fixed.

Suppressing less important structures speeds up the Hough transform computation time. Figure 5.6 illustrates the edge reduction for a given data set. On the left side the output image of the Sobel Filter is shown, and on the right side the thresholded images. Note that the image in all subfigures is the same, but shown in different planes: axial, coronal and sagittal.
Figure 5.6: Sobel filtered image
Chapter 6

Object Detection Using the Hough Transform

This chapter describes how to find one of the eight anatomical structures of interest in images so as to preposition the corresponding model to the individual anatomical structure, achieving an improvement of the results of the model-based approach (Section 8). To this end an algorithm, the Hough transform, is discussed, which finds the relevant shape in the image and subsequently prepositions the model in relation to the anatomical structure of the patient. Since in most applications data sets are successfully adapted if the models are properly prepositioned, the prepositioning of the model in the image of the patient as described here achieves an improvement of the model-based segmentation.

Normally, prepositioning is one of the first steps to automate the model-based approach. But prepositioning is challenging due to shape variability and differences in the alignment of the pelvis and the femur in the anatomy of different patients. Usually the rotation of pelvis and femur vary strongly from patient to patient. The particular anatomical form also depends to a large extent on the degree of hip dysplasia.

First the anatomical structure which can be most easily found in an image of the patient has to be determined. The structures to be examined are the various parts of pelvis, femur and cartilage. The pelvis cannot be taken as its complex form precludes description in simple mathematical formulae, while the cartilage is difficult to identify in the images. This leaves the femur. Since the femoral head has a spherical structure, this is most easily identifiable. Therefore, the Hough transform for spheres is employed to first localise the femoral head in the image. Figure 6.1(b) clearly illustrates the anatomical structure of a femoral head.

In this work a sphere is represented in a discrete form, because the images consist of discrete voxels. Rather than viewing the sphere as a whole and calculating its surface points with a 3D formula, its representation here depends on the Bresenham algorithm for circles. A discrete sphere can be represented as several circles layered above each other, as is shown in Figure 6.1(a).

The circles are approximated with the help of the Bresenham algorithm for circles. This chapter is organised as follows: The first section describes the Hough trans-
6.1 Hough Transform

The Hough transform is a well-known technique to locate specific shapes in images. In particular, it has been used for detecting lines, circles and ellipses. The classical Hough transform is only applicable to shapes which can be described in a parametrical form. The generalised version of the Hough transform [10] can be used for shapes which cannot be described by simple analytical terms.

The algorithm was originally introduced by Hough [30] in 1962 to locate bubbles, but not to find shapes, in images. In various publications [33] the Hough Transform (HT) is used for curve detection. Fast calculation of the candidate parameters is the important advantage of the Hough transformation as compared to other algorithms yielding similar results, like ”template matching” ([37], Princen et al. [31]).

Since, as mentioned above, specific shapes in images of patients can be localised with the HT, the input of the HT is an image. For each type of shape a specialised HT algorithm has to be implemented.

Using the a priori knowledge about the image, the grey values of the less important anatomical structures are removed, as the intensity levels of an anatomical structure have characteristic grey level intervals. Within the thresholding process a binary image is created, wherein only the high valued intensities calculated with the Sobel Filter (see section 5.4) are retained. In section 5.5 the thresholding process is explained in detail. This binary image is used as the input image.
The binary image is given in Cartesian coordinates and contains only ones and zeros, ones being the pixels lying on a contour. The shape has to be described in the same coordinate system as the input image. The parameters of the shape are its coordinates, but it can also have additional specific parameters. The points belonging to the shape are called the contour points.

The output of the HT is called accumulator space. The accumulator space has the dimension of the number of the parameters describing the shape. The accumulator space contains integer values, called votes, indicating the presence of the shape in the image.

The HT maps an image into the accumulator space. For each point in the image with value one, all shapes whose parameters contain the coordinates of this point are regarded. Note that the indicator image value can be chosen individually. For each of these shapes the values corresponding to its contour points are incremented in the accumulator space. Finally, a peak in the accumulator space is an indicator for the shape.

6.2 Classical Hough Transform for Circles

In this section the classical HT for circles as given in figure 6.2 is described in detail, as it illustrates very well the basic concept of any form of the HT. Furthermore, it allows straightforward generalisation to a 3D HT for spheres, which is employed in the application that is the central subject of this work [9] [15].

The HT for circles, which is sensitive only for these, is able to find circles in images if the points lie on the arc of the circle, irrespective of its radius. The input parameter of the HT for circles is a binary image. The output is the so-called accumulator space.

Let $w$ be the width and $h$ the height of the image. The binary image can be considered as a matrix of two intensities, $I \in \{0, 1\}^{h \times w}$. Note that the matrix notation is generally defined as (row × column), whereas an image is described as (column × row). In the given example in this section, the matrix notation is used.

In a 2D Cartesian coordinate system, a circle with centre $(x_0, y_0) \in \mathbb{R}^2$ and radius $r \in \mathbb{R}$ has the contour points $(x, y) \in \mathbb{R}^2$.

As the HT for circles works on discrete data, the coordinates for the circle are now $(x, y) \in \mathbb{Z}$ and $(x_0, y_0) \in \mathbb{Z}$. The radius also has to be given in discrete values. The set of all considered radii is $R = \{r_0 \ldots r_n\}, r_i \in \mathbb{N}, i, n \in \mathbb{N}$.

As mentioned above, the output of the HT is an accumulator space. In this example the accumulator space is defined as $A \in \mathbb{N}^{h \times w \times |R|}$.

The following formula is now used to describe the Hough Transform for the circle, whose shape is defined as $S(p_1, p_2, p_3)$, with $p_1 = w, p_2 = h$ and $p_3 \in |R|$:

$$Hough : \{ f_{S(p_1, p_2, p_3)} : \{0, 1\}^{w \times h} \rightarrow \mathbb{N}^{h \times w \times |R|} \quad (p_1, p_2, p_3) \quad \mapsto \#((w, h) \in I : S(p_1, p_2, p_3)(w, h) = 1) \}$$

(6.1)

This function 6.1 defines how an image is mapped onto the accumulator space.
In the following the HT for circles is mathematically deduced step by step. First the duality between the circle centre and its contour points is described through the definition of a circle 6.2. Using the duality, the points are then mapped onto the accumulator space. Finally the algorithm is explained with a given figure as example.

The circle can be described either in a parametric (6.3) or in an explicit form (6.2). The formula for a circle of radius \( r \in \mathbb{R} \) centred at \((x_0, y_0) \in \mathbb{R}^2\) in Cartesian coordinates is given by:

\[
(x - x_0)^2 + (y - y_0)^2 = r^2
\]  

(6.2)

Note that this formula 6.2 describes a continuous circle in \( \mathbb{R}^2 \). The \((x, y) \in \mathbb{R}^2\) describe the contour points of a circle of radius \( r \). The radius is assumed as a fixed value in the following explanations.

Figure 6.1: Duality between \((x, y)\) and \((x_0, y_0)\)

Equation 6.2 can be interpreted in two ways:

(i) For a fixed centre \((x_0, y_0)\) of a circle the interpretation contains all points \((x, y)\) within the distance of \( r \) belonging to the contour of the circle. In figure 6.2 the centre point \((x_0, y_0)\) is red and lies in the middle of the dotted circle. The radius \( r \) is dashed. All points \((x, y)\) lie in the dotted circle.

(ii) A fixed contour point \((x_0, y_0)\) belongs to all circles with centre point \((x, y)\) at a distance \( r \) to \((x_0, y_0)\). In figure 6.2 the centre point \((x, y)\) lies in the right blue circle and is green, while the contour point \((x_0, y_0)\) is red and lies in the dotted circle.

These two cases describe a duality that will be exploited in the following.

The HT algorithm evaluates the elements of the image matrix \( I \). When the algorithm starts, the accumulator is initialised with zeros. Each value "one" in \( I \) is considered as a contour point \((x, y)\). For each feature point in the image the
votes of all circles containing \((x, y)\), as described in case (ii), are incremented in the accumulator space.

Note that all these values belong to the contour of a circle in the accumulator space, e.g. the indices of the values can be calculated with a circle algorithm presented in section 6.3.

\begin{equation}
\begin{align*}
x &= x_0 + r \cos(\varphi) \\
y &= y_0 + r \sin(\varphi)
\end{align*}
\end{equation}

The advantage of formula 6.3 is to allow every contour point of the circle to be calculated. Therefore the formula can be defined for the parameters of a circle as
follows:

\[ x_0 = x - r \cos(\varphi) \]
\[ y_0 = y - r \sin(\varphi) \]

These equations define the points in the accumulator space. Note that \( \varphi \) is a fixed variable. \( \varphi, (0 \leq \varphi \leq 2\pi) \) defines the trace of the curve [28].

(a) Binary input image, containing the target shape to be localised by HT

(b) Accumulator space or Hough space

Figure 6.3: A binary image is mapped into the accumulator space

The example given in figure 6.2 provides a HT for circles. For the example that is shown in figure 6.3(a) a fixed radius is assumed. Figure 6.3 illustrates the application of the HT; the result is the accumulator space in figure 6.3(b).

The accumulator space is completed when all elements in \( I \) are evaluated. Matrix \( I \) represents the binary image in figure 6.3(a). The whole accumulator space, also called Hough space, is shown in figure 6.3(b). This space is given in 2D because the shape is described in 2D and the radius is a fixed value. Every contour point in the binary image 6.3(a) corresponds with an integer value, i.e. vote, at these coordinates, and these are assumed to be the parameter set of a circle in case of the HT for circles.

Figure 6.3(b) represents the accumulator space of the binary image in figure 6.3(a). This accumulator space’s coordinates have votes and are coded by colour. The dark blue values indicate that the possibility for a circle with these parameters in the image is low. The red colour values indicate that the possibility for a circle with these parameters in the image is high. The peak in the figure 6.3(b) corresponds to the parameters of the circle in the input binary image 6.3(a). In this case the peak or the most incremented value is dark red and indicates the parameters for the circle in the binary image.
With this, the basic concept of HT for circles is given and the algorithm mathematically explained. For an implementation of the classical HT for circles the circle will be approximated with the Bresenham algorithm (6.3).

### 6.3 Bresenham Algorithm for Circles

As mentioned in the previous section, the HT algorithm for circles needs a parameterisation of the circle, defined by its centre point \((x_0, y_0)\) and the radius \(r\). Usually the circle contour points can be represented for the Cartesian coordinates \(x, y\) with:

\[
    x = x_0 + r \cos(t_i), \quad y = y_0 + r \sin(t_i),
\]

where \(0 \leq t_0 \leq t_1 \leq \ldots \leq t_n \leq 2\pi\). But this representation of circle contour points, based on floating points and trigonometry, leads to a slowdown of computation time.

To speed up computation time integer arithmetic is preferred. In 1977 Jack E. Bresenham for the first time introduced a circle drawing algorithm (BCA) for incremental digital display of circular arcs [21]. The aim at that time was to optimise graphic algorithms for basic objects, e.g., lines and circles. In other words, the aim was to represent graphics on displays.

The circle algorithm complements an earlier line algorithm described in [5] and [6]. Bresenham describes the methodology for calculating the points closest to the true circle contour points. The proof of Bresenham’s method is given in the paper [21]. The BCA approximates the locus of the contour points of a circle on a given voxel grid.

In the following it is described how the contour points of a circle are calculated by the BCA. However, this will not be employed to plot a circle on any graphical device. Rather, the points are calculated internally and stored by the BCA for the specialised HT for circles.

The BCA is called an incremental algorithm because the position of the next pixel or voxel is calculated on the basis of the last calculated one. The repetitive incremental stepping loop for point selection requires only addition, subtraction and sign testing; neither quadratic nor trigonometric evaluations are required.

For determining a circle the BCA needs the centre point \((x_0, y_0)\) and the radius \(r\) as input parameters. But the following description of the BCA considers only the first quadrant and describes how the BCA works in this quadrant, since due to the symmetry of the circle one can deduce the whole circle from one quadrant.

The first quadrant is calculated by the approximate circle contour points on the positive x and y axes with:

\[
    (x - x_0)^2 + (y - y_0)^2 = r^2, \quad (x, y) \in \mathbb{Z} \text{ for } (x_0, y_0) \in \mathbb{Z}, \quad r \in \mathbb{N}.
\]

Further, because of the symmetry of the \(x\) and \(y\) and the 45 degrees diagonal axes, the BCA is here used to calculate only the coordinates of the contour points in the first 45 degrees.

The starting point is at \((x_{\text{start}}, y_{\text{start}}) = (0, r)\). BCA begins to approximate the contour points at these points until it reaches the coordinates \((x_{\text{start}}, y_{\text{start}}) = (r, 0)\). Each point is deduced from its predecessor.
Figure 6.4: The blue contour points are generated with $x = x_0 + r \cos(t_i), y = y_0 + r \sin(t_i)$, with $x_0 = 0, y_0 = 0$ and $0 \leq t_0 \leq t_1 \leq \ldots \leq t_n \leq 2\pi$. Further, the four quadrants of a circle in Cartesian coordinates are illustrated; $(x_{\text{start}}, y_{\text{start}}) = (0, r)$ indicates the starting point and the red arrow shows the work flow direction.

Figure 6.4 illustrates the regions of the quadrants and the initial setup. Note that the centre point of the circle is assumed as $(x_0, y_0) = (0, 0)$ and $r = 30$.

