Global and Local Adaption of Geometric Deformable Models for Medical Image Segmentation

Vom Fachbereich Ingenieurwissenschaften der

Universität Duisburg-Essen

zur Erlangung des akademischen Grades eines

Doktor der Naturwissenschaften

genehmigte Dissertation

von

Zhenyu Tang

aus Anhui, V.R. China

Referent: Prof. Dr. Josef Pauli
Korreferent: Prof. Dr. Wolfram Luther
Tag der mündlichen Prüfung: 19.04.2011
Acknowledgement

Behind every dissertation, there is years of hard work filled with the exaltation of success and the disappointment of failure. Certainly, this dissertation is no exception. It is impossible to be accomplished without the help from my advisor, my colleagues, and my friends. I owe a great deal of gratitude to them.

First of all, I would like to thank my advisor Prof. Dr. Josef Pauli, the head of the research group of intelligent system, for his inspiration and patient. His high academic standing and experiences in the field of image processing gave me the best guidance. I also would like to thank Prof. Dr. Wolfram Luther for giving me many valuable advices concerning the dissertation. Thank my colleagues Dipl. Inform. Johannes Herwig, Dipl. Inform. Michael Müller, Dipl. Inform. Maria Sagrebin, Dipl. Math. Anastasia Noglik, and Dipl. Inform. Kun Zhao, for their helps, encouragements and suggestions.

I would like to thank M. Phys. Oliver Kraff from the Universitätsklinikum Essen for providing me with all the MR image data and Prof. Dr. med. Franz Löer for giving me so much anatomical knowledge. Thank Prof. Dr. Andrés Kecskeméthy for proposing many valuable suggestions concerning the algorithms.

Many thanks to M. Sc. Yijun Guo, M. Sc. Xiaojie Liu, and M. Sc. Haider Albassam from the research group of mechanics and robotics at the Universität Duisburg-Essen for their helps and inspirations.

At last, I would like to thank my father Rushan Tang and mother Jinli Li, for their support and encouragement.
To my parents
# Contents

**Acknowledgement** III  
**Notation and Symbols** VIII  
- Acronyms ........................................ VIII  
- Symbols ........................................ IX  
**Abstract** XII  

## 1 Introduction  
1.1 Medical Image Segmentation .................................. 3  
  1.1.1 Difficulties .................................... 4  
  1.1.2 Segmentation Methods .................................. 4  
  1.1.3 Model-Based Image Segmentation Methods .............. 5  
1.2 Magnetic Resonance Imaging (MRI) .......................... 8  
  1.2.1 T1-weighted MRI .................................. 10  
  1.2.2 T2-weighted MRI .................................. 11  
  1.2.3 Proton-weighted MRI .................................. 12  
1.3 Motivation of Tasks ........................................ 13  
1.4 Tasks and Challenges ....................................... 14  
  1.4.1 Segmentation of Limb Bones and Extraction of FKF ... 15  
  1.4.2 Segmentation of Muscle with Tendon Attachment Sites (TASs) ... 17  
  1.4.3 Spine Curve Extraction .............................. 18  
1.5 Outline ........................................ 20  

## 2 Overview of Segmentation Methods  
2.1 Intensity Based Method .................................... 21  
  2.1.1 Thresholding .................................... 21  
  2.1.2 Region Growing .................................. 22  
2.2 Geometric Deformable Model Based Method .................. 23  
  2.2.1 Snakes: Active Contour Model ....................... 23  
  2.2.2 Active Shape Model (ASM) ......................... 27  
  2.2.3 Atlas Based Method .................................. 31  
2.3 Supervised Learning Methods ................................ 34  
  2.3.1 Support Vector Machine (SVM) ....................... 34

## 3 Segmentation of Rigid Body Organ: Femur and FKF  
3.1 Introduction .......................................... 37  
  3.1.1 State of the Art .................................. 38  
  3.1.2 Overview of Rigid Body Organ Segmentation .......... 38  
3.2 Pre-processing ........................................ 39
3.2.1 Mean-Shift Filter ........................................... 40
3.2.2 Thresholding and Region Growing ......................... 41
3.3 The 3D Femur Model ........................................... 42
3.4 Global Adaption ................................................. 43
  3.4.1 Rotation of Femur Axis .................................... 43
  3.4.2 Scaling of The Vertical Span Length ....................... 47
  3.4.3 Position Translation ...................................... 48
3.5 Local Adaption ................................................ 48
  3.5.1 Model Contour Extraction .................................. 49
  3.5.2 Affine Transformation and Non-Rigid Matching ............ 50
  3.5.3 The Extraction of FKF .................................... 52
3.6 3D Femur Reconstruction ....................................... 53
3.7 Evaluation ..................................................... 54
  3.7.1 Evaluation of Pre-processing Results ....................... 54
  3.7.2 Evaluation of Global Adaption ............................ 56
  3.7.3 Evaluation of Local Adaption ............................. 59
  3.7.4 Evaluation of FKF Results ............................... 61
3.8 Conclusions .................................................. 62

4 Segmentation of Soft Body Organ: Muscle and TAS ............. 64
  4.1 Introduction .................................................. 64
  4.1.1 State of the Art .......................................... 64
  4.1.2 Overview of Soft Body Organ Segmentation ................ 65
  4.2 Calculation of Active Shape Models of Bones and Sartorius Muscle .... 67
    4.2.1 Creation of two 2D ASM of Bones Cross Sections .......... 67
    4.2.2 Creation of 3D ASM of Muscle .......................... 70
  4.3 Global Adaption of the 3D ASM of Muscle .................... 71
    4.3.1 Vertical Span Length (VSL) ............................ 71
    4.3.2 Muscle Alignment ...................................... 74
  4.4 Local Adaption of 3D ASM of Muscle ........................ 74
  4.5 Tendon Attachment Sites (TASs) Extraction .................... 75
  4.6 Evaluation .................................................. 77
    4.6.1 Evaluation of Muscle VSL ............................... 77
    4.6.2 Evaluation of Alignment of the ASM of Muscle .......... 78
    4.6.3 Evaluation of Local Adaption ........................... 82
    4.6.4 Evaluation of TAS Extraction ........................... 85
  4.7 Conclusions ................................................ 85

5 Segmentation of Articulated Body Organ: Spine ................. 87
  5.1 Introduction ................................................ 87
    5.1.1 State of the Art ....................................... 88
    5.1.2 Overview of Articulated Body Organ Segmentation ........ 89
  5.2 Intervertebral Disk .......................................... 90
  5.3 Gradient-Based Intervertebral Disk Extraction ............... 91
  5.4 Global Adaption ............................................. 93
    5.4.1 Graph-Based Filter Combining with Intensity Profile ...... 93
    5.4.2 Active Shape Model (ASM) Based Non-Disk Filter .......... 97
## Contents

5.5 Local Adaption .............................................. 99  
5.5.1 Vertebra Registration and Spine Curve Extraction .......... 99  
5.6 Evaluation ................................................... 104  
5.6.1 Evaluation of the Disk Position Extraction Results ........ 104  
5.6.2 Evaluation of the Spine Curve Results .................... 105  
5.7 Conclusions ............................................... 106  

6 Applications ................................................... 108  
6.1 Bone Segmentation Application .................................. 108  
6.1.1 Bone Segmentation and 3D Reconstruction .................... 108  
6.1.2 Femur Cavity Volume Extraction ........................... 109  
6.1.3 Femur Segmentation of X-ray Image ......................... 110  
6.1.4 Superquadrics based Matching ................................ 111  
6.1.5 3D Gait Modeling ........................................... 112  
6.2 Muscle Segmentation Application ................................ 113  
6.3 Spine Curve Extraction Application ............................ 114  

7 Conclusions and Perspectives .................................... 115  
7.1 Rigid Body Organ Segmentation .................................. 115  
7.2 Soft Body Organ Segmentation .................................. 116  
7.3 Articulated Body Organ Segmentation ........................... 116  

A Appendix ......................................................... 118  
A.1 Disk Position Extraction using Different Scan Machine and Protocol . . 118  

Bibliography ......................................................... 121
# Notation and Symbols

## Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASM</td>
<td>Active Shape Model</td>
</tr>
<tr>
<td>CAD</td>
<td>Computer Assisted Diagnosis</td>
</tr>
<tr>
<td>CAS</td>
<td>Computer Assisted Surgery</td>
</tr>
<tr>
<td>CR</td>
<td>Correlation Ratio</td>
</tr>
<tr>
<td>CT</td>
<td>Computer Tomography</td>
</tr>
<tr>
<td>DAG</td>
<td>Directed Acyclic Graph</td>
</tr>
<tr>
<td>FKF</td>
<td>Functional Kinematic Feature</td>
</tr>
<tr>
<td>LM</td>
<td>Longitudinal Magnetization</td>
</tr>
<tr>
<td>MI</td>
<td>Mutual Information</td>
</tr>
<tr>
<td>MR</td>
<td>Magnetic Resonance</td>
</tr>
<tr>
<td>MRI</td>
<td>Magnetic Resonance Imaging</td>
</tr>
<tr>
<td>NMI</td>
<td>Normalized Mutual Information</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Components Analysis</td>
</tr>
<tr>
<td>PDE</td>
<td>Partial Differential Equations</td>
</tr>
<tr>
<td>PET</td>
<td>Positron Emission Tomography</td>
</tr>
<tr>
<td>RF</td>
<td>Radio Frequency</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine</td>
</tr>
<tr>
<td>TAS</td>
<td>Tendon Attachment Site</td>
</tr>
</tbody>
</table>
Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TE</td>
<td>Echo Time</td>
</tr>
<tr>
<td>TM</td>
<td>Transverse Magnetization</td>
</tr>
<tr>
<td>TR</td>
<td>Repetition Time</td>
</tr>
<tr>
<td>VRML</td>
<td>Virtual Reality Modeling Language</td>
</tr>
<tr>
<td>VSL</td>
<td>Vertical Span Length</td>
</tr>
</tbody>
</table>

Mathematical Symbols

- \( B_o \): External Magnetic Field Strength (MRI)
- \( C \): 2D Geometric Contour
- \( C(s) \): Parametric Representation of 2D Geometric Contour
- \( CR(A, B) \): Correlation Ratio of A and B (ATLAS)
- \( C_O \): Covariance Matrix (ASM)
- \( C_s(s) \): First Order Derivative of \( C(s) \)
- \( C_{model} \): Geometric Model
- \( C_{organ} \): Form of Organ
- \( C_{ss}(s) \): Second Order Derivative of \( C(s) \)
- \( E \): Deformation Model (SNAKES) (ASM) (ATLAS)
- \( H(A) \): Entropy of A (ATLAS)
- \( H(A, B) \): Joint Entropy of A and B (ATLAS)
- \( I \): Image Pixel Intensity (MEAN-SHIFT FILTER)
- \( I(x, y) \): Image Intensity Function
- \( J \): Jaccard Index
- \( L_f \): Real Length of Femur (RIGID BODY SEGMENTATION)
- \( L_m \): Real Length of Femur Model (RIGID BODY SEGMENTATION)
- \( MI(A, B) \): Mutual Information of A and B (ATLAS)
- \( V_f \): Vertical Span Length of Femur (RIGID BODY SEGMENTATION)
- \( V_m \): Vertical Span Length of Femur Model (RIGID BODY SEGMENTATION)
- \( X(\overrightarrow{d_i}) \): Function \( i \) Return the Value of the X Component of Vector \( \overrightarrow{d_i} \) (ASM)
- \( Y(\overrightarrow{d_i}) \): Function \( i \) Return the Value of the Y Component of Vector \( \overrightarrow{d_i} \) (ASM)
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Gamma$</td>
<td>Region within the Model Contour</td>
</tr>
<tr>
<td>$\Omega$</td>
<td>Region within the Manually Segmented Contour</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>Gyromagnetic Ratio (MRI)</td>
</tr>
<tr>
<td>$x_i$</td>
<td>Array Containing Position of Labeled Points of the $i$-th Training Instance (ASM)</td>
</tr>
<tr>
<td>$w$</td>
<td>Normal of the Hyperplane in SVM</td>
</tr>
<tr>
<td>$\nabla I(x, y)$</td>
<td>Gradient at Image Position $(x, y)$</td>
</tr>
<tr>
<td>$\nabla e_{\text{ext}}(C(s))$</td>
<td>Gradient of External Energy $e_{\text{ext}}(C(s))$ (SNAKES)</td>
</tr>
<tr>
<td>$\nu$</td>
<td>Larmor Frequency (MRI)</td>
</tr>
<tr>
<td>$\mathbf{x}$</td>
<td>Array Containing Mean Position of Labeled Points of Set of Training Instances $x_i$ (ASM)</td>
</tr>
<tr>
<td>$\mathbf{P}_i$</td>
<td>$i$-th Eigenvectors, $i = 1, 2, \ldots, N$ (ASM)</td>
</tr>
<tr>
<td>$\mathbf{d}_i$</td>
<td>Vector Pointing From Labeled Point of Model to Object Boundary (ASM)</td>
</tr>
<tr>
<td>$\overrightarrow{f}_{\text{ext}}$</td>
<td>Internal Force (SNAKES)</td>
</tr>
<tr>
<td>$\overrightarrow{f}_{\text{int}}$</td>
<td>Internal Force (SNAKES)</td>
</tr>
<tr>
<td>$\overrightarrow{n}_{\perp}$</td>
<td>Vector Perpendicular to Image Gradient (SNAKES)</td>
</tr>
<tr>
<td>$\mathbf{P}$</td>
<td>Array of Eigenvectors (ASM)</td>
</tr>
<tr>
<td>$\mathbf{b}$</td>
<td>Array of Weighting Factors (ASM)</td>
</tr>
<tr>
<td>$\theta(x, y)$</td>
<td>Angle of Gradient Direction at Position $(x, y)$ (SNAKES)</td>
</tr>
<tr>
<td>$\varphi(x, y)$</td>
<td>Implicit Function (LEVEL-SET)</td>
</tr>
<tr>
<td>$a$</td>
<td>Constant Value (LEVEL-SET)</td>
</tr>
<tr>
<td>$b_i$</td>
<td>$i$-th Element in $\mathbf{b}$ (ASM)</td>
</tr>
<tr>
<td>$c$</td>
<td>Image Pixel Coordinates (MEAN-SHIFT FILTER)</td>
</tr>
<tr>
<td>$c_{\text{oi}}$</td>
<td>Element at the $k$-th Row and the $l$ column of Covariance Matrix $C_o$ (ASM)</td>
</tr>
<tr>
<td>$dx_i$</td>
<td>Array Containing the Difference Between the Mean $\overline{x}$ and the $i$-th Instance $x_i$ (ASM)</td>
</tr>
<tr>
<td>$d_{\text{contour}}$</td>
<td>Contour Difference</td>
</tr>
<tr>
<td>$d_{\text{pose}}$</td>
<td>Pose Difference</td>
</tr>
<tr>
<td>$d_{\text{position}}$</td>
<td>Position Difference</td>
</tr>
<tr>
<td>$d_{\text{size}}$</td>
<td>Size Difference</td>
</tr>
<tr>
<td>$dr$</td>
<td>Direction of a Gradient Image Region</td>
</tr>
<tr>
<td>$e'$</td>
<td>Deformation Method (SNAKES) (ASM) (ATLAS)</td>
</tr>
<tr>
<td>$e(C_{\text{model}})$</td>
<td>Fitting Criterion Function (SNAKES) (ASM) (ATLAS)</td>
</tr>
</tbody>
</table>
Symbols

\( e_{\text{con}}(C(s)) \) External Constraints (SNAKES)
\( e_{\text{edge}}(C(s)) \) Edge Term of Image Forces (SNAKES)
\( e_{\text{img}}(C(s)) \) Image Forces (SNAKES)
\( e_{\text{int}}(C(s)) \) Internal Energy (SNAKES)
\( e_{\text{line}}(C(s)) \) Line Term of Image Forces (SNAKES)
\( e_{\text{term}}(C(s)) \) Termination Term of Image Forces (SNAKES)
\( g_I(x) \) Kernel Function (MEAN-SHIFT FILTER)
\( g_s(x) \) Kernel Function (MEAN-SHIFT FILTER)
\( h \) Plank Constant (MRI)
\( i \) Index, \( i = 1, 2, \ldots, N \)
\( p \) Image Pixel (MEAN-SHIFT FILTER)
\( r \) Circle Radius (LEVEL-SET)
\( w_s \) Weighted Factor of First Order Derivative of Contour (SNAKES)
\( w_s(s) \) Weighted Factor Function of First Order Derivative of Contour (SNAKES)
\( w_{\text{edge}} \) Weighted Factor of Edge Term of Image Forces (SNAKES)
\( w_{\text{line}} \) Weighted Factor of Line Term of Image Forces (SNAKES)
\( w_{ss} \) Weighted Factor of Second Order Derivative of Contour (SNAKES)
\( w_{ss}(s) \) Weighted Factor Function of Second Order Derivative of Contour (SNAKES)
\( w_{\text{term}} \) Weighted Factor of Constraint of Terminations of Image Forces (SNAKES)
Abstract

This thesis focuses on the research of medical image segmentation using geometric deformable models. A processing pattern called the global and local model adaption is proposed and applied to design a general scheme for segmentation. The principle of this processing pattern is a step by step, coarse to fine adaption process which makes the geometric deformable model match the corresponding object (organ) of interest. According to such a processing pattern, the segmentation process can be divided into three stages, which are the pre-processing stage, the global adaption stage, and the local adaption stage.

The pre-processing stage deals with, for example, the enhancement of the quality of the medical image and the pre-extraction of salient features of the object of interest. The result of the pre-processing is put into the global adaption stage, whereby the approximate features about the geometry and space (e.g., pose, size and location) of the object of interest are obtained from the pre-processing result. Then the geometric deformable model is adjusted to have the same features as the object. The local adaption stage tackles contour differences between the model and the object. These differences are to be removed in order to achieve the final segmentation result.

Three segmentation methods based on geometric deformable models are proposed to solve the problems of the segmentation of the rigid body organ, the soft body organ and the articulated body organ. All the methods are implemented and evaluated in three different segmentation tasks, which were demanded by a research project. These tasks focus on different kinds of human organs, which are bones (rigid body), muscles (soft body) and the spine (articulated body). The segmentation results of these tasks are used for gait modeling. The purpose of the gait modeling is to find the origins of the gait pathology of patients. To perform the gait modeling, 3D organs are virtually constructed based on the segmentation results of the organs of interest from the Magnetic Resonance (MR) image data. These 3D organs are combined with a gait data set, which is produced within a so-called Gait-laboratory environment. Each gait data indicate the motions of a part of the patient’s body. For each part of the patient’s body, there is a relevant 3D organ. By fusing each 3D organ with its corresponding gait data, the gait of the patient can be modeled and visualized in a 3D environment in order to give the medics an intuitional view that will alleviate the difficulties in the medical diagnosis.

The contribution of this thesis is to propose a global and local model adaption pattern for the medical image segmentation using geometric deformable models. Based on this framework, three methods are proposed for the semi- or fully automatic segmentation of the rigid body organ, the soft body organ, and the articulated body organ of the human from medical images.
1 Introduction

Due to the tremendous achievements in the field of physics as well as the highly developed computer technology, the application of medical imaging, which is a great invention in the 20th century, has arisen, and it plays an increasingly important role in medical diagnosis and science. Today it is an indispensable and powerful tool used in the noninvasive internal body imaging, disease diagnosis and biological study. However, it has some inherent problems too. The great challenges of using medical imaging in the clinical diagnosis are the large image data and the low image quality which make the manual image analysis a huge labor-intensity and inefficient work. Therefore, a new research field: the medical image processing was arisen, and it has been widely studied in the last two decades. At the beginning, only pixel-based methods were applied for the purpose of the image quality enhancement e.g., histogram transformation [46] (equalization and normalization) and the image noise reduction e.g., the low pass filter [30] such as the Gaussian filter [83] and the rank filter (e.g., the median filter [41]). Later, many high-technical methods, which are integrated with knowledge from the fields of artificial intelligence, statistics, physics etc., were proposed in succession. Most of them focused on the problem of the medical image segmentation which is still a hot research topic now. The medical image segmentation methods have been widely applied in Computer Assisted Diagnosis (CAD) and Computer Assisted Surgery (CAS). By means of the segmentation methods, damaged tissues and interesting organs can be detected and segmented automatically or semi-automatically. The workload of manual operations on medical images is significantly relieved. Furthermore, segmented tissues and organs of the human body can be visualized in a 3D environment in order to make a more intuitive view and even a navigation of the human body. Actually, such kind of technology has been already implemented in operating rooms. With the help of the 3D human body navigation, medics can easily reach to the expected position inside the human body or simulate an operation to design a suitable surgery plan.

Many methods dealing with the medical image segmentation have been proposed till now [70]. Some of them have been proved to be very effective. However, nearly all of these methods suffer from the low medical image quality and the ambiguous form of organs of interest. To solve this problem, a model-based medical image segmentation was proposed. The advantages of the model-based method are the inherent properties of image noise proof and highly adaptive capacity. Because of these advantages, the model-based segmentation method becomes the most studied subject in the medical image segmentation. Many sophisticated model-based segmentation algorithms [57, 91] have been proposed with success. The research described in this thesis works in the field of medical image segmentation. It focuses on the model-based segmentation method, more concretely, it focuses on the geometric deformable model based segmentation where the processing scheme is designed in the global and local model adaption pattern. The geometric deformable model mentioned in this thesis is defined as a combination of a geometric model of an object of interest (e.g., a 3D structure model of a human organ) noted as $C_{model}$, and a deformation
model noted as $E$. Strictly speaking, the deformation model $E$ is an algorithm defining the deformation strategy of each element (e.g., point) of the geometric model. Some well-known deformation models are, for example, the active contour model i.e., snakes [54], the Active Shape Model (ASM) [26] and the level-set [64]. Actually, in the original articles of the above mentioned algorithms, not only the deformation strategy but also the geometric contour is concerned. For instance, in snakes, there is a parametric function which defines a geometric contour. Since the essential content of these algorithms is about the deformation strategy while the geometric representation is not emphasized, these algorithms are regarded as the deformation models.

Commonly, the preliminary geometric model $C_{\text{model}}$ is fixed, and the corresponding organ to be segmented (noted as $C_{\text{organ}}$) varies between individuals. Therefore, differences between $C_{\text{model}}$ and $C_{\text{organ}}$ are inevitable, and these differences exist in size $d_{\text{size}}$, pose $d_{\text{pose}}$, position $d_{\text{position}}$ and contour $d_{\text{contour}}$. The relation between $C_{\text{model}}$ and $C_{\text{organ}}$ can be expressed as

$$C_{\text{organ}} = C_{\text{model}} + (d_{\text{size}} + d_{\text{pose}} + d_{\text{position}}) + d_{\text{contour}}.$$  (1.1)

This equation summarizes the idea and the objective of the medical image segmentation based on the geometric deformable model. $d_{\text{size}}$, $d_{\text{pose}}$ and $d_{\text{position}}$ in the brackets make up the global difference, while the $d_{\text{contour}}$ is known as the local difference.

The principle of the global and local model adaption pattern is a step by step coarse to fine adaption process, and the objective is to change the geometric model in order to make it match the corresponding organ in medical images. Based on the geometric deformable model and the processing pattern of global and local model adaption, three methods are proposed in this thesis to solve the problem of the segmentation of the rigid body, the soft body and the articulated body organs from medical images respectively. The rigid body organ is the organ which is not deformable e.g., the bones, while the soft body organ is dynamic, and its form always changes over time e.g., the lung and the heart. The articulated body organ e.g., the spine is composed of rigid and soft body organs, both of which are alternately connected. Globally, the articulated body organ behaves like the soft body organ, but locally the articulated body organ is similar to the rigid body organ.

According to the global and local model adaption pattern, each proposed method consists of three main stages (see Fig. 1.1). The first is the image pre-processing stage, where the image quality is enhanced or the features of the organ of interest are pre-extracted. The second stage is the global adaption stage, where the geometric model is adjusted to have the same size, pose and position as the organ. The third stage is the local adaption stage, where the contour differences between the geometric model and the organ are removed. According to equation 1.1, after the global and the local adaption stages, the geometric
model $C_{\text{model}}$ should be totally matched to the organ $C_{\text{organ}}$, thereby the segmentation is accomplished.

The main difference among these three proposed methods is in the dynamic level of the body of the interesting organ, and this difference causes the use of different approaches in the global and local adaption stages. Because the soft and the articulated body organs are more dynamic than the rigid body organ, in the global adaption stage, the approach for calculating the pose of the rigid body organ is easier than the soft and the articulated body organs. The same situation exists in the local adaption, where the variability of the contour of the rigid and the articulated body organs is lower than the soft body organ. It means that, in the local adaption stage, the deformation model $E$ for the soft body organ have to be more sophisticated than the other two kinds of organs.

The methods proposed in this thesis are implemented and evaluated in a research project called the PROREOP [71] which aims to develop a diagnosis system to assist the operation and rehabilitation measurement of the human musculoskeletal system. The methods for the segmentation of the rigid body, the soft body and the articulated body organs are applied in three applications of the project, each of which deals with the segmentation of human limb bones, muscles and the spine from MR images, respectively.

In the rest of this chapter, an introduction of the medical image segmentation methods is presented in Section 1.1, where different image segmentation methods, especially the methods based on the geometric deformable model are discussed. Because the type of the medical image used in this thesis is the Magnetic Resonance Imaging (MRI) image, it is necessary to give a short introduction about the principle of the MRI, and this introduction is written in Section 1.2. As has been noted, evaluations of all the methods proposed in this thesis are made by investigating the performance in the applications of the project PROREOP. The motivation of the project is described in Section 1.3. Then the objectives and the difficulties regarding the tasks of the three applications are discussed in Section 1.4. If no special notation, all of the medical images appearing in this thesis are produced by the Universitätsklinikum Essen.

1.1 Medical Image Segmentation

Because of the highly developed medical imaging technology, many kinds of imaging methods have been invented and used in clinical diagnosis, such as the commonly used X-ray imaging, Computer Tomography (CT), MRI and Positron Emission Tomography (PET). The organ of interest can be clearly inspected without invasive interventions in the human body. Generally, recognition and segmentation of interesting organs from medical images are done manually. However, common medical image data are often very large, e.g., a 3D MR data set of an adult lower limb is composed of more than 100 images, therefore the manual image segmentation is a highly time-consuming task. But with the help of the computer and the medical image segmentation methods, the boring slice by slice manual analysis of the image data can be significantly relieved from the medics. In the last two decades, many kinds of the medical image segmentation methods have been proposed and implemented with success. Today this research field still attracts more and more academics to make the further achievements.
1 Introduction

1.1.1 Difficulties

The medical image segmentation methods as compared with other segmentation methods using real-world images confront more challenges. Unlike the real-world images which are generated from visible light sensitive devices (e.g., CMOS and CCD), most medical images are produced from non-visible signals (having higher or lower frequency than the visible light). For example, the MRI detects the radio which is emitted from the protons. The PET uses the positrons which come from the radioisotope injected in the human body, and the ultrasonography focuses on the reflexed ultrasonic waves. Moreover, due to the consideration of security and human healthy, the signal strength is kept at a safe level. Therefore, a highly sensitive sensor is required to capture such weak signals. The problem is that the sensor is sensitive not only to the signals but also to the influences from the surroundings. Therefore, the image noise is one of the inherent properties of the medical images, and it is one of the main reasons why the medical image segmentation is more difficult than the real-world image segmentation. In the following, the major challenges to the medical image segmentation are listed.

1. The image noise is the inherent problem of the medical image. In the MR imaging, the noises come from many aspects, most of which are difficult or impossible to avoid, e.g., the interfered magnetic field, the interfered emitted signal from the hydrogen atom, and the signal encoding.

2. The partial volume effect [113] which means the intensity of each medical image pixel is a mixed value of different types of tissues. It is caused by the low scan resolution, and it could make the same tissue have nonhomogeneous gray values.

3. High variability of the anatomic structures between individuals and even the same person at different time.

4. Complex and highly irregular anatomic structures e.g., vessel trees in the angiogram.

5. Many kinds of image modalities (e.g., CT, MR) and scan protocols (e.g., T1, T2) are available for the medical image production. The same organ may have totally different appearances using different scan protocols.

1.1.2 Segmentation Methods

Based on the principle of the algorithm, the medical image segmentation methods can be divided into three classes:

1. The model-free method which only uses the information from the image to achieve the segmentation. (e.g., thresholding [41] and graph-based image cut method [101, 112]).

2. The model-based method which not only uses the content of the image but also integrates the prior knowledge of the organ of interest (e.g., the geometry [26], the appearance [23] and the location of the organ).

3. The classification method or the pattern recognition which can be further divided into the supervised method (e.g., the Support Vector Machine (SVM) [17, 43] and some kinds of the artificial neural network [2]) and the unsupervised method (e.g., the clustering K-Means [55], the Fuzzy C-means [9] and the K-NN [29]).
This thesis focuses on the geometric deformable model which belongs to the class of the model-based segmentation method. The basic knowledge of the model-based method and the components included in the geometric deformable model will be discussed in the following sections.

### 1.1.3 Model-Based Image Segmentation Methods

The word ”model” is not strange to most of us, and it can be heard in many occasions. But in different domains the model has different meanings. Referring to the Oxford dictionary, the model can be used to name

1. a representation of a person or thing, typically on a smaller scale
2. a mathematical description of a system or process, used to assist calculations and predictions
3. a person employed to display clothes by wearing them
4. a particular design or version of a product
5. an excellent example of a quality

Among these 5 definitions, the first two definitions are the closest descriptions of the model mentioned in this thesis. As noted before, the methods proposed in this thesis use the geometric deformable model which is composed of a geometric model $C_{\text{model}}$ and a deformation model $E$. The first definition of the model describes the geometric model, and the second definition is about the deformation model. To be exact, the word ”model” in this thesis refers to both the geometric model of an object of interest and the model deformation algorithm i.e., the deformation model.

The geometric model $C_{\text{model}}$ contains the general geometric information about the object of interest e.g., the structure, the size and the location of the object. Based on the geometric information, the object of interest can be segmented from the images by adjusting and deforming the initial geometric model to match the object in the images. This matching process is guided by the deformation model $E$. Concerning the deformation model $E$ in the model-based image segmentation, there are two main components which compose the deformation model $E$. The first component is the fitting criterion function noted as $e(C_{\text{model}})$, and it serves as a metric function of the matching quality between the geometric model and the corresponding object of interest in the images. The second component is the deformation method noted as $e'$, and it provides a deformation scheme for changing the initial geometric model. The reason why the second component is written as $e'$ is that, commonly, $e'$ is deduced from $e(C_{\text{model}})$ with the help of the Partial Differential Equations (PDE). The $e(C_{\text{model}})$ is commonly created by considering the image features of the object of interest (e.g., the object intensity and the boundary gradient), the knowledge introduced from physics (e.g., elasticity, rigidity) and the expert defined constraints. The image features and the expert defined constraints make the geometric model contour be attracted to the desired position (e.g., the object boundary), and the physical constraints make the geometric model contour act as an elastic string. The second component $e'$, which is deduced from the $e(C_{\text{model}})$, is applied to change $C_{\text{model}}$ to achieve an optimal solution of $e(C_{\text{model}})$.
which corresponds to the best segmentation result. These three elements ($C_{\text{model}}$, $e(C_{\text{model}})$ and $e'$) are foundations of the geometric deformable model based segmentation.