At each step BCA chooses a point so as to minimise the squared difference between $r^2$ and the square of the radius to that point, wherein the chosen point belongs to the approximate contour and is declared as $x$ and $y$, and $r$ is the given radius. The aim is to obtain an algorithm which uses simple arithmetic terms, i.e addition and subtraction. The algorithm for the calculation of the points should be executable iteratively. The concept of the BCA is to simplify these comparisons of the distances from the approximated contour points to the true circle points with respect to a simple calculation.

Beginning at the given point $(x, y) = (0, r)$, the approximated point can lie only either to the right of this point, i.e $(x+1, y)$, diagonal to this point, i.e $(x+1, y-1)$, or below this point $(x, y-1)$, as illustrated in figure 6.5(b), because the work-flow direction is clockwise in the first quadrant and all values are integer values.

To approximate the true circle arc all of these approximated contour points have to be evaluated. Therefore the squared distances of the approximated contour points to the true circle points are measured. But this evaluation needs square operation and has to evaluate the absolute value, too. This increases the computation time. To speed up computation time, this threefold consideration can be avoided. Intuitively choosing the approximated diagonal point as the beginning of the evaluation, the further consideration is reduced to a consideration of only two of these points. Either the distance of the true circle contour point to the diagonal point and to the point to the east, or the distance of the true circle contour point
to the diagonal point and to the point to the south have to be considered.

In both cases the decision on which side has to be considered is made by comparing
the sign of the squared distance of the diagonal point. The given error function
measures the squared distance of an approximated contour point.

The error function $P(x, y)$ is given by:

$$P(x, y) = x^2 + y^2 - r^2 \quad (6.4)$$

Further, the sign of the error function (6.4) provides information about the position
of the considered contour point.

$$P(x, y) = \begin{cases} 
\leq 0 & \text{x, y lies inside the circle} \\
> 0 & \text{x, y lies outside the circle} 
\end{cases} \quad (6.5)$$

Starting at the point $(x_{start}, y_{start}) = (0, r)$ in the first quadrant, $y$ is a monotonically
decreasing function of $x$. Clockwise movement in this quadrant can therefore
be accomplished by a sequence of moves involving only $m_0, m_1, m_2$: east, south-east
or south, as illustrated in figure 6.5(b).

Figure 6.5 illustrates the main idea of the BCA and figure 6.5(b) represents the
three possibilities of movement. By this means the true circle arc is approximated
through the evaluation of these movements by the BCA. In figure 6.5(c) the true
circle lies between the points $(x_i+1, y_i)$ and $(x_i+1, y_i-1)$. As this figure illustrates,
the true circle is closer to one of the approximated contour points; the closest
approximated point is at $(x_i+1, y_i)$, and therefore the BCA chooses this point and
as a result the movement $m_0$.

Similarly, the movement $m_1$ can be described. The possible position of the true
circle is illustrated in figure 6.5(d). If the true circle arc is closer to the approxi-
mated contour point $(x_i+1, y_i-1)$ on the voxel grid, then this point will be
chosen.

The movement $m_2$ is shown in figure 6.5(e). Here the true circle arc lies closer to
the point $(x_i, y_i-1)$.

The BCA chooses one of the successor points (i.e. the grey points in figure 6.5(c))
whose square distance is minimally distant from the true circle arc. By determining
the minimum of the following quantities the minimal squared distance can be found
within three possibilities. The absolute distance value for each possible movement
is given by the formula:

$$\begin{align*}
| (x_i + 1)^2 + y_i^2 - r^2 |, & \quad \text{for } m_0 \\
| (x_i + 1)^2 + (y_i - 1)^2 - r^2 |, & \quad \text{for } m_1 \\
| x_i^2 + (y_i - 1)^2 - r^2 |, & \quad \text{for } m_2 
\end{align*} \quad (6.6)$$

This threefold evaluation is modified to a twofold evaluation. The aim is to simplify
the whole evaluating process to a consideration of the sign, and to find another
way to calculate these formulae (6.6) by using only addition and subtraction op-
erators. The BCA simplifies it to the consideration of two points per step by first
observing the sign of the error $P$ function 6.5. Further, the successor points can
(a) The regions A and B indicate the movements \( m_0 \) and \( m_1 \); in A and in B \( m_1 \) and \( m_2 \) are possible.

(b) Three possibilities of movement: the red arrows show the direction of the movements: east, south-east or south.

(c) The first movement \( m_0 \) is to the east and is defined as \( x_{i+1} = x_i + 1, y_{i+1} = y_i \).

(d) The second movement \( m_1 \) is to the south-east is defined as \( x_{i+1} = x_i + 1, y_{i+1} = y_i - 1 \).

(e) The third movement \( m_2 \) is to the south and is defined as \( x_{i+1} = x_i, y_{i+1} = y_i - 1 \).

Figure 6.5: Movement possibilities (grey points) in each subfigure c, d and e for each successor of a point. The black bordered circle in subfigures c, d and e indicates the next point which is going to be chosen. Each subfigure c, d and e demonstrates a possible direction of movement.
be determined from their predecessor points iteratively. This method is highly efficient and decreases computation time.

At this point a new function is established which is called $\Delta$. This function evaluates the approximated contour points and delivers a value which can be positive, negative or zero. This evaluation isolates two points for further consideration. Considering the movement $m_1$ at the approximated point $(x_i + 1, y_i - 1)$, the function $\Delta$ may have a positive value, a negative value or be zero. If $\Delta$ is negative, only the movements $m_0$ and $m_1$ have to be considered. If $\Delta > 0$, only the movements $m_1$ and $m_2$ have to be considered.

Figure 6.6: The five possibilities for the circle’s intersection with the coordinate lines $x_i + 1$ and $y_i - 1$ are shown.

Figure (6.6) shows the five intersection possibilities of a circle with the coordinate lines of $x_i + 1$ and $y_i - 1$, which must be considered when choosing the next movement from the point $x_i, y_i$. The negative value indicates that the true circle arc can only intersect the coordinate lines of $(x_i + 1, y_i - 1)$ at intersection no. 1 or intersection no. 2. In the same manner, the positive value of $\Delta_i$ indicates intersections no. 4 and no. 5.

The error function for the diagonal movement $m_1$ in figure 6.5(d) is defined as:

$$\Delta_i = P(x_i + 1, y_i - 1) = (x_i + 1)^2 + (y_i - 1)^2 - r^2$$  (6.7)

The algorithm begins with evaluating the sign of the function $\Delta_i$. Based on the result the next possible point with a minimum square distance to the true circle arc is chosen.

A. First the case is considered in which $\Delta$ is negative.

If $\Delta_i < 0$, then the diagonal point $(x_i + 1, y_i - 1)$ lies inside the true circle. Necessarily, the point $(x_i, y_i - 1)$ also lies in the true circle, because this point is on the west of the diagonal point and its $x$ coordinate is 1 less than the $x$ coordinate of the diagonal point. Therefore the value of the error function $P$ for movement $m_2$ is even smaller than the value of the diagonal point. According to this analysis only the movement $m_0$ or $m_1$ needs to be considered, as the true circle can only lie...
Bresenham Algorithm for Circles

between either \((x_i+1, y_i)\) and \((x_i+1, y_i-1)\), i.e. intersection no. 2, or \((x_i+1, y_i+1)\) and \((x_i+1, y_i)\), i.e. intersection no. 1.

If we consider intersection no. 2 between \((x_i+1, y_i)\) and \((x_i+1, y_i-1)\), then the closest point to the true circle arc can be determined by evaluating the difference between the absolute error values, which can now be established as a new function \(\delta\).

The newly established function \(\delta\) is defined as \(\delta_i := 2\Delta_i + 2y_i - 1\). Similar to the function \(\Delta\), the new function measures the squared distances of the approximated contour points to the true circle contour points. Note that \(\delta\) considers only the absolute squared difference of the two points which are isolated by the function \(\Delta\), and calculates the absolute squared difference to the true radius which is evaluated with the error function \(P\).

Employing these points with \(\delta_i\) it follows:

\[
\delta_i = |P(x_i+1, y_i)| - |P(x_i+1, y_i-1)| = |(x_i+1)^2 + y_i^2 - r^2| - |(x_i+1)^2 + (y_i-1)^2 - r^2| \quad (6.8)
\]

As in the case of the function \(\Delta\), the function \(\delta\) can be simplified to a sign consideration, too. By omitting the absolute values the terms (a) and (b) in the formula 6.9 can be positive or negative.

\[
\delta_i = \underbrace{(x_i+1)^2 + y_i^2 - r^2}_a - \underbrace{(x_i+1)^2 + (y_i-1)^2 - r^2}_b \quad (6.9)
\]

Now the terms (a) and (b) in formula 6.9 have to be evaluated. The absolute value of the first term can be positive or negative.

Considering the case \(\Delta_i \leq 0\) for the second term (b) in formula 6.9, the absolute value of the approximated point at \((x_i+1, y_i-1)\) is negative or zero, as assumed at the beginning.

Employing this information and evaluating \(\delta_i\), it follows:

\[
\delta_i = \begin{cases} \underbrace{(x_i+1)^2 + y_i^2 - r^2}_a & \text{if } a < 0 \\ \underbrace{(x_i+1)^2 + (y_i-1)^2 - r^2}_b & \text{if } b \leq 0 \end{cases}
\]

\[
\delta_i = - \left\{ (x_i+1)^2 + y_i^2 - r^2 \right\} + \left\{ (x_i+1)^2 + (y_i-1)^2 - r^2 \right\}
\]

\[
\delta_i = -2y_i - 1 \quad (6.10)
\]

First the negative case for both terms will be considered.

The bottom line of 6.10 reveals that for \(a < 0\) whole expression for \(\delta_i\) is negative, since \(y_i\) is assumed to be in the first quadrant (i.e. \(y_i \geq 0\)). Consequently, the absolute value of term (a) is smaller than the absolute value of term (b). This tells that the point \((x_i+1, y_i)\) is closer to the true circle arc than the diagonal point \((x_i+1, y_i-1)\). Hence, the BCA decides in favour of the movement \(m_0\).
Now the positive case for the first term (a) will be considered, while the second term (b) is still negative as assumed at the beginning. When the absolute value is positive, the $\delta$ function in the formula 6.8 becomes

$$
\delta_i := 2\Delta_i + 2y_i - 1
$$

by omitting the absolute values.

Through this formula the sign of the function $\delta$ is determined. $\delta_i$ calculates the difference between the squared distances of the point at $(x_i + 1, y_i - 1)$ to the true circle arc, and the point at $(x_i + 1, y_i)$ to the true circle arc. In other words, the BCA has to choose between the movements $m_0$ and $m_1$, and take the one that is closer to the true circle arc, when $\Delta_i$ is negative.

Summing up the result of $\delta$ in the formula 6.11, we see that in the first case both terms are negative. This leads to the movement $m_0$. In the second case the first term is positive. This leads to the movement $m_1$. Consequently, $\delta_i$ obtained in the formula 6.11 is positive in the positive case. Therefore the movement $m_1$ is chosen. The true circle arc can thus intersect the coordinate lines at the circle’s intersection no. 2.

$$
\delta_i = \begin{cases}
\leq 0 & \text{movement for } m_0 \\
> 0 & \text{movement for } m_1 
\end{cases}
$$

B. Now the case is considered when the function $\Delta_i$ is positive.

If $\Delta_i > 0$, then the diagonal point $(x_i + 1, y_i - 1)$ lies outside the true circle, such that the point $(x_i + 1, y_i)$ does not need to be considered, as the error function for that point will have an even larger positive value. Instead, it has to be determined which of the two points given by $(x_i + 1, y_i - 1)$ and $(x_i, y_i - 1)$ is closer to the true circle arc.

The function $\delta'_i$ is introduced for this analysis. In the same manner as the function $\delta_i$, it calculates the difference between the squared distances of the point at $(x_i + 1, y_i - 1)$ to the true circle arc, and the point at $(x_i, y_i - 1)$ to the true circle arc; the calculated value can be positive, negative or zero. In this case the true circle passes either between the points $(x_i + 1, y_i - 1)$ and $(x_i, y_i - 1)$, or between $(x_i + 1, y_i)$ and $(x_i - 1, y_i - 1)$, giving the intersections no. 4 and no. 5. Thus only the movement $m_1$ or $m_2$ has to be considered.

Using the same methods as above the difference $\delta'_i$ is obtained, where $\delta'_i = 2\Delta_i - 1 - 2x_i$;
The positive case of both terms is evaluated first. If both terms \((a)\) and \((b)\) are positive,
\[
\delta_i' = \left\{ \left( x_i + 1 \right)^2 + \left( y_i - 1 \right)^2 - r^2 \right\} - \left\{ x_i^2 + \left( y_i - 1 \right)^2 - r^2 \right\}
\]
\[
\delta_i' = \frac{\left\{ x_i^2 + \left( y_i - 1 \right)^2 - r^2 \right\} - \left\{ x_i^2 + \left( y_i - 1 \right)^2 - r^2 \right\}}{b > 0}
\]
\[
\delta_i' = 2x_i + 1
\]

\(\delta_i'\) is positive too. This means that the point at \((x_i + 1, y_i - 1)\) also lies outside the circle, but is simultaneously the closest point to the true circle arc. Therefore, if \(\delta_i' > 0\), the movement \(m_2\) will be chosen.