For example, in an application of the knee segmentation using the active contour method (snakes), a geometric model $C_{\text{model}}$ of a standard knee is used to define an initial knee contour. The initial knee contour changes according to the deformation model $E$ which in this example is the snakes algorithm. Each point of the initial knee contour moves iteratively in the direction provided by $e'$. After a number of iterations, the contour of $C_{\text{model}}$ should be freeze at the boundary of the knee in the images, in other words, an optimal solution (minimum) of $e(C_{\text{model}})$ is achieved. In the snakes algorithm, $e(C_{\text{model}})$ is also called the energy function whose value depends on the curvature of $C_{\text{model}}$, the image gray values, the image gradients and the user interactive interpretations. The optimal solution of $e(C_{\text{model}})$ can only be achieved when the contour of $C_{\text{model}}$ fits the boundary of the knee in the images. In each iteration, the directions provided by $e'$ for moving the contour points of $C_{\text{model}}$ are always opposite to the gradient directions of the function $e(C_{\text{model}})$. It is clear that, $e'$ uses the gradient descend method to get the optimal solution of $e(C_{\text{model}})$. The iteration process will stop if no better solution can be found by the subsequent iteration or the predefined iteration number is exceeded. In theory, it should stop at the boundary of the knee in the images. The principle of the snakes algorithm will be discussed in Chapter 2 in detail. This example is a typical segmentation method based on the geometric deformable model. The geometric knee contour $C_{\text{model}}$, the fitting criterion function $e(C_{\text{model}})$ and the deformation method $e'$ are all included.

Model-Based Segmentation of Medical Images

The model-based segmentation algorithms have many advantages over its opposite way of segmentation namely the model-free method which based on the assumption of the self-contained image. The self-contained image [91] means that all the information, which is needed for identifying an organ of interest, is contained in the image. However, this assumption usually does not hold in the medical image. Because of the complex anatomic structures of the organ, and the limited medical image quality, the important information such as the organ boundary may be missing. Due to the lack of the necessary information about the organ, a pure image based segmentation method often leads to an unpredictable and unacceptable result. On the contrary, the segmentation method will be more robust and effective when the external information about the organ, which is absent in the image, is known as the prior knowledge. By taking the advantage of the prior knowledge e.g., the organ geometry $C_{\text{model}}$, the approximate contour of the organ can be obtained. Then with the help of a properly defined fitting criterion function $e(C_{\text{model}})$ and an effective contour deformation method $e'$, segmenting the organ from the image is not a big issue.

In the medical image segmentation, most of the model-based segmentation methods refer to the segmentation using geometric deformable models. As noted before, the geometric deformable model is composed of the geometric model $C_{\text{model}}$ and the deformation model $E$ (including $e(C_{\text{model}})$ and $e'$). In the following part, $C_{\text{model}}$, $e(C_{\text{model}})$ and $e'$ will be discussed.
Geometric Model

The purpose of using the geometric model $C_{\text{model}}$ is to provide the shape and spatial information of the object of interest. This kind of model is produced with the help of the method introduced from the discipline of numerical geometry [31, 35] (i.e., the geometric modeling). The numerical geometry methods are widely implemented in computer-aided design (CAD) applications, most of which are used in the industry to aid product designs and product test simulations. The main concern of the numerical geometry is to invent the algorithms which are suitable for the computer to represent and control the object geometry. The applied geometric modeling methods in the model-based image segmentation can be divided into the parametric method (e.g., the spline [6, 32]) and the non-parametric method (e.g., the level set [1, 63, 64]). For the purpose of simplicity and clear description of these two kinds of methods, a closed 2D contour $C$ is taken as an example.

In the parametric approach, the 2D geometric contour $C$ is expressed as $C(s) = (x(s), y(s))$, where $s \in [0, 1]$ is the parameter. $(x(s), y(s))$ can be considered as the position of a point on the 2D contour. For example, the parametric function of the circle centered at the origin with the radius of $r$ is represented as

$$x(s) = r \cos(2\pi s), \quad y(s) = r \sin(2\pi s). \quad (1.2)$$

In the model-based image segmentation, these points $((x(s), y(s)))$ are used to deform the geometric contour. Different from the parametric approach, the non-parametric approach describes the geometric contour without parameters i.e., in an implicit way where the $C$ is defined in the form of $C = \{(x, y)|\varphi(x,y) = a\}$ (a is constant). For example, $\varphi(x,y) = x^2 + y^2 = a$, and when $a = 1$, it represents a circle centered at $(0,0)$ with the radius of 1.

As mentioned above, the rest two components, which are the fitting criterion function $e(C_{\text{model}})$ and the deformation method $e'$, included in the deformation model $E$ are used to deform the geometric model $C_{\text{model}}$ to make it adapt to the organ of interest in the medical images. The adaption quality is defined by $e(C_{\text{model}})$ and the deformation method $e'$ guides the $C_{\text{model}}$ to achieve the optimal value of $e(C_{\text{model}})$, namely, to get the best segmentation result.

Fitting Criterion Function

The fitting criterion function $e(C_{\text{model}})$ is a crucial component of the deformation model $E$. It defines the criterion of how good the match between the geometric model $C_{\text{model}}$ and the organ is. For example, in the active contour method, the fitting criterion function i.e., the energy function decreases monotonically as $C_{\text{model}}$ approaches to the object boundary. Normally, the definition of $e(C_{\text{model}})$ is composed of two parts: the external energy $e_{\text{ext}}(C_{\text{model}})$, and the internal energy $e_{\text{int}}(C_{\text{model}})$, i.e., $e(C_{\text{model}}) = e_{\text{ext}}(C_{\text{model}})+e_{\text{int}}(C_{\text{model}})$. $e_{\text{ext}}(C_{\text{model}})$ is defined based on the image content. The purpose of introducing the external energy is to make the contour of $C_{\text{model}}$ move to the boundary of the organ of interest in the medical images. For example, if the contour of $C_{\text{model}}$ is required to be attracted to the boundary of an organ, then the value of $e_{\text{ext}}(C_{\text{model}})$ at each contour point of $C_{\text{model}}$ should be defined as $-|\nabla I(x,y)|$. It implies that $e_{\text{ext}}(C_{\text{model}})$ is small when all the contour points of $C_{\text{model}}$ are at the positions of high image gradients, and such positions are usually at the organ boundary. $I(x,y)$ is the image intensity at the image position of $(x,y)$, $\nabla$ stands for the image gradient. $e_{\text{int}}(C_{\text{model}})$, which is the other part of $e(C_{\text{model}})$, is defined by the
tension and stiffness of the contour of $C_{model}$. The purpose of $e_{int}(C_{model})$ is to make the contour behave like an elastic string. In theory, the energy function $e(C_{model})$ will achieve its minimum if and only if the contour is attracted to the object boundary and smooth.

### Deformation Method

The deformation method $e'$ is responsible for guiding each contour point of the geometric model $C_{model}$ to the desired position to make $C_{model}$ match to the organ of interest. Since the desired positions correspond to the optimal value of the fitting criterion function $e(C_{model})$, the deformation method can be regarded as an optimization algorithm. Still take the active contour as an example, to get the optimal solution (minimum) of the energy function $e(C_{model})$, the deformation of the contour $C_{model}$ must satisfy the Euler-Lagrange equation by which the important equation $\mathbf{f}_{int} + \mathbf{f}_{ext} = 0$ can be deduced, where $\mathbf{f}_{int}$ and $\mathbf{f}_{ext}$ stand for the internal and external forces (details are in Chapter 2). It means that the optimal solution of $e(C_{model})$ will be obtained when the internal and external forces ($\mathbf{f}_{int}$ and $\mathbf{f}_{ext}$) reach to an equilibrium state. According to this conclusion, in each iteration, each contour point moves in the direction of the resultant force of $\mathbf{f}_{int}$ and $\mathbf{f}_{ext}$ until the resultant force is zero i.e., in a balance state.

### 1.2 Magnetic Resonance Imaging (MRI)

Designing an effective segmentation method is desired, but acquiring good quality images is also important, because the image quality can significantly affect the segmentation result. That is why it has often been said that high quality images lead to the half success of image segmentation. In this thesis, a MRI [16, 44] scanner is used to produce the medical images because of no radiant harm and good resolution of soft tissues comparing with X-ray and CT. By using different MRI scan protocols, the produced MR image could have different contrasts and appearances of the human organs. In order to make suitable MR data for the research, it is necessary to understand the principle about MRI.

The most important role played in MRI is the proton from the hydrogen atom. The main source of the hydrogen atom in the human body is the water molecule which occupies about 60% weight of an adult human (with an average weight of 70 Kg). It is known that the proton has positive charges and spins all the time, thereby it can generate a magnetic dipole with an indefinite orientation (Fig. 1.2 left). When many protons are put into an external high magnetic field (noted as $B_o$), their magnetic directions will change to fit $B_o$. According to the energy state of the protons, there are two fitting types, and both types are illustrated in Fig. 1.2 right. The magnetic orientations of the protons, which are at a high energy state, are opposite to $B_o$ while the orientations of the protons at a low energy state are the same as $B_o$. Normally, the number of the protons at the low energy state is bigger than at the high energy state.

Most of us played tops in our childhood. The top not only spins but also precesses because of the gravity. In the similar manner, the spinning protons in an external magnetic field also precess at a certain frequency which is determined by the Larmor frequency equation [4]

$$\nu = \gamma B_o.$$  (1.3)
1.2 Magnetic Resonance Imaging (MRI)

**Figure 1.2**: The small circles indicate the protons, and the arrows stand for the magnetic orientations. Left: the protons without external magnetic field; Right: the protons in an external magnetic field $B_0$. The orientations of the high energy protons are opposite to $B_0$ while the orientations of the low energy protons are the same as $B_0$.

where $\gamma$ is the gyromagnetic ratio [8] (page 960), and $B_0$ is the strength of the external magnetic field. The knowledge discussed above can be seen as the foundation of MRI.

In MRI, the magnetic orientations of the protons can be changed by emitting a radio frequency (discussed later) periodically, then based on the strength of the feedback signals emitted from the protons, the MR images can be produced. To discuss this process in detail, instead of considering the magnetism of each proton, the net magnetism is focused. The magnetism can be seen as a vector which has direction and magnitude, and the net magnetism is the sum of the magnetism of a mass of protons. Assuming that the direction of the external magnetic field is the same as the $Z$-axis of a 3D coordinate system. Because more protons at the low energy state than at the high energy state, in the equilibrium state of the mass of protons, the net magnetism direction is the same as the $Z$-axis (see Fig. 1.3 left). In Fig. 1.3, the Longitudinal Magnetization (LM) and the Transverse Magnetization (TM) stand for the magnetic strength of the net magnetism in the longitudinal and the transversal directions respectively. In order to change the net magnetism of protons, MRI uses a radio pulse called the Radio Frequency (RF) pulse to alter the orientations of the protons. The RF pulse transmits the energy to the protons to make the protons at the low energy state jump to a higher energy state. But in order to make the protons absorb the energy, the frequency of the RF pulse must be equal to the precess frequency of the protons. Based on the precess frequency, the energy of the RF pulse can be determined by

$$E = h\nu,$$   \hspace{1cm} (1.4)

where $\nu$ is the Larmor frequency, and $h$ is the Plank constant. It is clear to see that the purpose of the RF pulse is to make the protons resonate. That is why it is called the Magnetic Resonance Imaging (MRI).

Assuming that a short RF pulse is emitted to a mass of protons, and because of the RF pulse, many protons are translated to a higher energy state. It means that the equilibrium state of the mass of protons is broken. As shown in Fig. 1.3 right, the LM is decreased while the TM is significantly increased. Then after removing the RF pulse, the equilibrium state will be restored, which means that the LM will grow to the same length as the one shown in Fig. 1.3 left while the TM will be disappeared. The time that the LM returns to the equilibrium state is called the spin-lattice relaxation time ($T_1$), and the time that the TM returns to the equilibrium states is called the spin-spin relaxation time ($T_2$). Because organs have different densities of protons, so each organ has distinct $T_1$ and
T2. This property is the core of making the MR images. In detail, the intensity of the MR image relies on the magnitude of the TM. Therefore, to get a high image contrast, the difference of the TM between organs should be large. To achieve this objective, two important parameters, which are Echo Time (TE) and Repetition Time (TR), are used in MRI. Combinations of different values of TE and TR can produce T1-weighted, T2-weighted and Proton-weighted MR images of different contrasts. TR is the time period between two RF pulses, and TE is the time period from the start of TR to the detection of TM. These two parameters should be set based on the T1 and T2 of the organ of interest to produce a high quality MR image.

1.2.1 T1-weighted MRI

If both TR and TE are short, the difference of the TM signal between two different organs is mainly caused by the difference of their T1. Therefore, in this case the produced image is called the T1-weighted image. Assuming that, a RF pulse can make a 90 degrees flip of the LM signal, which means that after the RF pulse the magnitude of the TM is equal to the LM (before the pulse), and the LM is decreased to zero (Fig. 1.4). Based on this RF pulse, the generation of the T1-weighted image is illustrated in Fig. 1.5. As shown in the image, there are two LM and TM curves (marked in different grey values), each of which corresponds to a different organ. After the first RF pulse, the LM signals of both organs are decreased to zero (at time 0). Then because the black curve has a shorter T1 than the gray curve, the black curve of the LM restores more quickly than the gray curve. By using the RF pulse, the LM difference is then flipped to the TM difference by which the T1-weighted MR image can be produced. Examples of the T1-weighted MR images of two
1.2 Magnetic Resonance Imaging (MRI)

<table>
<thead>
<tr>
<th>Time</th>
<th>Signal</th>
<th>LM</th>
<th>TM</th>
<th>TR</th>
<th>TE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 1.5:** Generation of T1-weighted images. There are two curves marked in different grey values. Each of them represents the change of the LM and TM of an organ.

Parts of the human lower limb are shown in Fig. 1.6.

**Figure 1.6:** The T1-weighted images. Left: the human abdomen; Right: the human legs.

1.2.2 T2-weighted MRI

If both TR and TE are long (Fig. 1.7), the T1 difference between both organs does not contribute to the difference of the TM signal. Because during long TR, the LM signals

**Figure 1.7:** Generation of the T2-weighted MR image.
of both organs have already been restored, and their TM signals after the RF pulse are the same. But using long TE can take advantage of the T2 difference which results in the difference between their TM signals. Therefore, the produced image is called the T2-weighted MR image. An example of the T2-weighted MR image is shown in Fig. 1.8.

![Figure 1.8: The T2-weighted images. Left: the human abdomen; Right: the human legs.](image)

**1.2.3 Proton-weighted MRI**

If TR is long and TE is short, the generated image is called the Proton-weighted image, because the difference of the TM signal between the two organs only depends on the different densities of the protons in the two organs. The illustration of the generation of the Proton-weighted MR image is shown in Fig. 1.9. Since this type of image is not commonly used, there is no example image to present here.

![Figure 1.9: Generation of the Proton-weighted images.](image)

To define the suitable values of TR and TE, a number of experiments have been made. Finally, 1070 milliseconds for the TR and 25 milliseconds for TE (T2-weighted) are determined to produce the MR images for this thesis. Using such configuration of TR and TE, the intensity of the bone is stable, and the intensity is darker than those of other organs. The stable intensity means that the intensity of the bone is robust (i.e., constant) to the influence of the marrow included in the bone.
1.3 Motivation of Tasks

As mentioned before, in this thesis, three methods, which use geometric deformable models and process in the global and local model adaption pattern, are proposed in order to do the segmentation of the rigid body organ, the soft body organ, and the articulated body organ from medical images. These methods are implemented and evaluated in a project, and the specified tasks are the segmentation of the limb bones, the muscles, and the spine. The full name of the project is "Development of a Diagnosis System to Optimize the Operation and Rehabilitation Measurements of Human Musculoskeletal System" (PROREOP for short).

The main task of this project is to model the gait of patients, and the gait is visualized in a 3D environment to give the medics an intuitive view of the motions of the relevant bones during the walk of patients. The purpose is to alleviate the difficulty and enhance the accuracy of the diagnosis in the medical gait pathology analysis. Fig. 1.10 shows the structure of the complete system of this project. Important walk-related parts of the musculoskeletal system of patients (here referring to the limb and the spine) are segmented from the 3D MR data including the axial view MR slices from the pelvis to the feet and the sagittal view MR slices of the spine. The segmented bones and muscles are reconstructed and analyzed to make the 3D geometric and mechanical models. The geometric models are used for the 3D visualization, and the mechanical models are applied for the gait modeling. However, only the mechanical models are insufficient to do the gait modeling, the motions of the mechanical models during the walk of patients are also needed. The motions are measured in a Gait-laboratory, where some tracking points (white balls) are attached to the patient, and the movement of each point during the walk of the patient is recorded by 7 cameras in different positions. The movement of each point indicates the motion of each corresponding part of the patient body. After that, a fusion of the tracking points and the corresponding mechanical models is carried out in order to define the motions.

![Diagram](image_url)

Figure 1.10: Structure of the gait pathology diagnosis system.
1 Introduction

of the mechanical models. By using the fusion, the force simulation and analysis of the mechanical models during the patient walk can be achieved. Finally, with the help of the result of the force simulation and analysis, the gait modeling can be completed and visualized in a 3D scene where the geometric models are used for displaying the appearance of the corresponding parts of the musculoskeletal system of the patient (Fig. 1.11). Further information about the project, please refer to [92]. Similar project can be found in [78] where a simulation model of human musculoskeletal system was created, but unfortunately, no evaluation result is provided.

It is clear that the precondition of the gait modeling is the segmentation of the bones and the muscles from the input medical images.

1.4 Tasks and Challenges

The segmentation tasks required by the project PROREOP are the segmentation of limb bones, muscles and the spine. Details about each task are listed below:

1. segmentation of limb bones and the Functional Kinematic Feature (FKF) which includes the radius and center of the femur head, the center of the femur neck, the trochanter major and the trochanter minor.

2. segmentation of muscles and Tendon Attachment Site (TAS).

3. extraction of the spine curve from the sagittal viewed MR image data.
1.4 Tasks and Challenges

To get the best understanding of these three tasks, basic anatomical knowledge about the above mentioned organs and challenges of each task are presented in the following sections.

1.4.1 Segmentation of Limb Bones and Extraction of FKF

Segmentation of Limb Bone   In the task of the segmentation of limb bones, the proposed method for segmenting the rigid body organ is applied. For the sake of better description of the principle of the method, a typical limb bone called the femur is chosen as the bone of interest. Because the femur is the longest and largest bone of the human body, and it is the main origin of the gait pathology. Furthermore, all the required elements of the FKF are about the femur.

The femur locates at the thigh of the human body. Its head is inserted into the acetabulum of the pelvis, and its bottom is a part of the knee (Fig. 1.12). Most of the challenges in the segmentation of limb bones can be encountered in the femur segmentation. The femur has three distinct parts, which have to be considered separately. The first part is at the top of the femur which include the femur head, the femur neck, the trochanter major, and the trochanter minor (Fig. 1.14). The second part is at the middle of the femur which contains the whole femur shaft. The last part is at the bottom of the femur which is the part of the knee. The main challenges concerning the femur segmentation are:

1. because the amount of images is large, the method should have a high computational efficiency. So that, it could be feasible for the routine clinical use.

2. because of the different complexity of the structure at the top, middle and bottom of the femur, the segmentation strategy should be able to work under all the three femur parts.
3. the femur head, which is the main origin of the gait pathology of the human, is inserted into the acetabulum of pelvis. In the axial MR image, the femur head and the part of the pelvis often have unclear boundaries (Fig. 1.13). So it is preferred that the method should be reliable, noise-proof, and edge-sensitive.

**Figure 1.13:** Example of the axial T2-weighted MR images of the femur head. The boundaries between the femur head and the pelvis are not clear.

**FKF extraction** The FKF mentioned here is a group of the characteristic parameters of the femur containing the radius and the center of the femur head, the center of the femur neck, the trochanter major, and the trochanter minor (Fig. 1.14). Comparing to the other elements in the FKF, the first two elements are relatively easy to be obtained, because the
femur head has a clear recognizable form: a circle. But for the center of the femur neck and the last two elements where many origins and insertions of muscles are located, there is no salient features for recognizing them in the MR images. Therefore, it is better to get the help from the geometric model, on which the locations of the elements in the FKF can be marked, and this is the only solution we can excogitate.

1.4.2 Segmentation of Muscle with Tendon Attachment Sites (TASs)

The muscle segmentation is one of the most challenging topics in medical image segmentation. Very few studies focus on the individual muscle segmentation from MR images. Concerning the segmentation of the muscle Tendon Attachment Site (TAS), only one paper, which is proposed by H. Seim [81], has been found, and it seems to be workable for the CT image only.

Segmentation of Muscle  For the task of the muscle segmentation, a muscle named the sartorius muscle is chosen as the organ of interest. The sartorius muscle is the longest muscle of the human body, and it runs through the whole human thigh. It starts from the anterior superior iliac spine (origin) and it ends at the anteromedial surface of the upper tibia in the pes anserinus (insertion).

Figure 1.15: The sartorius muscle [107] and its two Tendon Attachment Sites (TASs). Two images at right are the T2-weighted MR slices containing TAS1 (origin) and TAS2 (insertion).

The challenges confronted in the muscle segmentation are summarized below:

1. the huge workload. Because of the length of the muscle, the average number of the MR slices of each input MR data used in this thesis is over 100. Therefore, the high computational efficiency of the segmentation method is required.

2. the muscle belongs to the soft body organ. It has a highly dynamic body i.e., no fixed form, and it changes not only between individuals but also over time.
3. the boundaries between the neighboring muscles are almost invisible in the image (Fig. 1.16).

![Figure 1.16: Examples of the sartorius muscle. Two images at the top show the muscle near its two TASs, and the other two images at the bottom illustrate the muscle at its middle parts.](image)

Considering the challenges described above, a proper geometric model of the muscle combining with an efficient and highly adaptive deformation method is needed. Moreover, the geometric model should be capable of representing as many various forms of the muscle as possible.

**TASs** The second subtask of the muscle segmentation is the extraction of the muscle TASs. The tendon is a strip of fibrous connective tissue, which connects the muscle to the bone. The power for the human motions are supplied by the muscle, and the tendon is responsible for the transmission of the power from the muscle to the bone. Therefore, positions of the tendons are important to the force simulation and analysis. Concerning the sartorius muscle, the TASs are referred to the positions of the anterior superior iliac spine and the pes anserinus. The problems of this subtask are:

1. there is no salient image feature of the TAS in the T2-weighted MR images.
2. some parts of the muscle tendons can not be seen in the image. The intuitive idea that tracking the running of the muscle tendon to reach the TAS is infeasible.

### 1.4.3 Spine Curve Extraction

The spine has an articulated body, and the spine curve discussed here is composed of three different sub-curves which are the cervical curve, the thoracic curve, and the lumbar curve (Fig. 1.17). The cervical is composed of 7 vertebrae, the thoracic is composed of 12 vertebrae, and the lumbar is composed of 5 vertebrae.
The spine is composed of the rigid vertebrae and the soft intervertebral disks, therefore it has the properties of both the rigid body organ and the soft body organ. Globally, it has a dynamic body, and locally each vertebra is an rigid organ. So the challenges of the spine curve extraction can be found in the segmentation of the rigid body and the soft body organs. The challenges of the spine curve extraction are:

1. the spine curve is dynamic, and it is neither simple as a straight line nor can be expressed by a second/third order function, therefore a piecewise polynomial method (e.g., the spline) should be applied to describe the curve.

2. the input MR data of the spine contain around 15 sagittal MR slices, among which only 1-3 slices include the whole spine and are suitable for processing (Fig. 1.18).

3. the vertebrae of the spine have different sizes and orientations, and the intensity of each vertebra is also a little different (Fig. 1.18 right).

As mentioned at the beginning of this chapter, three segmentation methods are proposed in this thesis, they use the geometric deformable models and work in the pattern of the global and local model adaption. The general idea of these methods is to adjust and deform the 2D or 3D geometric models to make them adapt to the corresponding organs in the medical images in order to achieve the segmentation. According to the adaption pattern, each segmentation method consists of three stages (pre-processing, global, and local adaption stages), and the geometric model adaption process is proceeded in the global and the local adaption stages. In the global adaption, the pose, size, and position of the organ of interest in the medical images are calculated based on the result from the pre-processing stage. The objective of the global adaption is to make the geometric model match to the organ of interest at the level of size, pose and location. Then the result of the global adaption is refined by the local adaption, which focuses on the local contour differences between the model and the organ of interest. Ideally, after the local adaption stage, the model should totally adapt to the organ of interest.

In order to give a clear and concrete description about the principle and the performance of the three proposed methods, especially, the essence of the geometric deformable
1 Introduction

Figure 1.18: Example of the sagittal T2-weighted MR images of the spine. The left image contains no spine at all. The middle image contains an incomplete and unclear spine. The right image contains the complete spine and is suitable for processing.

model based segmentation method and the global and local model adaption pattern, an application of each method is described. Each application focuses on the one of the three segmentation tasks (described above) required by the project PROREOP. In addition, since the MR imaging is radiant harmless to the human body, the main medical image data used in this research are the MR image data.

1.5 Outline

Overview of the relevant segmentation methods are discussed in Chapter 2, where the classic medical image segmentation methods, which are investigated and applied in this thesis, are introduced. After that, the applications of the three proposed segmentation methods are discussed in three chapters respectively. Specifically, Chapter 3 presents the rigid body organ segmentation method working on the task of the segmentation of the limb bone and the FKF (concerning femur as a representative bone). Chapter 4 presents the segmentation method of the soft body organ applied to complete the task of the muscle segmentation. The task of the spine curve extraction using the segmentation method of the articulated body organ is described in Chapter 5. More applications using the proposed methods are briefly introduced in Chapter 6. Conclusions are made in Chapter 7.
2 Overview of Segmentation Methods

In this chapter, classic medical image segmentation methods are discussed to help readers to better understand the proposed methods in this thesis. The involved segmentation methods are classified into three categories. These are intensity-based methods, methods based on geometric deformable models, and supervised learning methods. As examples of intensity-based methods, thresholding \[41\] (page 443) and region growing \[41\] (page 458), which can be found in many segmentation methods, are briefly described. In the discussion of methods based on geometric deformable models, geometric models and some well-known deformation models e.g., snakes \[54\], Active Shape Model (ASM) \[25, 26\], and atlas-based methods \[90\] are presented. Finally, as a popular example of supervised learning methods, Support Vector Machine (SVM) \[17, 43\], which attracts more and more attentions in medical image segmentation, is discussed.

2.1 Intensity Based Method

The pure image intensity-based method for image segmentation is a quite straightforward method which considers the pixel intensity only. Due to its simplicity and effectiveness, it can be found in many applications. Methods belonging to this class are often used to do the pre-segmentation, whose result needs to be further refined. Two typical intensity-based methods, which are thresholding and region growing, are discussed. Within the methods proposed in this thesis, these two methods are often applied during the pre-processing stage to obtain coarse segmentation results from which information concerning pose, size and position of the organ of interest can be extracted.

2.1.1 Thresholding

The idea of thresholding is simple. When the method uses a single threshold, pixels can be divided into two classes by checking whether their intensities are bigger or smaller than the threshold. Thresholding with a single threshold is used to implement a binary operation, while for the division of more than two classes, a multi-threshold method using two or more thresholds is required. In medical image segmentation, the thresholding method can be often found in applications of bone segmentation from CT images as the intensity of bones is distinct from other organs (Fig. 2.1). The key point of the thresholding method is the threshold value which can be determined either by hand or automatically with the help of the gray value histogram analysis e.g., the Otsu method \[65\].

The advantages of the thresholding method are that it is easy to implement and requires little processing power. Moreover, it has a high compatibility, which means that it can be applied in many kinds of applications. However, since there is no constraint concerning the spatial and structural information of the object of interest, segmentation results are usually
2 Overview of Segmentation Methods

Figure 2.1: Left: original CT image of the bottom of the pelvis and the middle of the femur head; Right: results of the thresholding with gray value 156.

not acceptable, especially when the intensity of the object in an image is not homogenous and accompanied with a strong noise.

2.1.2 Region Growing

The region growing method is better than thresholding because it considers not only the image intensity but also spatial relations. This method starts with a so-called seed (one pixel or region) which is often selected manually or from pre-segmented results. Pixels, which are adjacent to the seed, are checked whether their intensities are similar to the seed. If the similarity meets a predefined criterion, then the pixels are merged into the seed. The region growing method proceeds iteratively until no more suitable pixels are found.

The region growing method is more robust to the image noise than the thresholding method. Since it doesn’t consider the whole image but only neighbor pixels, the region growing method is highly efficient. The region growing method is capable of segmenting both 2D and 3D images and the result is much better as compared with the result using the thresholding method. An example of the region growing method for the segmentation of 3D bones is shown in Fig. 2.2. But due to the lack of the structural information about the organ of interest, region growing will not work well if adjacent organs consist of the same tissue. In this case, there is no intensity difference between the interesting organ and the adjacent organs. E.g., muscles which are adjacent to each other can not be separated by region growing.

Figure 2.2: Left: original MR images of a part of the knee (3D volume display); Middle: result of the thresholding with 40; Right: result of the region growing method.
2.2 Geometric Deformable Model Based Method

Due to the complex shape of human organs and the low quality of medical images (e.g., biases and artifacts), it is hard to produce satisfactory segmentation results using model-free segmentation methods only (e.g., use the thresholding method). If information like the organ structure, which can hardly be obtained from algorithms based on the image content alone, were known at the beginning, then segmentation difficulties would be greatly alleviated. Therefore, the geometric deformable model based image segmentation method appeared. The geometric deformable model is composed of a geometric model \( C_{\text{model}} \) of an object of interest, and a deformation model \( E \) which guides the deformation of the geometric model to fit to the corresponding object in the image. The details of \( C_{\text{model}} \) and \( E \) have been discussed in Section 1.1. The geometric model \( C_{\text{model}} \) usually contains the geometric information including the shape, the size and the location, etc. of the object of interest. The deformation model \( E \) is the core of the geometric deformable model, because the geometric model adapts itself to the object according to a deformation method \( \varepsilon' \) which is established based on the fitting criterion function \( \varepsilon(C_{\text{model}}) \), and both \( \varepsilon' \) and \( \varepsilon(C_{\text{model}}) \) are members of \( E \). So \( E \) gives the geometric model the properties of flexibility and adaptability. These properties make the method using the geometric deformable model suitable for medical image segmentation as the geometric model is capable of fitting the corresponding human organ whose shape is irregular and varies between individuals and over time.

As noted in the first chapter, the methods proposed in this thesis use geometric deformable models. Furthermore, the segmentation process of these methods is designed in the global and local model adaption pattern. According to this pattern, each method is composed of three processing stages which are the image pre-processing stage, the global adaption stage and the local adaption stage. Geometric and spatial information like size, pose and position of interesting organs in images are calculated in the pre-processing stage. Then, in the global stage, geometric models are rotated, scaled and translated based on the information extracted in the first stage. The final segmentation result is obtained after the local adaption stage, where geometric models are deformed based on the deformation model \( E \) to match the corresponding organ in images. The geometric model is a 2D or 3D representation of the organ of interest. In the proposed methods of this thesis, all geometric models are integrations of 2D or 3D points. The geometric models has been discussed in Section 1.1, so in this chapter, further discussion is not needed, and the main focus is the currently used deformation models. In this thesis, three kinds of deformation models are investigated and applied. These are the active contour model i.e., snakes [54], the Active Shape Model (ASM) [25, 26], and the atlas-based method [90]. In the following sections, the original algorithms of the deformation models, that mentioned above, are described.