Now the second term \((b)\) is considered as negative, the first term \((a)\) still being positive.
\[
\delta_i' = \left\{ \left( x_i + 1 \right)^2 + \left( y_i - 1 \right)^2 - r^2 \right\} - \left\{ x_i^2 + \left( y_i - 1 \right)^2 - r^2 \right\}
\]
\[
\delta_i' = \frac{\left\{ x_i^2 + \left( y_i - 1 \right)^2 - r^2 \right\} - \left\{ x_i^2 + \left( y_i - 1 \right)^2 - r^2 \right\}}{a < 0}
\]
\[
\delta_i' = 2\Delta_i - 2x_i - 1
\]

Because the sign of \(\delta_i'\) is negative, the approximated contour point \((x, y_i - 1)\) lies inside the true circle. Therefore only the movement \(m_1\) remains. So, if \(\delta_i' \leq 0\), the movement \(m_1\) will be chosen.

The analysis for \(\delta_i\) may be summed up as follows:
\[
\delta_i' = \begin{cases} 
< 0 & \text{movement for } m_1 \\
> 0 & \text{movement for } m_2 
\end{cases}
\]

C. The last case is now considered, in which \(\Delta_i\) is zero.
If \(\Delta_i = 0\), then the diagonal point \((x_i + 1, y_i - 1)\) lies on the true circle, so the movement is \(m_2\).
Based on this analysis of the three possible values of \(\Delta_i\) and the binomial theorem, it is observed that the coordinates \(x_i, y_i\) and \(\Delta_i\) have the following recurring relations:

- for the movement \(m_0\) (\(\Delta_i < 0\) and \(\delta_i \leq 0\)):
  \[
x_i+1 := x_i + 1 \\
y_{i+1} := y_i \\
\Delta_{i+1} := \Delta_i + 2x_{i+1} + 1
\]

- for the movement \(m_1\) (\(\Delta_i \leq 0\) and \(\delta_i > 0\) or \(\Delta_i \geq 0\) and \(\delta_i' \leq 0\)):
  \[
x_i+1 := x_i + 1 \\
y_{i+1} := y_i - 1 \\
\Delta_{i+1} := \Delta_i + 2x_{i+1} - 2y_{i+1} + 2
\]
• for the movement $m_2$ ($\Delta_i > 0$ and $\delta_i' > 0$):
  \[ \begin{align*}
  x_{i+1} &:= x_i \\
  y_{i+1} &:= y_i - 1 \\
  \Delta_{i+1} &:= \Delta_i - 2y_{i+1} + 1
  \end{align*} \]

The description of the BCA is now finished. In accordance with the given mathematical definition of the BCA, it can now be used for drawing a circle.

For the first quadrant of the circle in figure 6.4 this BCA can calculate only the three movements $m_0, m_1, m_2$. Though for the remaining quadrants it is necessary to define other possible movements, by using the geometrical advantages of a circle, i.e. the symmetry of the $x$ axis, the symmetry of the $y$ axis and the symmetry of the diagonal axis, the whole approximated circle contour points can be calculated with the BCA.

Figure 6.7 illustrates a circle drawn with the BCA overlaid with a circle generated by the trigonometrical formula. In this study the circle is drawn for the first 45 degrees clockwise starting at the point $(x, r) = (0, r)$, as the figure illustrates. By using the symmetry to the $x$-axis and the $y$-axis, and to the 45 degrees diagonal axis, the circle has been completed. Only the movements $m_0$ and $m_1$ in the eighth of the circle (A) need to be considered.

The algorithm 1 summarises a BCA in a pseudo code. In this work the BCA stores the contour points coordinates internally.

Starting the BCA for drawing a circle at $(x_{\text{start}}, y_{\text{start}}) = (0, r)$, the function $\Delta_0$ is
initialised by:
\[
\Delta_0 = (x_{\text{start}} + 1)^2 + (y_{\text{start}} - 1)^2 - r^2 \\
= 1 + (r - 1)^2 - r^2 \\
= 1 + r^2 - 2r + 1 - r^2 \\
= 2 - 2r
\]

\(
\delta \) and \( \delta' \) are defined as:
\[
\delta_0 = ((x_i + 1)^2 + y_i^2 - r^2) + \Delta_0 \quad \text{signed difference}
\]
\[
\delta'_0 = ((x_i)^2 + (y_i - 1)^2 - r^2) + \Delta_0 \quad \text{signed difference}
\]

\[
\delta_0 = [(x_{\text{start}} + 1)^2 + y_{\text{start}}^2 - r^2] + \Delta_0 \\
= 0 + 1 + \Delta_0 \\
\delta'_0 = [(x_{\text{start}})^2 + (y_{\text{start}} - 1)^2 - r^2] + \Delta_0 \\
= -2r + 1 + \Delta_0
\]

In each step \( x_i \) and \( y_i \) is incremented by 2 (e.g. 2\( x_{\text{start}} + 1 \) and \( -2y_{\text{start}} + 1 \)).

6.4 Using the Bresenham Algorithm for Circles to Generate Spheres

The femoral head has an approximately spherical form. The aim is to calculate the sphere surface points simply, without using trigonometrical and 3D formulae. To achieve this the calculation of the surface points is led back to a calculation in simple arithmetic. To this end the knowledge obtained from the Bresenham algorithm for circles (BCA) is used. In this study, the sphere is not considered as a whole sphere; rather, it is considered as a conglomerate of circles layered parallelly on top of each other. These circles are determined with BCA, which has been deduced for circles in the previous section (6.3).

Figure 6.8 illustrates the surface rendering of the right femur. Figure 6.8(c) illustrates a sphere of several layered circles for a better overview.

The 3D formula for creating a sphere is given in the following formula
\[
(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = r^2
\]
with the centre point at \((x_0, y_0, z_0)\) and radius \(r\). All points \((x, y, z)\) are called surface points. These surface points are now approximated by BCA, and they are discrete.

The sphere generating algorithm using BCA, or SBCA, builds the sphere using circles layered parallelly on top of each other in the \(x, y\) plane.
Algorithm 1 BresenhamCircle($x_0, y_0, radius$)

**Input:** Centre of the circle $x_0, y_0$, radius $radius$

**Output:** Plotting the contour points $x, y$

compute the error $\Delta_0 = 1 - radius$

$\delta_0 = 0$, $\delta'_0 = -2*radius$

$x = 0$

$y = radius$

$i = 0$;

plot($x_0, y_0 + radius$)

plot($x_0, y_0 - radius$)

plot($x_0 + radius, y_0$)

plot($x_0 - radius, y_0$)

**while** $x < y$ **do**

**if** $\Delta_i \geq 0$ **then**

$y = y - 1$

$\delta'_i = \delta'_i + 2$

$\Delta_i = \Delta_i + \delta'_i$

**end if**

$x = x + 1$

$\delta_i = \delta_i + 2$

$\Delta_i = \Delta_i + \delta_i$

plot($x_0 + x, y_0 + y$)

plot($x_0 - x, y_0 + y$)

plot($x_0 + x, y_0 - y$)

plot($x_0 - x, y_0 - y$)

plot($x_0 + y, y_0 + x$)

plot($x_0 - y, y_0 + x$)

plot($x_0 + y, y_0 - x$)

plot($x_0 - y, y_0 - x$)

$i++$

**end while**
First, a circle in the $x, z$ plane is calculated at the centre point $(x_0, y_0, z_0) = (0, 0, 0)$ and the radius $r = 30$ (see figure 6.9(a)). In this example the radius is a fixed value and the calculation of the sphere is given in the first quadrant (figure 6.9(a)). In accordance with this circle the parameters for the circles in the $x, y$ plane can now be determined. The $z$-coordinate is used as the distance from the centre of the sphere along the $z$-axis as the centre point of the corresponding layered circle, while the $x$-coordinate is used as its radius. That means that in the first quadrant, for every positive $(x_i, z_i)$, the BCA calculates the circle in the $x, y$ plane, with $x_i$ as its radius $r_i = x_i$ and $(x_0, y_0, z_i)$ as its centre point, with $0 \leq x_i, z_i \leq r$, $i \in \mathbb{N}$. The sphere is complete when for all $x_i \in (0, \ldots, r)$ and $z_i \in (-r, \ldots, r)$ a circle is generated.

Figure 6.9(a) illustrates the circle in the $x, z$ plane. Figure 6.9(b) illustrates the work flow direction of the main circle and the other layered circles. The whole sphere generated with the method mentioned above is shown in figure 6.9(c).

The description of the SBCA is now finished. In accordance with the given mathematical definition of the SBCA, it can now be used for drawing a sphere. The input parameters of the SBCA are the radius and its centre point. For the following explanations the centre is at $(x_0, y_0, z_0) = (0, 0, 0)$, and $r = 30$. The SBCA delivers the discrete surface points of the sphere. In this study the surface points are not plotted, but are stored internally.

Unlike the BCA pseudo code, the SBCA has to use step by step evaluation of the movements used in BCA. In accordance with these movements the $x$ and $z$ coordinates and $\Delta$ are calculated. These points are the new input parameters for the BCA.
Using the Bresenham Algorithm for Circles to Generate Spheres

(a) A circle in the $x, z$ plane, which provides the parameters of the circles in the $x, y$ plane. The work flow direction of the first circle lies in the $x, z$ plane.

(b) Several layered circles along the $z$ axis are shown for better overview.

(c) Full sphere.

Figure 6.9: Generating a sphere using the BCA.
Starting at \((x, z) = (0, r)\), \(\Delta\) can be initialised thus in the BCA:

\[
\Delta = (x + 1)^2 + (z - 1)^2 - r^2 = (1) + (r - 1)^2 - r^2 = (1) + r^2 - 2r + 1 - r^2 = 2 - 2r
\]

SBCA 2 is initialised with \(\Delta = 1 - r, (x, y, z) = (0, 0, \text{radius})\). In this study the SBCA determines the surface points in the first quadrant only, and by using the symmetry to the \(x\) axis the sphere is completed. Figure 6.10 illustrates the process of sphere generation.

![Figure 6.10: Completing the sphere using the symmetry to the \(x, y\) plane through its centre point](image)

6.5 Hough Transform for Spheres

Previous studies ([42], [13] [38]) have shown an improvement of the segmentation results when the models are prepositioned. In this study the aim is, first, to localise the femoral head in the image of the patient, second to preposition the other models in accordance with the femur model and to achieve good segmentation results. The femoral head has a spherical shape. Therefore the Hough Transform (HT) for spheres is employed to detect the sphere in the image of the patient for
Algorithm 2 SBCA(radius)

Input: radius
Output: Plotting the sphere surface points $x, y, z$

Initialise: $y = 0$
Initialise: $x = 0$
Initialise: $z = radius$

Compute: $\Delta = (1 - radius)$

while $z \geq 0$ do
  if $\Delta < 0$ then
    $\delta = 2\Delta + 2z - 1$
    if $\delta > 0$ then
      BresenhamCircle($y, z, x$)
      $x = x + 1$
      $z = z - 1$
    else
      $x = x + 1$
      $\Delta = \Delta + 2x + 1$
      BresenhamCircle($y, z, x$)
  end if
  else if $\Delta > 0$ then
    $\delta' = 2\Delta - 2z + 1$
    if $\delta' > 0$ then
      BresenhamCircle($y, z, x$)
      $z = z - 1$
      $\Delta = \Delta + 2x + 1$
    else
      BresenhamCircle($y, z, x$)
      $x = x + 1$
      $z = z - 1$
      $\Delta = \Delta + 2x - 2z + 2$
    end if
  else
    BresenhamCircle($y, z, x$)
    $x = x + 1$
    $z = z - 1$
    $\Delta = \Delta + 2x - 2z + 2$
  end if
  BresenhamCircle($y, z, x$)
end while
prepositioning the whole femur model; this is called the image sphere. The sphere obtained through the sphere parameterisation as used by the HT is called the Hough sphere.

The HT for spheres algorithm works in the same manner as the HT for circles, but has one additional dimension. As in the HT for circles in section 6.2, the HT for spheres generates the accumulator space. Through the additional dimension computational time increases markedly, because the HT for spheres has to process more votes for each contour point. To minimise computation time a priori knowledge about the directions of the gradients delivered by the Sobel filter is utilised.

The gradients are approximately normals of the surface of the image sphere, and point away from this. The negative gradients point in the direction of the centre point of this sphere. As only the centre point of this sphere has to be found, it suffices if only the votes in its immediate vicinity are considered. Therefore it is not necessary to evaluate the whole Hough sphere; only the votes for a small cupola of this sphere passing through the centre point of the image sphere have to be considered.

A larger cupola permits a larger inaccuracy of the direction of the gradient, while a smaller cupola reduces computation time. For producing the shape of the cupola, in this study only those points of the Hough circle are considered whose $z$-coordinates are larger than $\frac{4}{5}$. This shape is rotated depending on the direction of the gradient.

Figure 6.11 illustrates the grey level image in the axial plane and its Hough-transformed image.

The Hough or accumulator space in figure 6.11(a) has the same dimension as the number of parameters describing the sphere $S_{sphere}(p_1, p_2, p_3)$, with centre point $(p_1, p_2, p_3)$ and a fixed radius. Here the radius is $r = 21$ voxels.
Figure 6.11: Grey level image in the axial plane and its Hough-transformed image
Chapter 7

Applying Point-based Registration by Computing Subspaces with Singular Value Decomposition and Computing the Rotation of Subspaces

Computer-assisted surgical systems often use the registration of images to compensate for the deviations of these when the patient moves[14]. The registration problem is normally categorised into rigid and non-rigid registration. Rigid registration denotes the registration of a rigid body, one in which the distance between points does not change in time. Rigid registration techniques have been successfully used in guidance systems designed for neurosurgery, hip surgery and spine surgery. These systems often rely on feature- or point-based registration to determine the registration transformation: rotation, translation and scaling. Features or landmarks can be either extracted from the anatomy and imaging data sets being registered, or created artificially by implanting fiducial markers on the surgical site.

Point-based registration (PBR) aims to find the rotation, the translation and the scaling factor between two tomographic images to accomplish the image registration. The positions of the features or landmarks are linked to the tomographic image position through the registration information.

Usually point-based registration is used between images to fit the first image into the second one by minimising the error between the landmarks. But in this case point based registration will be used to point out the relationship between the initial model and the adapted model. After calculating rotation and translation between the models, the matrix $T$ is obtained by the formula 7.2. The remaining models are multiplied by the matrix $T$. The aim is to achieve an improvement of the models’ alignment in the image.

In other words, point based registration is employed in this study to preposition the models in the image of the patient by exploiting the obtained information.
about the adaptation. Therefore an improvement of the model-based approach is also obtained.

The first section 7.1 describes the point-based registration and explains the defined matrix $T$. The second section 7.2 describes the calculation of the rotation matrix by solving the orthogonal Procrustes problem through the use of the singular value decomposition.

### 7.1 Point-based Registration

The formal definition of this registration is the determination of one-to-one mapping between the landmarks, and of the position in the tomographic image; in this way the points are mapped to each other.

The mathematical definition of such mapping is given by:

$$T : X_A \rightarrow X_B \iff T(X_A) = X_B$$

(7.1)

where $T$ is the transformation matrix, $X_A$ are the feature points and $X_B$ are the positions in the tomographic image. Feature points are points specially marked by specialists. These marked points highlight the important regions, or help to figure out the anatomical object in the tomographic image.

Rigid body registration is appropriate only when the imaged object is itself rigid. Anatomical objects such as bones are rigid, for example the human head, the vertebrae, the pelvis, the femur and other bones. The rigid body transformation requires that the real Euclidean distance between any two coordinate locations remains unchanged by the transformation. A rigid transformation preserves angles as well as distances.

![Rigid transformation](image)

**Figure 7.1: Rigid transformation**

Figure 7.1 illustrates a rigid transformation for two objects. The first figure 7.1(a) shows the original objects, the second figure 7.1(b) illustrates a global rigid transformation and in the third figure 7.1(c) a local rigid transformation is shown: the first object is rotated counterclockwise and the second object remains unchanged.
In this work the relationship between the initial models and the adapted models is investigated. The vertices $v = (v_1^Tv_2^T \ldots v_n^T)^T$ of the initial model and the vertices of the adapted model $\hat{v} = (\hat{v}_1^T\hat{v}_2^T \ldots \hat{v}_k^T)^T$ are called landmarks.

There are many variations in the size and shape of the landmarks, their number and their configuration. In each case it is required to find a 3D translation $t$ and rotation $R$ that aligns the landmarks from the initial model to the adapted model.

A singular value decomposition (SVD) is performed to calculate the rotation, the translation and the scaling factor between these models. In the matrix $T$ the rotation matrix $R$, translation $t$, the scaling factor $s$ are combined together, and $T$ is given with the formula:

$$ T = \begin{pmatrix} sr_1 & sr_2 & sr_3 & t_1 \\ sr_1 & sr_2 & sr_3 & t_2 \\ sr_1 & sr_2 & sr_3 & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (7.2) $$

The orthogonal Procrustes problem (OPP) provides the rotation matrix $R$ by a SVD, where $R$ rotates the initial model to the adapted model. Therefore the image registration problem can be solved by OPP using the SVD.

The procedure is useful for investigating the relationship between the initial model and the adapted model. The mathematical definition is explained in the next section.

Figure 7.1 illustrates the way of using point-based registration for aligning the models in the image of the patient. Figure 7.2(a) shows the initial model in light grey and the adapted model in blue. The following figure 7.2(b) shows the perspective view of the two models.

Figure 7.2(c) shows the registrated pelvis model in red. The initial pelvis model is multiplied by the $T$ matrix, therefore its alignment is obtained. The light grey pelvis model is the initial model. Figure 7.2(d) illustrates the adaptation of the aligned pelvis model. The blue pelvis is the adapted model.

### 7.2 Orthogonal Procrustes Problem and Computing the Rotation of Subspaces using Singular Value Decomposition

The data matrix $V \in \mathbb{R}^{m \times p}$ is obtained from the initial model. The other data matrix $\hat{V} \in \mathbb{R}^{m \times p}$ is obtained from the adapted model, where $m, p \in \mathbb{N}$. The image registration problem can be reduced to the orthogonal Procrustes problem by letting row $i$ of $V$ be the elements of the demeaned vector $v_i = v_i - v_{mean}$, and simultaneously for $\hat{V}$ the rows are demeaned [14] [34], where $v_{mean}$ is the mean value and $i \in \mathbb{N}$. The orthogonal Procrustes problem solves the problem if the matrix $V$ can be rotated into the matrix $\hat{V}$. This problem is expressed by:
Orthogonal Procrustes Problem and Computing the Rotation of Subspaces using Singular Value Decomposition

Figure 7.2: Point-based registration of the left pelvis. $T$ is calculated between the initial femur model in grey and the adapted femur model in blue.
Orthogonal Procrustes Problem and Computing the Rotation of Subspaces using Singular Value Decomposition

\[ \text{minimise} \| V - \hat{V}Q \|_F, \quad \text{wherein} \quad Q^TQ = E \]  
(7.3)

where \( F \) is the Frobenius norm and is \( \| A \|_F = \sqrt{\sum_i \sum_j |a_{ij}|^2} \), and \( E \) is the identity matrix.

A \( m \times m \) matrix \( Q \) is an orthogonal matrix if

\[ QQ^T = E \quad Q^TQ = E \]

In particular, an orthogonal matrix is always invertible:

\[ Q^{-1} = Q^T \]

The orthogonal matrices are precisely those matrices which preserve the inner product \( \langle v, w \rangle = \langle Qv, Qw \rangle \), all the angles remaining unchanged after a multiplication with an orthogonal matrix. \( \| Q\vec{x} \|_2 = \| \vec{x} \|_2 \) means that all Euclidean distances remain unchanged through multiplying with an orthogonal matrix. The matrix product of two orthogonal matrices is an orthogonal matrix.

The trace of a matrix is the sum of the diagonal entries, and therefore \( tr(C^TC) = \| C \|_F^2 \). It follows that if \( Q \in \mathbb{R}^{p \times p} \) is orthogonal, then:

\[
tr((V - \hat{V}Q)^T(V - \hat{V}Q)) = tr(V^TV) - tr(V^T\hat{V}Q) - tr(Q^T\hat{V}^TV) + tr(Q^T\hat{V}Q) \]  
(7.4)

In Schönemann’s original work [34], which was unrelated to image registration, the aim was to find the orthogonal matrix \( Q^{p \times p} \) that optimally transforms one set of \( m \geq p \) observations \( V \) into another set observations \( \hat{V} \). Note that ”observations” in this case mean the points of the models, whereas ”optimally” means that \( tr((V - \hat{V}Q)^T(V - \hat{V}Q)) \) is minimised.

The minimising problem is changed to a maximising problem because the sign of the term is changed \( -2tr(Q^T\hat{V}^TV^T) \) in equation 7.4. A solution for maximising \( -2tr(Q^T\hat{V}^TV^T) \) can be obtained by a SVD of \( \hat{V}^TV^T \). \( Q^T \) is an orthogonal matrix, and it has to remain an orthogonal matrix after the multiplication by \( \hat{V}^TV^T \). Therefore the \( V^TV^T \) has to be orthogonalised by a mathematical decomposition.

However, if \( A \in \mathbb{R}^{m \times p} \) is an \( m \times p \) real matrix with \( m > p \), \( A \) can be written using a so-called singular value decomposition of the form

\[ A = U\Sigma W^T \]  
(7.5)

where the orthogonal matrices are \( U \in \mathbb{R}^{m \times m} \) and \( W \in \mathbb{R}^{p \times p} \), and the diagonal matrix is \( \Sigma = (\sigma_i \delta_{i,j})_{i,j} \in \mathbb{R}^{m \times p} \) [32].
The solution using the SVD was an improvement over the solution published by Green in 1952 [17]. It was based on the concept of the square root of a symmetric matrix and required that $\hat{V}^T \hat{V}^T$ be nonsingular. But SVD does not require this. By applying the SVD 7.5 to $\hat{V}^T \hat{V}^T$ we obtain

$$\hat{V}^T \hat{V}^T = U \Sigma W^T$$

and then

$$tr(Q^T \hat{V}^T \hat{V}^T) = tr(Q^T U \Sigma W^T)$$

$$= tr(W^T Q^T U \Sigma)$$

$$= tr(Z \Sigma) = \sum_{i=1}^{p} z_{ii} \sigma_i \leq \sum_{i=1}^{p} \sigma_i$$  \hspace{1cm} (7.6)

Note that with $tr(ab) = tr(ba)$ and using the orthogonal matrix $Z = W^T Q^T U$ the formula 7.6 is obtained.

Since matrix $\Sigma$ is a diagonal matrix, only the diagonal elements of matrix $Z$ in the formula 7.6 contribute to the trace. $Z$ is an orthogonal matrix because the matrix product of two orthogonal matrices is another orthogonal matrix. All three matrices $W^T, Q^T, U$ are orthogonal. Therefore $Z$ becomes maximal if $Z = E$. So rewriting the definition of $Z$ for $Q^T$, it follows:

using $Q^T = WZU^T$

$$WZU^T = WIU^T$$

$$WW^T Q^T UU^T = WU^T$$

$$Q^T = WU^T$$  \hspace{1cm} (7.7)

where $WW^T = E$ and $UU^T = E$. Only the orthogonal transformation matrix $Q^T$ remains.

As to the minimising problem, given the data matrix $V \in \mathbb{R}^{m \times p}$ and a second data matrix $\hat{V} \in \mathbb{R}^{m \times p}$, the following algorithm finds an orthogonal $Q \in \mathbb{R}^{p \times p}$ such that $\left\| V - \hat{V} Q \right\|_F$ is minimal:

(i) $C = \hat{V}^T V^T$

(ii) compute the SVD of $UCW^T = \Sigma$ and store the matrices $U$ and $W$

(iii) $Q^T = WU^T$

This orthogonal matrix $Q^T$ rotates the initial model $V$ to the adapted model $\hat{V}$. Figure 7.1 illustrates the rotation; the $V$ matrix represents the grey initial model and the $\hat{V}$ represents the rotated model in red. This red model will be adapted by the model-based approach.

This disposes of the minimising problem.
Chapter 8

Adaptation of Deformable Models to Anatomical Structures

The adaptation of deformable models to anatomical structures is one of the central subjects of this work. The aim of this adaptation is a precise segmentation, explained in section 2, of the anatomical object, as required for automated image analysis. A model-based adaptation is performed by accurately matching the boundaries of the patient’s anatomy [12]. Deformable models offer a unique and powerful approach to image analysis that combines geometry and physics. They are capable of capturing the significant variability of biological structures. With a manual segmentation the obtained deformable anatomical models provide a priori knowledge about structures. In this study an initial deformable model is adapted to the female pelvis, femur and its cartilage, as well as the cartilage in the acetabulum. In this chapter the adaptation algorithm is given in section 8.1. The parameter settings are discussed in 8.2, while the simulated annealing algorithm is explained in 8.3. The improvement of the model generation by mesh refining is introduced in 8.4.

8.1 Mesh Adaptation to the Anatomical Structure

The mesh adaptation in this work aims to segment the pelvis, the femur and its cartilage and the cartilage in the acetabulum of a female hip. In order to adapt the deformable shape models to these, the model-based adaptation is applied to the image. The input parameters are the image and the deformable model, also called mesh. The output is another mesh which has the same size as the input mesh (i.e., no triangles are inserted or removed). The image can be expressed mathematically as a three dimensional matrix having intensities as picture elements. The height is described with \( h \), the weight with \( w \) and the depth with \( d \). Therefore the image with the intensity range of 0 to 65535 is defined as \( I_{\text{grey}} \in \{0,65535\}^{h\times w\times d} \).
The mesh is a vector of dimension one with \( \mathbf{v} = (v_1^T v_2^T \ldots v_n^T)^T \), wherein a vertex is described as \( v_i = (v_{i,x}, v_{i,y}, v_{i,z})^T \).

The model-based adaptation can be expressed as an energy term which consists of an external energy term and an internal energy term. By minimising the sum of both terms the iterative procedure of mesh deformation is performed, wherein the value of \( E_{ext} \) and \( E_{int} \) increase with respect to the deviation of the mesh from the image features of the anatomical object.

\[
E(I_{grey}, \mathbf{v}) = E_{ext}(I_{grey}, \mathbf{v}) + \alpha E_{int}(\mathbf{v}) \quad (8.1)
\]

The parameter \( \alpha \) balances the contribution of the external energy (attraction to boundary) and the internal energy (shape constraints).

The parameter settings are discussed in section 8.2. These parameters limit the range of the intensities and contain the values for external and internal energies, and are hence defined and explained while discussing these terms.

The new vertices can be calculated with the formula: \( \tilde{\mathbf{v}} = A^T \mathbf{b} \). The Matrix \( A \) has the entries \( E_{ext}(I_{grey}, \mathbf{v}) \) and \( \alpha E_{int}(\mathbf{v}) \). The vector \( \mathbf{b} \) contains the scaled and rotated model vertices with respect to the internal and external energies.