2.2.1 Snakes: Active Contour Model

As mentioned in Chapter 1, the snakes algorithm is a kind of deformation model for deforming the geometric contour defined by explicit functions. The general form of the explicit function is \( y = f(x) \) where the output \( y \) is determined explicitly by the input \( x \), while in the implicit function, \( y \) is obtained by solving an equation like \( f(x, y) = 0 \). In the snakes algorithm, contours are defined by a parametric function denoted as \( C(s) \). In this thesis, \( C(s) \) represents the 2D contour only. So if the geometric model \( C_{\text{model}} \) is a 3D
object, then $C(s)$ represents the 2D contour of each cross section of $C_{model}$. The snakes algorithm has been widely and successfully used in different kinds of image segmentation applications. However, as mentioned in [54], the original snakes algorithm has a major drawback, because the initial model contour has to be close to the boundary of the desired object. In other words, model contours using the snakes algorithm have a short capture range. If a model contour is not well placed in the initialization, some parts of the contour may be far away from the boundary of the organ of interest, and during the process of the original snakes algorithm, the contour may not be attracted by the organ boundary. Therefore, to overcome this drawback of snakes algorithm, many improvements were added to extend the capture range, e.g., using multi-scale images [51] or introducing an expanding force [19, 20]. In the following of this subsection, the original snakes algorithm and some well-known derivatives of the snakes will be presented.

**Energy Function**

The snakes algorithm works in the framework of energy function minimization. As noted before, the energy function is an instance of the fitting criterion function $e(C_{model})$. In the snakes algorithm, the energy function is defined based on both the object of interest and the parametric contour $C(s)$, and it is denoted as $e(C(s))$. $e(C(s))$ includes a set of prior knowledge about the interesting object and constraints of the contour. The main task of the snakes algorithm is to find an optimal solution of $e(C(s))$ (usually the minimal value), and the minimum can only be achieved if the contour reaches to a desired position e.g., the interesting object boundary. The basic form of $e(C(s))$ in the snakes algorithm is defined as

$$e(C(s)) = \int_0^1 e_{int}(C(s)) + e_{img}(C(s)) + e_{con}(C(s)) ds$$  \hspace{1cm} (2.1)$$

where $C(s) = (x(s), y(s))$ with $s \in [0,1]$ is the parametric representation of the 2D geometric contour. The energy function can be divided into two parts, which are the internal and the external energy parts. The internal energy (denoted as $e_{int}$) comes from $C(s)$. The external energy includes $e_{img}$ which comes from the image content and the user defined energy $e_{con}$ (e.g., attractive or repelled points).

$e_{int}(C(s))$ is responsible for the guaranty of smoothness of the 2D geometric contour and it is defined as

$$e_{int}(C(s)) = \int_0^1 \left[w_s(s)|C_s(s)|^2 + w_{ss}(s)|C_{ss}(s)|^2\right] ds$$  \hspace{1cm} (2.2)$$

where the weighting factors $w_s(s)$ and $w_{ss}(s)$ control the contributions of each term. Since $w_s(s)$ and $w_{ss}(s)$ are usually constant functions, both are often denoted as $w_s$ and $w_{ss}$, instead. An implementation with non-constant weighting factors can be found in [109]. The first order derivative $C_s(s)$ allows the contour to deform as an elastic ring. If $w_s$ is negative the contour is encouraged to expand while the contour tends to shrink when $w_s$ is positive. The second order derivative $C_{ss}(s)$ measures the contour curvature, by which unsmooth contour parts can be observed and removed.

The image energy $e_{img}(C(s))$, which is a part of the external energy, consists of three
2.2 Geometric Deformable Model Based Method

\[ e_{\text{img}}(C(s)) = \int_{0}^{1} [w_{\text{line}}e_{\text{line}}(C(s)) + w_{\text{edge}}e_{\text{edge}}(C(s)) + w_{\text{term}}e_{\text{term}}(C(s))] ds \] (2.3)

where weighting factors \( w_{\text{line}}, w_{\text{edge}} \) and \( w_{\text{term}} \) define the contribution of their corresponding elements. \( e_{\text{line}}(C(s)) \) can be simply defined as the pixel intensity i.e., \( e_{\text{line}}(C(s)) = I(x, y) \). Thereby the contour tends to move to either higher or lower intensity regions according to the sign of \( w_{\text{line}} \).

\( e_{\text{edge}}(C(s)) \) is defined as \(-|\nabla I(x, y)|^2\) which is small at positions with a high image gradient. The last element, i.e., \( e_{\text{term}}(C(s)) \), which is defined as

\[ e_{\text{term}}(C(s)) = \frac{\partial \theta(x, y)}{\partial n_{\perp}}, \] (2.4)

is used to attract the contour to the terminations of line segments and corners. \( \theta(x, y) \) is the direction of the gradient at point \((x, y)\), and \( n_{\perp} = (-\sin\theta, \cos\theta) \) is a unit vector perpendicular to the gradient direction.

\( e_{\text{con}}(C(s)) \) is the other part of the external energy, and it is interactively defined by the user, e.g., user can set springs (attraction and repulsion forces) between contour points and other image pixels to force the 2D contour to deform in an expected region.

**Energy Function Minimization**

As mentioned above, the energy function can be divided into two parts which are the internal and external energy parts, i.e., \( e(C(s)) = e_{\text{int}}(C(s)) + e_{\text{ext}}(C(s)) \). The internal energy part includes \( e_{\text{int}}(C(s)) \) while the external energy part \( e_{\text{ext}} \) contains \( e_{\text{img}}(C(s)) \) and \( e_{\text{con}}(C(s)) \). The objective of the snakes is to deform the contour \( C(s) \) to make the energy function \( e(C(s)) \) minimal and this objective is equivalent to finding the deformation of \( C(s) \) which satisfies the Euler-Lagrange equation [36]

\[ w_s(C_s(s))_s - w_{ss}(C_{ss}(s))_{ss} - \nabla e_{\text{ext}}(C(s)) = 0. \] (2.5)

Here, \( w_s \) and \( w_{ss} \) are considered to be constant values. The first two items at the left side of this equation represent the stretching and bending forces

\[ \overrightarrow{f}_{\text{int}} = w_s(C_s(s))_s - w_{ss}(C_{ss}(s))_{ss}. \] (2.6)

The third item at the left side of this equation represents the external force

\[ \overrightarrow{f}_{\text{ext}} = -\nabla e_{\text{ext}}(C(s)). \] (2.7)

It is clear that to satisfy the Euler-Lagrange equation, \( C(s) \) should make both vectors \( \overrightarrow{f}_{\text{int}} \) and \( \overrightarrow{f}_{\text{ext}} \) have equal magnitude, but in contrary directions. In other words, the resulting force of \( \overrightarrow{f}_{\text{int}} \) and \( \overrightarrow{f}_{\text{ext}} \) should be zero, i.e., \( \overrightarrow{f}_{\text{int}} + \overrightarrow{f}_{\text{ext}} = 0 \).
Snakes in Discrete Domain

By now, it is clear that the 2D geometric contour is deformed to reach to the minimal value of the energy function and according to the Euler-Lagrange equation, at each iteration, points of the 2D contour move in the direction of the resulting force. Next, realization of the snakes algorithm for image segmentation, which works in a discrete domain, is discussed. As shown in Fig. 2.3, at each iteration of the snakes algorithm, each contour point has 8 neighbor pixels in the image, and for each position of the 8 neighbor pixels, the energy can be calculated using the energy equation. After the calculation of energies at all the 8 pixel positions, the contour point will move to its new position where the energy is the lowest of all (including the energy at the current position). The iteration stops when there is no position change for each contour point, or when the predefined maximum iteration number is exceed.

Figure 2.3: Illustration of the neighborhood search (3 × 3) of each contour point (grey point). The energy calculation at the middle contour point needs the help of the left and right contour points to calculate the internal energy.

Active Contour in Multi-Resolution

Based on the original snakes algorithm many derivatives were proposed [7, 18, 20, 51, 66]. Some of them focus on the improvement of the energy function in order to get a precise energy value. However, the major disadvantage of snakes is that the contour will not be able to approach the object boundary if the initialized position is far away from the object. Cohen [18] introduced an additional internal pressure to make the contour act as a balloon. The initial contour should be placed inside the object of interest. Then no matter how far the distance between the contour and the object boundary, the contour has the capacity to expand to the object boundary. Leroy [51] proposed a solution for improving the capture range. In his approach, the initial contour was specified on a coarse-scale image where object boundary was blurred to increase the capture range, so the initial contour did not need to be placed near the object boundary. However, the result was not accurate using the coarse-scale image, because the object boundary was highly blurred. Therefore, the result was then used as the initial contour for the next finer-scale image. This method proceeds iteratively until the original image is reached.
2.2 Geometric Deformable Model Based Method

2.2.2 Active Shape Model (ASM)

The principle of ASM [25, 26] is to use the statistic method to extract the distribution of contour points from training instances of the object of interest. The distribution of contour points is used to build the ASM, which acts as the deformation model \( E \), to deform the geometric model \( C_{\text{model}} \) of the object of interest. Unlike the snakes algorithm, the ASM, which profits from the statistic study of contours of training instances, deforms \( C_{\text{model}} \) in a rational way. That is why in [25], Cootes called the ASM the smart snakes. The ASM has many advantages over the snakes algorithm in the segmentation of dynamic body objects, where the deformation of the geometric contour could be arbitrary and irrational when using the snakes algorithm.

**ASM Calculation**

Assume that the contour of an object of interest consists of \( n \) labeled 2D contour points which represent the object boundary. To calculate the ASM of the object, a training set which contains \( m \) contour instances of the object is needed (see Fig. 2.4 as an example). Each contour instance in the training set is represented as an array \( x_i \) (\( i \)-th instance) which has 2\( n \) elements, i.e., \( x_i = (x_0, y_0, x_1, y_1, \ldots, x_{n-1}, y_{n-1})^T \), where \( (x_0, y_0), (x_1, y_1), \ldots, (x_{n-1}, y_{n-1}) \) are 2D coordinates of the \( n \) 2D points of the contour instance. First of all, one instance in the training set is chosen as a reference. Then the rest instances are translated to have their centroids at the same position as the reference’s. This step is called the instance alignment. Let \( \bar{x} \) be the mean contour of the aligned instances, i.e.,

\[
\bar{x} = \frac{1}{m} \sum_{i=0}^{m-1} x_i, \quad (2.8)
\]

Then \( \bar{x} \) and instances of the training set are used to calculate the distribution of each labeled point by Principal Components Analysis (PCA) [87]. To construct the covariance matrix used by PCA, difference between each instance and \( \bar{x} \) is calculated as \( dx_i = x_i - \bar{x} \) first. Then each element of the \( 2n \times 2n \) covariance matrix \( C_O \) is obtained by

\[
c_{Okl} = \frac{\sum_{i=0}^{m-1} dx_{ik} * dx_{il}}{m - 1}, \quad (2.9)
\]

![Figure 2.4: 10 (\( m = 10 \)) triangle instances are used as an examples of training set for creating an ASM of triangle. Here, each instance is described by three vertex points i.e., \( n = 3 \).](image-url)
where \( c_{kl} \) is the element at the \( k \)-th row and \( l \)-th column of the matrix \( C_O \), and \( k = [0, 2n-1], l = [0, 2n-1] \). \( dx_{ik} \) and \( dx_{il} \) indicate the \( k \)-th and \( l \)-th elements of \( dx_i \). Supposing that \( P = (\overrightarrow{P}_0, \overrightarrow{P}_2, \ldots, \overrightarrow{P}_{2n-1}) \), where \( \overrightarrow{P}_i, i = 0, 1, \ldots, 2n - 1 \), were eigenvectors of \( C_O \), then the ASM of the object could be presented as

\[
\mathbf{x} = \mathbf{x} + \mathbf{Pb},
\]

where \( \mathbf{b} \) is a \( 2n \) elements array \((b_0, \ldots, b_{2n-1})^T\), and each element \( b_i \), \( i = 0, 1, \ldots, 2n - 1 \) defines the magnitude (i.e., weighting factor) of the deformation of the model contour in the direction of the \( i \)-th eigenvector \( \overrightarrow{P}_i \) (see Fig. 2.5).

![Image](image-url)

**Figure 2.5:** Top: deformation based on the first eigenvector which has the corresponding largest eigenvalue; Middle: deformation based on the second eigenvector; Bottom: deformation based on the third eigenvector.

### Adaption of ASM

In the original article of ASM, the adaption process in the segmentation by using ASM is operated in an iterative way (see [25] for detail). The adaption of ASM involves the affine transformation, such as rotation, translation, scaling (without shear), and a non-rigid transformation which is realized by changing the weighting array \( \mathbf{b} = (b_0, \ldots, b_{2n-1})^T \) in equation 2.10. Fig. 2.6 illustrates a situation where a gray object (\( x_o \) denotes its boundary) is included within an initial contour (green line) of the ASM (the mean contour \( \bar{x} \) with an initial pose and size estimation). \( \overrightarrow{d}_i, i = [0, n - 1] \) is a vector pointing from the \( i \)-th landmark of the model contour to the object boundary \( x_o \) in the direction of the model contour normal. \( dx = (X(\overrightarrow{d}_0), Y(\overrightarrow{d}_0), \ldots, X(\overrightarrow{d}_{n-1}), Y(\overrightarrow{d}_{n-1}))^T \) is the difference matrix, where \( X(\overrightarrow{d}_i) \) and \( Y(\overrightarrow{d}_i) \) are the values of \( \overrightarrow{d}_i \) in \( x \) and \( y \) directions respectively, therefore

\[
\mathbf{x}_o \approx (s\mathbf{R}(\theta)\mathbf{x} + \mathbf{x}_c) + dx,
\]

where \( s\mathbf{R}(\theta)\mathbf{x} + \mathbf{x}_c \) represents the initially estimated model contour that is adjusted by the estimated rotation angle \( \theta \), scaling factor \( s \) and the \( 2n \times 1 \) translation matrix \( \mathbf{x}_c = (x_c, y_c, \ldots, x_c, y_c)^T \). \( \mathbf{R}(\theta) \) represents the rotation matrix. These parameters can be
2.2 Geometric Deformable Model Based Method

Figure 2.6: The green line represents the current ASM contour. The gray region stands for the object of interest that is to be fitted. The ASM contour is composed of many points, i.e., landmarks. The red dash lines indicate the differences between the object boundary $x_o$ and the corresponding landmarks.

estimated using exhaustive search within a certain range of parameters. Equation 2.11 implies that the main task of the ASM adaption is to reduce or eliminate $d\mathbf{x}$. As mentioned above, the adaption process of ASM contains affine and non-rigid transformations. Equation 2.12 shows how both kinds of transformations act on the initially estimated model contour

$$x_o \approx s(1 + ds)R(\theta + d\theta)(\mathbf{x} + d'\mathbf{x}) + \mathbf{x}_c + dx_c$$

(2.12)

where $\mathbf{d}x$ is decomposed into an affine transformation part including $ds$, $d\theta$, $dx_c$, and a non-rigid part which is $d'\mathbf{x}$. $dx_c = (dx_c, dy_c, \ldots dx_c, dy_c)^T$ is a $2n \times 1$ matrix where $dx_c$ and $dy_c$ are the differences between the center of the model contour and the object center in $x$ and $y$ directions, respectively. $d'\mathbf{x}$ indicates the residual difference that cannot be eliminated by the affine transformation, and it can be calculated by combining equation 2.11 and 2.12 resulting in the following equation

$$d'\mathbf{x} = \frac{1}{s(1 + ds)}R(-(\theta + d\theta))(sR(\theta)\mathbf{x} + dx - dx_c) - \mathbf{x},$$

(2.13)

where the first part on the right side of this equation represents the approximated $x_o$ which is transformed and normalized to the origin of the coordinate system of $\mathbf{x}$. According to equation 2.10, we get $d'\mathbf{x} = \mathbf{P}(d\mathbf{b})$, therefore,

$$d\mathbf{b} = \mathbf{P}^{-1}d\mathbf{x} = \mathbf{P}^T d'\mathbf{x}.$$  

(2.14)

Cootes [25] proposed an iterative process for the ASM adaption. For example, at the beginning of the $k$-th iteration, $\mathbf{x}_{(k-1)} = s_{(k-1)}R(\theta_{(k-1)})\mathbf{x}_{(k-1)} + \mathbf{x}_{c(k-1)}$ and $d\mathbf{x} = \mathbf{x}_{(k-1)} - \mathbf{x}_o$. Then at each iteration, elements of $dx_c$ are calculated by

$$dx_c = \frac{1}{n} \sum_{i=0}^{n-1} X(\overrightarrow{d}_i) \quad \text{and} \quad dy_c = \frac{1}{n} \sum_{i=0}^{n-1} Y(\overrightarrow{d}_i).$$

(2.15)

Then $\mathbf{x}_{(k-1)}$ is updated to $\mathbf{x}_{(k-1)} + dx_c$, and $d\mathbf{x}$ is updated to $d\mathbf{x} - dx_c$. Fig. 2.7 illustrates how to obtain the $d\theta$ and $ds$ in the iteration. As shown in Fig. 2.7 left, the length of the vector $\overrightarrow{d}_{ir}$ equals to the inner product of the normalized vector
2 Overview of Segmentation Methods

![Illustration of calculating ds (left) and dθ (right).](image)

Figure 2.7: Illustration of calculating $ds$ (left) and $d\theta$ (right).

\[
(-x_i/\sqrt{x_i^2 + y_i^2}, -y_i/\sqrt{x_i^2 + y_i^2}) \text{ with the vector } \vec{d}_i, \text{ and the direction of } \vec{d}_{ir} \text{ is the same as } (-x_i/\sqrt{x_i^2 + y_i^2}, -y_i/\sqrt{x_i^2 + y_i^2}). \text{ Therefore, } \vec{d}_{ir} \text{ is calculated by}
\]

\[
\vec{d}_{ir} = \frac{-x_i X(\vec{d}_i) - y_i Y(\vec{d}_i)}{\sqrt{x_i^2 + y_i^2}}(-x_i/\sqrt{x_i^2 + y_i^2}, -y_i/\sqrt{x_i^2 + y_i^2})^T.
\] (2.16)

The scale factor $ds_i$ for scaling the $i$-th ASM landmark is

\[
ds_i = \frac{\|\vec{d}_{ir}\|}{\sqrt{x_i^2 + y_i^2}} = \frac{-x_i X(\vec{d}_i) - y_i Y(\vec{d}_i)}{x_i^2 + y_i^2}.
\] (2.17)

The rotation angle (Fig. 2.7 right) for each ASM landmark is calculated by $\vec{l} / \sqrt{x_i^2 + y_i^2}$, where $l$ (red dashed line in Fig. 2.7 left) is the distance from the object contour point $(x_i, y_i)$ to the line segment $(0,0)$ and $(x_i, y_i)$. It is easy to deduce that

\[
l = \|\vec{d}_i - \vec{d}_{ir}\|
\] (2.18)

and

\[
d\theta_i = \frac{l}{\sqrt{x_i^2 + y_i^2}}.
\] (2.19)

d$\theta_i$ and $ds_i$ are the rotation angle and scale factor for the $i$-th landmark of the ASM contour, respectively. Therefore, in each iteration the final rotation angle and scale factor are the average value of $d\theta_i$ and $ds_i$, $i = (0, 1, \ldots, n - 1)$.

\[
d\theta \approx \frac{1}{n} \sum_{i=0}^{n-1} d\theta_i
\] (2.20)

\[
ds \approx \frac{1}{n} \sum_{i=0}^{n-1} ds_i
\] (2.21)

Then with the calculated $d\theta$ and $ds$, $\vec{d}b$ can be obtained by using equations 2.13 and 2.14. Then we update $x_{c(k)} = x_{c(k-1)} + w_c \vec{d}x_c$, $\theta_{(k)} = \theta_{(k-1)} + w_d d\theta$, $s_{(k)} = s_{(k-1)} + w_s ds$,
\( x_{(k)} = x_{(k-1)} + P \mathbf{w}_b \mathbf{d} \) and \( x_{(k)} = s_{(k)} R(\theta_{(k)}) x_{(k)} + x_{c_{(k)}} \). \( w_c, w_\theta, w_s \) and \( w_b \) are weighting factors in the range of \([0,1]\). Then \( x_{(k)} \) is used as the input for the next \( k+1 \)-th iteration. The iterative process stops when the iteration number exceeds a pre-defined number or \( dx \) is smaller than a given threshold.

### 2.2.3 Atlas Based Method

Atlas based method is a hot topic in medical image segmentation. There are many studies focusing on it [21, 48, 59, 60]. An atlas is a labeled image and can be regarded as the geometric model \( C_{\text{model}} \). In the atlas not only the shapes but also the spatial information of different objects of interest are described. Usually, an atlas contains more than one object, different objects are marked in different intensities (as labels). The atlas is commonly generated by manual segmentation from images or synthetic shapes of objects (Fig. 2.8). Image segmentation by using an atlas is strongly connected to image registration [12] whereby the atlas coordinates are transformed to the coordinate system of the image that contains the object of interest. The transformation involves rigid and non-rigid deformations. The deformation method works based on certain similarity measurements (act as the fitting criterion function \( e(C_{\text{model}}) \)), e.g., Mutual Information (MI) and Correlation Ratio (CR) which will be discussed later. After the registration, the class of objects (in case of more than one object in the atlas), which each image pixel belongs to, is determined by looking for the label at the corresponding position of the atlas.

**Figure 2.8:** Example of the atlas. Left: the original MR image showing a part of the spine; Middle: the original MR image showing one vertebra of the spine; Right: the atlas of the vertebra. The vertebra and two intervertebral disks are marked with different intensities (i.e., labels).

### Similarity Measurements

In the atlas-based method, similarity measurements, which are used to evaluate the image registration quality of the atlas, are equivalent to the fitting criterion functions \( e(C_{\text{model}}) \). Good registration corresponds to the optimization (minimum or maximum) of the \( e(C_{\text{model}}) \). A commonly used measurement method is the so called MI [56] which is defined as

\[
MI(A,B) = H(A) + H(B) - H(A,B),
\]  

(2.22)
where $A$ and $B$ represent two images. The entropy of image $H(A)$ is calculated by

$$H(A) = -\sum_ap_A(a)\log[p_A(a)],$$

(2.23)

where $a$ is the gray value and $p_A(a)$ is the probability of $a$ which is equal to the number of pixel whose gray value is $a$ divided by the number of pixel of $A$. The joint entropy of $A$ and $B$: $H(A, B)$ is defined as

$$H(A, B) = -\sum_{a,b}p_{AB}(a,b)\log[p_{AB}(a,b)],$$

(2.24)

where $p_{AB}(a,b)$ is the probability of index $(a, b)$ in the joint histogram (Fig. 2.9) of the overlay part of $A$ and $B$. $p_{AB}(a,b)$ is calculated by summing up the number of intensity value pairs whose intensity in $A$ is $a$ and where at the corresponding position in $B$ the value is $b$. Then the sum is divided by the number of all pixel pairs that are located in the overlap part of $A$ and $B$. In Fig. 2.9, if $A$ and $B$ are correctly registered (matched), the distribution of their joint histogram will be small (a diagonal line) which means $H(A,B)$ is small, and in other words the $MI(A,B)$ is large.

![Figure 2.9: Example of the joint histogram. Assuming that $A$, $B$ are two identical images, and at the beginning, they are completely overlapped as shown in the leftmost image. The right 4 images are 256*256 joint histogram image of the overlapped part of $A$ and $B$. From the second left to the rightmost image: joint histogram when $A$ and $B$ are completely overlapped; $B$ shifted in 1 pixel; $B$ shifted in 5 pixel, and $B$ shifted in 10 pixel (shift in the right direction).](image)

$MI$ is successfully applied in image registration, however it has a big drawback because it is influenced by the size of overlapping of the two images. An illustration of the overlapping influence is shown in Fig. 2.10. Image $A$ is bigger than $B$ and the radius of the white circles in both images are the same. In the case shown at the top right in Fig. 2.10, $A$ and $B$ are correctly matched (expected result), and the $MI$ of the overlapping part actually equals to $H(B)$, which is 0.806. However, 0.806 is not the largest $MI$ in the view of the computer. Rather, the largest $MI$ appears in the case shown at the bottom right in Fig. 2.10, where the amount of black and white pixels of the overlapping part is almost the same. In theory, the largest $MI$ of binary images is 1, which is when the amount of black pixels equals the white pixels.

The overlapping problem has been solved by Studholme [89] in 1999. In his paper a method called the Normalized Mutual Information (NMI) is proposed, which is defined as

$$NMI(A, B) = \frac{H(A) + H(B)}{H(A, B)},$$

(2.25)

However, $MI$ or NMI works on the intensity of pixels, but in medical images, intensities of the region of an organ may be nonhomogeneous. Therefore, Roche [74] proposed a new
2.2 Geometric Deformable Model Based Method

Figure 2.10: Example of the overlapping influence of MI. The size of the white circles in images A and B is the same, but the length of A is longer than B. On the right, two cases of registration results and the corresponding MI values are shown.

similarity measurement, namely the CR method, which does not focus on the intensity of each pixel but the mean intensity of each region. Suppose that there are \( i \) different regions in an atlas image \( A \), and \( r_i \) indicates the \( i \)-th region. \( I_{r_{ij}} \) is the intensity value of the \( j \)-th pixel in image \( B \) and belongs to region \( r_i \). \( n_{r_i} \) is the size of \( r_i \) in \( B \). Then CR is calculated by

\[
CR(A, B) = \frac{Var(E[B|A])}{Var(B)} = \frac{\sum_i n_{r_i} \cdot (\bar{T}_{r_i} - \bar{T})^2}{\sum_{r_{ij}} (I_{r_{ij}} - \bar{T})^2}
\]

(2.26)

where

\[
T_{r_i} = \frac{\sum_j I_{r_{ij}}}{n_{r_i}}, \bar{T} = \frac{\sum_i n_{r_i} T_{r_i}}{\sum_i n_{r_i}}.
\]

(2.27)

The CR takes values between 0 and 1. A high CR value indicates a good registration result.

Deformation Method

The deformation method \( e' \) used in the atlas-based image registration can be divided into the rigid and the non-rigid classes. The rigid deformation method includes translation, rotation, and sometimes scaling is also regarded as a rigid deformation. Details about the performance using the rigid deformation method can be found in the paper written by Studholme [88] where a comparison of different fitting criterion functions under a pure rigid deformation (only translation and rotation) was conducted for an application scenario that adapts MR images to PET images. The rigid deformation method proposed in [88] finds the optimal solution for each kind of the fitting criterion function applied to hierarchically resampled images (multi-resolution).

Since the high adaption capability when compared to the rigid method, the non-rigid deformation methods are preferred in most atlas-based image registration applications. The non-rigid method is normally used to solve the problem of inter-individual shape differences, which cannot be achieved by the pure rigid method. However, the great disadvantage of the non-rigid method is its low computational efficiency. For example, in [12], the registration of a bee brain image (749×496 pixels) with an bee brain atlas took around
2 Overview of Segmentation Methods

10 minutes using the non-rigid deformation method proposed in [76] (using B-Splines) on a PC equipped with an Intel Pentium 4, 3.0GHz, hyperthreading CPU.

2.3 Supervised Learning Methods

In this thesis, machine learning methods [3] are not applied but some of them are tested in the experiments during the research in order to compare them with the methods proposed in the thesis. Machine learning methods have shown a strong capacity in pattern recognition. Currently, machine learning methods are intensively studied in the medical image segmentation. Therefore, it is necessary to take some time to discuss them here.

Machine learning methods can be generally divided into supervised and unsupervised learning methods. The main difference between these two is the training data which are labeled for the former but unlabeled for the latter. Unsupervised learning methods such as K-means [55] and fuzzy C-means [9], try to classify the unlabeled input data based on the differences of data attributes (e.g., the pixel intensity and position). Since the weak classification capacity, they are often used for the image pre-segmentation in some medical image segmentation methods. Supervised learning methods use labeled data to deduce a function which describes the relations between the attributes of the data and their labels. The deduced function is used to determine the label for new previously unseen data (unlabeled).

Up to date, the supervised learning method called SVM [17] is the most popular method and has been applied in many different research fields. In this thesis, performance of medical image segmentation using SVM was tested in order to compare the results with the model-based segmentation method proposed in this thesis.

2.3.1 Support Vector Machine (SVM)

The SVM was developed for classification and regression. The objective of the SVM is to find an optimal hyperplane that separates the training data (presented as vectors) according to their labels. The hyperplane should maximize the margin which is determined by some vectors (i.e., support vectors) from the training data. Imaging that a set of training data \((x_i, l_i), i = 1, \ldots, n, x_i \in \mathbb{R}^d,\) and \(n\) is the amount of the training data. \(l_i \in \{−1, 1\}\) (two classes) is the label of \(x_i\). According to the distribution of the training vectors, SVM builds a hyperplane which separates the training vectors into two classes \((l_i = 1\) or \(−1\)) (Fig. 2.11).

Obviously, vector \(x\) which lays on the hyperplane satisfies \(x \cdot w + b = 0\) and the training vectors \(x_i\) and its \(l_i\) satisfy

\[
l_i(x_i \cdot w + b - 1) \geq 0, \quad (2.28)
\]

which is equivalent to

\[
x_i \cdot w + b \geq 1, l_i = 1 \text{ and } x_i \cdot w + b \leq -1, l_i = -1.
\]

The margin equals to \(2/\|w\|\) (proved on p129 of [17]). To maximize the margin is equivalent to minimize \(\|w\|\). Normally, we solve

\[
MIN\left(\frac{1}{2}\|w\|^2\right) \quad (2.29)
\]
subject to eq. 2.28, instead of $\|w\|$. Furthermore, considering the overfitting and the non-separable problems, the final objective of SVM is to solve

$$MIN\left(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i\right) \tag{2.30}$$

subject to

$$l_i(x_i \cdot w + b) \geq 1 - \xi_i \tag{2.31}$$

where $\xi_i$ are the slack variables (Fig. 2.12), and $C$ is the cost factor which is the weight factor of the sum of slack variables. A higher $C$ describes the tendency that all vectors
satisfy eq. 2.28, where an extreme example is an infinite $C$ which can cause the overfitting problem.

Another important conception of SVM is the kernel function which is used to convert the linear non-separable vectors to a higher dimensional space to make them separable. Since SVM is not the emphasis of this thesis, the kernel functions are not described here. For more detailed information about the SVM, please refer to [17] by Christopher.

From the next chapter on, the implementations of the proposed methods for the segmentation of the rigid body organ (limb bones), the soft body organ (muscles) and the articulated body organ (the spine) will be described.
In this chapter, a segmentation method for the rigid body organ will be presented. To discuss the principle and the performance of this method, an application of this method is described. The task of the application is to segment lower limb bones such as the femur, tibia, and fibula. Considering that the Functional Kinematic Feature (FKF), which is required in the application, concerns the femur, so the femur is chosen as the bone of interest in this chapter. More segmentation results of other limb bones will be briefly presented in Chapter 6. The FKF mentioned here is a group of the characteristic parameters of the femur including the femur ball center, the femur ball radius, the femur neck center, the trochanter major and the trochanter minor. Publications concerning the previous work can be found in [94, 95].