### 8.1.1 External Energy

The external energy demonstrates the energy obtained when the mesh triangles move towards the image features. It minimises the weighted sum of the quadratic distances between the target points \( \tilde{x}_i \) and the triangle centres \( c_i \). The input parameters are the vertices \( \mathbf{v} \), the input image \( I_{grey} \) and a mesh. The output is another weighted mesh vertices \( \tilde{\mathbf{v}} \) of the same size as the input.

For every triangle \( x_i \) at its barycentre \( c_i \) and in the direction of its normal \( n_i \) a compatible feature point is searched for at the discrete position \( j \).

\[
c_j = j\delta n_i \quad \text{with} \quad j = -l, \ldots, l \quad (8.2)
\]

The parameter \( \delta \) defines the sampling distance of the profile. The number of sample points \( c_j \) on the triangle normal \( n_i \) is \((2l + 1)\), and the profile length is \( l \). The triangle centre can be defined as \( c_i(v) = \frac{v_1 v_2 + v_3}{3} \).

While searching along the sample points \( c_j \) the feature function \( F(x) \) is evaluated. The function yields high response at object boundaries, and it is an edge detector. The new candidate point \( \tilde{x}_i \) is that point \( \tilde{x}_i + c_i \) which best combines feature value and distance \( j\delta \) with the current point \( \tilde{x}_i \).

A target point \( \tilde{x}_i \) can be selected according to

\[
\tilde{x}_i = \tilde{x}_i + \arg \max_{j=-l, \ldots, l} \left[ F_i(\tilde{x}_i + j\delta n_i) - D j^2 \delta^2 \right] \delta n_i
\]

where \( D \) controls the weight of the penalty term, which leads to a search in the nearer neighbour points. The edge detector calculates for each discrete value \( j \) the response of the point \( x_{i,j} \), wherein the target point having the minimal quadratic distance to the triangle centre and a high edge response is chosen.
Depending on the model, the control parameters $\delta$, $D$, and $l$ can be set with varying values for yielding the most accurate result. These parameters are chosen for $l = 10$, $0.1 \leq D \leq 5.0$ and $\delta = 1$.

The edge detector $F(x)$ evaluates the feature strength at the point $x$ based on the image gradient $5.4$. The boundary of the region is defined as the transition of grey values. Usually the anatomical object of interest has a complex structure, therefore the range of feature values depends on different parts of it. To overcome this difficulty, feature values from different data sets can be evaluated, and the feature evaluation function trained. In this approach a global feature function is used for all triangles of the models.

A straightforward approach is the projection of the gradient $\nabla I(x)$ onto the triangle normal $n_i$.

$$ F(x) = n_i^T \nabla I(x) \quad (8.3) $$

Using the a priori knowledge about the image, the range intensity including the anatomical structures can be limited by some threshold, as the intensity levels of an anatomical structure have characteristic grey level intervals. Information on the grey levels is obtained from the image; one can, for instance, define the gradient direction, which can be consigned to the algorithm. Therefore the direction of the gradient can be included in the formula of the external energy as a parameter $\sigma \in \{-1, 1\}$, where $-1$ defines the transition from dark to bright. With this modification the formula can be improved to

$$ F(x) = \begin{cases} \sigma n_i^T \nabla I(x) \left[ \frac{g_{\text{max}}(g_{\text{max}} + \|\nabla I(x)\|)}{g_{\text{max}} + \|\nabla I(x)\|} \right] & : I_{\text{min}} < I(x) < I_{\text{max}} \\ 0 & : \text{otherwise} \end{cases} \quad (8.4) $$

The second term in squared brackets restricts the magnitude of $F(x)$ to the range of $\{I_{\text{min}}, I_{\text{max}}\}$. With the most promising point $\hat{x}_i$ and the current triangle centre $c_i$, the external energy controlling the impact of the image structures in the adaptation process can be formulated as

$$ E_{\text{ext}}(v) = \sum_{i \in T} w_i \left( \frac{g_i}{\|g\|} \hat{x}_i - \hat{x}_i \right)^2 \quad (8.5) $$

where the weight $w_i$ is defined as

$$ w_i = \max \left\{ 0, F_i(\hat{x}_i) - D_f^2 \delta^2 \right\} \quad (8.6) $$

With this formula only the projection of the gradient $g$ onto the surface normal $n_i$ is penalised. The triangle centre can slide along the iso-contour. With this approach, the mesh is prevented from being attached to false object boundaries, which are frequently detected at the beginning of an adaptation process [20].
8.1.2 Internal Energy

The internal energy term controls the deviation of the vertices of the adapted mesh and depends only on the model. It penalises the edge deformation of the triangles with respect to the input model.

\[
E_{\text{int}}(\mathbf{v}) = \sum_{(j,k)\in \text{edgesM}} (\mathbf{\hat{v}}_j - \mathbf{\hat{v}}_k - s\mathbf{R}(\mathbf{v}_j - \mathbf{v}_k))^2 \quad (8.7)
\]

The internal energy preserves shape similarity of all adapted model vertices \(v_j\) to the model vertices \(\mathbf{\hat{v}}_j\) [18].

A singular value decomposition computes the parameters \(s\) and \(\mathbf{R}\). Only rigid registration between the vertices of the initial mesh and of the adapted mesh is allowed because the Euclidean distances between the vertices have to be preserved.

Vertex Optimisation

With the results of the external energy which searches for the most promising features, and the internal energy which calculates the constraints on the model vertices, the new mesh \(\mathbf{\tilde{v}}_i\) can be calculated. These coordinates of the new mesh are determined by solving a linear equation system, resulting in a least square solution for the constraints formulated by both energies [42].

The Equation System

The external energy defined by 8.3 drives each triangle barycentre \(\mathbf{\hat{x}}_i\) towards the iso-surface normal at the image position of the candidate position \(\mathbf{\tilde{x}}_i\) obtained from the feature search. The resulting constraint for the triangle \(i\) can be expressed for the vertices \(\mathbf{\tilde{v}}_i^1, \mathbf{\tilde{v}}_i^2, \mathbf{\tilde{v}}_i^3\) forming the triangle by using the following equation:

\[
M_i \mathbf{\tilde{x}}_i = M_i \frac{1}{3}(\mathbf{\tilde{v}}_i^1 + \mathbf{\tilde{v}}_i^2 + \mathbf{\tilde{v}}_i^3). \quad (8.8)
\]

with the matrix \(M_i\) projecting the candidate point \(\mathbf{\tilde{x}}_i\) onto the iso-surface normal.

The global transformation \(s\mathbf{R}\) defined between the initial model and the current mesh configuration introduced with 8.7 defines the following constraint for two neighboured vertices \(\mathbf{\tilde{v}}_j, \mathbf{\tilde{v}}_k\):

\[
s\mathbf{R}(\mathbf{\tilde{v}}_j - \mathbf{\tilde{v}}_k) = (\mathbf{\hat{v}}_j - \mathbf{\hat{v}}_k). \quad (8.9)
\]

By rearranging 8.8 and using 8.9, constraints for each triangle and each edge of the mesh on the new vertices are obtained. Following these considerations, a sparse and overdetermined equation system for the new vertex coordinates \(\mathbf{\tilde{v}}_v (v \in \{1, ..., V\})\) can be defined. Assuming triangle \(i\) with its barycentre \(\mathbf{\hat{x}}_i\) connected by vertices \(\mathbf{\tilde{v}}_j, \mathbf{\tilde{v}}_k, \mathbf{\tilde{v}}_l\), the equation system can be written as:
The Solver

To calculate the least square solution of the overdetermined equation system, $8.10$ is multiplied by $A^T$:

$$A^T A \hat{\tilde{v}} = A^T b. \tag{8.11}$$

By considering its sparse characteristic, the resulting symmetric and positive definite matrix can be solved in a fast and robust way with the conjugate gradient.

8.2 Feature Parameter Settings

The choice of parameters is very difficult especially in this case because the number of parameters influencing the model-based approach is large. This makes the evaluation of the results difficult. The choice of parameters depends upon the image values, the anatomical structure to be segmented and the accurate prepositioning of the models too. The dependency of the parameters upon each other further complicates the issue. All this variability influences the quality of the segmenting result considerably, and so the parameters have to be set individually for every segmentation.

To this end, a heuristic method (simulated annealing) is applied. The parameters of the model-based approach have to be combined so as to obtain the best segmentation result. The range of parameters used in the model-based approach varies from 0.001 to 5000. The algorithm needs eleven input parameters. An exhaustive search for the best parameter settings is, however, precluded because it would take too much computing time and would have high complexity; with a brute force method the number of iterations would be $1000^{11}$ if each parameter be sampled at 1000 values along its range. This number of iterations is too large.

To obtain the desired parameters the method should, therefore, fulfil the requirements of short computation time and targeting compatible values. This is exactly what simulated annealing does. This algorithm attempts to find the optimal parameter setting to improve segmentation. All such parameters have already been mentioned in the previous section in different formulae, and will be explained in detail in the next subsection.
The Input Parameters of the Model-based Approach

The input parameters of the model-based algorithm can be extracted from the formula of the external energy 8.5 and the formula of the internal energy 8.7. In the energy term which controls the performance of the model-based approach given in formula 8.1, the parameter $\alpha$ controls the balance between the external energy and the internal energy. If $\alpha$ is too small then the influence of the shape constraints (internal energy) is too weak to prevent the triangles from being attracted towards misleading boundaries. If $\alpha$ is too large, then the model is not flexible enough to adapt to the specific shape of the image. The optimal value thus has to be determined experimentally [12].

Considering the external energy term in formula 8.5, the weight $w$ reflects the reliability of the detected boundary. The weight in formula 8.6 has parameters that influence the external energy, and thereby the model-based approach. These are:

- $\alpha$, controlling the balance between the external energy and the internal energy
- $D$, controlling the weight of the penalty term, which biases the search to nearby points
- $\delta$, sampling distance for the feature search along the normals
- $l$, number of sampling points for the feature search
- $g_{\text{max}}$, controlling the range of the intensities around the anatomical object

Different sets of values for the parameters $D$, $\delta$, $l$ can be used depending on how far the model is from the object in the image. If there is no prepositioning of the anatomical models in the image of the patient, the parameter $l$ should be large enough to have a longer profile length for capturing the range of the anatomical structure that is possibly far from the model, and $D$ should have a smaller bias to nearby points. The parameter $\delta$ is the sampling distance on this profile length. The range of intensities around the anatomical object is controlled by the parameter $g_{\text{max}}$. This is the damping factor for the gradient in the feature function 8.4. It consists of a range of intensity values. Therefore the search for a feature value is confined only to this region.

8.3 Simulated Annealing

In this work the simulated annealing algorithm (SA) is applied for targeting the optimal parameter set to obtain the best result of the model-based approach. SA is a generic probabilistic meta-algorithm for global optimisation problems and was originally introduced by Kirkpatrick in 1983 [24] [22]. SA can achieve an optimisation of a set of parameters and finds a good approximation to the global optimum in a large search space.
For certain problems, SA may be more effective than exhaustive enumeration. It provides an acceptable good solution in a fixed amount of time. Annealing is a physical process, a technique involving heating and controlled cooling of a material to increase the size of its crystals and reduce their defects. The heat causes the atoms to become unstuck from their initial positions (a local minimum of the internal energy) and wander randomly through states of higher energy; the slow cooling gives them more chances of finding configurations with lower internal energy than the initial one.

In analogy to this physical process, in each step the SA algorithm changes the current set of parameters to another set of parameters, which is called the neighbouring parameter set. A neighbouring parameter set is another valid parameter set that differs randomly only in one parameter. The neighbouring parameter set is chosen with a probability that depends on the difference between two different sets of parameters, evaluated through the evaluation function $E(\cdot)$, and on the parameter $T$, called the temperature, that is gradually decreased step by step during the process. A large temperature changes the current solution, a so-called uphill. A decreasing temperature, nearly zero, leads to a so-called downhill. The allowance for uphill moves saves the method from becoming stuck at local minima which are the bane of Greedier methods [2].

The change of the neighbouring parameter set depends on a uniform distribution that with a uniform probability chooses a parameter from the parameter set, and changes its value by choosing the next or previous number from its range lying on its sample grid. This uniform distribution, sometimes also known as a rectangular distribution, is a distribution that has constant probability. $Params_i \in \mathbb{R}, i \in \mathbb{N}$ describes each set of the parameter range. A parameter set is defined as

$$set_p \in \{s_1, s_2, s_3, \ldots, s_{11}\}, s_i \in Params_i$$  \hspace{1cm} (8.12)

with $p \in \mathbb{N}$.

The neighbouring parameter sets $set_1, set_2$ differ in only one parameter.

$$E_{set_p}(v, \hat{v})$$  \hspace{1cm} (8.13)

The function 8.13 evaluates the quality of the result of segmentation with the parameter set $set_p$, where $\hat{v}$ represents the adapted mesh. The quality measure is the squared distance of two meshes. In each step a parameter set $set_i$ is generated, and the initial model is adapted with the model-based approach. The output of the model-based approach is also a mesh. Therefore the function $E_{set_p}(v, \hat{v})$ measures the distance of the adapted model to the reference or ground-truth model. The ground-truth meshes are manually segmented and an orthopedist confirms the reliability of the meshes.