3.1 Introduction

The femur is one of the limb bones and plays an important role in the human motion. It connects the human upper and lower bodies. The femur is one of the most easily injured bones. From an investigation we found that in most cases the origin of the gait pathology is the injury of the hip joint (e.g., arthritis of the hip joint), and it is treated by replacing the femur head with a prosthesis (Fig. 3.1). Obviously, the size and shape of the hip joint

![Figure 3.1: Example of a hip joint prosthesis. The prosthesis is composed of four parts shown in the left and the middle images. Left image (from top to bottom): the socket which is fixed in the acetabulum, the inlay which is inserted into the socket, the ball which is embedded into the inlay. Middle image: the prosthesis shaft. Right image: the X-ray image of a patient whose femur head has been replaced with the prosthesis.](image-url)
vary from individual to individual. The production of the hip joint prosthesis must be customized for each patient. Therefore, the FKF is a part of the task in this application.

### 3.1.1 State of the Art

Many methods have been proposed for bone segmentation. However, most of them are designed for CT data not MR data. The intensity of bones in CT images is high and quite distinct from other tissues. The segmentation result would even be acceptable by using the simple thresholding method. But in MR images, the contrast between the bone and other tissues is low. Furthermore, the intensity of the bone is usually nonhomogeneous as the result of the partial volume effect. Therefore, simple low level image processing methods such as thresholding are insufficient. Nowadays, methods such as the active contour model and level set are those which are mostly applied to the bone segmentation from MR data. For example, Krátký [50] presented a 3D bone segmentation method for MR data by using the fast level set [86] and plenty of encouraging results were obtained. However, the substantial parameters for the level set function had to be adjusted according to the scan protocol of the input MR data. Pettersson [69] proposed an approach by means of the morphons algorithm [49] to register the prototype to the bone in CT images. It provided a good result using CT data but for MR data it was difficult to get such a nice result. Concerning the extraction of the femur FKF, to the best of our knowledge, there is very few literatures and studies dealing with the FKF. In addition, among those, most of them bear no relation to medical image segmentation.

### 3.1.2 Overview of Rigid Body Organ Segmentation

As noted before, a segmentation method for the rigid body organ is implemented, and this method works in the global and local model adaption pattern. Three stages are contained in this method, which are the pre-processing, the global adaption and the local adaption stages. Fig. 3.2 shows a flow chart of this method. In the pre-processing stage, the input

---

**Figure 3.2:** Flow chart of the segmentation of the rigid body organ.

MR image data are denoised using the Mean-Shift filter [22]. Then the thresholding and region growing methods are applied to do the pre-segmentation of the organ of interest. The pre-segmented result is sent to the global adaption stage where a 3D geometric model
of the organ of interest is adjusted according to the spatial information (pose, size, position) extracted from the pre-segmented result. The adjustment includes rotation, scaling, and translation. Since the organ has a rigid body, its pose can be represented by its body axis. The objective of rotation is to make the axis of the model the same as that of the organ of interest. For the scaling, the length of the organ of interest has to be calculated and the model is scaled to the same length. For the translation, a position of a reference point of the organ of interest is needed and a translation vector which is from the corresponding point of the model to the reference point can be obtained and used to move the model. So after the global adaption stage, differences of pose, size, and position between the model and the organ can be removed. Finally, in the local adaption stage, the local contour differences between the model and the organ can be eliminated using an improved active contour algorithm. In addition, for the task of the FKF extraction, the positions of the FKF elements are marked (except for the radius of the femur head) on the 3D geometric model of the femur manually. Then after the local adaption, all the FKF markers are at the corresponding positions of the segmented femur.

The description of this method is divided into three parts each of which corresponds to one of the three stages. The first part is the pre-processing involving the Mean-Shift filter, thresholding and region growing, and it is discussed in Section 3.2. The applied geometric model is briefly described in Section 3.3. The second part deals with the global adaption and is presented in Section 3.4. The third part is the local adaption and is discussed in Section 3.5. Section 3.6 shows the methods for the 3D reconstruction of the segmented femur. Evaluations are given in Section 3.7.

The input MR image data of the femur are produced by a SIEMENS 1.5 Tesla MR scanner and are composed of axial MR slices with a voxel size of $0.39 \times 0.39 \times 3 \text{ mm}$. Each input MR image data covers a patient body from the beginning of the femur head to the end of the knee. The 3D coordinate system of the MR image data is defined such that the $X$-axis is along the image width, the $Y$-axis is along the image height, and the $Z$-axis is the patient body axis (MR scanning orientation).

### 3.2 Pre-processing

Fig. 3.3 shows three MR slices each of which contains a section of the femur to give an impression of the appearances of the femur in the MR data. Because of the connection

![Figure 3.3: MR slices of three typical sections of femur. From left to right: femur head, femur shaft and knee. Structure and intensity of the head and knee are complex and nonhomogeneous as compared to the shaft.](image-url)
with other bones and the nonhomogeneous intensity, the bones shown in the left (the femur head section) and the right (the knee section) images of Fig. 3.3 are relatively hard to segment as compared to the femur shaft section shown in the middle of Fig. 3.3.

At the beginning of the pre-processing stage, the Mean-Shift filter is applied to denoise the input MR data. Next, the bones are coarsely segmented from these denoised MR data by the thresholding method. Then, the region growing method is used to eliminate small non-bone tissues being falsely classified as of the bone class by the thresholding method.

### 3.2.1 Mean-Shift Filter

For image denoising, the most popular filter is the Gaussian filter [83] by which image noises can be effectively removed. However the disadvantage is that object boundaries in the image are blurred. Since blurred boundaries may cause inaccurate segmentation results, an edge preserving filter is preferred in this method. The bilateral filter [100] and the Mean-Shift filter [22] are two typical edge preserving filters. The principles of both methods are almost the same and the main difference is that the Mean-Shift is an iterative process while the bilateral filter is not. In this method, the Mean-Shift filter is used instead of the Gaussian filter.

Unlike the Gaussian filter where the weight for each neighbor pixel is defined by the Gauss distribution function according to the distance from each neighbor pixel to the central pixel, in the Mean-Shift filter, the weight of each neighbor pixel depends not only on the distance but also on the gray value difference compared to the center pixel. Suppose that \( p = (c, I) \) represents an image pixel where \( c = (x, y) \) are the image coordinates of \( p \), and \( I \) is the grey value of \( p \). In the Mean-Shift algorithm, the new intensity of pixel \( p \) is iteratively computed by using

\[
    r_{i+1} = \frac{\sum_{j=1}^{m} p_j \cdot g_s(||r_{i,1} - p_{j,1}||^2) \cdot g_I(||r_{i,2} - p_{j,2}||^2)}{\sum_{j=1}^{m} g_s(||r_{i,1} - p_{j,1}||^2) \cdot g_I(||r_{i,2} - p_{j,2}||^2)}
\]

where \( r_{i+1} = (c_{r,i+1}, I_{r,i+1}), (i = 1, 2, \ldots, N) \) is the result of the 2D image position and the grey value after the \( i \)-th iteration. \( r_{i,1} \) and \( r_{i,2} \) indicate the first and the second elements of \( r_i \), i.e., \( c_{r,i} \) and \( I_{r,i} \). In the beginning, \( r_1 \) is set to the position and intensity of the pixel \( p \) (i.e., \( r_{1,1} = c \) and \( r_{1,2} = I \)). \( m \) is the total number of image pixels. \( p_j = (c_j, I_j) \) includes position \( c_j \) and intensity \( I_j \) of the \( j \)-th pixel of the image. \( p_{j,1} \) and \( p_{j,2} \) are the first and the second elements of \( p_j \), i.e., \( c_j \) and \( I_j \). \( g_s(x), g_I(x) \) are kernel functions which are usually used in kernel density estimation [67]. Many kinds of kernels can be chosen for the Mean-Shift filter (e.g., normal or uniform distribution). Considering the convergence speed, both of the kernels \( (g_s(x) \) and \( g_I(x) \)) are uniform kernel functions:

\[
    g(x) = \begin{cases} 
    \frac{1}{2t}, & \text{if } |x| \leq t, \\
    0, & \text{else}.
    \end{cases}
\]

In \( g_s(x) \), \( t \) is the distance in pixels (40 in this method), while in \( g_I(x) \), \( t \) is the grey value offset (20 in this method). The iterative process stops when it converges or exceeds the defined number of iterations, then the new intensity for pixel \( p \) is \( I_{r,n} \) (\( n \) is the total number of iteration).
3.2 Pre-processing

Fig. 3.4 shows an example of the Mean-Shift filter on a 1D signal which represents the intensities of each frame of a video sequence at the same pixel position. We can see that edges are well preserved, and each region is more homogenous than the original one.

![Figure 3.4: Example of the Mean-Shift filter. Left: the original 1D signal. Right: the result of the Mean-Shift filtering. The time range is 40 and the intensity value offset is 20.](image)

3.2.2 Thresholding and Region Growing

In this method the threshold is set manually, and by using the thresholding method the denoised MR data of the femur are converted into a set of binary images which contain only two classes: bone and non-bone classes. These two classes are marked in black (bone) and white (non-bone) in the binary images respectively. For the thresholding method, a proper selection of the threshold is crucial to the quality of the segmentation result. Since the Mean-Shift filter makes the intensity of the bones more homogenous, and the intensity of the bones in the T2-weighted MR data is darker than other tissues, a suitable threshold is not difficult to determine. However, it can not be guaranteed that all regions classified as bone class in the binary images are correct, because some small non-bone tissues (e.g., the tendon) may have the same gray value as the bone, and they could be falsely classified as bone class. Therefore, the region growing method is used to remove these fake bones from the binary images. Theoretically, one starting seed is enough to process the whole femur and the seed position is only required to be set at any part of the femur. But in order to make a better result, more seeds set at different femur parts are recommended.

Fig. 3.5 shows three example results of the thresholding and region growing methods. Because of the simple form and its homogeneous intensity, the results of the femur shaft after the region growing method (Fig. 3.5 middle bottom) are satisfactory and can be directly used as the final segmentation results. For the femur head in Fig. 3.5 left and the knee in Fig. 3.5 right, the situation is not as good as the femur shaft. According to the knowledge of the human anatomy, there is a joint between the femur and the pelvis where the femur head is inserted into the acetabulum of the pelvis. Therefore, the segmentation results of the femur head contain both the bones which come from the femur head and those from the pelvis (Fig. 3.5 left bottom). The segmentation result of the knee is easily interfered with by muscle tendons because the origins or the insertions of many muscles...
are located on the knee, and the intensity of the muscle tendon is similar to the bone. Although the pre-segmentation result of the femur has low accuracy, it is enough for the extraction of the spatial information in the next stage i.e., the global adaption stage.

Figure 3.5: Three images at the top show the results of the thresholding method (From left to right: the femur head, the femur shaft and the knee), bottoms are the corresponding results after the region growing method.

After the pre-processing stage, a 3D geometric model of the femur will come on the scene, and this 3D geometric model will be adjusted and deformed in the global and local adaption stages to achieve the final segmentation of the femur.

3.3 The 3D Femur Model

The 3D geometric model of the femur consists of more than 10,000 3D points, and all of them come from a Virtual Reality Modeling Language (VRML) [14] file of a standard femur. In the 3D geometric model (Fig. 3.6), more points are employed to represent the femur parts containing complex structures e.g., the femur head and knee, while fewer points are used to depict the femur parts with simple structures e.g., the femur shaft. The length of the model is 453 mm, and the model axis is the same as the Z-axis. In this femur segmentation task, the femur model is used not only to do the segmentation but also to get the FKF. The idea of getting the FKF is to insert markers into the femur model by adding some 3D points whose coordinates are at the desired FKF positions on the femur model. Then if the femur model is fitted to the femur in the MR data, all FKF markers on the femur will lie on the positions of the corresponding elements of the FKF. In addition, the insertion of the FKF markers is done by hand, and this work needs to be executed only once.

The length of the femur model is fixed, however different people have different femur lengths. Furthermore, the orientation of the femur axis in the human body is not the same as the MR scan direction. The MR scan direction is regarded as the direction of the Z-axis of the 3D coordinate system of the MR data. This means that the orientation of the femur axis is different from the Z-axis. But the axis of the femur model is set to the Z-axis. Moreover, the orientation of the femur axis is different between individuals. Fig. 3.7 left shows an example of the initial femur model and a 3D-volume-displayed pre-segmented result of a femur. Both are in the 3D coordinate system of the MR data.
Figure 3.6: Left: the 3D femur model. Many points are at the femur head and the knee where the anatomic structures are complex, while few points are at the femur shaft which has a simple structure.

above mentioned two differences are obvious. Furthermore, these two femurs have different locations. So to make the femur model fit the femur in the MR data, first, the differences in the length, axis orientation (i.e., pose), and position between the model and the femur in the MR data have to be removed. This work is done in the global adaption stage.

In addition, because the segmentation result of the femur shaft from the pre-processing stage is satisfactory, moreover no FKF locates are at the femur shaft, so only the femur head and the knee of the femur model are focused in the global and local adaption stages. The pre-segmentation result of the femur shaft is regarded as the part of the final segmentation result. By ignoring the femur shaft of the model in the global and local adaption stages, the computational efficiency can be enhanced.

3.4 Global Adaption

The process sequence for the geometric model of the femur in the global adaption stage is rotation (pose adjustment), scaling (size adjustment), and translation (position adjustment). To rotate the model, the axis of the pre-segmentation of the femur is required. To scale the model, the length of the pre-segmentation of the femur has to be obtained. To translate the model, the position of the pre-segmentation of the femur is needed. All the required adjustment parameters are determined in the global adaption stage. In this task of the femur segmentation, the process in the global adaption stage is fully automatic.

3.4.1 Rotation of Femur Axis

Fig. 3.7 right illustrates the approach of the rotation of the femur model. Vector $\overrightarrow{AD}$ represents the axis of the pre-segmented femur in the MR data. As mentioned above, the axis of the initial femur model is set to the Z-axis. As shown in Fig. 3.7 right, making the orientation of the model axis the same as that of the femur axis in the MR data is
Figure 3.7: Left: the 3D femur model (grey object) whose axis is set to the $Z$-axis (marked in blue). The pre-segmented femur (black object) and the 3D model are in the same coordinate system, their lengths, axis orientations, and positions are different. Right: the illustration of the axis orientation of the femur in the MR data. Line $AD$ represents the femur axis.

equivalent to rotating the $Z$-axis to $\overrightarrow{AD}$. It is clear that to make the model axis the same as $\overrightarrow{AD}$, the model axis should be first rotated around the $X$-axis by $\alpha$ degrees and then rotated around the $Z$-axis by $\beta$ degrees. Both rotation directions are clockwise, and the mathematical expression is

$$\frac{\overrightarrow{AD}}{|\overrightarrow{AD}|} = \begin{pmatrix} \cos \beta & \sin \beta & 0 \\ -\sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha \\ 0 & -\sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix},$$ (3.3)

where the $Z$-axis is represented by a unit vector $\overrightarrow{Z}$ i.e., $(0, 0, 1)$. $\alpha$ equals the angle between the vectors $\overrightarrow{AD}$ and $\overrightarrow{AB}$. Vector $\overrightarrow{AB}$ is parallel to the $Z$-axis. Therefore, $\alpha$ can be obtained by

$$\alpha = \arccos\left(\frac{\overrightarrow{Z} \cdot \overrightarrow{AD}}{|\overrightarrow{Z}| \cdot |\overrightarrow{AD}|}\right).$$ (3.4)

$\beta$ is the angle between the vector $\overrightarrow{BD}$ which is the projection of $\overrightarrow{AD}$ on the X-Y plane, and the vector $\overrightarrow{BC}$ which is parallel to the $Y$-axis. Therefore,

$$\beta = \arctan\left(\frac{X(\overrightarrow{AD})}{Y(\overrightarrow{AD})}\right)$$ (3.5)

where $X(\overrightarrow{AD})$ and $Y(\overrightarrow{AD})$ stand for the $x$ and $y$ components of the vector $\overrightarrow{AD}$.

It is clear that in order to calculate the angles $\alpha$ and $\beta$, $\overrightarrow{AD}$ must be extracted. As noted before, $\overrightarrow{AD}$ is the axis of the pre-segmented femur. The idea of getting $\overrightarrow{AD}$ is to get the vector of the distribution direction of the femur shaft centers, and this vector can be regarded as the femur axis. First of all, the femur shaft centers must be localized. Since the
3.4 Global Adaption

Contours of all the pre-segmented femur shaft sections presented in the binary images are circle-like, the Hough circle detection algorithm [34] is applied to find the centers of these femur shaft contours in the binary images. The 3D coordinates of the detected centers are defined in the coordinate system of the MR data. The vector of the femur axis can be deduced from these detected femur shaft centers by the Principal Component Analysis method (PCA). In this femur segmentation task, the detected Hough circles are either at the external contours or at the inner room of the femur shaft, and the circle centers of both cases are acceptable (Fig. 3.8).

**Figure 3.8:** Three examples of the Hough circle detection of the pre-segmented femur shaft. Examples are selected (top row from left to right) at the top, middle, and bottom of the femur shaft. Images at the bottom row show the corresponding detected circles (marked in red) and the circle centers (marked as blue dots).

Because the pre-segmentation results include a whole femur, it is unknown that which part is the femur head and which belongs to the femur shaft. So the Hough circle detection method is applied to the whole pre-segmented femur. The detected circle centers are drawn in the 3D coordinate system of the MR data (see Fig. 3.9). From Fig. 3.9, it is clear that the detected circle centers are distributed in three main femur parts: the femur head, the femur shaft, and the knee. Most centers are located in the femur shaft, and they

**Figure 3.9:** 3D-volume-displayed front and side views of the pre-segmented femur (gray object). The extracted circle centers are marked in violet.
are expected to lie in a line. However, due to the deviation of the center extraction and
the deformation present in the femurs of old people, the detected femur shaft centers are
usually arranged in a curve.

As noted before, the femur axis is the vector of the distribution direction of the detected
femur shaft centers and the PCA is used to get the vector. However, the detected centers
are not only from the femur shaft but also from the femur head and the knee. Therefore,
before using the PCA, each detected circle center must be checked whether it belongs to
the femur shaft or not. Centers located at the femur shaft part are preserved and centers
at the femur head and the knee must be filtered out. To achieve this filtration task, the
Directed Acyclic Graph (DAG) \([47, 62, 99]\) is applied. In the DAG algorithm, all detected
centers are regarded as nodes and each two adjacent nodes are connected by an edge. The
weight for each edge is defined as the multiplicative inverse of the projection distance of
the two adjacent nodes. For each two nodes \((x_1, y_1, z_1)\) and \((x_2, y_2, z_2)\), if \(|z_1 - z_2| < d_t\) \((d_t\) is a pre-defined distance) then the weight \(w\) is defined as

\[
w = \frac{1}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}} \tag{3.6}
\]

In this method \(d_t\) is equal to the thickness of two MR slices. The objective of the DAG
is to find the longest connection sequence. The DAG starts from the node at the top of
the pre-segmented femur and searches the nodes below within the distance \(d_t\). If there is
more than one corresponding node, the DAG chooses the one with the biggest weight edge.
Then the DAG turns to the searching for the next node below the new selected node in
the same way until no further nodes are available. If within the distance \(d_t\), no node below
a new selected node, then the up to now searched nodes and edges will be regarded as a
connection sequence. After that the DAG continues to search a new start node from the
rest of the nodes below in order to begin a new round of searching for the next connection
sequence. Fig. 3.10 shows the result of the DAG for a pre-segmented right femur. The

![Figure 3.10: The result of the DAG algorithm. The detected circle centers (marked in violet)
are regarded as nodes, and each two nodes which satisfy the distance constraint are connected
by a weighted edge. The edge weight is represented by the edge width (a higher weight is given
a wider link).](image-url)
violet points are the detected circle centers from the Hough circle detection method, and they can be regarded as the nodes. An edge is connected to each two nodes if they satisfy the distance constraint. The weight of each edge is represented visually by the edge width (a higher weight receives a greater width).

It is clear that the longest connection sequence of the DAG belongs to the femur shaft part, and the nodes (i.e., detected centers) belonging to the longest connection sequence are preserved and used to get the femur axis via the PCA. Fig. 3.11 shows the result of the preserved centers (left) and the calculated femur axis (right).

![Figure 3.11: Left: the preserved centers (violet points), all of which are located in the femur shaft. Right: the calculated femur axis (the red line).](image)

After getting $AD$, the angles $\alpha$ and $\beta$ can be calculated by using equation 3.4 and 3.5. By means of the values of $\alpha$, $\beta$ and equation 3.3, the rotation of the axis of the femur model to the axis of the pre-segmented femur can be achieved. After the rotation, the next step in the global adaption stage is the femur model length scaling where the Vertical Span Length (VSL) of the pre-segmented femur is used to adjust the femur model.

### 3.4.2 Scaling of The Vertical Span Length

The reason why the VSL is used not the real length is that after the rotation of the femur model, the orientation of the femur model axis and pre-segmented femur axis are the same. Assuming that the angle between the femur axis and the horizontal plane is $\delta$, the VSLs of the femur model and the pre-segmented femur are $V_m$ and $V_f$. The real lengths of the model $L_m$ and the pre-segmented femur $L_f$ should be equal to $V_m / \sin(\delta)$ and $L_f = V_f / \sin(\delta)$ respectively. So scaling to the same VSL is equivalent to the scaling to the same length. Moreover, the VSL is much easier to obtain. It is calculated by multiplying the MR slice thickness $t$ by the number of MR slices $n$ contained in the MR data of the femur. In this femur segmentation task, thickness of the input MR data is 3 $mm$ ($t = 3 \, mm$). Since the MR femur data used in this task are from the top of the femur head to the end of the knee, $n$ is the total number of the MR slices included in the MR data. Therefore, the model is scaled simply to the VSL of $t \times n \, mm$. 

47
3.4.3 Position Translation

The last work in the global adaption stage is to translate the femur model to the same position of the pre-segmented femur. The center position of the middle of the femur is considered as the reference position. For the pre-segmented femur, the image at the middle of the set of the binary images containing the pre-segmented femur is regarded as the middle of the femur. Since the femur shaft occupies the largest part of the femur body, the middle image always contains a cross section of the femur shaft. Furthermore, the center of each femur shaft cross section is available, because in the adjustment of the femur model axis (the first process of the global adaption), the center of each femur shaft cross section was already extracted by the Hough circle detection method. Assuming that the center of the cross section of the femur shaft in the middle image is \( (x_i, y_i, z_i) \). For the femur model, the middle part, which corresponds to the middle of the pre-segmented femur, is the middle level of the femur model VSL and the center position is the average position of the model points located at the middle level. Assuming that the model center position is \( (x_m, y_m, z_m) \). Then the translation vector is \( (x_i - x_m, y_i - y_m, z_i - z_m) \), and the result of the global adaption is shown in Fig. 3.12.

![Figure 3.12: Result of the global adaption in the 3D coordinate system of the MR data of the femur. The grey object is the 3D-volume-displayed pre-segmented femur, and the white object is the femur model after the global adaption.](image)

3.5 Local Adaption

It is impossible to make the femur model totally match the femur in the MR data by using only the global adaption as there are many local differences between the model and the pre-segmented femur. Furthermore, the pre-segmented femur result is inaccurate and contains not only the femur but also some noises. Therefore, a local adaption is required to accomplish a good matching.
3.5 Local Adaption

The objective of the local adaption stage is to deform the femur model contour to completely match the femur in the MR data. Since the model is aligned with the pre-segmented femur after the global adaption, for each binary image containing the pre-segmented femur and each MR slice of the femur, there is a corresponding 2D cross section contour of the femur from the femur model. This 2D cross section contour of the model is adjusted and deformed according to the information from the MR slice and the binary image which contain the corresponding cross section of the femur. The adjustment and deformation in the local adaption stage include the affine transformation and the non-rigid (improved active contour [54]) matching.

Because the femur model is constructed by 3D points, a method which is capable of making a 2D contour from the 3D points at a desired level of the femur model is required.

3.5.1 Model Contour Extraction

In order to get the 2D contour of the cross section at each level of the femur model, all 3D points, which are at the desired level of the femur model, are projected onto a 2D image plane. Since the thickness of the input MR images is 3 mm, for each level $\rho$, the 3D points, which are within the range of $\rho - 1.5 \text{ mm } \sim \rho + 1.5 \text{ mm}$, are considered to be at the same level. In order to project the 3D points whose units are millimeters onto the 2D image whose unit is pixel, the pixel size of the 2D image is set to 0.39 × 0.39 mm which is the same as that of the input MR data. According to the pixel size, the 2D image coordinates $(x_{2D}, y_{2D})$ of each 3D point $(x_i, y_i, z_i)$ are computed by $x_{2D} = [x_i/0.39 + 0.5]$ and $y_{2D} = [y_i/0.39 + 0.5]$.

Normally, the number of the 3D points at each level is more than 200. Some projection examples are shown in Fig. 3.13. The result for the femur shaft is not acceptable as few points are located on it. But as mentioned above, the femur shaft of the model is not used and it is shown here just for giving an impression.

![Figure 3.13: Five examples of the cross sections of the femur model (gray object). Each level is indicated by a red line.](image-url)
The results of the projected 3D points at each level can not be directly used as the 2D contour for the local adaption, because the projection results usually have gaps and are not smooth. Therefore, the morphological closing operation \[ [82] \] is applied to fill up the gaps and make them smooth. The 2D contour of each level of the femur model is defined as the external contour, so the inner room are filled by the flood-fill method \[ [103] \]. Then by means of the chain code method \[ [37] \], the 2D contour of the femur model at each level can be obtained. Fig. 3.14 illustrates this contour extraction process. The top left of Fig. 3.14 is the projection result of the 3D points. It is clear that many gaps are present in both contours. The top right is the result after the closing operation and the floodfill, and both contours have no gaps and are smooth. The bottoms are the two contours, which are obtained by using the chain code on the top right image.

Figure 3.14: An example of the contour extraction. The top left image shows the result of the 3D points projection. The top right image shows the result after the morphological closing operation and the floodfill. The two images at the bottom are the 2D contours obtained by the chain code method.

In addition, during the 3D to 2D projection of each level of the femur model, the 3D FKF markers are also included at some levels and the projected FKF 2D points do not participate in the contour extraction as they only indicate the FKF positions.

3.5.2 Affine Transformation and Non-Rigid Matching

In this part, the extracted 2D contours from the femur model will deform automatically to match the corresponding cross sections of the femur in the MR data. The local adaption process is composed of an affine transformation and a non-rigid adaption which uses the improved active contour method. The affine transformation makes the 2D contour generally fit the corresponding cross section of the pre-segmented femur, and the improved active contour method corrects the former result at some small parts of the contour according to the information from both the MR data (containing the original femur images) and the pre-segmented femur (in the binary images).

Affine Transformation

The affine transformation involves scaling, rotation and translation (no shear). Because of the global adaption, the differences in size, pose and location between the 2D contour of
the model and the corresponding cross section of the pre-segmented femur are not so big. Therefore, a drastic adjustment in the affine transformation is not needed. The quality of the affine transformation is estimated by the number of the pre-segmented femur boundary points which lie on the 2D contour of the model. In order to achieve the global optimization, the exhaustive searching method is used to get the parameters for the scaling, rotation and translation. In the exhaustive searching process, the range of the scaling factor is set to \([0.9, 1.1]\) (step size 0.1), the rotation range is \([-30, 30]\) degrees (step size 10 degrees), and the translation range is \([-10, 10]\) pixels for both \(x\) and \(y\) directions (step size 1).

### Non-Rigid Matching

The next step after the affine transformation is the non-rigid matching which uses an improved active contour method. The traditional active contour model or snakes [54] (or see Section 2.2.1) uses the energy function

\[
e = \int_0^1 e_{\text{int}}(C(s)) + e_{\text{img}}(C(s)) + e_{\text{con}}(C(s)) ds. \tag{3.7}
\]

The big disadvantage is that it suffers from the short capture range, which means that if the initial contour is not near the desired object boundary, the contour may not be able to approach the right boundary. Motivated by [51] which uses a multi-resolution method to extend the capture range, the edges of the binary images containing the pre-segmented femur are calculated by the Canny method [13], then the edge images are blurred by the Gaussian filter with a large deviation \(\sigma\). Since the accuracy of the pre-segmented femur is not high, using only the pre-segmented femur to guide the contour is not a good idea. Therefore, besides the large scaled edge images of the pre-segmented femur, the Mean-Shift filtered MR slices which are produced in the pre-segmentation stage are also considered. So the image energy \(e_{\text{img}}(C(s))\) used here is composed of four elements:

\[
e_{\text{img}}(C(s)) = w_1 e_{\text{line}}^{(1)}(C(s)) + w_2 e_{\text{line}}^{(2)}(C(s)) + w_3 e_{\text{edge}}^{(1)}(C(s)) + w_4 e_{\text{edge}}^{(2)}(C(s)) \tag{3.8}
\]

where \(e_{\text{line}}^{(1)}\) and \(e_{\text{edge}}^{(1)}\) stand for the image gray value and the modified image gradient \(-|\nabla I(x, y)|^2\) of the large scaled edge images of the pre-segmented femur. \(e_{\text{line}}^{(2)}\) and \(e_{\text{edge}}^{(2)}\) stand for the same features but of the Mean-Shift filtered original MR slices. To attract the initial model contour to the femur boundary, \(w_1\) should be set to a negative value, because in the large scaled edge image of the pre-segmented femur, the femur boundary is brighter than any other position. In contrast, in the Mean-Shift filtered MR slices the femur boundary is dark, so \(w_2\) is set to a positive value. Both \(w_3\) and \(w_4\) are set to positive values. The intrinsic energy \(e_{\text{int}}(C(s))\) here contains not only the first and second derivatives \((C, C)\) but also a new element \(e_{\text{exp}}(C(s))\) which prevents the model contour from shrinking when \(e_{\text{img}}(C(s))\) is not dominant. \(e_{\text{exp}}(C(s))\) is defined as a normalized inverse of the distance between each model contour point and the contour center. The user defined constraint \(e_{\text{con}}(C(s))\) is not needed here so it is not included in the applied energy function.

Fig. 3.15 shows an example of the deformation of the model contour by using the improved active contour method. At this level, these are two contours (one at the femur head and the other one at the trochanter major). The image at the top left contains the
result of the model contour of the trochanter major (marked in green) after the affine transformation. It is clear that some local differences exist. The refinement result of the improved active contour method is shown in the top right image where the local differences are removed. The remaining two images at the bottom show the same results of the model contour of the femur head after the affine transformation (bottom left) and the improved active contour method (bottom right).

Figure 3.15: The improved active contour algorithm for deforming the contour to fit the femur in binary images.

In addition, if the corresponding model contours at some levels are composed of more than one sub-contour (as the case shown in Fig. 3.15), then each contour will be processed independently, and if the model contour contains the FKF markers, the positions of the FKF markers will be changed along with the model contour in the affine transformation.

3.5.3 The Extraction of FKF

After the global and the local adaption stages, the femur model is totally adapted to the femur in the MR data, and all the FKF markers are also at the corresponding positions. Fig. 3.16 shows the final result of the FKF markers (marked as red points) in three different views.

Figure 3.16: Different views of the extracted FKF. The black points are from the segmented femur and the red points are the FKF markers including the femur head center, the femur neck center, the trochanter major, and the trochanter minor. The red line is the femur axis.
3.6 3D Femur Reconstruction

The purpose of the reconstruction of the 3D femur from the segmented femur is to produce the 3D view of the segmented femur. There are two 3D reconstruction methods used in this method, the first one is the Iso-surface [106] and the other one is the Delaunay triangulation [39].

**Iso-surface** The Iso-surface is widely used in computer graphics. The surface is determined by the properties (e.g., similar gradient or gray value) of the input image pixels. In this method, the binary images of the segmented femur are used as the input images to get the surface. To construct the surface, a widely used method called the marching cubes [53] is applied. First of all, the unit of the coordinates of each pixel in the binary images is converted from pixel to millimeters based on the voxel size of the input MR data. Then according to the algorithm of the marching cubes, each 8 adjacent pixels in the binary images are inspected to check the intensity of each pixel of the 8 adjacent pixels (Fig. 3.17 left). In the binary images, pixels belonging to the femur have the intensity of 0, and pixels belonging to non-femur have the intensity of 255. Based on the intensity of the 8 pixels, a surface, which separates the 8 pixels into two parts, can be determined. In theory, there are $2^8 = 256$ kinds of surfaces each of which corresponds to one combination of the 8 pixel intensities. After using the marching cubes through the whole binary images, all surfaces can be determined, and all of them should be located at the surface of the segmented femur (Fig. 3.18 right).

**Delaunay triangulation** In the Delaunay triangulation method, edge points of the segmented femur are extracted using the Canny algorithm. Then, the units of the coordinates of each edge are converted from pixel to millimeters. According to the algorithm of the
Delaunay triangulation, the triangulation is produced from each three edge points in a 3D coordinate system under the constraint that no edge is contained in the circumscribed circle of the triangle. Finally, all produced triangulations form the 3D surface of the segmented femur (Fig. 3.18 left).

Both methods (Iso-surface and Delaunay) are included in a medical image visualization library called the Visualization Toolkit (VTK) [80] which is applied here for the 3D reconstruction (Fig. 3.18).

Figure 3.18: 3D reconstruction of the segmented femur. The left shows the reconstruction result using the Delaunay method and the right shows the result of the iso-surface method.

3.7 Evaluation

10 sets of the T2-weighted axial MR femur data are used to evaluate the method for segmenting the femur, all of the MR data are provided by the Universitätsklinikum Essen from 10 volunteer patients. The method is composed of three stages: the pre-processing stage, the global adaption stage, and the local adaption stage. In this evaluation, the result from each stage will be evaluated, because it is helpful to know which stage is relatively unreliable and the possible origin of the inaccurate result.

3.7.1 Evaluation of Pre-processing Results

The thresholding and region growing methods are carried out in the pre-processing stage. For the thresholding method, the threshold is set manually. The intensity of bones in the T2-weighted MR data is distinct from that of other organs, and the main challenges to the thresholding method are the image noise and the partial volume effect. Fortunately, both influences can be greatly reduced by the Mean-Shift filter and the quality of the thresholding result is significantly enhanced. Fig. 3.19 shows some examples of the thresholding
Figure 3.19: The thresholding results of the femur head (top row), the knee (middle) and the femur shaft (bottom). The three images at the top of each row (left to right) are the original image, the original images after 1 iteration, and that after 3 iterations of the Mean-Shift filter. The three images at the bottom of each row are the corresponding results of thresholding. The threshold is 40.

results of the images with and without the Mean-Shift filter. There are three rows each of which shows the femur head, the knee and the femur shaft respectively. Each row is composed of three image pairs. Each pair contains a MR slice and a corresponding thresholding result as a binary image. In each row, the image pairs from left to right are the images of the original MR slice, the MR slice after 1 iteration and that after 3 iterations of the Mean-Shift filter. The thresholding result of the original MR slice is influenced by
the noise and the partial volume effect, many outliers are included in the binary image. But the thresholding results of the two Mean-Shift filtered MR slices are good, and more iterations of the Mean-Shift filter make the result better. The problem of the thresholding is that there exist some small non-bone tissues which have the same intensity as bones. Those tissues are also classified as bone class. Therefore, the 3D region growing (26-connected) method is used after the thresholding to eliminate those non-bone tissues. In the experiments, the region growing works well and reliably with at least one given seed which can be set at anywhere inside the femur.

3.7.2 Evaluation of Global Adaption

After the pre-processing, the pre-segmented femur is put into the global adaption stage where a 3D geometric femur model is adapted to it. The global adaption stage includes three steps: axis adjustment, length scaling, and position translation. The length scaling is done by scaling the femur model to have the same VSL as the pre-segmented femur. The calculation of the VSL only needs the number of MR slices and the voxel size of the input MR data, and both are exactly given by the MR data, so the length scaling has no need to be evaluated. For the translation of the femur model, the middle points of the model and the pre-segmented femur are used to get the translation vector. The process is too simple, and high accuracy is not required. Therefore, the evaluation of the global adaption only focuses on the femur axis adjustment and the final result of the global adaption.

Evaluation of Femur Axis Results

The calculation of the femur axis is a fully automatic process. For the pre-segmented femur, a Hough circle detection is applied to extract the centers of the femur shaft sections. By using all the extracted shaft centers, the femur axis can be obtained through the PCA. Fig. 3.20 shows some extracted femur axes in a 3D coordinate system.

![Figure 3.20: Results of the femur axis extraction. The gray objects are the pre-segmented femurs. The violet points are the extracted circle centers and the red lines are the femur axes.](image-url)
3.7 Evaluation

To evaluate the accuracy of the result of the femur axis, the ground truth of the femur axis is produced by using the PCA on the manually selected centers of some cross sections of the femur shaft in the MR data. The manually selected centers of the femur shaft sections have to be evenly distributed from the beginning to the end of the femur shaft. The axis deviation is defined as the angle (in degrees) between the automatic extracted femur axis and the ground truth. Table 3.1 shows the details of the evaluation of the femur axis extraction. For each one of the 10 MR data sets, the automatically extracted femur axis and the corresponding ground truth are represented as the normalized 3D vectors. The deviation is the angle between these two vectors in the unit of degree. As shown in Table 3.1, the mean deviation is 1.0037 degrees.

**Table 3.1:** Comparison of the automatically and manually extracted femur axes. 10 patient data sets (labeled 01-10) are used and the deviation is the angle between the two extracted axes in degrees.

<table>
<thead>
<tr>
<th>Data ID</th>
<th>Calculated Axis</th>
<th>Manual Axis</th>
<th>Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>-0.04623, -0.1617, -0.9858</td>
<td>-0.04785, -0.1606, -0.9859</td>
<td>0.1100</td>
</tr>
<tr>
<td>02</td>
<td>-0.05133, -0.1530, -0.9868</td>
<td>-0.05381, -0.1467, -0.9877</td>
<td>0.3901</td>
</tr>
<tr>
<td>03</td>
<td>-0.08754, -0.1472, -0.9852</td>
<td>-0.08442, -0.1396, -0.9866</td>
<td>0.4782</td>
</tr>
<tr>
<td>04</td>
<td>-0.05699, -0.1511, -0.9869</td>
<td>-0.06924, -0.1414, -0.9875</td>
<td>0.8966</td>
</tr>
<tr>
<td>05</td>
<td>-0.08391, -0.1328, -0.9875</td>
<td>-0.06637, -0.09007, -0.9937</td>
<td>2.6719</td>
</tr>
<tr>
<td>06</td>
<td>-0.1082, -0.09117, -0.9899</td>
<td>-0.1097, -0.07146, -0.9914</td>
<td>1.1356</td>
</tr>
<tr>
<td>07</td>
<td>-0.1091, -0.08978, -0.9900</td>
<td>-0.1237, -0.07175, -0.9897</td>
<td>1.3295</td>
</tr>
<tr>
<td>08</td>
<td>-0.1046, -0.05706, -0.9929</td>
<td>-0.1016, -0.05082, -0.9935</td>
<td>0.3973</td>
</tr>
<tr>
<td>09</td>
<td>-0.1465, -0.2064, -0.9674</td>
<td>-0.1415, -0.2096, -0.9675</td>
<td>0.3401</td>
</tr>
<tr>
<td>10</td>
<td>-0.02094, -0.1592, -0.9870</td>
<td>-0.05615, -0.1404, -0.9885</td>
<td>2.2881</td>
</tr>
<tr>
<td>Mean</td>
<td>–</td>
<td>–</td>
<td>1.0037</td>
</tr>
</tbody>
</table>

**Evaluation of Global Adaption Results**

After the axis adjustment, the length scaling and the position translation, the global adaption stage is complete. To evaluate the result of the global adaption, the Jaccard similarity coefficient (Jaccard index) [93] is used to measure the similarity between the femur model and the manually segmented femur. The Jaccard index considers not only the precision but also the recall of the result, and it is proved to be a good choice for measuring the segmentation accuracy [11]. Supposing that $\Gamma$ is the region within the contour of the femur model and $\Omega$ is the region of the manually segmented femur in the
corresponding MR slice. Then the Jaccard index $J$ is calculated by

$$J = \frac{\Gamma \cap \Omega}{\Gamma \cup \Omega}.$$ (3.9)

The value of the Jaccard index of the two regions is between 0 (no overlap) and 1 (identity).

Table 3.2 shows the evaluation results of the global adaptation of the 10 MR data sets. Since each MR data set has more than 100 MR slices i.e., more than 100 cross sections of the femur, the Jaccard index $J$ can not be given for each MR slice. In this table, only the maximum and minimum values of $J$ ($J_{\text{max}}$ and $J_{\text{min}}$) for each MR data set, and the mean $J$ ($J_{\text{mean}}$), which is the average value of $J$ of all MR slices in each MR data set, are provided. A statement of the evaluation has to be announced here. Except the influence of the specific method, the Jaccard index of each test data also heavily depends on the test image quality (e.g., image contrast and resolution) and the manually delineated ground truth which can not be reproduced. Furthermore, the number of test image data is small. Therefore, the evaluation results can only give an approximate impression of the performance of the method, and they do not have statistical significance. Such problem exists in all the evaluations in this thesis.

**Table 3.2:** Evaluation of the overlap between the model contour after the global adaption and the manually segmented femur. $J_{\text{max}}$ and $J_{\text{min}}$ are the maximum and the minimum values of the Jaccard index in each MR data set. $J_{\text{mean}}$ is the average value of the Jaccard index of each data set.

<table>
<thead>
<tr>
<th>Data ID</th>
<th>$J_{\text{max}}$</th>
<th>$J_{\text{min}}$</th>
<th>$J_{\text{mean}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>0.66</td>
<td>0.37</td>
<td>0.51</td>
</tr>
<tr>
<td>02</td>
<td>0.75</td>
<td>0.42</td>
<td>0.68</td>
</tr>
<tr>
<td>03</td>
<td>0.76</td>
<td>0.50</td>
<td>0.67</td>
</tr>
<tr>
<td>04</td>
<td>0.77</td>
<td>0.43</td>
<td>0.66</td>
</tr>
<tr>
<td>05</td>
<td>0.71</td>
<td>0.27</td>
<td>0.54</td>
</tr>
<tr>
<td>06</td>
<td>0.72</td>
<td>0.44</td>
<td>0.56</td>
</tr>
<tr>
<td>07</td>
<td>0.70</td>
<td>0.17</td>
<td>0.50</td>
</tr>
<tr>
<td>08</td>
<td>0.52</td>
<td>0.43</td>
<td>0.46</td>
</tr>
<tr>
<td>09</td>
<td>0.72</td>
<td>0.51</td>
<td>0.68</td>
</tr>
<tr>
<td>10</td>
<td>0.61</td>
<td>0.43</td>
<td>0.52</td>
</tr>
<tr>
<td>Mean</td>
<td>0.69</td>
<td>0.40</td>
<td>0.58</td>
</tr>
</tbody>
</table>

The average maximum, minimum and mean values of $J$ of the 10 MR data sets are 0.69, 0.40 and 0.58 which meet the expected requirement, and they are good enough to be used as the input of the local adaption stage.
3.7 Evaluation

3.7.3 Evaluation of Local Adaption

The local adaption stage contains a local affine transformation including rotation, scaling and translation of the model contour and a local non-rigid matching which uses the improved active contour method. The local adaption is needed to solve the contour differences which can not be removed by the global adaption.

Evaluation of Local Affine Transformation Results

Since the pose, size, and location of the femur model after the global adaption is close to the pre-segmented femur, the affine parameters concerning the degrees of rotation, the scaling factor, and the translation do not need to cover a large range. As noted before, the rotation parameter is set to the range of \([-30, 30]\) degrees. The range of the scaling factor is set to \([0.9, 1.1]\). The translations in both \(x\) and \(y\) directions are set to the range of \([-10, 10]\) pixels. The strategy of searching for the best affine parameters is an exhaustive searching process. The step sizes of the rotation, the scaling factor, and the translations in both \(x\) and \(y\) directions are 10 degrees, 0.1 and 1 pixel, respectively. Table 3.3 shows the evaluation results of the local affine transformation. The evaluation still uses the Jaccard index. The items of this table are the same as those of Table 3.2. It is clear that the results are highly improved by the local affine transformation as compared to the results of the global adaption.

<table>
<thead>
<tr>
<th>Data ID</th>
<th>(J_{\text{max}})</th>
<th>(J_{\text{min}})</th>
<th>(J_{\text{mean}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>0.80</td>
<td>0.61</td>
<td>0.68</td>
</tr>
<tr>
<td>02</td>
<td>0.89</td>
<td>0.69</td>
<td>0.76</td>
</tr>
<tr>
<td>03</td>
<td>0.88</td>
<td>0.70</td>
<td>0.73</td>
</tr>
<tr>
<td>04</td>
<td>0.83</td>
<td>0.67</td>
<td>0.70</td>
</tr>
<tr>
<td>05</td>
<td>0.81</td>
<td>0.55</td>
<td>0.67</td>
</tr>
<tr>
<td>06</td>
<td>0.85</td>
<td>0.61</td>
<td>0.69</td>
</tr>
<tr>
<td>07</td>
<td>0.80</td>
<td>0.53</td>
<td>0.66</td>
</tr>
<tr>
<td>08</td>
<td>0.76</td>
<td>0.60</td>
<td>0.64</td>
</tr>
<tr>
<td>09</td>
<td>0.86</td>
<td>0.71</td>
<td>0.79</td>
</tr>
<tr>
<td>10</td>
<td>0.79</td>
<td>0.69</td>
<td>0.71</td>
</tr>
<tr>
<td>Mean</td>
<td>0.83</td>
<td>0.64</td>
<td>0.70</td>
</tr>
</tbody>
</table>
Evaluation of Local Non-Rigid Matching Results

The local non-rigid matching of the model contour is the last step of the local adaption stage. The improved active contour method is used to solve the contour differences which cannot be removed by the global adaption and the local affine transformation. The detailed evaluation results of the local non-rigid matching are shown in Table 3.4. Since the local non-rigid matching deals with the contour difference which occupies a small part of each femur cross section, the improvement of the value of $J$ is not so significant. For some cross sections of the femur, the values of $J$ are even decreased as the expanding of the model contour during the non-rigid matching (Fig. 3.21). But in most cases, the values of $J$ are

Table 3.4: Evaluation of the overlapping between the manually segmented bone and the model contour after the local non-rigid matching.

<table>
<thead>
<tr>
<th>Data ID</th>
<th>$J_{\text{max}}$</th>
<th>$J_{\text{min}}$</th>
<th>$J_{\text{mean}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>0.86</td>
<td>0.62</td>
<td>0.71</td>
</tr>
<tr>
<td>02</td>
<td>0.92</td>
<td>0.67</td>
<td>0.77</td>
</tr>
<tr>
<td>03</td>
<td>0.90</td>
<td>0.71</td>
<td>0.75</td>
</tr>
<tr>
<td>04</td>
<td>0.83</td>
<td>0.69</td>
<td>0.73</td>
</tr>
<tr>
<td>05</td>
<td>0.82</td>
<td>0.65</td>
<td>0.70</td>
</tr>
<tr>
<td>06</td>
<td>0.89</td>
<td>0.62</td>
<td>0.73</td>
</tr>
<tr>
<td>07</td>
<td>0.88</td>
<td>0.56</td>
<td>0.71</td>
</tr>
<tr>
<td>08</td>
<td>0.85</td>
<td>0.65</td>
<td>0.71</td>
</tr>
<tr>
<td>09</td>
<td>0.87</td>
<td>0.76</td>
<td>0.81</td>
</tr>
<tr>
<td>10</td>
<td>0.82</td>
<td>0.75</td>
<td>0.78</td>
</tr>
<tr>
<td>Mean</td>
<td>0.86</td>
<td>0.67</td>
<td>0.74</td>
</tr>
</tbody>
</table>

Figure 3.21: Because of the influence of the pelvis bone, the model contour is attracted to the pelvis (right) during the non-rigid matching which causes the decrease value of $J$. 
improved by the non-rigid matching. Fig. 3.22 shows some typical results of the model contour after the global adaption, the local affine transformation and the local non-rigid matching.

![Figure 3.22: Results of the model contours after the global adaption (left), the local affine transformation (middle), and the local non-rigid matching (right). The images in each row contain different parts of the femur. The green lines indicate the model contours.]

As a summary of the evaluation of the femur segmentation, a diagram of the mean values of $J$ at the global adaption, the local affine transformation and the local non-rigid matching is shown in Fig. 3.23. It is clear that the quality of the segmented femur is improved step by step.

### 3.7.4 Evaluation of FKF Results

The femur FKF results are obtained after the final adaption of the femur model. Table 3.5 shows the evaluation details about the extracted FKF. The extracted FKF results of the 10 MR data sets are compared with the corresponding manually selected results to get the disparity (the distance). The disparity is given in millimeters.

The maximum and minimum of the average disparities of the FKF results are 4.28 mm and 0.72 mm. 4.28 mm is the average disparity of the position of the Trochanter major, and 0.72 mm is the average disparity of the radius of the femur head. Because surrounding the trochanter major there is no special anatomic structure which can be used as a clue for locating the Trochanter major. So the model contour adaption only focuses on the fitting of the boundary, and the relevant position for the trochanter major on the model contour is ignored.
3 Segmentation of Rigid Body Organ: Femur and FKF

Figure 3.23: Comparison of the segmentation results after the global adaption, the local affine transformation, and the local non-rigid matching of the 10 MR data sets.

Table 3.5: Disparities (in millimeters) between the calculated and the manually selected FKF results.

<table>
<thead>
<tr>
<th>Data ID</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.86</td>
<td>0.78</td>
<td>3.20</td>
<td>5.20</td>
<td>2.96</td>
</tr>
<tr>
<td>2</td>
<td>1.44</td>
<td>0.65</td>
<td>1.34</td>
<td>5.67</td>
<td>4.42</td>
</tr>
<tr>
<td>3</td>
<td>3.86</td>
<td>0.39</td>
<td>2.24</td>
<td>4.00</td>
<td>2.47</td>
</tr>
<tr>
<td>4</td>
<td>3.42</td>
<td>1.12</td>
<td>3.04</td>
<td>5.33</td>
<td>2.31</td>
</tr>
<tr>
<td>5</td>
<td>3.23</td>
<td>1.46</td>
<td>3.30</td>
<td>3.10</td>
<td>3.44</td>
</tr>
<tr>
<td>6</td>
<td>2.29</td>
<td>0.46</td>
<td>2.73</td>
<td>3.24</td>
<td>1.52</td>
</tr>
<tr>
<td>7</td>
<td>1.80</td>
<td>0.61</td>
<td>1.05</td>
<td>3.81</td>
<td>2.24</td>
</tr>
<tr>
<td>8</td>
<td>0.94</td>
<td>0.55</td>
<td>3.39</td>
<td>3.80</td>
<td>2.55</td>
</tr>
<tr>
<td>9</td>
<td>1.28</td>
<td>0.65</td>
<td>2.05</td>
<td>3.47</td>
<td>3.00</td>
</tr>
<tr>
<td>10</td>
<td>0.39</td>
<td>0.55</td>
<td>2.30</td>
<td>5.15</td>
<td>2.40</td>
</tr>
</tbody>
</table>

| Mean    | 2.25 | 0.72 | 2.46 | 4.28 | 2.73 |

(a: center of femur head. b: radius of femur head. c: center of femur neck. d: Trochanter major. e: Trochanter minor. Units: \textit{mm})

3.8 Conclusions

In this chapter, the femur segmentation, which uses the segmentation method of the rigid body organ, has been discussed. Furthermore, this method is capable of extracting the FKF of the femur. By evaluating the experimental results, the accuracy of the segmented
femur and the extracted FKF are at a high level.

For the future improvement of this segmentation method, a bias correction algorithm [42, 52, 77] on the input MR data could be used to produce uniformly illuminating MR slices. By using the uniformly illuminating MR slices, the pre-segmentation result obtained by the thresholding and region growing methods could be more accurate. Considering the low variation and good image contrast of the rigid body organ in MR images, the active contour model is selected as the deformation model. In order to make the method more robust and accurate, more intelligent deformation model could be used. An interesting deformation model used to segment the femur can be found in [33] where a shape-based level-set method was used to do the femur segmentation.

Because the variability of the rigid body organ as compared to the soft body organ is low, the model adaption strategy is simpler than that for the soft body organ. In the next chapter, the performance of a geometric deformable model based segmentation method of the soft body organ, which is more challenging than the rigid body organ segmentation, will be discussed.
4 Segmentation of Soft Body Organ: Muscle and TAS

Soft body organ segmentation is one of the most challenging topics in medical image processing as the soft body organ has no regular forms and deforms non-rigidly during human movements. Furthermore, the boundaries between soft body organs in common medical image modalities, e.g., CT and MR, are difficult to identify even with the naked eye. In this chapter, a method for the segmentation of the soft body organ is going to be discussed, and the method is semi-automatic which means human interactions are needed. As the previous chapter, an application of this method is presented. This application focuses on the segmentation of muscles and their Tendon Attachment Sites (TASs). Both of the results are used in the 3D modeling of the gait of patients, and the results of the TASs are also needed by the mechanics to do a force analysis during the walk of patients. The muscle of interest, which is used to demonstrate the process of this method, is called the sartorius muscle. In theory, if the muscle has no branches and its TASs are located at distinguished bone sections, it can be segmented by this method. Previous publications about this method can be found in [96, 98].

4.1 Introduction

Unlike the rigid body organ, whose form does not vary much between individuals e.g., the femur, the form of the soft body organ changes not only between individuals but also over time e.g., the heart. Moreover, as the result of limitations of current medical imaging technologies, the image contrast of soft body organs is usually too low to distinguish their boundaries. The muscle, which is a kind of the soft body organ, is concerned here. The task is to segment the muscle and its TASs, and a segmentation method of the soft body organ is proposed and applied to complete this task. A muscle called the sartorius muscle is used as the muscle of interest to demonstrate the segmentation process. The sartorius muscle (Fig. 4.1) is the longest muscle of the human, and it participates in hip flexion and lateral rotation as well as the flexion of the knee. This muscle begins from the anterior superior iliac spine (i.e., one TAS of the sartorius muscle) and ends at the anteromedial surface of the upper tibia (i.e., the other TAS). In this chapter, the former and the latter mentioned TASs are named TAS1 and TAS2.

4.1.1 State of the Art

Up to now, very few methods have been proposed for muscle segmentation. Wang [102] introduced a method for segmenting the leg muscle which only works with the type of MR T1-weighted data. Furthermore, the segmentation result is a mass of different muscles, and these muscles can not be separated. Seim [81] presented a method for the extraction
4.1 Introduction

Figure 4.1: Illustration of the sartorius muscle [107] and its two tendon attachment sites (TAS1 and TAS2). Two images at right are T2-weighted MR slices of TAS1 and TAS2.

of the ligament attachment site (LAS) from CT images, where the grey value property of the LAS is used as a clue. However there is no such property in MR data. Ng [61] proposed a method using the Gradient Vector Flow (GVF) [110] to segment the muscle from MR T1-weighted data. But for the muscle in MR T2-weighted data, the GVF does not work well. To the best of our knowledge, there is no paper concerning the segmentation of the sartorius muscle or similar muscles.

4.1.2 Overview of Soft Body Organ Segmentation

Similar to the method discussed in the previous chapter, the segmentation method of the soft body organ uses a geometric deformable model, and the segmentation scheme is designed in the global and local model adaption pattern. But the processes in the global and local adaption stages are more complicated than for the rigid body organ because the form of the soft body organ is more dynamic. The main objective of the global adaption stage is to remove the differences of pose, size and location between the geometric model and the organ in the input medical images. Because the form of the soft body organ is flexible, to complete the global adaption, the curve of the organ has to be extracted first. Currently, in this proposed method, the extraction of the curve of the organ is achieved by interpolating a certain number of manually selected center positions of the organ. Information concerning the pose, size and location of the organ can be obtained from the curve. Then the geometric model can be simply aligned with the curve of the organ to finish the task of the global adaption. In the local adaption stage, because each cross section of the organ has a corresponding 2D contour from the 3D geometric model, the 2D contour of the model is deformed based on a certain deformation model. Considering that the soft body organ of the human usually varies in forms within a limited range which can be modeled. Therefore in this method, the ASM is selected as the deformation model to change the geometric model in order to match the organ. From the calculation of the distribution of the landmarks for making the ASM, not only the deformation model but
also the mean geometric shape of the organ can be obtained, and this mean shape of the organ can be used as the geometric model. Therefore, the ASM stands for not only the deformation model but also the geometric model.

As noted before, to present the segmentation method of the soft body organ, an application of this method is discussed, and the task is to segment the muscle and its TASs. The sartorius muscle is chosen as the muscle of interest, and its two TASs are denoted as TAS1 and TAS2. These two TASs are located at the surfaces of different bones. Before the segmentation process starts, the geometric model, which is the 3D ASM of the sartorius muscle, is needed. Furthermore, considering that the shapes of the two bone cross sections containing TAS1 and TAS2 are distinguishable, and the variations of the two shapes can be modeled, so it is possible to detect this two bone cross sections from the input MR data automatically. The advantage is that if this two bone cross sections are detected, the approximate TAS1 and TAS2 can be obtained (both TASs are located at fixed positions on the corresponding bone cross sections) and used as a part of the points for the interpolation of the muscle curve i.e., reducing the human interactions. Moreover, the approximate Vertical Span Length (VSL) of the muscle, which is required by the global adaption, can also be calculated (discussed later). Therefore, besides the 3D ASM of the muscle, two 2D ASMs of the two bone cross sections are created and used to search for the two bone cross sections containing TAS1 and TAS2 from the input MR data. Fig. 4.2 shows a flow chart of the process for muscle segmentation. In the pre-processing stage, the bones are

![Flow chart of the segmentation of the muscle and its TASs.](image)

coarsely segmented using the thresholding and region growing methods. The segmented
bones are put into the global adaption stage where the two bone cross sections containing TAS1 and TAS2 are searched using the two 2D ASMs of the two bone cross sections. To get the muscle curve, a number of points locating at the center of some cross sections of the sartorius muscle are set manually in the MR data. Based on these points and the approximate TAS1 and TAS2 obtained from the detected two bone cross sections, the muscle curve can be interpolated. After that the 3D ASM of the sartorius muscle can be aligned to the interpolated curve. In the local adaption stage, for each MR slice containing a cross section of the sartorius muscle, there is a corresponding 2D ASM of the muscle contour from the 3D ASM of the sartorius muscle. According to the deformation method defined by the ASM, each 2D ASM muscle contour is adapted to the corresponding cross section of the sartorius muscle in the MR slice to complete the segmentation. Accurate TAS1 and TAS2 are obtained by checking the relations between the ends of the segmented muscle and the bones (from the pre-processing stage). Details will be presented in the following sections. The MR data used in this application is composed of axial T2-weighted slices from the pelvis to the foot with the voxel size of $0.39 \times 0.39 \times 3.0 \, \text{mm}$.

The calculation of the ASMs of the 3D sartorius muscle and the two 2D bone cross sections is presented in Section 4.2. Because the pre-processing stage contains only the thresholding and region growing methods, which are the same as those discussed in the previous chapter, the description of the pre-processing stage will not be given. The global and local adaption of the ASM of the muscle are discussed in Section 4.3 and 4.4 respectively. In Section 4.5 the strategy of the extraction of TAS1 and TAS2 is described. The evaluation is given in Section 4.6.

### 4.2 Calculation of Active Shape Models of Bones and Sartorius Muscle

The calculation of the ASMs of the sartorius muscle and the two bone cross sections is based on [26]. In brief, the ASM is calculated by analyzing the landmarks of the input training instances from the population of the object of interest, and the ASM is composed of a mean geometric shape of the training instances and the distribution of their landmarks. The calculation of the ASM contains three steps:

1. Landmarks selection for each training instance.
2. Alignment of all training instances.
3. Computing the distributions of the landmarks of the training instances using Principal Components Analysis (PCA) [87].

The training instances, which are used to calculate the required ASMs, are 8 MR data sets of 8 volunteers aged 20-70, 4 males and 4 females.

#### 4.2.1 Creation of two 2D ASM of Bones Cross Sections

Before discussing the calculation of the two 2D ASMs of the two bone cross sections, examples of MR slices which contain TAS1 and TAS2 are presented in Fig. 4.3. TAS1 located at the part of the pelvis is a small region and can be regarded as a point. TAS2
Figure 4.3: Examples of the varieties of the bone cross sections containing TAS1 or TAS2. The images at the top are four varieties of the bone cross section where TAS1 is located. The images at the bottom are four varieties of the bone cross section where TAS2 is located.

located at the upper tibia is a big region and considering the feasibility of this method, TAS2 is defined as any point that lies within the region. From Fig. 4.3, it is clear that the bone cross sections containing TAS1 or TAS2 have a high variability. To calculate the ASMs of the two bone cross sections, the landmarks of the training instances of the two bone cross sections are required. Usually, the landmarks should be set at some salient positions of the object, e.g., at the corners or tips of the object, and they can be found in all instances of the object. But the problem is that there are no such positions in both the bone cross sections, therefore the landmarks are set in a special manner described below.

Bone Cross Section Containing TAS1

For setting the landmarks of the bone cross section containing TAS1, the top and the bottom points of the bone cross section are required. Then the vertical distance between the two points is equally divided into 15 intervals by 14 parallel lines (Fig. 4.4 left). These parallel lines intersect the boundary of the bone cross section, and the landmarks are set at these intersections manually. It means that each instance of the TAS1 bone cross section is composed of 30 landmarks (28 intersections plus the top and the bottom points).