The SA starts with a predefined set of parameters which provide a good adaptation result. The input of the SA is at first the ground-truth mesh, which has to be adapted to the anatomical object of interest in the image of the patient, the MR image of the patient and a valid parameter set. The algorithm internally uses the
Simulated Annealing

evaluating function, a physical term which controls the temperature $T$ and the starting and stopping value. First, the initial mesh is adapted with the model-based approach and the adapted mesh evaluated with the evaluation function $E_{set_p}(v, \hat{v})$. $E_{set_p}(v, \hat{v})$ provides the mean square error between the ground-truth and the adapted meshes.

If the last calculated mean square error is less then the current mean square error, then the current parameter set does not change.

During SA the current parameter set is replaced by a neighbouring parameter set, that is here called mutation. To mutate a parameter set a parameter is randomly chosen, and its value is changed by choosing the next value from its range in the set, or its previous value. Then the adaptation of the initial mesh is executed. The result is evaluated with the evaluation function 8.13.

The physical term leads the SA to change the current location.

$$ r < e^{(E_{set_p}(v, \hat{v}) - E_{set_{p+1}}(v, \hat{v})) / T} $$

The variable $r \in [0, 1]$ is a random variable.

The term $e^{(E_{set_p}(v, \hat{v}) - E_{set_{p+1}}(v, \hat{v})) / T}$ provides a value which at a high temperature is nearly 1 and at a small temperature nearly 0. The high temperature corresponds with the up-hill and the small temperature corresponds with the down-hill.

The algorithm repeats the mutation of the parameters, applying the model-based approach and evaluating the result until the stopping value for the $T$ is reached. SA aims to find the best possible set of parameters close to the initial parameter set, but the model-based approach yields a better result for segmentation. The temperature $T$ should be initialised with very high value to ensure that the up-hill is also possible. In each step the temperature decreases percentally with respect to the variable $\alpha$. The SA algorithm is implemented as a tool in this work 3; the parameters are chosen as follows; the temperature is $T = 10000$, the border temperature is $T_{stop} = 0.01$ and the variable $\alpha$ is $\alpha = 0.99$.

The number of steps can be calculated with $n = \frac{\log(T_{stop})}{\log(\alpha)}$.

With this description it is now possible to calculate a parameter set for the model-based approach to obtain good segmentation results.

Figure 8.1 illustrates the iterations of the particular parameters in each subfigure. Some particular parameter were fixed to speed up the computation time. The subfigure ”MSE between meshes ” shows the graph of the number of the iterations to the square distance to the reference mesh. The set of Parameters are chosen having the minimal distance to the reference set.

The SA algorithm 3 begins with the initialisation by evaluating the function $E_{set_{current}}(v, \tilde{v}_{ref})$. The evaluation function delivers the mean square distance of the initial model to the reference model. In the while loop a new neighbouring parameter set is generated. This is the mutation. In each step of this loop the initial mesh is adapted to the anatomical structure with the neighbouring parameter set. Every adapted model’s distance to the reference model is measured by the function $E_{set_{current}}(\hat{v}, \tilde{v}_{ref})$. If the value is better than the current value, then this parameter set is chosen as the best current parameter.
Figure 8.1: Simulated Annealing results
Algorithm 3 SimulatedAnnealing\( (T, T_{\text{stop}}, \alpha) \)

**Input:** Temperature, the stop temperature and the variable \( \alpha \)

**Output:** Best parameter set

- Initialise: \( E_{\text{set, current}}(v, \tilde{v}_{\text{ref}}) \);  
- Initialise: \( \text{set}_{\text{best}} = \text{set}_{\text{current}} \)

while \( (T > T_{\text{stop}}) \) do

- \( \text{set}_{\text{neighbour}}.\text{Mutate}() \)
- Adaptation \( (I_{\text{grey}}, v, \text{set}_{\text{neighbour}}, \tilde{v}) \)
- \( E_{\text{set, current}}(\tilde{v}, \tilde{v}_{\text{ref}}) \)

if \( E_{\text{set, current}}() < E_{\text{set, best}}() \) then

- \( E_{\text{set, best}}() = E_{\text{set, current}}() \)
- \( E_{\text{set, best}}() = E_{\text{set, current}}() \)

else if \( r > e^{(E_{\text{set, best}}())-E_{\text{set, current}}()}/T \) then

- \( \text{set}_{\text{current}} = \text{set}_{\text{neighbour}} \)

end if

- \( T = \alpha T \)
- \( \text{Freemesh}(\tilde{v}) \)

end while

return \( \text{set}_{\text{best}} \)

The loop goes on until the border temperature is reached. The SA delivers the globally best parameter set close to the initial parameter set. Therefore the mean square error is minimal.

### 8.4 Refining of Surfaces

In general, surface meshes are used for the representation of objects in computer graphics applications. Many of these meshes are generated by scanning devices such as X-Ray or MRI-scanning. With the help of specific graphical programmes the anatomical structure in the image of the patient is manually segmented, and with respect to the marching cubes algorithm the triangulated mesh is generated. The number of triangles used for describing the shape of the anatomical object can vary strongly. But for a good approximation of curves of the anatomical structure, i.e the femoral head, the number of triangles has to be increased, the whole mesh becoming finer. The curved objects can be represented more smoothly. In many approaches the refining has an important role.

Let us assume that a triangle mesh approximates a curved shape, possibly with sharp edges and corners. The process which refines the triangle mesh in this region is called mesh refining in this work. That means that the number of triangles in curved regions is increased by dividing the triangle into three subtriangles.

The refining aims to obtain a high resolution surface mesh which better approximates the curved regions, namely the pubis, the ischium and femoral head. Figure 8.2 illustrates the pelvis and the femur, whereby the curved regions are
shown.

Figure 8.2: Proximal right femur and pelvis in front view
Chapter 9

Work Flow

In this chapter the tools utilised are introduced. All programmes used in this study have been developed object oriented in C++. The structures to be segmented are: right pelvis, left pelvis, right femur, left femur, right cartilage, left cartilage. The segmentation of cartilages is not possible on the basis of the given MRT images, as they are difficult to differentiate in the image. Therefore they are segmented together, as one object.

9.1 Prepositioning

The results of segmentation show an improvement when the models are prepositioned in the image. Various approaches used in this study to permit automatic prepositioning in the image of the patient are explained below.

Detection of the Femoral Heads with the Hough Transform for Spheres

The femoral heads can be detected very reliably in the image with the Hough transform, allowing the initial models to be moved to these positions. The tool developed for this first converts the image into a black-and-white edge image. Then the whole image is remapped in an accumulator space with the HT. Subsequently the maximum votes in this Hough space are determined. Because of anatomical factors the centre of the femoral head is searched for first in the left and then in the right half of the Hough space.

The edge image is generated with the Sobel filter as in section 5.4. During this process the image is convolved with a $3 \times 3 \times 3$ Sobel filter kernel. The convolution is explained in section 5.2.

Since the image of the patient is isotropic only in two directions one has to scale the coordinates used. To be able to carry out such algebraic operations comfortably an algebra library has been developed for C++. This is able to carry out certain operations on the models. One can, for instance, inter alia scale, move and rotate...
Refinement of Model Topology

all points of a mesh very easily at will. Using this library the models can be scaled with ease.

The HT requires a parameterised sphere as the input parameter. This is localised in the image. The HT is optimised for speed by using the gradient information supplied by the Sobel filter.

The difference between the femoral heads in the mesh and the image detected by the HT gives the translation vector, used for prepositioning the mesh in the image. Subsequently, the anatomical models are adapted through model-based segmentation.

Point-based Registration

The second approach is to preposition the initial models through point-based registration. Registration requires a rotation matrix, a translation vector and a scaling factor as input. These are first determined as described in chapter 7, and combined in a matrix $T$. The relative Euclidean distances are not changed through registration. Presupposing that the other anatomical structures in the body have the same rotation as the femur, the matrix $T$ is applied to other models too: right pelvis, left pelvis and both cartilages.

Two Level Segmentation

Another prepositioning method is more dexterous employment of the model-based segmentation parameters by attempting to segment anatomical strutures in two adaptations. In the first segmentation a parameter set is chosen that allows the models to move within the image during adaptation. In the second segmentation the parameter set is chosen so that the models adapt themselves locally to the anatomical structure. The values for the parameters are:

1. step: $\alpha = 3.0, l = 20, D = 0.1$

2. step: $\alpha = 0.001, l = 10, D = 5$

9.2 Refinement of Model Topology

Beside automatic prepositioning a further approach for optimisation of segmentation results is changing the topology of the input model. The adjacent faces of sharp edges of the model are refined, so that at these loci a more precise adaptation to the anatomical structures that normally show only rounded surfaces is possible.
Figure 9.1: Hough Workflow
9.3 Adaptation, Evaluation and Generation of Mean-models

After adaptation of the mean model to the anatomical structure the results are evaluated by specialists. If success is determined, then new mean models are calculated on the basis of the hitherto successfully segmented models – including the one just generated.

9.4 Summary

The complete workflow of a segmentation encompasses the following successive steps:

1. application of the mentioned prepositioning algorithms
2. model refinement if applicable
3. adaptation
4. evaluation
5. renewed computation of the mean model if segmentation is successful

One-time preparations have to be made so that the workflow can be executed.

• The first models have to be manually segmented to obtain the first initial model.

• The adaptation parameters are determined by means of the simulated annealing algorithm in chapter 8.3.
Chapter 10

Results

The performance of the whole hip segmentation was evaluated in terms of accuracy and robustness. The clinical study delivered the images of different patients having hip dysplasia, which had to be segmented. To obtain a representative quality measure, the so-called ground truth model had to be generated, and reference models and every particular adapted model had to be evaluated in accordance with a distance.

Because of the high resolution of the patient’s image data manual segmentation was precluded. Instead, the initial models, i.e the mean models, were manually prepositioned and adapted. These adapted models were used as reference models. Besides, some important points were marked by experts. These landmarks pointed out some characteristic regions of the anatomical structures. The minimal distances to these were also measured.

Both methods measured the minimal squared distance of points to each other. First the model-based approach was applied to the initial models. They were generated by manual segmentation of one of the images, a male hip. These models complicated the model-based approach, because the male hip has a different extent and male bones have different characteristics. As the number of female data sets was increased and became more than three, mean models more suitable than a male hip model were generated. Using these improved the result of the model-based approach considerably.

This study aimed to automate the whole segmentation chain. Therefore several methods were combined to obtain the best result. In each of these combinations every intermediate step was also evaluated.

Metrics

Assuming that the adapted model and the reference model have the same topology, the mean square distance was simply measured by calculating the mean square distance from the adapted model vertices to the reference model vertices. The second measure was the minimal distance between the landmarks introduced in section 7 and the reference model, the units of measurement used being mm and
voxtels.

Figure 10.1: Distance between two vertices. The line represents the distance from a reference vertex (left side) to an adapted mesh vertex (right side). The mean distance is given subsequently.

The minimal distance between a landmark and the model was calculated by scanning the surroundings of the landmark within a cube with an initially small edge length. If there was no hit within the cube the edge length was increased. This procedure could make no statement about whether the landmark lay within or outside the model. Landmarks are marked in the acetabulum.

10.1 Evaluation

To evaluate the quality of the model-based segmentation the adaptation results were compared to the reference meshes. These were generated by adaptation of manually prepositioned meshes to the particular anatomical structures. First the initial models were adapted. Note that these were generated using a male hip. The particular alignment of the patient during the imaging was noticed in his image and was also used for aligning the models.

Without using any prepositioning the anatomical structures were segmented with initial models (male hip models). The mean square differences are given in table 10.1 and table 10.2.

As some models were positioned with insufficient accuracy the model-based segmentation did not deliver satisfactory results.

Simulated Annealing

The parameters used for segmentation are determined by the simulated annealing algorithm discussed in section 8.3. The following parameter set is used for the segmentation of the pelvis:

Figure 10.2 illustrates the alteration of the parameters over time. As the tomographic images are supposed to have similar characteristics, the chosen parameter set is suitable for all other images, too.
Evaluation

<table>
<thead>
<tr>
<th>name</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>$0.01$</td>
</tr>
<tr>
<td>maxGradientValue ($g_{max}$)</td>
<td>$381$</td>
</tr>
<tr>
<td>profileLength ($l$)</td>
<td>$9$</td>
</tr>
<tr>
<td>intensityRange</td>
<td>$[0.400.0]$</td>
</tr>
<tr>
<td>distWeight ($D$)</td>
<td>$1.0$</td>
</tr>
</tbody>
</table>

Note that during this experiment some of the parameters have been constrained to a fixed value. These are the number of iterations, the "featureweight" and the intensity range. The aim was to observe the influence of specific parameters on the result. It was figured out that $\alpha$, $D$ and $l$ have a high impact on the segmentation results. The red asterisks mark parameter values which lead to the best adaptation.

The parameter set for the segmentation of the femur is chosen as:

<table>
<thead>
<tr>
<th>name</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>$0.01$</td>
</tr>
<tr>
<td>maxGradientValue ($g_{max}$)</td>
<td>$300$</td>
</tr>
<tr>
<td>profileLength ($l$)</td>
<td>$15$</td>
</tr>
<tr>
<td>intensityRange</td>
<td>$[0.300.0]$</td>
</tr>
<tr>
<td>distWeight ($D$)</td>
<td>$1.0$</td>
</tr>
</tbody>
</table>

**Model-based adaptation**

The tomographic images have recurrent high intensities at the location of the bladder, which can lead to misadaptations. To suppress this source of irritation, intensities above a specified threshold are scaled down for the segmentation.