To compute the ASM of the bone cross section, all the training instances, each of which is represented by 30 landmarks, must be aligned. The alignment is achieved by translating every instance to make its center, which is the average position of the instance landmarks, be at a given point. Next, all the instances are represented as vectors and put into a high dimensional space. As mentioned above, each training instance contains 30 landmarks and each landmark is defined by a 2D coordinate \((x_i, y_i), i = 0 \ldots 29\). Therefore each instance can be represented as a 60 dimensional vector which is structured as \((x_0, y_0, x_1, y_1, \ldots, x_{29}, y_{29})\), and all the instances are put into a 60-dimensional space. Then by means of PCA, the distribution of these vectors, which are expressed as the eigenvectors of the covariance matrix of all the instances, can be obtained. Theoretically, 60 eigenvectors in all can be calculated by PCA, but in this method, the first three eigenvectors whose
4.2 Calculation of Active Shape Models of Bones and Sartorius Muscle

Figure 4.4: Strategies of setting the landmarks for the bone cross sections containing TAS1 (left) and TAS2 (right) are illustrated. Left: the 14 parallel lines divide the bone cross section into 15 equal vertical sections. The landmarks include the 28 intersections between the 14 lines and the bone boundary plus the top and the bottom points of the bone. Right: the 30 rays are emitted from the manually selected point, and the landmarks are the intersections between the rays and the bone boundary.

corresponding eigenvalues are the biggest are preserved. Deformations of the ASM of the bone cross section are produced by changing the mean shape of the bone cross section in the direction of these three eigenvectors. Fig. 4.5 shows the deformations of the ASM of the bone cross section containing TAS1. The deformations at each row are obtained by

Figure 4.5: Deformations of the ASM of the bone cross section containing TAS1. The deformations at the top row are obtained by changing the mean shape only in the direction of the eigenvector with the maximal eigenvalue. The middle and the bottom rows show the deformations in the directions of the eigenvectors with the next two largest eigenvalues, separately.
changing one of these three main eigenvectors. The middle shape at each row is the mean shape of the bone cross section.

**Bone Cross Section Containing TAS2**

Considering the shape of the bone cross section containing TAS2, which is an ellipse-like region, the landmarks are determined by manually setting the center of the bone in each training instance, then 30 rays, which divide the circle into 30 equal regions (the angle between each neighbor rays is $\pi/15$), are emitted from the center. The landmarks are the intersections between each ray and the bone boundary (Fig. 4.4 right). The calculation of the ASM of the bone containing TAS2 is the same as that of the ASM of the bone cross section containing TAS1.

### 4.2.2 Creation of 3D ASM of Muscle

The creation of the 3D ASM of the sartorius muscle is almost the same as the two 2D ASMs of the bone cross sections containing TAS1 and TAS2. The only difference is that the number of MR slices containing the sartorius muscle is often more than 100, thereby tracing the landmarks of the muscle in each one of the 100 slices is a highly labor intensive. In this method, for the construction of each 3D muscle instance, 20-30 slices are selected from the total muscle slices of an instance, and 30 landmarks are set manually in each selected slice in the same manner as for the bone cross section containing TAS2. The landmarks of the rest slices are produced by interpolating the landmarks of the selected slices. Fig. 4.6 left shows an 3D instance of the sartorius muscle. First of all, all the 3D muscle instances are aligned by setting one instance as a reference and the other instances are scaled to make their VSLs the same as that of the reference. Then, the beginning and end of the rest 3D muscle instances are translated to the corresponding positions of the reference. The alignment result is shown in Fig. 4.6 right.

![Figure 4.6: Left: a 3D instance of the sartorius muscle. Right: the alignment result of all the 3D sartorius muscle instances (every instance is marked in a distinct color).](image-url)
Strictly speaking, the 3D ASM of the muscle in this method can not be called a 3D ASM, because the 3D ASM of the muscle is not calculated and used to do the muscle adaption in 3D. The reason is that the landmarks of each 3D muscle instance are from more than 100 MR slices, and each slice contributes 30 landmarks. Therefore, each 3D muscle instance, which is used in PCA to obtain the eigenvectors, is expressed as at least a $100 \times 30 \times 2 = 6000$ dimensional vector. It would be inefficient to use the 6000 dimensional eigenvectors to deform the 3D ASM of the muscle to match the muscle in the MR data. Therefore instead of computing the 3D ASM of the muscle, a series of 2D ASMs of the muscle cross sections is used to do the muscle adaption. Each 2D ASM of the muscle cross section is calculated by doing PCA on the corresponding 2D muscle cross sections from all the 3D muscle instances. So the 3D ASM of the muscle here contains only the 3D mean geometric shape of the muscle. Details will be discussed in the local adaption stage.

After the calculation of the ASMs of the two bone cross sections and the muscle, the next step is to adapt the 3D ASM of the muscle to the muscle in the MR data, globally and locally.

### 4.3 Global Adaption of the 3D ASM of Muscle

Generally speaking, the objective of the global adaption is to make the 3D ASM of the muscle have the same length, pose and position as the muscle in the MR data. In this method, equal muscle length is achieved by scaling the VSL of the ASM of the muscle to be the same as the VSL of the muscle in the MR data. Identical pose and position are attained by aligning the 3D ASM of the muscle according to the curve of the muscle in the MR data. The processing sequence in the global adaption is first the VSL scaling and then the alignment of the muscle curve.

#### 4.3.1 Vertical Span Length (VSL)

Since the form of the sartorius muscle is neither a straight line nor a simple curve, its length is difficult to measure accurately. But its VSL can be easily obtained by getting the index numbers of two MR slices which have the two bone cross sections containing TAS1 and TAS2. Based on these two index numbers, the total number of the MR slices containing the sartorius muscle can be obtained. Because the thickness of the MR slice is included in the input MR data, so the VSL of the sartorius muscle is equal to the number of the MR slices multiplied by the slice thickness (in mm).

Therefore, the method of getting the muscle VSL is to search for the two MR slices containing the bone cross sections in which TAS1 and TAS2 are located. The ASMs of the two bone cross sections are available, and both can be used to search for the two corresponding bone cross sections in the MR data. The search strategy is that bones are first segmented from the MR data using the thresholding and region growing methods. The process of thresholding and region growing has already been discussed in Section 2.1 and 3.2.2, so there will be no further description here. The segmented bones are stored in the form of binary images where bones are black and others are white. The segmented bone cross section in each binary image is compared with both the 2D ASMs of the bone cross sections via the contour scanning code method. The search results of the two bone cross sections containing TAS1 and TAS2 are determined by choosing the bone cross section.
whose contour is most similar to one of the 2D ASMs.

**Contour Scanning Code**

Since each cross section of the segmented bone has to be compared with the two 2D ASMs of the bone cross sections to measure the similarity, and the number of images containing the segmented bones (from pelvis to foot) is more than 300, the workload for the comparison is huge. To make the comparison automatic and efficient, a contour scanning code algorithm is proposed. For each cross section of the segmented bones, a contour scanning code is given based on the following steps.

1. Scale the bone cross section region to the length of $n$ pixels in the vertical direction.
2. Record the width (in pixels) of the scaled region at each height level from the top to the bottom of the region, pixel by pixel (Fig. 4.7 left).
3. Scale the original size bone cross section region to the length of $n$ pixels in the horizontal direction and record the height as in step 2 but from the leftmost to the rightmost of the region (Fig. 4.7 right).

![Figure 4.7](image)

**Figure 4.7**: Left: the vertical scaled object and $w_i$ is the width at the $i$-th level that started from the top object. Right: the horizontal scaled object and $h_i$ is the $i$-th line height that started from the leftmost of the scaled object.

Therefore, the contour scanning code of a bone cross section is a $2n$ element array and is structured as $(w_0, \ldots, w_{n-1}, h_0, \ldots, h_{n-1})$. The contour dissimilarity between each bone cross section and the two 2D ASMs is quantified by the difference of their contour scanning codes:

$$s = \sum_{i=0}^{2n-1} \frac{|C_{r,i} - C_{m,i}|}{C_{m,i}},$$

where $C_{r,i}$ and $C_{m,i}$ are the $i$-th elements of the contour scanning codes of a segmented bone cross section and one of the two 2D ASMs respectively.

Considering that the bone cross section containing TAS1 or TAS2 varies between individuals (as shown in Fig. 4.3). Comparing each segmented bone cross section with only one deformation of each 2D ASM of the bone cross section makes no sense. That’s why in this method, a set of deformations of each 2D ASM of the bone cross section is used for the contour comparison. By investigating many examples of the bone cross sections
containing TAS1 and TAS2, the set of the deformations of each 2D ASM of the bone cross section contains 5 typical deformations which are produced by changing the eigenvectors of each 2D ASM. Fig. 4.8 shows the set of deformations of the 2D ASM of the bone cross section containing TAS1.

**Figure 4.8:** 5 deformations of the bone cross section containing TAS1.

Because each segmented bone cross section has to be compared with 5 deformations of each 2D ASM of the bone cross section, so each segmented bone cross section has 5 values indicating the difference of the contour scanning code \( S = (s_1, \ldots, s_5) \) for each 2D ASM of the bone cross section. In this case, the minimum value \( (s_{\text{min}} | s_{\text{min}} \leq s_i, i = 1 \ldots 5, s_{\text{min}}, s_i \in S) \) is regarded as the final contour difference between the segmented bone cross section and the corresponding 2D ASM of the bone cross section. Two images each of which contains the segmented bone cross section with the smallest contour difference to one of the two 2D ASMs of the bone cross sections are regarded as the final search results. The time of the comparison of 300 images is around 30 seconds (P4 3.0GHz processor, 1G memory, WinXP). The whole process of the VSL calculation is illustrated in Fig. 4.9.

**Figure 4.9:** The VSL calculation. The inputs are MR data (left). Then bones are segmented using thresholding and region growing (middle), and the segmentation results are represented as binary images. By using the contour scanning code, two images each of which contains the most similar bone cross section to one of the two ASMs are found (right), and the muscle VSL can be calculated using the index numbers of these two images.
4.3.2 Muscle Alignment

Muscle alignment is the last step in the global adaption stage where the 3D ASM of the muscle is attached to the curve of the muscle in the MR data. In order to get the curve of the sartorius muscle in the MR data, the centers of the sartorius muscle sections are selected manually in the MR data. These centers are needed to make a complete muscle curve to which the 3D ASM of the muscle is attached. In this method, the centers of all sartorius muscle sections are not required, and only a part of the muscle centers are needed. Then with the help of the interpolation method or the 3D ASM of the muscle to reconstruct the muscle curve. There are two feasible methods using the part of the muscle section centers to make the muscle curve.

The idea of the first method is to translate the corresponding section center of the ASM of the muscle to each selected muscle section center in the MR data. The curve between two selected centers is defined by the curve of the ASM of the muscle. Assuming that \((x_a, y_a, z_a)\) and \((x_b, y_b, z_b)\) are two selected centers contained in two slices of the MR data, \((x_{am}, y_{am}, z_a)\) and \((x_{bm}, y_{bm}, z_b)\) are the corresponding section centers of the ASM of the muscle. \((x_{rm}, y_{rm}, z_r)\) represents the section centers of the ASM of the muscle in the interval of \([z_a, z_b]\) i.e., \(z_a \leq z_r \leq z_b\). Then the curve between the \((x_a, y_a, z_a)\) and \((x_b, y_b, z_b)\) is defined by

\[
\begin{align*}
x & = x_{rm} + x_a - x_{am} + \frac{z - z_a}{z_b - z_a} \cdot (x_b - x_{bm} - x_a + x_{am}) \\
y & = y_{rm} + y_a - y_{am} + \frac{z - z_a}{z_b - z_a} \cdot (y_b - y_{bm} - y_a + y_{am}) \\
z & = z_r,
\end{align*}
\]

where \((x, y, z)\) represents any point located on the muscle curve between the interval \([z_a, z_b]\). This makes the ASM of the muscle keep its original curve between each two selected muscle centers in the MR data.

The idea of the second method is to interpolate these selected muscle centers via the cubic spline where an elaborate scheme for the selection of the muscle centers in the MR data is required in order to obtain a satisfactory interpolation of the muscle curve. In this chapter, both methods are tested and compared with each other. Details can be found in Section 4.6.2.

4.4 Local Adaption of 3D ASM of Muscle

After the global adaption, each MR slice containing a cross section of the sartorius muscle has a corresponding 2D contour from the 3D ASM of the muscle. Because the global adaption stage focuses on the differences of pose, size and location, there still exist differences in the muscle contour between the muscle in the MR data and the ASM of the muscle. Therefore, a local adaption is needed where the contour of the 3D ASM of the muscle is deformed to match the muscle boundaries in the MR data.

As mentioned in Section 4.2.2, the deformation of the 3D ASM of the muscle in the local adaption stage does not use the at least 6000-dimensional eigenvectors because it would be inefficient. In this method, the 3D muscle adaption is converted into a series of 2D muscle contour adaption. For each MR slice containing a muscle cross section, a corresponding
2D ASM of the muscle cross section is calculated from the 3D ASM of the muscle. The 2D ASM of the muscle cross section is adapted to the muscle section in the MR slice by affine transformation (rotation, scaling and translation) and non-rigid adaption (deformed by three main eigenvectors of the 2D ASM of the muscle section). The matching quality is indicated by the number of image edges (produced by the Canny algorithm [13]) located on the contour of the 2D ASM of the muscle cross section. For each MR slice, the deformation of the 2D ASM of the muscle section, which has the maximum number of image edges, is regarded as the final adaption result i.e., the muscle segmentation result. Fig. 4.10 shows some examples of the adaption results. The white contours are the segmentation results of the method and the black contours are from manual segmentation. Further segmentation results and evaluations are discussed in Section 4.6.

4.5 Tendon Attachment Sites (TASs) Extraction

Compared with the muscle segmentation, the TAS extraction is relatively difficult to achieve because there are no distinct features of the TAS in the MR data. To get an accurate TAS, the segmentation result of the muscle is used. Because the two terminal vertices of the segmented muscle are attached on the bone surfaces which are the TASs. However, due to the inaccurate muscle VSL calculation (Section 4.3.1), the VSL of the segmented muscle may be shorter or longer than that of the muscle in the MR data. Through the experiments of this method, the calculated VSL was seen to have $0 - 12$ mm deviation from the true VSL. Therefore, there are three possible cases of the segmented muscle: (1) shorter, (2) equal or (3) longer compared to the true muscle. Fortunately these three cases can be detected automatically from the segmentation result of the muscle.

1. If the segmented muscle is shorter than the true length, the end of the segmented muscle will not attach to a bone.

2. For the case 2 which is the optimal result, the end will be at the bone surface.

3. If the segmented muscle is longer than the true length, the muscle will reach to the bone before its ends.

Fig. 4.11 illustrates strategies for case 1 (left) and 3 (right). In case 1, the end of the segmented muscle has a small distance to the true TAS. Suppose that $p_1$ and $p_2$ are center...
Figure 4.11: The red curves stand for the segmented muscles and the blue circles indicate the correct TAS. Left: the calculated VSL is shorter than the true one. $p_1$ and $p_2$ are the two consecutive center positions at the end of the segmented muscle. Right: the calculated VSL is longer than the true one and the segmented muscle has already attached to the bone before reaching the muscle end.

positions at the end and near the end of the segmented muscle. The true TAS is obtained by moving $p_1$ in the direction of the vector $p_1 - p_2$ until reaching the bone surface. In case 3, the position of the segmented muscle near its end has already reached the bone surface and this position will be regarded as the correct location of the TAS. Examples of the extracted TASs are shown in Fig. 4.12. There are four pairs of TAS1 and TAS2 with each pair placed in one column.

Figure 4.12: TAS results. The top row shows the results of TAS1 and the bottom row contains the results of TAS2. The green points indicate the corresponding TAS positions and the red contours are the optimal deformations of the 2D ASMs of the two bone cross sections containing TAS1 and TAS2.
4.6 Evaluation

The evaluation of the segmentation result of the sartorius muscle and its TASs is done by using the MR T2-weighted data of 7 patients. The global adaption results, which contain the muscle VSL calculation and the alignment of the 3D ASM of the muscle, are inspected first. After that is the evaluation of the local adaption which produces the final segmentation results of the muscle. At last, the extracted TASs are compared with the manually selected positions to measure the disparity.

4.6.1 Evaluation of Muscle VSL

Evaluation of the muscle VSL focuses on the deviation between the searched indexes and the true indexes (manually selected) of the images containing the bone cross sections of TAS1 and TAS2. The evaluation results are shown in Table 4.1. $d_{\text{TAS1}}$ and $d_{\text{TAS2}}$ stand for the deviations of the indexes of the images containing TAS1 and TAS2 respectively. Since the images from the beginning of the pelvis to the end of the foot are indexed in ascending order, a positive value of $d_{\text{TAS1}}$ and $d_{\text{TAS2}}$ means the searched image is below the true one while a negative value means it is above the true one. Since the thickness of the MR slice of the input MR data is 3.0 mm, the deviation of the VSL $d_{\text{VSL}}$ is calculated by $(d_{\text{TAS2}} - d_{\text{TAS1}}) \times 3.0$ with the sign which indicates whether it is shorter (negative) or longer (positive) than the true VSL. $r_{\text{VSL}}$ is the relative error. The mean values of $d_{\text{TAS1}}$, $d_{\text{TAS2}}$, $d_{\text{VSL}}$ and $r_{\text{VSL}}$ are computed by averaging the absolute values of the corresponding item. The calculated VSL of the sartorius muscle has an average disparity of 7.29 mm

<table>
<thead>
<tr>
<th>Patient ID</th>
<th>$d_{\text{TAS1}}$</th>
<th>$d_{\text{TAS2}}$</th>
<th>$d_{\text{VSL}}$</th>
<th>$r_{\text{VSL}} \times 100%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>+1</td>
<td>-2</td>
<td>-9.0</td>
<td>1.9</td>
</tr>
<tr>
<td>2</td>
<td>+3</td>
<td>-1</td>
<td>-12.0</td>
<td>2.2</td>
</tr>
<tr>
<td>3</td>
<td>+2</td>
<td>-1</td>
<td>-9.0</td>
<td>1.7</td>
</tr>
<tr>
<td>4</td>
<td>-1</td>
<td>-3</td>
<td>-6.0</td>
<td>1.0</td>
</tr>
<tr>
<td>5</td>
<td>+2</td>
<td>+1</td>
<td>-3.0</td>
<td>0.5</td>
</tr>
<tr>
<td>6</td>
<td>-2</td>
<td>-2</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>7</td>
<td>+1</td>
<td>-3</td>
<td>-12.0</td>
<td>2.2</td>
</tr>
<tr>
<td>Mean</td>
<td>1.71</td>
<td>1.86</td>
<td>7.29</td>
<td>1.4</td>
</tr>
</tbody>
</table>

with an relative error of 1.4%. The main origin of the VSL deviation is the similarity of bone cross sections between neighboring images (see Fig. 4.13).
4.6.2 Evaluation of Alignment of the ASM of Muscle

The global alignment is achieved by manually selecting a certain number of muscle section centers in the MR data and using them to make a complete muscle curve. Based on the muscle curve, the 3D ASM of the muscle is aligned with the muscle in the MR data. As mentioned in Section 4.3.2, two methods are available to produce the muscle curve.

1. Keeping the curve of the ASM of the muscle at each interval of two selected centers.

2. Interpolating the muscle curve from the selected centers by means of the cubic spline.

For both methods, the biggest issue is the selection scheme of the muscle section centers. In order to obtain a good muscle curve, the centers have to be set according to the complexity of the muscle curve. More centers should be set at complex curve parts while fewer centers are needed at simple curve parts. Fig. 4.14 shows a result of the muscle curve by using the curve information of the 3D ASM of the muscle for a short part of the sartorius muscle (35 slices, VSL = 105 mm). The muscle is represented as a white strip.

Figure 4.14: The muscle curve result using the curve of the ASM of the sartorius muscle (front and side views). The result of the muscle curve is shown as a gray line and the part of the sartorius muscle is displayed as a white stripe region. Because of the curve of the ASM of the muscle, the curve result is not smooth.
region. Two centers at the top and the bottom slices of the part of the sartorius muscle are selected, and the result is shown as a gray curve. In the experiment, the muscle VSL of each test MR data is in the range of 483-555 mm and for each test MR data, about 8 centers are needed for making a complete muscle curve.

Fig. 4.15 shows an example of the cubic spline interpolation for the same muscle part as used in the first method. The centers are selected from 3 slices which are at the top, middle, and bottom of the muscle part. The gray curve indicates the muscle curve result. The curve is smooth and totally within the muscle. In the experiment, for the production of the complete muscle curve, the minimum number of the centers required in the cubic spline method should be more than 20. Otherwise, the curve result is not satisfactory.

Comparing the results in Fig. 4.14 and Fig. 4.15, it is clear that the muscle curve result from the cubic spline method is smoother than that from the method using the curve of the ASM of the muscle (Fig. 4.16). But the former requires more manually selected centers (more than 20) than the latter (about 8). The main reason is that the cubic spline method does not take advantage of the curve information from the muscle model, while the information is fully utilized by the other method. In the experiments both methods are tested and compared by checking the disparity which is the distance (in millimeters) between the position of the curve result on each MR slice and the corresponding manually
selected center. Furthermore, to comprehensively compare these two methods, the number of manually selected centers for curve reconstruction is also considered. Fig. 4.17 and Fig. 4.18 represent the comparison result of the disparity and the number of selected centers between both methods in the form of a bar chart respectively. Clearly, in Fig. 4.17 the disparity resulting from using the cubic spline is much smaller than that from using the curve of the ASM of the muscle. However, the number of manually selected centers for the cubic spline is bigger which means more human interactions are required (Fig. 4.18). Details of the evaluation of both muscle curve reconstruction methods are shown in Table 4.2. Considering that the disparity of using the ASM curve is acceptable and can be corrected in the local adaption stage, the ASM curve method is therefore preferred.

After the global adaption of the ASM of the muscle, each contour of the muscle cross section in the MR data has a corresponding 2D contour from the 3D ASM of the muscle.
Table 4.2: Disparity between each muscle center by using cubic spline/ASM curve and manually selected center positions. $d_c$ and $d_m$ are the average values of the disparity (in millimeters) of using cubic spline and ASM curve of each test data set (1-7). $n_c$ and $n_m$ are the number of manually selected centers for the cubic spline and ASM curve.

<table>
<thead>
<tr>
<th>Patient ID</th>
<th>$d_c$</th>
<th>$d_m$</th>
<th>$n_c$</th>
<th>$n_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>1.2</td>
<td>3.8</td>
<td>20</td>
<td>5</td>
</tr>
<tr>
<td>02</td>
<td>1.5</td>
<td>3.9</td>
<td>25</td>
<td>8</td>
</tr>
<tr>
<td>03</td>
<td>1.2</td>
<td>5.0</td>
<td>23</td>
<td>6</td>
</tr>
<tr>
<td>04</td>
<td>1.3</td>
<td>3.5</td>
<td>26</td>
<td>8</td>
</tr>
<tr>
<td>05</td>
<td>1.2</td>
<td>3.9</td>
<td>26</td>
<td>8</td>
</tr>
<tr>
<td>06</td>
<td>1.1</td>
<td>4.5</td>
<td>25</td>
<td>6</td>
</tr>
<tr>
<td>07</td>
<td>1.5</td>
<td>4.5</td>
<td>25</td>
<td>6</td>
</tr>
<tr>
<td>Mean</td>
<td>1.3</td>
<td>4.2</td>
<td>24</td>
<td>7</td>
</tr>
</tbody>
</table>

But there is a small difference between these two contours, and the small difference is presented in Table 4.3 where the overlap ratios between the contour of the muscle section in each MR slice and the corresponding contour from the 3D ASM of the muscle are shown. The overlap ratios are calculated using the Jaccard index.

Table 4.3: Evaluation of the overlap ratio between the ASM contour after the global adaption and the muscle section contour in each MR slice. $J_{max}$, $J_{min}$ and $J_{mean}$ are the maximum, the minimum and the average values of the Jaccard index in each test MR data (1 – 7).

<table>
<thead>
<tr>
<th>Patient ID</th>
<th>$J_{max}$</th>
<th>$J_{min}$</th>
<th>$J_{mean}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>0.68</td>
<td>0.31</td>
<td>0.53</td>
</tr>
<tr>
<td>02</td>
<td>0.66</td>
<td>0.30</td>
<td>0.50</td>
</tr>
<tr>
<td>03</td>
<td>0.61</td>
<td>0.26</td>
<td>0.41</td>
</tr>
<tr>
<td>04</td>
<td>0.73</td>
<td>0.41</td>
<td>0.60</td>
</tr>
<tr>
<td>05</td>
<td>0.68</td>
<td>0.38</td>
<td>0.58</td>
</tr>
<tr>
<td>06</td>
<td>0.63</td>
<td>0.30</td>
<td>0.49</td>
</tr>
<tr>
<td>07</td>
<td>0.61</td>
<td>0.28</td>
<td>0.48</td>
</tr>
<tr>
<td>Mean</td>
<td>0.66</td>
<td>0.32</td>
<td>0.51</td>
</tr>
</tbody>
</table>
4.6.3 Evaluation of Local Adaption

In the local adaption stage, each 2D ASM of the muscle cross section is derived from the 3D ASM of the muscle and is deformed affinely and non-rigidly in order to correct the contour difference. The affine transformation involves scaling, translation, and rotation. Then in the non-rigid adaption, each 2D ASM of the contour of the muscle cross section is changed by the three main eigenvectors (obtained by PCA) which indicate the main changing directions of each muscle cross section.

Evaluation of Affine Transformation

The exhaustive searching strategy is used in the affine transformation where rotation, scaling, and translation are performed in predefined ranges which are $[0.9, 1.1]$ for scaling (0.1 step size), $[-5, 5]$ pixels in the $X$ and $Y$ directions for translation (1 pixel step size), and $[-10, 10]$ degrees for rotation (5 degrees step size). Table 4.4 shows the evaluation of the overlap ratio (Jaccard index) after the affine transformation. Obviously, the overlap ratio is highly improved as compared to the result after the global adaption. However, the contour difference still cannot be eliminated completely. So to solve this problem, in the next step, the non-rigid adaption is carried out.

**Table 4.4:** Evaluation of the overlap ratio of the 2D ASM of the muscle cross section to the corresponding muscle cross section in the MR data after the local affine transformation.

<table>
<thead>
<tr>
<th>Patient ID</th>
<th>$J_{\text{max}}$</th>
<th>$J_{\text{min}}$</th>
<th>$J_{\text{mean}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>0.88</td>
<td>0.40</td>
<td>0.70</td>
</tr>
<tr>
<td>02</td>
<td>0.87</td>
<td>0.47</td>
<td>0.69</td>
</tr>
<tr>
<td>03</td>
<td>0.85</td>
<td>0.50</td>
<td>0.70</td>
</tr>
<tr>
<td>04</td>
<td>0.88</td>
<td>0.60</td>
<td>0.78</td>
</tr>
<tr>
<td>05</td>
<td>0.85</td>
<td>0.48</td>
<td>0.73</td>
</tr>
<tr>
<td>06</td>
<td>0.83</td>
<td>0.48</td>
<td>0.72</td>
</tr>
<tr>
<td>07</td>
<td>0.81</td>
<td>0.46</td>
<td>0.70</td>
</tr>
<tr>
<td>Mean</td>
<td>0.85</td>
<td>0.48</td>
<td>0.72</td>
</tr>
</tbody>
</table>

Evaluation of Non-Rigid Adaption

The changing of the 2D ASM of each muscle cross section in the non-rigid adaption is based on

$$x = x + b_1p_1 + b_2p_2 + b_3p_3.$$  \hspace{1cm} (4.5)

where $x$ is the mean geometric shape of the 2D ASM of the muscle cross section, $p_1$, $p_2$ and $p_3$ are three main eigenvectors. To achieve the global optimization, the exhaustive...
searching method is applied. In order to make the process more efficient, ranges of $b_1$, $b_2$ and $b_3$, each of which defines the changing magnitude along each eigenvector, are set based on the corresponding eigenvalues. The higher the eigenvalue, the more important is its corresponding eigenvector. In the experiment, $b_1$, $b_2$ and $b_3$ are set to $[-10, 10]$, $[-10, 10]$ and $[-5, 5]$ respectively, and the step size in the exhaustive searching is 1.

Table 4.5 shows the evaluation of the overlap ratio (Jaccard index) after the non-rigid adaption. The overlap ratio as compared to the result of the local affine transformation is not so significantly improved. The reason is that the objective of the non-rigid adaption is to solve the local contour differences which occupy a small proportion of the whole region of the muscle cross section. Fig. 4.19 shows a summary of the mean overlap ratios after the global and local (containing affine and non-rigid) adaption stages of each test MR data.

**Table 4.5:** Evaluation of the overlap ratio of the 2D ASM of the muscle cross section to the muscle cross section in the MR data after the local non-rigid adaption.

<table>
<thead>
<tr>
<th>Patient ID</th>
<th>$J_{max}$</th>
<th>$J_{min}$</th>
<th>$J_{mean}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>0.90</td>
<td>0.55</td>
<td>0.78</td>
</tr>
<tr>
<td>02</td>
<td>0.92</td>
<td>0.55</td>
<td>0.81</td>
</tr>
<tr>
<td>03</td>
<td>0.90</td>
<td>0.61</td>
<td>0.83</td>
</tr>
<tr>
<td>04</td>
<td>0.93</td>
<td>0.68</td>
<td>0.86</td>
</tr>
<tr>
<td>05</td>
<td>0.92</td>
<td>0.50</td>
<td>0.78</td>
</tr>
<tr>
<td>06</td>
<td>0.88</td>
<td>0.55</td>
<td>0.79</td>
</tr>
<tr>
<td>07</td>
<td>0.87</td>
<td>0.55</td>
<td>0.79</td>
</tr>
</tbody>
</table>

| Mean       | 0.90      | 0.57      | 0.81       |

**Figure 4.19:** Improvement of the segmented muscle results stage by stage.
to give an impression of the improvement of segmentation results of the muscle stage by stage. Examples of the final results of the muscle segmentation which contain the muscle cross sections from the beginning to the end of the sartorius muscle are shown in Fig. 4.20. The images are arranged in the order of left to right and top to bottom. The red curves indicate the final adaption results of the 2D ASMs of the muscle cross sections and the red points are the centers of the muscle cross sections. Ideally, the red points in the images at the top left and bottom right are the beginning and end of the sartorius muscle i.e., TAS1 and TAS2. From Section 4.3.1 it is known that these two images are searched by comparing the contours of the segmented bone cross sections in each image to the two 2D ASMs of the bone cross sections. Because of the high similarity of the bone cross sections contained in the neighbor images, these two images may not contain the true TAS1 and TAS2. Therefore, the TAS position extraction strategy which is described in Section 4.5 is needed.

Figure 4.20: Example of the final muscle adaption result which contain the muscle distributed from the beginning to the end in the order of left to right and top to bottom. Red curves and points indicate the model contours and centers.
4.6.4 Evaluation of TAS Extraction

The evaluation of the extraction of TAS1 is shown in Table 4.6. Since TAS1, which is at the anterior superior iliac spine, is a small region, it can be regarded as a point. Therefore the disparity is the distance between the extracted TAS1 and the manually selected position in the 3D coordinate system of the MR data (units in millimeters). For the large region of TAS2, which is at the anteromedial surface of the upper tibia, we can only check whether the extracted position lies within the TAS2 region. In the experiment, all the extracted TAS2 are within the manually segmented TAS2 regions.

Table 4.6: Evaluation of TAS1 extraction. $d_i (i = 1, 2, \ldots, 7)$ represents the disparity between TAS1 of the $i$-th patient and corresponding manually selected TAS1. $d_{\text{mean}}$ is the mean disparity. The units are millimeters.

<table>
<thead>
<tr>
<th>Dev.</th>
<th>$d_1$</th>
<th>$d_2$</th>
<th>$d_3$</th>
<th>$d_4$</th>
<th>$d_5$</th>
<th>$d_6$</th>
<th>$d_7$</th>
<th>$d_{\text{mean}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAS1</td>
<td>3.4</td>
<td>4.2</td>
<td>3.18</td>
<td>3.24</td>
<td>6.46</td>
<td>7.8</td>
<td>4.6</td>
<td>4.7</td>
</tr>
</tbody>
</table>

4.7 Conclusions

A segmentation method for the soft body organ and its application have been presented in this chapter. The method uses a geometric model which is the mean geometric shape of the organ of interest produced by the ASM. The segmentation process is designed in the global and local model adaption pattern. Because the form of the soft body organ is dynamic, the global adaption method for removing the differences of pose, length and position used for the rigid organ does not work. For example, the pose of the rigid organ is determined by calculating its body axis which is infeasible for the soft body organ. Therefore instead, the curve of the soft body organ is extracted in the global adaption stage where a set of manually selected positions located at the centers of some cross sections of the organ of interest is required and used to interpolate the complete organ curve. Then in the local adaption, each cross section of the geometric model adapts the corresponding cross section of the organ of interest using the affine transformation and the eigenvectors included in the ASM.

The task of the application is to segment muscles and their corresponding TASs. An example muscle of interest called the sartorius muscle, which is the longest muscle of the human body, was focused. According to the anatomic features of the sartorius muscle, in the global adaption stage, the begin and end positions of the sartorius muscle can be extracted automatically using two 2D ASMs of the two bone cross sections each of which contains either TAS1 or TAS2. In addition, an efficient contour comparison algorithm called the contour scanning code was proposed for the process of searching the images containing the bone cross sections on which TAS1 and TAS2 are located. Comparing the results with manual segmentation by an expert, the mean deviation of the extraction results of TAS1 was 4.7 mm and the accuracy of segmentation results of muscle was 0.81 (Jaccard Index). The evaluations were made by using 7 sets of the MR T2-weighted data.
In fact besides the T2-weighted data, this method also accepts the MR T1-weighted data as input.

An alternative segmentation method of the muscle is to use the classification algorithm e.g., the Support Vector Machine (SVM) which was introduced in Chapter 2. In this research, segmentation of the muscle using the SVM was evaluated to make a comparison with the proposed method. A library called LIBSVM \cite{15} was applied and three classes: bone, fat and muscle were defined. Each training vector contains features of mean, deviation, entropy, and co-occurrence matrix (for texture) of manually selected image patches of each class. Examples of the results are shown in Fig. 4.21 where the three kinds of organ are classified and marked in three different colors. The background is detected using the region growing method and marked in red. In the experiment of the SVM, the classification accuracy, which depends largely on the quality of the training set and applied features, is high. However, the low computational efficiency, which takes more than 10 minutes to finish the classification of one image (768×564 pixels), makes the SVM algorithm inconvenient for medical image segmentation. Furthermore, muscles can not be individually classified by the SVM.

In this and the previous chapters, two geometric deformable model based segmentation methods, which work in the global and local model adaption pattern, were described. They focused on the segmentation of the rigid body organ (e.g., the femur) and the soft body organ (e.g., the muscle), respectively. To further discover the capacity of the geometric deformable model based method designed in the global and local model adaption pattern, segmentation of a new kind of organ which has the property of both rigid and soft body organs will be presented in the next chapter.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{image.png}
\caption{Examples of the results using the SVM. Three classes are used: the bone (green), the fat (yellow), and the muscle (blue). The background is marked in red.}
\end{figure}
5 Segmentation of Articulated Body Organ: Spine

In this chapter, a segmentation method for the articulated body organ using geometric deformable models is proposed. The articulated body organ is composed of both rigid and soft tissues which make this kind of organ have the properties of both rigid body and soft body organs. To discuss this method, an application is presented. The organ of interest in the application is the human spine which is a typical articulated body organ. As noted in the first chapter, all implementations discussed in this thesis are devoted to the gait modeling. The spine plays an important role in the walking of human, therefore in the gait modeling, deformations of the spine are investigated. The task of the application is to extract the spine curve from MR data. In the gait modeling, the extracted spine curve changes correspondingly with the motions of tracking points attached along the spine of the patient (Fig. 5.1). Publication about this spine extraction is [97].

![Figure 5.1: Spine curve in the gait modeling (see [75] for detail).](image)

5.1 Introduction

Human spine is composed of 33 vertebrae, of which from top to bottom 7 vertebrae are in the cervical region (noted as $C_1 \ldots C_7$), 12 belong to the thoracic ($T_1 \ldots T_{12}$), 5 are in the lumbar ($L_1 \ldots L_5$), 5 compose the sacrum and the rest 4 are at the coccygeal. Since the sacrum and coccygeal are fused into a rigid bone, the spine mentioned here refers to the composite of the cervical, the thoracic and the lumbar. Besides the vertebrae, the spine contains intervertebral disks which are made up of soft tissues. Each intervertebral disk connects two adjacent vertebrae. Because of no intervertebral disk between $C_1$ and...
C2, there are 23 intervertebral disks in all. The MR data used in this method contains T2-weighted sagittal spine slices (Fig. 5.2).

![Image of T2-weighted sagittal MR spine data. There are 4 slices, which are at different spatial layers in the MR spine data. Three red points in the rightmost slice, which contains the complete spine, indicate the positions of three intervertebral disks (totally 23 disks). The name of the disk is defined by its two connected vertebrae, e.g., C2-C3 means the disk between the second and the third cervical vertebrae.](image)

**Figure 5.2:** Example of T2-weighted sagittal MR spine data. There are 4 slices, which are at different spatial layers in the MR spine data. Three red points in the rightmost slice, which contains the complete spine, indicate the positions of three intervertebral disks (totally 23 disks). The name of the disk is defined by its two connected vertebrae, e.g., C2-C3 means the disk between the second and the third cervical vertebrae.

### 5.1.1 State of the Art

In the last decade, spine image segmentation and analyzing was extensively studied and many methods were proposed. Most of them are focused on the vertebrae and intervertebral disk segmentation. Huang [45] presented a method which uses ID3 decision tree [73] and adaboost [38] method to pre-locate all the vertebrae candidates. Then a spine curve fitting approach is used to eliminate the outliers of the vertebrae candidates. Finally, each vertebra is segmented using normalized cut method [84]. The method from Peng [68] uses an intervertebral disk model to find all intervertebral disks. Then an intensity profile is produced by recording pixel intensity through the center of each detected disks. This intensity profile is applied to fix the intervertebral disks which are not detected by using the intervertebral disk model. Finally, the vertebrae are segmented by connectivity tracing of edges of the vertebrae. Shi [85] proposed a method which uses a statistic spine cord model to extract the spine cord. Then based on the spine cord position, intervertebral disks can be segmented. Michopoulou [58] presented an atlas-based method to match the disks and Yao [111] adopted watershed to extract the spine cord. But unfortunately, all the methods above are no fully automatic. Only two papers [27, 79], which applied
5.1 Introduction

probabilistic disk information to achieve fully automatic segmentation, have been found. However, the probabilistic disk information is gained by learning the spine location and appearance from huge number of training sets, which means the process is low efficient.

Commonly used MR spine data of a patient is composed of many sagittal view slices each of which contains image of different position of the patient inner body (Fig. 5.2). Within these slices, only 1 or 2 of them, which include a complete spine (Fig. 5.2 leftmost), are suitable for processing. Currently, most methods require to manually point out the suitable slice (e.g., in [85]). Furthermore, most methods use very complicated algorithms to obtain candidate positions of vertebrae (e.g., in [45]) or intervertebral disks (e.g., in [68]) and even manual selection (e.g., in [58]). In the following of this chapter intervertebral disk is called disk for short.

5.1.2 Overview of Articulated Body Organ Segmentation

The extraction of spine curve discussed here uses the segmentation method for the articulated body organ and the extraction process is fully automatic. The whole process of the method still can be divided into three stages which are the pre-processing stage, the global adaption stage and the local adaption stage.

Globally, the body of the articulated organ is dynamic like a soft body organ, and the idea for the global adaption is similar to the one used for the soft body organ. In the global adaption stage of the soft body organ segmentation method, the interesting organ curve is produced by interpolating points which are selected by hand. But in this method, such points can be detected automatically. It is know that the articulated body organ consists of a number of rigid components (e.g., the vertebra) and these components are connected by soft joints (e.g., the spine disk), and normally the joints are made up of a single homogenous tissue, moreover the joints show slight variation in structure. In this case, the joint positions are possible to be localized automatically. But of course, some articulated body organs may have easier detectable rigid components as compared to its joints. In other words, detecting the joint or the rigid component depends on the properties of the given organ. Since joints and rigid components in the organ are alternately combined, if the joint positions are known, then the rigid component positions are known too, and vice versa.

Locally, the articulated organ is rigid as a rigid body organ, because the organ contains a number of rigid components. In the local adaption, the objective is to segment each rigid component using a geometric model. Based on the detected joint or rigid component positions from the global adaption stage, parameters for initializing the geometric model before the segmentation of each rigid component in images can be determined. Then according to a suitable deformation model, each rigid component in images can be segmented.

The whole process sequence is that in the pre-processing stage, possible joint or component positions are detected first. Then in the global adaption stage, outliers in these possible positions are filtered out. Optionally, an approximate curve of the organ can be produced from the reserved positions (depending on given organs). But actually, the curve is often unnecessary, because spatial information of rigid components, which are used to initialize the model in the local adaption, can be extracted directly from the reserved positions. In the local adaption stage, because of the rigid body of each component, the deformation model, which is used to deform the model to adapt to each rigid component in images, does not need to be complex (e.g., affine transformation only).
This method is applied to extract the spine curve from MR data. Fig. 5.3 shows a flow chart of the extraction of the human spine curve. In the pre-processing stage, a gradient-based method is used to localize positions of all possible disks, i.e., joints. Then in the global adaption stage, non-disk positions are filtered out by means of a graph-based method and an Active Shape Model (ASM) [24] based filters. In the local adaption stage, based on the disks extracted from the global adaption stage, each vertebra of the spine is segmented using an atlas-based image registration method. The final result of the spine curve is interpolated from all the segmented vertebrae centers using the cubic spline [6]. The MR data are produced by a Siemens 1.5T MR scanner and the slice size in pixels is around 580 × 1100 (depending on individual) with the voxel size of 0.586 × 0.586 × 3.3 mm.

In the following sections, disk features in MR T2-weighted data are discussed first in Section 5.2. Based on these features, a gradient-based method for the extraction of possible disk positions in the pre-processing stage is presented in Section 5.3. The global adaption stage, where outliers of the extracted possible disk positions are filtered out to get a approximate spine curve, is discussed in Section 5.4. Section 5.5 describes the process in the local adaption stage where an atlas-based registration method is applied to segment each vertebra of the spine. Evaluations are given in Section 5.6.

### 5.2 Intervertebral Disk

It is known that the spine is composed of vertebrae (rigid components) and disks (joints). The reason why choosing the disks as the targets for the global adaption but not the vertebrae are that the vertebrae are much more variable than the disks (e.g., in size and
5.3 Gradient-Based Intervertebral Disk Extraction

_pose) and gray values of the disks in MR data are more homogeneous than the vertebrae. Both factors make the disks easy and stably detectable as compared to the vertebrae.

**Image Feature of Intervertebral Disk**

Fig. 5.4 shows some examples of the disks in MR data at different parts of the spine. The disk in the T2-weighted MR data is a dark strip-like region and the intensity of its adjacent vertebrae is much higher than the disk. Furthermore, the dark strip region orientates around horizontal in MR data. Based on these features, positions of possible disks can be found in MR data.

![Image of disks in MR slice](image)

**Figure 5.4:** Examples of disks in MR slice at different parts of the spine. From left to right, the disks from cervical (left), thoracic (middle) and lumbar (right).

**Extraction of Intervertebral Disk Positions**

The method for extracting positions of the spine disks is highly efficient and it contains three steps. In the first step, disk position candidates are extracted using a gradient-based method and then outliers are filtered out in the followed two steps which are the graph-based filter combining with intensity profile and the ASM-based non-disk filter. The first step belongs to the pre-processing stage and the rest two steps compose the stage of global adaption.

**5.3 Gradient-Based Intervertebral Disk Extraction**

First of all, the input MR spine data are denoised using a $3 \times 3$ median filter and then for each pixel the intensity gradient is calculated using the Sobel operator. Since common cases are considered, ranges of the gradient direction on the disk boundary are set to $[\pi/6, 5\pi/6]$ and $[-\pi/6, -5\pi/6]$. Gradient of each pixel, whose direction is not within these two defined ranges, is not considered in this method. In other words, pixel, whose gradient direction is within the defined ranges, is preserved otherwise the pixel is deleted. Example result of preserved pixels are shown in Fig. 5.5 right where the preserved pixels are divided into two classes (represented in different intensities) based on their gradient directions. Pixel whose gradient direction is in $[\pi/6, 5\pi/6]$ belongs to one class, and direction in $[-\pi/6, -5\pi/6]$ is classified to the other class.
Segmentation of Articulated Body Organ: Spine

Figure 5.5: Left: part of the original MR spine image where 6 disks at the thoracic are included. Right: result of preserved pixels. Pixels are divided into two classes (marked in different gray values) according to the pixel gradient direction ([\(\pi/6, 5\pi/6\]) in light gray and \([-\pi/6, -5\pi/6]\) in dark gray). Two red dashed rectangles show that most preserved pixels come from the disk boundaries.

Adjacent pixels of the same class are grouped into one region, and the region center is the average position of all included pixels. Small regions, which are defined as pixel number less than 20, are regarded as noises or non-disk regions and are deleted. Then for each preserved region a direction (noted as \(dr\)) is given by

\[
dr = \frac{\sum_{i=0}^{n-1} \| \nabla I(p_i) \| \cdot \text{Dir}(\nabla I(p_i))}{\sum_{i=0}^{n-1} \| \nabla I(p_i) \|}
\]

(5.1)

where \(p_i\) is the position of the \(i\)-th image pixel which belongs to a region. \(\| \nabla I(p_i) \|\) and \(\text{Dir}(\nabla I(p_i))\) are the image gradient magnitude and the direction at the \(i\)-th image pixel position. \(n\) is the number of pixels included in the region. The leftmost image in Fig. 5.6 shows an example result of the calculated region centers (red points) and corresponding orientations (blue lines).

Figure 5.6: Left: red points indicate centers of preserved regions and blue lines are region directions. Middle: region centers and directions are mapped to the original MR slice. Right: possible disk centers, each of which is a middle point of two adjacent region centers which have contrary directions.

As seen from the middle image of Fig. 5.6, each disk has two region centers which locate respectively at two sides of the disk. Furthermore, directions of the two region centers are
contrary. Therefore, if two adjacent region centers have contrary directions then the middle point between these two region centers will be the possible disk position. So for each pair of adjacent region centers, if the direction of the top region center is within \([-\pi/6, -5\pi/6]\) and the bottom region center direction is within \([\pi/6, 5\pi/6]\) then a middle point between these two region centers is regarded as a possible disk position. The rightmost image in Fig. 5.6 shows the result of calculated possible disk positions (marked by red points). Fig 5.7 shows some results of extracted possible disk positions of 6 patients. As seen from Fig 5.7, most of the calculated disk positions are at correct disk positions and there are also many outliers most of which locate at the top and the bottom parts of the images.

![Figure 5.7: Examples of calculated possible disk positions (marked as red points) of 6 patients using the gradient-based method.](image)

5.4 Global Adaption

As noted before, the extracted disk positions, which are the input of the global adaption stage, have many outliers, and the task in the global adaption stage is to filter out these outliers. The process in the global adaption consists of two steps which are the graph-based filtering step and the ASM-based filtering step.

5.4.1 Graph-Based Filter Combining with Intensity Profile

Considering the smoothness and the length of the spine which can be used as constraints to get correct disk positions, an undirected graph \(G = (V, E)\) is created. Each possible disk position is regarded as a node. Starting from the top node, each node is connected to all its nearby nodes which locate below. Then the task is to find the longest path without circle in the whole graph under the constraint of smoothness. The length of the path is
indicated by the number of nodes it goes through. The smoothness is defined by the angle between two vectors at each node on the path (discussed later).

In order to find the desired path in the graph, for each node, its upper and lower nodes within a certain range are selected. The range is defined by a circle whose center is at the node position and the radius is \( r \ \text{mm} \). Considering the possibility of missing disks (discussed later), the larger the \( r \) is the more robust the path search is. Setting the radius of the circle to the height of the MR slice is the safest solution, however, processing in such a large range is computational inefficient and the result does not benefit that much. Therefore, the radius is defined about 2-3 times larger than the height of the average vertebra of an adult (\( r = 100 \ \text{mm} \) in this method). Within the circle centered at each node, more than one upper or lower nodes can be found. Assume that for the current node \( N_c \), \( N_u_i \) (\( i = 0 \ldots n \)) and \( N_l_j \) (\( j = 0 \ldots m \)) are the satisfied upper and lower nodes of \( N_c \) respectively. Then each node from \( N_u_i \) is paired with each node from \( N_l_j \), it means that \( N_c \) has \( i \times j \) node pairs. According to all the node pairs of \( N_c \), a table of \( N_c \) is created. In the table, each entry is corresponding to a node pair (e.g., \( (N_u_i; N_l_j) \)), and the entry contains four items: the upper node index \( ID_u = i \), the lower node index \( ID_l = j \), the smoothness \( s \) and the number of following nodes \( k \). Assume that \( \vec{V}_{u_i,c} \) and \( \vec{V}_{c,l_j} \) are vectors which are equal to \( N_u_i - N_c \) and \( N_c - N_l_j \) respectively. Then the \( s \) is simply calculated by

\[
s = \left| \arccos \frac{\vec{V}_{u_i,c} \cdot \vec{V}_{c,l_j}}{\| \vec{V}_{u_i,c} \| \| \vec{V}_{c,l_j} \|} \right| = \left| \arctan \frac{\vec{V}_{u_i,c}.y}{\vec{V}_{u_i,c}.x} - \arctan \frac{\vec{V}_{c,l_j}.y}{\vec{V}_{c,l_j}.x} \right|
\]

which is the absolute angle between \( \vec{V}_{u_i,c} \) and \( \vec{V}_{c,l_j} \). The smaller the \( s \) is the more smooth the path is. Fig. 5.8 shows an example of a graph and the corresponding table of each node to illustrate the calculation of \( k \). There are 9 tables (shown right) corresponding to

![Figure 5.8](image)
5.4 Global Adaption

termination node. For each table (except the termination nodes), only one entry which has the minimum \( s \) value is left, others are removed (Fig. 5.9 left). For each entry in each table the \( k \) is calculated from the most bottom node, i.e., the 8th node in the example graph. For the most bottom node the \( k \) is equal to 1 which means that it has no following node lower than itself. Then for each upper node \( N \), the \( k \) is calculated by looking for the \( ID_l \) item in the entry of its own table. With the \( ID_l \), the corresponding \( ID_u \) node can be found. If the \( ID_u \) item in the entry of the \( ID_l \) node is equal to the index of the node \( N \), the \( k \) in the entry of \( N \) will be set to \( k_{ID_l} + 1 \) where \( k_{ID_l} \) is the \( k \) item value of the \( ID_l \) node. Fig.5.9 right shows the calculated \( k \) for each entry.

<table>
<thead>
<tr>
<th>Table</th>
<th>Entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(-1,1,-,-),(-2,-,-)</td>
</tr>
<tr>
<td>1</td>
<td>(0,4,45,-)</td>
</tr>
<tr>
<td>2</td>
<td>(0,3,2,-)</td>
</tr>
<tr>
<td>3</td>
<td>(2,5,2,-)</td>
</tr>
<tr>
<td>4</td>
<td>(1,6,2,-)</td>
</tr>
<tr>
<td>5</td>
<td>(3,7,40,-)</td>
</tr>
<tr>
<td>6</td>
<td>(4,7,2,-)</td>
</tr>
<tr>
<td>7</td>
<td>(6,8,2,-)</td>
</tr>
<tr>
<td>8</td>
<td>(7,-,-,1)</td>
</tr>
<tr>
<td>9</td>
<td>(-,1,-,6),(-,2,-,4)</td>
</tr>
</tbody>
</table>

Figure 5.9: Left: tables which only contain the entry with the minimum \( s \). Right: the calculated \( k \) of each entry.

Finally, the desired path of the graph is determined by checking all the termination nodes to see which entry has the highest \( k \). In this example graph, it is the entry of \((-1,-,6)\) of 0th node. Then the path is defined by getting values of item \( ID_l \) of all the involved nodes. In this example, indexes of all the involved nodes are 0, 1, 4, 6, 7, 8. Nodes, which lie on the desired path, are preserved and the rest nodes are deleted. In addition, if values of \( s \) in all the entries of a node table is bigger than 90 degrees (greater than 90 is not on the spine curve for sure), this node is regarded as a termination node (no upper node connected to it). The important thing is that it just needs to choose the termination node whose \( k \) is the biggest as the top beginning of the spine curve. Fig. 5.10 shows an example of the result using the graph-based filter on a slice of the MR spine data.

Due to the noise and the partial volume effect in the MR data, it is unavoidable that in the result of the gradient-based disk extraction, positions of some disks are missing (seldom) or more than one position is extracted on one disk (occasional). Motivated by [68, 111], an intensity profile along the longest path obtained from the graph-based filter is produced and a gaussian filter is applied to the profile to make it smooth. The intensity profile is drawn on an image whose height is the same as the height of the MR spine slice, and the width is set to 256 pixel long which represent the intensity values from 0 to 255. Since the intensity of the disk is lower than the vertebra, disk regions are indicated by valleys in the intensity profile. Therefore, the task is to find all the valleys, and in this method each valley is determined by checking whether it is a local minimum or not. Then each disk position of the MR spine slice is redefined as the point which lie on the longest path (red curve in the third image of Fig. 5.10) and at the same height \( Y \) as one of the valleys in the intensity profile image. Results of using the intensity profile are shown in Fig. 5.11 where two mentioned errors are shown in the left. Two gray lines marked in the left two images indicate parts of the longest path along which two intensity profiles are
Figure 5.10: An example of the graph-based filter process. From left to right, the possible disk positions from the first step, the undirected graph (red nodes and green edges), the longest path (marked in red) and the preserved disk positions.

made. The middle two images show the two intensity profiles corresponding to the left two images respectively. Right two images show the result of corrected disk positions where the two errors are solved.

Figure 5.11: Two errors are shown in the left images. One has a missing disk (top left) and the other has two positions on one disk (bottom left). Middle two images show intensity profiles along the corresponding paths (marked as gray lines in the left images) and the red points are valleys of the intensity profiles. Right images are corrected disk positions.
Till now it still cannot say that the extracted disk positions have no outliers as only local constraints have been applied and such constraints are inadequate to eliminate all outliers. Therefore, an Active Shape Model (ASM) of the spine curve is computed and used to define a global constraint.

5.4.2 Active Shape Model (ASM) Based Non-Disk Filter

The method for computing the ASM can be found in [26] and will not be described here. 10 MR spine slices, each of which contains a complete spine of a different patient, are used as training instances. Landmarks of the ASM are defined as the disk center positions. Therefore, each of these 10 training instances is composed of 23 disk centers. By using the Principal Component Analysis (PCA) on the 10 training instances, three eigenvectors, which have the highest corresponding eigenvalues and represent the main changing trends of the spine curve in these training instances, are chosen to produce possible spine curve variations. The objective is to produce a certain number of the spine curve variations which cover as many typical kinds of the spine curve as possible. In this method, 15 spine curve variations (Fig. 5.12), each of which exhibits a spine curve of different curvatures, are generated as the spine curve models by which the global constraint is defined.

Figure 5.12: 15 spine curve variations generated by the ASM. Red points indicate the landmarks which are the disk centers.

The idea is to use the 15 spine curve models and the extracted disk positions from the last step to define all the possible spine curves in MR data. Each two extracted disk positions in the MR slice are selected and regarded as the beginning and end of the spine. Then the 15 spine curve models are translated according to the two positions to define the possible spine curves. This means that for each two extracted disk positions, 15 possible spine curves can be defined. Considering the processing speed and the logical spine length, all the extracted disk positions are divided into 4 regions according to their $y$ coordinates, i.e., height. The position for the spine beginning can only be selected from the top region and the spine end position comes from the bottom region only. According to the positions from the top and bottom regions, all possible spine curves $SC_i$ ($i = 0, \ldots, 15 \times n_t \times n_b - 1$) can be defined. $SC_i$ is the $i$-th possible spine curve, $n_t$ and $n_b$ are numbers of extracted disk positions in
the top and bottom regions respectively. For each $SC_i$, its 23 landmarks (disk centers) $MS_i = (m_0, \ldots, m_{22})$ search for their nearest extracted disk positions in the MR slice. Therefore, every $SC_i$ has 23 corresponding extracted disk positions $PS_i = (p_0, \ldots, p_{22})$ of the MR slice. Then for each $SC_i$ and $PS_i$, the reliability is evaluated. The final result of the extracted disk positions are from the $PS_i$ which have the highest reliability. The reliability $Re$ is defined as

$$Re = \alpha \cdot \sum_{j=0}^{21} \frac{|d(m_j, m_{j+1}) - d(p_j, p_{j+1})|}{d(m_j, m_{j+1})} + \beta \cdot \sum_{j=0}^{22} d(m_j, p_j)$$

(5.3)

where $d(a, b)$ is the distance between positions $a$ and $b$. $\alpha$ and $\beta$ are weighting factors. The lower the value of $Re$ is the higher the reliability of $PS_i$ and $MS_i$ is. It is possible that identical positions are contained in the same $PS_i$. In this case, the first item of $Re$ is big which makes the value of $Re$ big. Fig. 5.13 shows three results of the ASM-based non-disk filter used in the MR spine slices of three patients. The left part of each image contains the extracted disk positions after the graph-based filter (red points) and a possible spine curve (a green curve), with which the corresponding possible disk positions set has the lowest $Re$. The right part of each image shows the disk positions contained in this set.

![Figure 5.13](image_url)

**Figure 5.13:** The left part of each image contains the extracted disk positions after the graph-based filter (red points) and a possible spine curve (a green curve), with which the corresponding possible disk positions set has the lowest $Re$. The right part of each image shows the disk positions contained in this set.

extracted disk positions after the graph-based filter (red points) and a possible spine curve $SC$ (a green curve), with which the correspond $PS$ has the lowest $Re$. The right part of each image shows the disk positions contained in $PS$. It is clear that all the outliers which are at the topmost and the bottommost of the images are eliminated.

The MR slice, which contains the final result of the 23 extracted disk positions, implies that it includes a complete spine. Therefore, in this method, the MR slice which contains
the complete spine can be automatically selected from the MR spine data (commonly containing circa 20 MR slices) by checking each slice whether the final extracted disk positions are included in the slice or not.

5.5 Local Adaption

In the local adaption stage, the disk positions extracted in the global adaption stage are used by an atlas-based image registration process to initialize the vertebra atlas for segmenting each vertebra in the MR slice.

5.5.1 Vertebra Registration and Spine Curve Extraction

Actually, the spine curve can be already produced from the extracted disk positions, however since the extracted disk positions can not be guaranteed to be at the disk centers, the produced spine curve could be inaccurate. In order to achieve a highly accurate result of the spine curve as well as to segment each vertebra, in the local adaption stage, an atlas-based vertebra registration is applied. The idea is to use the vertebra atlas and the result of the disk positions to do the registration for each vertebra in the MR slice. Based on the segmented vertebrae, an accurate spine curve can be obtained by interpolating the centers of the segmented vertebrae. Furthermore, the orientation and form of each vertebra can also be got. In this method, the vertebra atlas is simplified as a combination of three rectangles, of which two rectangles at top and bottom represent adjacent disks and the middle one stands for the vertebra (Fig. 5.14).

Figure 5.14: The vertebra atlas used in this method. It contains the vertebra (white region) and two adjacent disks (two dark gray regions). The gray values indicate different regions.

For the atlas-based image registration, the most important thing is the similarity measurement, and the commonly used similarity measurement methods are Correlation Ratio (CR) [74] and Mutual Information (MI) [56]. Following are their definitions.

\[ CR(A, B) = \frac{Var(E[B|A])}{Var(B)} = \frac{\sum_i n_i \cdot (\bar{T}_{r_i} - \bar{T})^2}{\sum_{r_{i,j}} (I_{r_{i,j}} - \bar{T})^2}, \]  
(5.4)

\[ MI(A, B) = H(A) + H(B) - H(A, B) \]  
(5.5)

where \(A\) and \(B\) represent the atlas and the overlapped part of the MR slice respectively. \(E[B|A]\) is the conditional expectation of \(B\) given \(A\). Different regions are discriminated by different labels (gray values) in the atlas. \(r_i\) is the \(i\)-th region defined by the atlas. \(I_{r_{i,j}}\) is the overlapped MR slice intensity at the \(j\)-th pixel which belongs to the \(r_i\) region. The
region to which the overlapped MR slice pixel belongs is determined by looking up the corresponding overlapped label in the atlas. \( r_i \) is the average intensity of the overlapped MR slice pixels which belong to the \( i \)-th region. \( T \) is the average intensity of the whole overlapped MR slice. \( n_r \) is the number of the overlapped MR slice pixels belonging to the \( i \)-th region. To the equation of MI, \( H(A) \) and \( H(B) \) are the entropy of the \( A \) and \( B \) respectively. \( H(A, B) \) is the joint entropy of \( A \) and \( B \).

**Comparison of Correlation Ratio and Mutual Information**

Assume that the atlas has the same size as a vertebra in a MR slice and is fixed at the center of the vertebra of the MR slice. Then the atlas rotates itself counter-clockwise with each step of 1 degree (Fig. 5.15). For each step, corresponding values of CR and MI are calculated. Pixels of the rotated atlas may not coincide with the pixels of the overlapped MR slice. The nearest neighbor method is applied to solve this problem. The pixel of the atlas always corresponds to the nearest pixel of the overlapped MR slice. The experiment results are shown in Fig. 5.16. Since the atlas is longitudinal symmetry, there are two peaks which are at 150 and 330 degrees on the CR and MI curves. It is clear that the CR curve is much more smooth than the MI curve. Maes [56] suggested to use the trilinear partial volume distribution instead of the nearest neighbor method to smooth
the MI curve. However, the processing time is much longer. Therefore, in this atlas-based image registration, CR is chosen as the similarity measurement method.

**Problem of Changes in Overlap**

Let’s continue with the experiment of CR and MI. Fig. 5.17 illustrates four possible registration results of the atlas. The dashed red line represents the atlas. The synthetic white and black regions stand for the vertebra and its two adjacent disks respectively. Obviously, the leftmost result is expected. However, all the four results have the same MI or CR values which means in the view of computer each of the four cases can be seen as the final registration result. Considering these four cases in the real MR slice, the result is even worse. Assume that the atlas has the same orientation as a vertebra in a MR slice, but the width of the atlas is longer than the vertebra (Fig. 5.18 left). Then we decrease the width of the atlas with the step of 1 pixel (Fig. 5.18). For each step, corresponding values of CR and MI are calculated. Fig. 5.19 shows the experiment results where the black vertical line indicates the true width of the vertebra (corresponding to the middle image in Fig. 5.18). However when the atlas width continues decreasing, both CR and MI values are increasing.