Tables 10.3 to 10.12 show the measured distances between the adapted meshes and the reference meshes. There is one column for each data set and one row for each anatomical structure. For each method there is one table listing the average point wise distances and one listing the standard deviations of these, where the considered points are the vertices of the meshes. The reference meshes were obtained by repeated segmentation of manually prepositioned models. In cases were the repetition count is one, the reference meshes equal the adapted meshes with entries "0" in table 10.3.

Tables 10.1 and 10.2 clearly show a misadaptation of the anatomical structures with the initial models obtained from a male hip. The misadaptation results from skipping the prepositioning phase. Furthermore the initial models are not capable of capturing the female hip characteristics. The results are regarded as not useful. The following results in Table 10.3 are measured between the manually prepositioned models and the reference models. It is obvious that prepositioning leads to improved results. The disadvantage of this method is that the prepositioning is carried out manually, i.e. the method is semi-automatic.
Figure 10.2: Parameter search using simulated Annealing

Table 10.1: Initial models without prepositioning: average distance

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>17,9</td>
<td>16,4</td>
<td>67,3</td>
<td>27,5</td>
<td>59,3</td>
<td>23,6</td>
<td>23,5</td>
<td>41,1</td>
<td>53,4</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>14,2</td>
<td>19,3</td>
<td>63,2</td>
<td>28,0</td>
<td>56,7</td>
<td>22,0</td>
<td>22,2</td>
<td>29,8</td>
<td>44,7</td>
</tr>
<tr>
<td>Femur R</td>
<td>22,9</td>
<td>18,3</td>
<td>44,1</td>
<td>25,8</td>
<td>41,7</td>
<td>17,3</td>
<td>17,5</td>
<td>29,8</td>
<td>24,9</td>
</tr>
<tr>
<td>Femur L</td>
<td>21,7</td>
<td>22,6</td>
<td>32,3</td>
<td>23,5</td>
<td>45,3</td>
<td>18,4</td>
<td>36,1</td>
<td>18,0</td>
<td>36,3</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>6,1</td>
<td>16,6</td>
<td>56,6</td>
<td>26,7</td>
<td>52,0</td>
<td>20,7</td>
<td>24,1</td>
<td>38,4</td>
<td>48,4</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>7,8</td>
<td>11,2</td>
<td>51,0</td>
<td>22,8</td>
<td>56,7</td>
<td>13,9</td>
<td>21,0</td>
<td>26,7</td>
<td>46,5</td>
</tr>
</tbody>
</table>

Table 10.2: Initial models without prepositioning: standard deviation

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>7,4</td>
<td>5,3</td>
<td>11,3</td>
<td>7,9</td>
<td>7,1</td>
<td>6,5</td>
<td>8,6</td>
<td>10,4</td>
<td>5,6</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>6,6</td>
<td>7,7</td>
<td>10,6</td>
<td>11,2</td>
<td>3,3</td>
<td>7,7</td>
<td>7,6</td>
<td>5,4</td>
<td>4,3</td>
</tr>
<tr>
<td>Femur R</td>
<td>4,5</td>
<td>4,8</td>
<td>6,5</td>
<td>9,4</td>
<td>9,7</td>
<td>4,7</td>
<td>6,5</td>
<td>7,6</td>
<td>8,4</td>
</tr>
<tr>
<td>Femur L</td>
<td>8,2</td>
<td>4,4</td>
<td>8,4</td>
<td>10,2</td>
<td>13,0</td>
<td>4,7</td>
<td>6,1</td>
<td>10,2</td>
<td>12,0</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>2,2</td>
<td>2,7</td>
<td>1,9</td>
<td>1,2</td>
<td>1,3</td>
<td>1,9</td>
<td>1,2</td>
<td>1,7</td>
<td>1,2</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>21,6</td>
<td>1,4</td>
<td>2,0</td>
<td>1,4</td>
<td>1,3</td>
<td>2,1</td>
<td>1,1</td>
<td>1,7</td>
<td>1,4</td>
</tr>
</tbody>
</table>
Table 10.3: Manual prepositioning: average distance

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>1,8</td>
<td>3,3</td>
<td>0,0</td>
<td>5,6</td>
<td>0,0</td>
<td>1,6</td>
<td>5,2</td>
<td>3,2</td>
<td>6,4</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>1,5</td>
<td>4,7</td>
<td>1,8</td>
<td>2,1</td>
<td>0,0</td>
<td>1,5</td>
<td>3,4</td>
<td>3,4</td>
<td>2,2</td>
</tr>
<tr>
<td>Femur R</td>
<td>0,0</td>
<td>15,6</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>5,6</td>
<td>0,0</td>
<td>10,1</td>
<td>8,7</td>
</tr>
<tr>
<td>Femur L</td>
<td>0,0</td>
<td>10,2</td>
<td>0,0</td>
<td>4,8</td>
<td>0,0</td>
<td>5,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
</tr>
</tbody>
</table>

Table 10.4: Manual prepositioning: standard deviation

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>1,0</td>
<td>3,1</td>
<td>0,0</td>
<td>3,9</td>
<td>0,0</td>
<td>0,8</td>
<td>4,6</td>
<td>2,1</td>
<td>6,6</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>0,7</td>
<td>7,1</td>
<td>1,3</td>
<td>1,0</td>
<td>0,0</td>
<td>0,8</td>
<td>3,0</td>
<td>2,5</td>
<td>1,5</td>
</tr>
<tr>
<td>Femur R</td>
<td>0,0</td>
<td>9,1</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>4,8</td>
<td>0,0</td>
<td>6,7</td>
<td>6,9</td>
</tr>
<tr>
<td>Femur L</td>
<td>0,0</td>
<td>6,1</td>
<td>0,0</td>
<td>4,0</td>
<td>0,0</td>
<td>3,8</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
<td>0,0</td>
</tr>
</tbody>
</table>

Figure 10.3: Segmentation example
Figure 10.3(a) shows the results of the segmentation of data set number 3 using the initial meshes without prepositioning. Figure 10.3(b) shows the meshes resulting from prepositioned initial models. The pelvis has a very complex anatomical structure which makes the segmentation difficult. Within the segmentation some regions have to be segmented, which are very important for further processing systems. For the most important anatomical regions reference points are marked manually by a specialist. They are used for measuring the quality of the segmentation by calculating their distance to the mesh. The segmentation results using mean models are shown in table 10.5 and in table 10.6

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>5.9</td>
<td>1.2</td>
<td>50.8</td>
<td>2.2</td>
<td>30.9</td>
<td>0.9</td>
<td>5.3</td>
<td>4.9</td>
<td>18.9</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>5.3</td>
<td>4.4</td>
<td>47.8</td>
<td>1.2</td>
<td>42.9</td>
<td>1.2</td>
<td>17.2</td>
<td>1.8</td>
<td>31.9</td>
</tr>
<tr>
<td>Femur R</td>
<td>9.6</td>
<td>8.7</td>
<td>37.7</td>
<td>21.2</td>
<td>41.3</td>
<td>6.2</td>
<td>35.2</td>
<td>11.1</td>
<td>35.9</td>
</tr>
<tr>
<td>Femur L</td>
<td>3.3</td>
<td>6.9</td>
<td>13.3</td>
<td>6.5</td>
<td>47.5</td>
<td>6.0</td>
<td>34.4</td>
<td>4.0</td>
<td>36.2</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>22.7</td>
<td>15.9</td>
<td>41.3</td>
<td>28.9</td>
<td>41.4</td>
<td>12.0</td>
<td>34.8</td>
<td>23.3</td>
<td>36.0</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>27.6</td>
<td>9.0</td>
<td>35.9</td>
<td>21.3</td>
<td>48.6</td>
<td>1.8</td>
<td>35.0</td>
<td>21.3</td>
<td>33.9</td>
</tr>
</tbody>
</table>

Table 10.5: Mean models: average distance

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>9.3</td>
<td>0.6</td>
<td>5.4</td>
<td>1.5</td>
<td>6.7</td>
<td>0.6</td>
<td>4.6</td>
<td>4.3</td>
<td>5.7</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>8.8</td>
<td>6.9</td>
<td>8.0</td>
<td>0.8</td>
<td>4.2</td>
<td>0.9</td>
<td>4.8</td>
<td>1.0</td>
<td>10.3</td>
</tr>
<tr>
<td>Femur R</td>
<td>9.7</td>
<td>6.4</td>
<td>9.5</td>
<td>2.6</td>
<td>10.4</td>
<td>6.2</td>
<td>3.2</td>
<td>8.9</td>
<td>5.2</td>
</tr>
<tr>
<td>Femur L</td>
<td>2.7</td>
<td>5.1</td>
<td>11.0</td>
<td>5.7</td>
<td>15.8</td>
<td>5.5</td>
<td>2.4</td>
<td>3.5</td>
<td>5.7</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>1.4</td>
<td>0.8</td>
<td>0.5</td>
<td>1.1</td>
<td>0.4</td>
<td>0.9</td>
<td>0.5</td>
<td>0.4</td>
<td>0.5</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>19.9</td>
<td>3.7</td>
<td>3.4</td>
<td>2.1</td>
<td>1.2</td>
<td>3.7</td>
<td>2.0</td>
<td>2.0</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Table 10.6: Mean models: standard deviation

The results of the segmentation using mean models were not as good as expected. However they are better then the results of the segmentation using the point based registration method. As mentioned above the problem roots in inaccurately prepositioning. The mean models are tested with the "leave one out" method. The segmentation results are shown in table 10.5. If the initial models are located close to the anatomical structures the PBR works quite good, else the segmentation results are not useful. The PBR leads to no improvements compared to the other methods.

The blue and red colored meshes in figure 10.4 are the initial meshes of pelvis and femur. The meshes adapted to the anatomical structures are shown in green and yellow. Unfortunately the adaptation quality is not very good.

Using 2 Level segmentation the following results are obtained. This method has six out of nine with high accuracy segmented.

The algorithm is employed to preposition the models in the image. The segmentation result of the HT is in the table 10.11 and 10.12.
<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>3.3</td>
<td>52.3</td>
<td>11.4</td>
<td>66.4</td>
<td>21.1</td>
<td>53.1</td>
<td>77.3</td>
<td>42.6</td>
<td>41.2</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>3.1</td>
<td>48.7</td>
<td>16.8</td>
<td>60.8</td>
<td>25.9</td>
<td>10.2</td>
<td>80.2</td>
<td>57.5</td>
<td>19.0</td>
</tr>
<tr>
<td>Femur R</td>
<td>7.7</td>
<td>47.0</td>
<td>5.7</td>
<td>50.8</td>
<td>13.4</td>
<td>51.0</td>
<td>63.3</td>
<td>29.8</td>
<td>57.5</td>
</tr>
<tr>
<td>Femur L</td>
<td>4.7</td>
<td>38.0</td>
<td>5.4</td>
<td>47.7</td>
<td>32.2</td>
<td>5.6</td>
<td>63.9</td>
<td>45.1</td>
<td>27.5</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>23.0</td>
<td>42.2</td>
<td>43.3</td>
<td>81.3</td>
<td>43.1</td>
<td>37.7</td>
<td>86.2</td>
<td>59.0</td>
<td>48.9</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>23.3</td>
<td>42.7</td>
<td>31.8</td>
<td>39.5</td>
<td>32.1</td>
<td>33.7</td>
<td>46.5</td>
<td>47.2</td>
<td>40.7</td>
</tr>
</tbody>
</table>

Table 10.7: Point based registration of models: average distance

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>2.5</td>
<td>12.3</td>
<td>8.8</td>
<td>19.8</td>
<td>10.3</td>
<td>11.7</td>
<td>18.0</td>
<td>4.8</td>
<td>9.1</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>1.6</td>
<td>11.4</td>
<td>9.0</td>
<td>23.9</td>
<td>11.1</td>
<td>7.5</td>
<td>23.4</td>
<td>15.8</td>
<td>7.2</td>
</tr>
<tr>
<td>Femur R</td>
<td>7.6</td>
<td>5.6</td>
<td>4.8</td>
<td>9.4</td>
<td>11.6</td>
<td>5.5</td>
<td>12.9</td>
<td>9.8</td>
<td>23.9</td>
</tr>
<tr>
<td>Femur L</td>
<td>3.0</td>
<td>7.8</td>
<td>4.7</td>
<td>11.6</td>
<td>11.9</td>
<td>4.4</td>
<td>11.7</td>
<td>13.5</td>
<td>11.0</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>1.8</td>
<td>2.2</td>
<td>0.8</td>
<td>2.3</td>
<td>1.8</td>
<td>2.1</td>
<td>2.3</td>
<td>2.3</td>
<td>2.8</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>20.4</td>
<td>3.9</td>
<td>3.8</td>
<td>3.9</td>
<td>3.5</td>
<td>2.5</td>
<td>3.6</td>
<td>3.9</td>
<td>19.3</td>
</tr>
</tbody>
</table>

Table 10.8: Point based registration of models: standard deviation

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>1.1</td>
<td>1.3</td>
<td>68.7</td>
<td>1.6</td>
<td>23.4</td>
<td>0.8</td>
<td>3.1</td>
<td>2.9</td>
<td>1.6</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>1.2</td>
<td>1.6</td>
<td>69.5</td>
<td>1.0</td>
<td>55.0</td>
<td>0.9</td>
<td>3.7</td>
<td>2.4</td>
<td>1.5</td>
</tr>
<tr>
<td>Femur R</td>
<td>7.6</td>
<td>5.2</td>
<td>26.6</td>
<td>5.0</td>
<td>42.8</td>
<td>4.4</td>
<td>10.9</td>
<td>8.5</td>
<td>30.6</td>
</tr>
<tr>
<td>Femur L</td>
<td>2.0</td>
<td>9.0</td>
<td>21.1</td>
<td>6.3</td>
<td>26.9</td>
<td>4.6</td>
<td>6.6</td>
<td>4.0</td>
<td>10.8</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>28.6</td>
<td>1.2</td>
<td>42.9</td>
<td>29.6</td>
<td>42.0</td>
<td>1.0</td>
<td>38.9</td>
<td>21.4</td>
<td>37.0</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>23.0</td>
<td>1.3</td>
<td>25.5</td>
<td>2.2</td>
<td>50.4</td>
<td>2.3</td>
<td>30.8</td>
<td>2.8</td>
<td>42.4</td>
</tr>
</tbody>
</table>

Table 10.9: 2 Level segmentation: average distance

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>0.7</td>
<td>0.6</td>
<td>17.3</td>
<td>1.2</td>
<td>12.4</td>
<td>0.5</td>
<td>2.6</td>
<td>2.2</td>
<td>0.9</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>0.6</td>
<td>0.7</td>
<td>24.0</td>
<td>0.5</td>
<td>25.7</td>
<td>0.6</td>
<td>3.6</td>
<td>1.5</td>
<td>0.9</td>
</tr>
<tr>
<td>Femur R</td>
<td>6.7</td>
<td>4.9</td>
<td>9.7</td>
<td>5.5</td>
<td>8.9</td>
<td>4.7</td>
<td>7.4</td>
<td>6.8</td>
<td>5.9</td>
</tr>
<tr>
<td>Femur L</td>
<td>1.5</td>
<td>4.9</td>
<td>13.2</td>
<td>5.6</td>
<td>8.3</td>
<td>4.1</td>
<td>6.6</td>
<td>3.6</td>
<td>7.3</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>2.0</td>
<td>0.4</td>
<td>1.0</td>
<td>2.2</td>
<td>0.7</td>
<td>0.5</td>
<td>1.6</td>
<td>0.7</td>
<td>0.9</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>19.8</td>
<td>4.4</td>
<td>4.2</td>
<td>4.3</td>
<td>2.0</td>
<td>21.6</td>
<td>2.9</td>
<td>22.3</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Table 10.10: 2 Level segmentation: standard deviation

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>1.0</td>
<td>1.1</td>
<td>4.7</td>
<td>1.8</td>
<td>3.9</td>
<td>2.3</td>
<td>2.8</td>
<td>2.4</td>
<td>1.5</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>1.1</td>
<td>1.3</td>
<td>1.7</td>
<td>0.9</td>
<td>2.5</td>
<td>0.7</td>
<td>3.3</td>
<td>1.7</td>
<td>1.3</td>
</tr>
<tr>
<td>Femur R</td>
<td>4.3</td>
<td>2.4</td>
<td>6.8</td>
<td>4.6</td>
<td>6.9</td>
<td>2.8</td>
<td>2.1</td>
<td>4.6</td>
<td>5.8</td>
</tr>
<tr>
<td>Femur L</td>
<td>4.8</td>
<td>3.4</td>
<td>2.7</td>
<td>3.4</td>
<td>6.2</td>
<td>3.0</td>
<td>3.4</td>
<td>4.4</td>
<td>6.7</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>0.7</td>
<td>1.1</td>
<td>0.7</td>
<td>1.1</td>
<td>0.7</td>
<td>0.8</td>
<td>0.9</td>
<td>1.0</td>
<td>0.4</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>2.5</td>
<td>2.2</td>
<td>3.8</td>
<td>1.2</td>
<td>1.0</td>
<td>0.4</td>
<td>0.2</td>
<td>0.6</td>
<td>0.6</td>
</tr>
</tbody>
</table>

Table 10.11: Segmentation using HT: average distance
Figure 10.4: Segmentation result with an the PBR method

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>0.8</td>
<td>0.6</td>
<td>4.4</td>
<td>1.3</td>
<td>3.5</td>
<td>3.3</td>
<td>2.5</td>
<td>1.7</td>
<td>0.8</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>0.5</td>
<td>0.6</td>
<td>0.7</td>
<td>0.5</td>
<td>1.1</td>
<td>0.5</td>
<td>3.3</td>
<td>0.9</td>
<td>0.8</td>
</tr>
<tr>
<td>Femur R</td>
<td>3.2</td>
<td>1.6</td>
<td>6.1</td>
<td>3.9</td>
<td>7.1</td>
<td>2.2</td>
<td>1.7</td>
<td>3.2</td>
<td>5.0</td>
</tr>
<tr>
<td>Femur L</td>
<td>4.0</td>
<td>1.7</td>
<td>2.1</td>
<td>2.8</td>
<td>5.8</td>
<td>2.5</td>
<td>3.8</td>
<td>3.0</td>
<td>5.7</td>
</tr>
<tr>
<td>Cartilage R</td>
<td>0.4</td>
<td>0.4</td>
<td>0.3</td>
<td>0.5</td>
<td>0.3</td>
<td>0.4</td>
<td>0.5</td>
<td>0.4</td>
<td>0.2</td>
</tr>
<tr>
<td>Cartilage L</td>
<td>21.4</td>
<td>1.3</td>
<td>1.5</td>
<td>0.4</td>
<td>2.9</td>
<td>2.0</td>
<td>0.3</td>
<td>0.3</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Table 10.12: Segmentation using HT: standard deviation
Table 10.13: Landmark distances to the segmented pelvis: average distance

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>aep l</td>
<td>0.45</td>
<td>1.49</td>
<td>0.67</td>
<td>0.80</td>
<td>0.81</td>
<td>1.16</td>
<td>0.63</td>
<td>0.73</td>
<td>0.53</td>
</tr>
<tr>
<td>ais l</td>
<td>0.57</td>
<td>0.33</td>
<td>0.25</td>
<td>0.35</td>
<td>0.30</td>
<td>0.14</td>
<td>0.26</td>
<td>0.06</td>
<td>0.19</td>
</tr>
<tr>
<td>aep r</td>
<td>0.47</td>
<td>0.55</td>
<td>0.58</td>
<td>1.44</td>
<td>0.72</td>
<td>0.52</td>
<td>0.66</td>
<td>0.49</td>
<td>1.06</td>
</tr>
<tr>
<td>ais r</td>
<td>0.97</td>
<td>0.51</td>
<td>1.22</td>
<td>1.14</td>
<td>0.73</td>
<td>0.72</td>
<td>1.46</td>
<td>0.28</td>
<td>0.72</td>
</tr>
</tbody>
</table>

Table 10.14: Landmark distances to the segmented pelvis: standard deviation

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>aep l</td>
<td>0.08</td>
<td>1.28</td>
<td>0.42</td>
<td>0.50</td>
<td>0.44</td>
<td>0.60</td>
<td>0.33</td>
<td>0.28</td>
<td>0.39</td>
</tr>
<tr>
<td>ais l</td>
<td>0.14</td>
<td>0.11</td>
<td>0.12</td>
<td>0.14</td>
<td>0.11</td>
<td>0.07</td>
<td>0.14</td>
<td>0.04</td>
<td>0.08</td>
</tr>
<tr>
<td>aep r</td>
<td>0.36</td>
<td>0.28</td>
<td>0.39</td>
<td>0.73</td>
<td>0.57</td>
<td>0.11</td>
<td>0.31</td>
<td>0.19</td>
<td>0.74</td>
</tr>
<tr>
<td>ais r</td>
<td>0.42</td>
<td>0.27</td>
<td>0.35</td>
<td>0.98</td>
<td>0.24</td>
<td>0.62</td>
<td>0.14</td>
<td>0.10</td>
<td>0.38</td>
</tr>
</tbody>
</table>

All evaluations have been carried out on a workstation based on an Intel Core2Duo CPU with 2.2 GHz and 2Gbyte RAM. The processing time of the prepositioning using HT and adaptation are given in the table 10.15. The adaptation takes over 300 seconds.

Table 10.15: Processing time of the HT

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>HT of model</td>
<td>1.38</td>
<td>1.39</td>
<td>1.20</td>
<td>1.34</td>
<td>1.28</td>
<td>1.35</td>
<td>1.30</td>
<td>1.44</td>
<td>1.44</td>
<td>3.81</td>
</tr>
<tr>
<td>HT of Image</td>
<td>11.97</td>
<td>11.64</td>
<td>10.06</td>
<td>11.93</td>
<td>11.58</td>
<td>11.27</td>
<td>11.17</td>
<td>11.27</td>
<td>11.29</td>
<td>11.32</td>
</tr>
<tr>
<td>Hough Vote</td>
<td>230.06</td>
<td>231.99</td>
<td>228.72</td>
<td>237.67</td>
<td>224.67</td>
<td>231.77</td>
<td>227.63</td>
<td>230.34</td>
<td>230.30</td>
<td>231.02</td>
</tr>
<tr>
<td>Total</td>
<td>248.61</td>
<td>243.95</td>
<td>239.11</td>
<td>240.08</td>
<td>237.63</td>
<td>239.67</td>
<td>243.33</td>
<td>243.05</td>
<td>242.52</td>
<td>242.93</td>
</tr>
</tbody>
</table>

These values show that the computation time of the HT of the image as well as the HT of the femur model has acceptable range.

Figure 10.5 illustrates the HT prepositioned meshes.

The following tables show the results of a visual evaluation of different segmentation workflows. The symbols used to rate the quality are:

- ++ mesh matches the anatomical structure
- + mesh matches the anatomical structure with local deviation
- – mesh deviates from the anatomical structure in greater parts
- −− total misadaption of the mesh

10.2 Conclusion

Even though the aim of this study was the automatisation of the segmentation of the female hip, in the beginning only five data sets of male patients were available. For this reason only the model of a male hip could be prepared at that time. The
Figure 10.5: Prepositioning using HT

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>++</td>
<td>–</td>
<td>++</td>
<td>++</td>
<td>–</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>–</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Femur R</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Femur L</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
</tbody>
</table>

Table 10.16: Visual evaluation of the segmentation results using manual prepositioning, translation and Rotation

<table>
<thead>
<tr>
<th></th>
<th>001</th>
<th>002</th>
<th>003</th>
<th>004</th>
<th>007</th>
<th>008</th>
<th>009</th>
<th>010</th>
<th>011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pelvis R</td>
<td>++</td>
<td>–</td>
<td>++</td>
<td>++</td>
<td>–</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>–</td>
</tr>
<tr>
<td>Pelvis L</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>++</td>
<td>++</td>
<td>+</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Femur R</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Femur L</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
<td>++</td>
</tr>
</tbody>
</table>

Table 10.17: Visual evaluation of the segmentation results using HT prepositioning
model-based segmentation was carried out based on this model. The result of the model-based segmentation can be influenced by adapting a set of input parameters. Hitherto these parameters had been chosen based on experience. The aim now was to optimise the parameters by means of a search algorithm. Since the range of values of the parameters is very large, a heuristic was used to keep the search time within acceptable limits. With the help of the simulated annealing algorithm parameters could be determined which led to better segmentation results.

Gradually nine images of female patients also became available. Manual segmentation of these would have been too laborious because of the large mass of data. Therefore the female hips and femurs were segmented using the male models as input mesh.

Since the male model cannot capture the geometry of a female model adequately, the first results of segmentation were relatively unsatisfactory. Only few data sets were segmented properly. The number of images of female patients was at that time not enough to generate a representative mean model. But manual prepositioning of the male models improved the results markedly. Segmentation through this method was successful in the case of 7 of 9 data sets, even though once again a male model was used as the input mesh. For this reason the first approach was to develop an automated prepositioning algorithm. This task was taken up by the Hough transform.

As more data sets became available mean models of all anatomical structures were generated from the individual models. These models now had the characteristics of a female hip. To test the quality of the model leave-one-out cross-validation was applied. In this, the respective individualised models of the data sets to be segmented were left out while generating the mean models. Using the Hough transform prepositioning with mean models all data sets are segmented properly.

Apart from the Hough transform, point-based registration was also tested as a prepositioning algorithm.
A two level segmentation was tested as another possibility for prepositioning. In the first segmentation cycle the parameters were set up so that the model as a whole moved, but hardly changed its form. By contrast, in the second cycle the model adapts itself locally to the structure of the anatomical object. None of the data sets was segmented properly.

To improve the segmentation an algorithm that changes the geometry of the models was tested. Adjacent faces of the model, whose surface normals were such that the angle between them exceeded a certain value, were refined. It has been observed that refining the used meshes does not lead to better segmentation results.
Chapter 11

Outlook

The potential employment of computer assisted surgical procedures is being investigated in the context of the project "Triple/MR-Analyse der Beckendreifachosteotomie". In the course of this investigation the triple pelvic osteotomy (TPO) operation is being accompanied by the navigation system "OrthoPilot". The aim of this operation is the covering of the femoral head in as perfect a manner as possible in female patients with hip dysplasia.

The navigation system "OrthoPilot" protocols the coordinates of certain reference points on the hips during the operation. Further, the orientation of the acetabulum before and after the operation is presented as a rotation matrix. The anteversion and inclination are determined by evaluating this information (see figure 3.3). This allows subsequent determination of the angle of correction of the acetabulum.

The findings on segmentation obtained through the present study could be utilised for 3D visualisation in computer assisted surgery. Specialists would thus be given the opportunity to simulate the result of an operation beforehand. Apart from this, the computer would gain exact information on the geometry and location of anatomical structures, which it could use to propose the course of an operation.
Bibliography