The reason is that in the middle of the vertebra in the MR slice, the image intensity is more homogeneous, so the image intensity deviation within each region \( r_i \) \((i = 0 \ldots n)\) becomes smaller. Therefore \( \sum_i n_{r_i} \cdot (I_{r_i} - \bar{T})^2 \) is more and more closer to \( \sum_{r_{i,j}} (I_{r_{i,j}} - \bar{T})^2 \), and the CR value is increased. For the MI, as the atlas width decreases, the overlapped part on the MR slice becomes smaller. The varieties of the image intensity in the overlapped
5 Segmentation of Articulated Body Organ: Spine

Figure 5.19: Values of CR (blue) and MI (red) along with the decrease of the atlas width. The black line indicate the true width of the vertebra of MR slice.

MR slice are reduced, thereby the entropy of the overlapped MR slice $H(B)$ and the joint entropy $H(A, B)$ decreases. However, the decreased rate of the $H(B)$ is slower than $H(A, B)$, and $H(A)$ is constant. Therefore the MI value becomes bigger. Actually, even using the improved algorithm of MI: the Normalized Mutual Information (NMI) [89], the situation is still the same.

To solve this problem, image gradients at the MR slice pixels, which are overlapped with the boundary of the middle rectangle region of the atlas, are considered. So the similarity measurement in this method is defined as

$$S(A, B) = \lambda_1 \cdot CR(A, B) + \lambda_2 \cdot G(A, B)$$

(5.6)

where $CR(A, B)$ is the correlation ratio between the atlas $A$ and the overlapped MR slice $B$, $G(A, B)$ is the sum of gradient magnitudes at the pixels of $B$ which are overlapped with the boundary of the middle rectangle region of the atlas. $\lambda_1$ and $\lambda_2$ are weighting factors. In the definition of $S(A, B)$, $CR(A, B)$ prevents the atlas from shifting to the boundary of adjacent vertebra, and $G(A, B)$ prevents the atlas from falsely decreasing its width when use only $CR(A, B)$. Fig. 5.20 shows the diagram of $S(A, B)$ and MI values versus the

Figure 5.20: Values of $S(A,B)$ (blue) and MI (red) along with the decrease of the atlas width. The black line indicate the true width of the vertebra of MR slice.
5.5 Local Adaptation

atlas width, and there is only one peak of $S(A, B)$ which occurs at the true vertebra width in the MR slice.

Registration and Interpolation

For each vertebra in the MR slice, its top and bottom adjacent disk positions, which are extracted in the global adaption stage, are available. By connecting the two disk positions, the length, the middle point and the orientation of the connection line are approximately equal to the height, the center position and the orientation of the corresponding vertebra. These three parameters are used to initialize the vertebra atlas (atlas width is equal to its height). Then the atlas is deformed using the affine transformation to do the registration, and the optimal transformation parameter set (scaling factors, rotation and translations) is obtained by the exhaustive searching method. Because the vertebra atlas has been properly initialized, so the exhaustive searching method can work in small searching ranges. In this method, the scaling factor ranges of the atlas width and height are $[0.5, 1]$ and $[0.9, 1]$ (step size 0.1), the rotation range is $[-3, 3]$ in degree (step size 1) and the translation range for both X and Y directions is $[-3, 3]$ in pixels (step size 1). Therefore, there are 4116 possible parameter sets, and for each set the $S(A, B)$ value is calculated. The optimal transformation parameter set always has the highest $S(A, B)$ value. After the registration, the final result of the spine curve is obtained by interpolating centers of all the segmented vertebrae using the cubic spline. Fig. 5.21 shows the spine curve results of three patients. Each image contains the segmented vertebrae and the final spine curve.

![Figure 5.21](image_url)

**Figure 5.21:** Three example results each of which contains the vertebrae (red rectangles) segmented by the atlas-based image registration and the final spine curve (right part of each image) produced by interpolating the vertebrae centers (green points).
5.6 Evaluation

Image Data used for the evaluation are T2-weighted sagittal MR spine data of 13 patients and each MR data contains circa 20 MR slices. The evaluation contains two parts. The first part is the evaluation of the disk positions extraction, and the second part is the evaluation of the final spine curves which are obtained by the vertebra registration using the improved correlation ratio.

5.6.1 Evaluation of the Disk Position Extraction Results

The disk position extraction is composed of three steps: the gradient-based extraction of the disk candidates, the graph-based filter combining with the intensity profile and the ASM-based non-disk filter. Each step was evaluated by calculating the ratio of the number of the true positive disk positions with the number of all extracted disk positions (i.e., the precision). The results of the gradient-based extraction and the graph-based filter with the intensity profile are shown respectively in Table 5.1 and Table 5.2.

<table>
<thead>
<tr>
<th>ID</th>
<th>$n_{total}$</th>
<th>$n_{true}$</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>38</td>
<td>22</td>
<td>0.58</td>
</tr>
<tr>
<td>2</td>
<td>37</td>
<td>23</td>
<td>0.62</td>
</tr>
<tr>
<td>3</td>
<td>41</td>
<td>23</td>
<td>0.56</td>
</tr>
<tr>
<td>4</td>
<td>41</td>
<td>23</td>
<td>0.56</td>
</tr>
<tr>
<td>5</td>
<td>45</td>
<td>23</td>
<td>0.51</td>
</tr>
<tr>
<td>6</td>
<td>42</td>
<td>23</td>
<td>0.55</td>
</tr>
<tr>
<td>7</td>
<td>39</td>
<td>23</td>
<td>0.59</td>
</tr>
<tr>
<td>8</td>
<td>44</td>
<td>23</td>
<td>0.52</td>
</tr>
<tr>
<td>9</td>
<td>40</td>
<td>23</td>
<td>0.57</td>
</tr>
<tr>
<td>10</td>
<td>35</td>
<td>23</td>
<td>0.66</td>
</tr>
<tr>
<td>11</td>
<td>41</td>
<td>23</td>
<td>0.56</td>
</tr>
<tr>
<td>12</td>
<td>40</td>
<td>23</td>
<td>0.57</td>
</tr>
<tr>
<td>13</td>
<td>40</td>
<td>23</td>
<td>0.57</td>
</tr>
</tbody>
</table>

Table 5.1: Evaluation of the gradient-based disk extraction. $n_{total}$ is the total number of extracted disk positions and $n_{true}$ is the number of the true positive disk positions. Proportion of the correct disk positions is shown in the item Ratio.

<table>
<thead>
<tr>
<th>ID</th>
<th>$n_{total}$</th>
<th>$n_{true}$</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>25</td>
<td>23</td>
<td>0.92</td>
</tr>
<tr>
<td>2</td>
<td>24</td>
<td>23</td>
<td>0.96</td>
</tr>
<tr>
<td>3</td>
<td>27</td>
<td>23</td>
<td>0.85</td>
</tr>
<tr>
<td>4</td>
<td>23</td>
<td>23</td>
<td>1.0</td>
</tr>
<tr>
<td>5</td>
<td>30</td>
<td>23</td>
<td>0.77</td>
</tr>
<tr>
<td>6</td>
<td>26</td>
<td>23</td>
<td>0.88</td>
</tr>
<tr>
<td>7</td>
<td>26</td>
<td>23</td>
<td>0.88</td>
</tr>
<tr>
<td>8</td>
<td>28</td>
<td>23</td>
<td>0.82</td>
</tr>
<tr>
<td>9</td>
<td>25</td>
<td>23</td>
<td>0.92</td>
</tr>
<tr>
<td>10</td>
<td>24</td>
<td>23</td>
<td>0.96</td>
</tr>
<tr>
<td>11</td>
<td>25</td>
<td>23</td>
<td>0.92</td>
</tr>
<tr>
<td>12</td>
<td>25</td>
<td>23</td>
<td>0.92</td>
</tr>
<tr>
<td>13</td>
<td>24</td>
<td>23</td>
<td>0.96</td>
</tr>
</tbody>
</table>

Table 5.2: Evaluation of the graph-based filter with the intensity profile. $n_{total}$ is the total number of extracted disk positions and $n_{true}$ is the number of the true positive disk positions. Proportion of the correct disk positions is shown in the item Ratio.
As shown in these two tables, the ratio of the first step was around 0.5, and after the second step the ratio was increased to around 0.9. In addition, the extracted disk positions of each test MR data referred in these two tables were from the MR slice, which contains the complete spine in the test MR data. Such MR slice in each test MR data was automatically selected. In this evaluation, the final disk positions of each test MR data after the ASM-based non-disk filter were at the true disks in the MR slice, i.e., the ratio after the third step was 1.0.

The computer, which was used to run the application, has a $2 \times 2.66$ GHz CPU, 4G memory and the operating system is WinXP. The processing time of the disk position extraction is shown in Table 5.3. The slice number $n$ of each tested MR data was around 20. The processing time $t_{total}$ for each MR data was between 16-30 s and the average processing time $t_{mean}$ for one MR slice was around 1 s. In this method, when the number of the extracted disk positions of a MR slice after the first two steps (gradient-based and graph-based steps) was less than 23, then the third step, which is the ASM-based non-disk filter, would not be executed.

**Table 5.3:** The processing time of the disk extraction. $n$ is the total number of MR slices in each MR data. $t_{total}$ and $t_{mean}$ (in second $s$) represent the processing time for each MR spine data, and the average processing time for one MR slice of each MR data respectively.

<table>
<thead>
<tr>
<th>ID</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n$</td>
<td>20</td>
<td>26</td>
<td>21</td>
<td>24</td>
<td>25</td>
<td>22</td>
<td>21</td>
<td>16</td>
<td>23</td>
<td>18</td>
<td>20</td>
<td>18</td>
<td>20</td>
</tr>
<tr>
<td>$t_{total}$</td>
<td>16</td>
<td>30</td>
<td>20</td>
<td>18</td>
<td>29</td>
<td>22</td>
<td>17</td>
<td>13</td>
<td>21</td>
<td>15</td>
<td>15</td>
<td>17</td>
<td>14</td>
</tr>
<tr>
<td>$t_{mean}$</td>
<td>0.8</td>
<td>1.2</td>
<td>1.0</td>
<td>0.8</td>
<td>1.2</td>
<td>1.0</td>
<td>0.8</td>
<td>0.8</td>
<td>0.9</td>
<td>0.8</td>
<td>0.8</td>
<td>0.9</td>
<td>0.7</td>
</tr>
</tbody>
</table>

To show the compatibility of the disk extraction, two sets of MR data produced using different scan protocols and MR scanners were also tested. Both MR data were provided by the University of Cincinnati College of Medicine. Results are shown in Appendix A.1.

### 5.6.2 Evaluation of the Spine Curve Results

For the evaluation of the spine curve which was obtained by using the atlas-based vertebra registration, center and orientation of each segmented vertebra were compared with the results segmented by an expert. The evaluation results are shown in Table 5.4. The mean disparity $d_{c_{mean}}$ of the vertebra center of the 13 tested MR data was between 1.2-1.7 mm, and the mean orientation deviation $d_{o_{mean}}$ was around 3 degrees. The disparity is the distance (in mm) between the automatically extracted and manually segmented vertebra centers. Since the pixel size of all the tested MR data was $0.586 \times 0.586$ mm, the mean disparity of the vertebra center was just 2-3 pixels. The processing time for the registration of each vertebra was almost the same which was around 5 seconds.
### Table 5.4: Evaluation of the spine curves which were obtained by using the atlas-based vertebra registration method. The maximum, minimum and average disparities of the vertebra centers are represented by $d_{c_{\text{max}}}$, $d_{c_{\text{min}}}$, and $d_{c_{\text{mean}}}$ (in mm). The deviations of the vertebra orientation are represented in the same manner ($d_{o_{\text{max}}}$, $d_{o_{\text{min}}}$, and $d_{o_{\text{mean}}}$ in degree).

<table>
<thead>
<tr>
<th>ID</th>
<th>$d_{c_{\text{max}}}$</th>
<th>$d_{c_{\text{min}}}$</th>
<th>$d_{c_{\text{mean}}}$</th>
<th>$d_{o_{\text{max}}}$</th>
<th>$d_{o_{\text{min}}}$</th>
<th>$d_{o_{\text{mean}}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.5</td>
<td>0</td>
<td>1.4</td>
<td>9</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>1.9</td>
<td>0</td>
<td>1.3</td>
<td>8</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>2.3</td>
<td>0</td>
<td>1.2</td>
<td>9</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>3.5</td>
<td>0.6</td>
<td>1.5</td>
<td>8</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>3.5</td>
<td>0.6</td>
<td>1.4</td>
<td>9</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>3.7</td>
<td>0.6</td>
<td>1.4</td>
<td>8</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>3.8</td>
<td>0</td>
<td>1.7</td>
<td>6</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>2.3</td>
<td>0</td>
<td>1.5</td>
<td>9</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>2.9</td>
<td>0</td>
<td>1.2</td>
<td>8</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>3.5</td>
<td>0</td>
<td>1.4</td>
<td>9</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>11</td>
<td>1.9</td>
<td>0</td>
<td>1.2</td>
<td>8</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>2.7</td>
<td>0</td>
<td>1.3</td>
<td>10</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>3.7</td>
<td>0</td>
<td>1.4</td>
<td>8</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

### 5.7 Conclusions

A fully automatic segmentation of the spine curve using the segmentation method for the articulated body organ has been described. The segmentation process consists of three stages. In the first stage i.e., the pre-processing stage, possible joint positions i.e., the disks are extracted using a gradient-based method. In the global adaption stage, outliers of the possible joint positions are filtered out to obtain the spatial information about the spine. In the local adaption stage, an atlas-based registration method, which uses an improved correlation ratio based similarity measurement, is applied to guide the atlas to adapt to each component i.e., the vertebra of the spine. By means of the improved similarity measurement, the problem of changes in overlap is solved. In the experiment, the average disparity of the segmented vertebra centers as compared to the manual segmentation results by an expert is around $1.4$ mm. So the final spine curve, which is produced by interpolating centers of the segmented vertebrae, is at a highly accurate level.

The main advantage of the spine curve segmentation discussed here is that it accepts MR data produced using different kinds of scan protocols without changing parameters or requiring additional work. The only constraint of this method concerning the image is that the disk intensity must be different from the vertebra intensity. But this requirement is almost always satisfied. Furthermore, the disk extraction process in the pre-processing and global adaption stages is highly efficient and accurate. According to the experiment of the extraction of all the spine disks in a MR spine slice (resolution of $512 \times 1012$ pixels), the average time is around 1 second (C2D 2×2.66GHz CPU, 4G RAM), and results of the extracted spine disks are all correct. Furthermore, this spine curve segmentation method can be used by other segmentation methods to achieve fully automatic 3D spine segmentation. For example, Ghebreab etc. [40] proposed a method using string and necklace models to do 3D segmentation of spine, however, initial positions of the spine have to be set manually. If they use this spine curve segmentation method, no human interaction will be needed.

Concerning the influence of parameters which include the disk region size, the disk
region direction range, and the searching range of the graph, the function of the last two parameters is to enhance the processing efficiency, and they can be set extremely large without affecting the final result. However, the processing speed will be seriously dropped. The most important parameter is the first parameter i.e., the disk region size. Too small region size causes the increasing number of possible disks which make the outlier filtering inefficient and inaccurate while too large region size makes the missing of true disks. Since in the body of normal adults the variation of the disk size is small, setting a proper value for the disk region size is not a big issue.

Based on the segmentation method for the articulated body organ, many other objects having articulated bodies can also be segmented. For example, the arm of the digger which can be seem as an articulated body. According to the form of the joint e.g., a circle, the possible joints of the digger arm can be detected using e.g., the Hough circle detection method. To filter out the outliers of the detected joints, possible postures of the digger arm can be used as a global constraint. The possible postures could be determined based on the degree of freedom of the components of the digger arm. Normally, the movement of each component of the digger arm is restricted within a limited range, and most of the digger arms have just 4 components. Therefore, getting the possible postures of the digger arm is much easier than the spine. After obtain the joint positions, each component can be segmented with ease.
6 Applications

In this chapter, the applications using the methods proposed in this thesis are presented. Most of the applications are currently used in the project PROREOP [71]. The objective of this project is to develop an environment for the gait modeling, by which a diagnosis system for the gait pathology can be constructed.

6.1 Bone Segmentation Application

This application applies the method for segmenting the rigid body organ (discussed in Chapter 3), and it has been used to do the limb bone segmentation. Fig. 6.1 shows the graphical user interface of this application. Besides the bone segmentation, this application is capable of extracting the femur cavity volume. In the following subsections, the tasks, which are relevant to this application, are described.

6.1.1 Bone Segmentation and 3D Reconstruction

In order to achieve the gait modeling, the pelvis, the femur, the tibia, the fibular and the foot are required to be segmented. These requirements can be fulfilled by employing this

![Figure 6.1: Application of the limb bone segmentation.](image)
6.1 Bone Segmentation Application

application. Fig. 6.2 shows an example of the segmented bones of the lower limb, and different bones are marked in different colors.

![Figure 6.2: Example of Segmented lower limb. Different bones are marked in different colors.](image)

6.1.2 Femur Cavity Volume Extraction

The femur cavity volume is needed to do the surgery of the hip joint replacement. The idea is to use a hip joint prosthesis to replace the damaged femur head (Fig. 6.3). To make sure that the stem of the prosthesis, which has to be inserted into the femur, fits the femur inner room, the cavity volume of the femur has to be calculated.

To measure the femur cavity volume, the segmentation of the femur from the input MR data has to be done first, then the region growing method is applied to obtain the inner region of the segmented femur. According to the area of the inner region and the voxel

![Figure 6.3: Left: the X-ray image of the planted prosthesis. Right: the simulation of the planted prosthesis in the segmented femur.](image)
size of the MR data, the femur cavity volume can be calculated (in $mm^3$). Fig. 6.4 shows an example of the extracted cavity of the femur.

**Figure 6.4:** Example of the segmented femur cavity. The left two images show the 3D volume representation of a segmented femur (white) and its cavity (black tubule). The right image shows a 3D femur cavity constructed using the Iso-surface method.

### 6.1.3 Femur Segmentation of X-ray Image

Segmentation of the femur from a 2D X-ray image is a difficult task, because the boundary between the femur head and the pelvis is very low. To achieve this task, the femur of the same patient from the MR data is segmented first, then the segmented femur is used as a model which can be projected on the X-ray femur image of this patient (Fig. 6.5). The initial orientation of the model is determined based on the orientation of the femur shaft in the X-ray image. Then the exhaustive searching method is applied to find the

**Figure 6.5:** The left window shows the X-ray image, the two red lines indicate the femur orientation which is needed to adjust the 2D projection of the segmented 3D femur (the right window).
6.1 Bone Segmentation Application

best projection parameters e.g., rotation and translation for the model. For each test parameter, a value which indicates the projection quality is calculated (Fig. 6.6). The projection quality is determined by checking the number of the X-ray image edges which are located at the boundary of the projected model. The final projection parameters correspond to the highest projection quality.

Figure 6.6: Left: an example of a projected femur model. Right: illustration of the projection quality.

6.1.4 Superquadrics based Matching

The idea is to use the superquadrics algorithm [5] to model the segmented femur which is composed of a huge number of 3D points. According to the matched superquadrics, an accurate result of the FKF could be obtained (Fig. 6.7).

Figure 6.7: Example of the modeling of the 3D points from the segmented femur by the superquadrics algorithm. The left object is the 3D points of the segmented femur. The right object is the modeling result using the superquadrics.

For example, to obtain the radius of the femur head, the 3D points, which belong to the segmented femur head, are focused. These 3D points are modeled with a 3D sphere using
the superquadrics algorithm, and the radius of the femur head is equal to the radius of the matched 3D sphere. Details about this work can be found in [28].

6.1.5 3D Gait Modeling

As mentioned in the very beginning of this thesis, the purpose of the project PROREOP is the 3D gait modeling. The segmented bones of a patient are put into a virtual 3D scene where the simulation of the gravitational field is embedded. The motion of each bone during the walk of the patient is obtained under a Ganglabor environment. The patient walks under 7 cameras, which are used to capture the movements of the tracking points attached on the patient body (Fig. 6.8).

![Figure 6.8: Example of the Ganglabor. The patient walks straightly, and the tracking points are attached on his lower limb.](image)

The movement of each tracking point indicates the motion of the corresponding bone. Based on all the motions, the gait of the patient can be modeled (Fig. 6.9). Details can be found in [92].

![Figure 6.9: Example of the gait modeling. The segmented bones are colored in yellow. The white balls indicate the positions of the tracking points on the patient body.](image)
6.2 Muscle Segmentation Application

The application using the method for segmenting the soft body organ was presented in Chapter 4. The objective of the application is to segment the muscle. In that chapter, the sartorius muscle is taken as an example to describe the segmentation process. But besides the sartorius muscle, the application can be used to segment other muscles. In principle, the muscle, whose origin and insertion are located on different bones and have no branches, can be segmented by using this application. Fig. 6.10 shows an example of the segmentation of the gracilis muscle [104].

![Figure 6.10: Example of the segmentation of the gracilis muscle. The red contours indicate the cross sections of the muscle model.](image)

Fig. 6.11 shows a 3D reconstruction result of the segmented gracilis muscle using the delaunay method.

![Figure 6.11: Example of the 3D reconstruction of the segmented gracilis muscle.](image)

In the 3D gait modeling, the segmented muscles can be added. There are two manners of the muscle visualization. The first manner uses only the origin and insertion points of the muscle, and a simple line connecting these two points is visualized (Fig. 6.12). The
other manner uses the complete segmented muscle to do the visualization, and an example method can be found in [10].

![Figure 6.12: Example of the muscle visualization using the line manner.](image)

### 6.3 Spine Curve Extraction Application

Segmentation of the spine curve could be applied to do the analysis of the deformation of the spine in the gait modeling of the patient (see Fig. 6.13).

![Figure 6.13: Example of the analysis of the spine deformation in the gait modeling. The original graphic is in [75].](image)

Previously, the spine curve is depicted manually from some kinds of the medical images e.g., X-ray, and MRI. By using the method of the spine curve extraction (discussed in Chapter 5), getting a spine curve from the input MR data takes only a few seconds, and no human interaction is needed.
7 Conclusions and Perspectives

In this thesis, the main concern is medical image segmentation using the method based on the geometric deformable model. The segmentation proceeds in a global and local model adaption pattern. According to the adaption pattern, the method can be divided into three main stages which are the pre-processing, the global adaption and the local adaption stages. Totally, three methods have been proposed to solve the problems involving the segmentation of the rigid body, the soft body and the articulated body organs, respectively. All the proposed methods are as general as possible, and actually they can be regarded as the frameworks. Because in the practical applications, different kinds of the organ of interest, different medical image modalities and different patients could be encountered, and highly specific segmentation algorithms may not be suitable.

7.1 Rigid Body Organ Segmentation

Concerning the segmentation method of the rigid body organ, in the pre-processing stage, The Mean-Shift filter is applied to do the edge-preserved image denoising. Then the thresholding and region growing methods are used to get the pre-segmentation of the organ of interest. Based on the pre-segmentation result, pose, size, and position of the organ of interest, which are needed in the global adaption stage, can be extracted. The objective of the global adaption is to make the geometric model of the organ of interest have the same pose, size and position as the pre-segmented organ. Since the body of the organ is rigid, the adjustment of the model pose can be achieved by rotating the model to make its axis the same as that of the pre-segmented organ. The adjustment of the model size focuses on the Vertical Span Length (VSL) of the pre-segmented organ, because calculating the VSL is more accurate and efficient as compared with the calculation of the length of the organ body. Furthermore, the necessary information (e.g., MR slice thickness and the number of slices) for getting the organ VSL is easy to obtain. The identical size between the model and the pre-segmented organ can be achieved by scaling the VSL of the model to the VSL of the pre-segmented organ. The difference of the position between the model and the pre-segmented organ is measured by a translation vector, which starts from a predefined reference position of the model (e.g., the femur shaft center in the femur segmentation) to the corresponding position of the pre-segmented organ. Then the difference of the position can be removed by adding the translation vector to each point of the model. The model after the global adaption is then put into the local adaption stage. In this stage, an improved snakes algorithm is developed to remove the contour differences between the model and the organ. After the local adaption, the model should be totally adapted to the organ in the input medical images.

Because the main purpose of this method is to do medical image segmentation for the routine clinical use, it is required to be able to work with many kinds of image modalities (e.g., CT, MR). The thresholding and region growing methods are considered as the all-
around algorithms for image segmentation, therefore they are adopted in the pre-processing stage. However, both algorithms are not elegant enough, because much user interaction is needed. So in the future work, the amount of the user interaction should be reduced.

7.2 Soft Body Organ Segmentation

The second method proposed in this thesis is the segmentation of the soft body organ. The segmentation strategy is similar to the method of the rigid body organ. The pre-processing stage deals with the image quality enhancement and the pre-segmentation. Then, in the global adaption stage, a geometric model of the organ of interest is adjusted in the purpose of eliminating the differences of pose, size, and location between the model and the organ of interest. However, due to the dynamic body of this kind of organ, the global adaption process is completely different from that of the rigid body organ. For example, the body axis is used to deal with the pose difference in the method of the rigid body organ, but for the soft body organ, the axis can not be fixed. Instead, the curve of the organ of interest is used to achieve the global adaption. Based on some manually selected points located at some parts of the organ of interest, the complete curve of the organ of interest can be calculated using an interpolation algorithm (e.g., the cubic spline). According to the calculated curve, the model can be aligned to the organ of interest. Therefore, the differences of pose, size and location can be removed. In the local adaption stage, the contour differences are considered, and they are eliminated by means of a deformation model. The selection of a proper deformation model should depend on the properties of the organ of interest. The deformation model used in the application of the muscle segmentation is the ASM, because in the MR data the muscle often has obscure boundary, methods like snakes can be easily confused by the insignificant image gradient at the muscle boundary. Considering that the variations of the muscle cross sections are within a limited range which can be modeled, so the ASM is selected for the segmentation of the muscle.

Regarding the future improvements, the interpolation of the organ curve requires some reference points which are currently set by hand, and it is better to make it automatic. A small progress has been made in the application of the muscle segmentation, where the approximate beginning and end positions of the sartorius muscle i.e., TAS1 and TAS2 are automatically extracted using the contour scanning code method. But the complete curve of the sartorius muscle can not be interpolated using only two points, and the rest points are still designated manually. Therefore, a more intelligent approach is required. A possible way is to employ the probability atlases containing some cross sections of the organ of interest and other adjacent organs. Then by registering the atlases to the corresponding MR slices. The centers of the organ of interest can be obtained from the registered atlases. These centers can be used to produce the curve of the organ of interest.

7.3 Articulated Body Organ Segmentation

Globally, the articulated body organ has a dynamic body like the soft body organ. But locally, the articulated body organ is composed of a number of components which belong to the rigid body organ.

As the dynamic body in the global point of view, the processing scheme for the global
adaption is similar to that for the soft body organ segmentation where the organ curve is considered. The only different is that the points, which are used to interpolate the organ curve, are selected manually in the soft body organ segmentation, while such points in the articulated body organ segmentation are detected automatically. Because the articulated body organ consists of a number of rigid components and soft joints, both are regularly and alternately combined with each other. Based on this special structure of the organ, it is possible to detect the joints or components of the organ automatically. For example, in the application for the extraction of the spine curve, the spine disks can be automatically localized. Furthermore, in the global adaption of this method, the organ curve is not needed to be interpolated, because the pose, the size, and the position of each rigid component of the articulated body organ can be directly calculated using the detected points.

With the help of the detected points from the global adaption stage, the segmentation of each organ component, which is the task of the local adaption, can be achieved with ease. Normally, the structure of the organ components is similar e.g., the form of the vertebrae of the spine is a rectangle (sagittal view). So using one geometric model possessing a typical structure of the organ component is enough. Because each component has a rigid body and the joint positions are already detected in the global adaption stage, the parameters (e.g., pose, size and position) for initializing the geometric model of the organ component can be determined. Then the further adaption process, which uses a simple deformation model (e.g., the affine transformation), is carried out to achieve the final segmentation of each component.

Concerning the improvement in the future, strengthening the applicability and robustness of this method is a primary task. Regarding the applicability, the method should be able to work with different modalities of the input medical images, e.g., CT, MR, and with different scan protocols. The encouraging thing is that, in the application of the extraction of the spine curve, the method has shown a potential capacity of solving the MR data produced using different scanners and protocols (see appendix A). Considering the robustness, the constraint of the current method is that each component of the articulated body organ should be roughly coplanar. For example, in the spine curve extraction, at least one MR sagittal slice must contain a whole spine. Although normally this requirement is always satisfied, but it is preferred that the method is capable of solving very unusual cases e.g., heavily distorted spines. Fortunately, according to the scheme of this method, just a little modification in the pre-processing and the local adaption stages can make the method relieve from the constraint. Assuming that the articulated body organ is heavily distorted, it is impossible to make an image containing a whole body of the organ. The modification is that in the pre-processing stage, the joint positions are extracted not only from one image but also from a certain number of adjacent images, and all the extracted joint positions are gathered together. In the global adaption stage, the outliers of the extracted joint positions are filtered out. Based on the remaining joint positions, the approximate curve of the organ can be obtained. It is possible that, the remaining joint positions after the global adaption may be from different images. In the original local adaption stage, the geometric model of the organ component is initialized according to the positions of the two adjacent joints of the component. If the two adjacent joint positions are from different images, a vector starting from one joint position to the other can be calculated. Based on the vector, an image containing the both joint positions can be interpolated from the original image data. Finally, the adaption of the geometric model is carried out based on the interpolated image.
A Appendix

A.1 Disk Position Extraction using Different Scan Machine and Protocol

In the evaluation of the spine curve extraction, all MR data are provided by the Universitätsklinikum Essen using the same scan protocol and MR scanner (SIEMENS 1.5 Tesla MR scanner). To show that the disk position extraction is capable of processing different medical image modalities. Two images, which are produced using an experimental pulse sequence and a GE MR scanner, are tested. These two images are provided by The University of Cincinnati College of Medicine. Due to the small MR scan range, each image is composed of two separated half-spine frames which result in a line at the middle of the image. Each frame is of size 350 mm × 350 mm (512 × 512 pixels). Fig. A.1 and Fig. A.2 show the intermediate and final disk position results of the two images, respectively.
Figure A.1: The disk position results of the MR data produced using different scan protocol and scanner. Top left: result of the region centers and directions; Top middle: result of the possible disk positions; Top right: result of the possible disk positions after the graph-based filter; Bottom left: result of the disk positions after the intensity profile; Bottom right: result of the final disk positions after the ASM-based filter. The green curve represents the best fitted spine curve model.
Figure A.2: The disk position results of the MR data produced using different scan protocol and scanner. Top left: result of the region centers and directions; Top middle: result of the possible disk positions; Top right: result of the possible disk positions after the graph-based filter; Bottom left: result of the disk positions after the intensity profile; Bottom right: result of the final disk positions after the ASM-based filter. The green curve represents the best fitted spine curve model.
Bibliography


122


Bibliography


