DIPLOMARBEIT

# Automatic detection of landmarks and axes at the human knee joint

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Diplom-Ingenieurs unter der Leitung von:

Assoc.Prof. Dipl.-Ing. Dr.techn. Dieter H. Pahr

Mitwirkend:

Univ.Ass. Dipl.-Ing. Dr.techn. Andrea Lorenz

E317
Institut für Leichtbau und Struktur-Biomechanik

eingereicht an der Technischen Universität Wien
**Fakultät für Maschinenwesen und Betriebswissenschaften**

von

Clemens Freidhager
1325793
Pülslgasse 19/13
1230 Wien

Wien, December 12, 2018

# Abstract

**Motivation:** Anatomical landmarks are of great importance for many medical fields. Especially at the knee joint, landmarks are used for orientation during surgery or to define axes. Consequently, knowledge of their exact location is crucial. Previous methods to detect these points are very time consuming or highly complex. In 2009, a fully automatic approach to detect peak landmarks based on surface curvatures was presented by Subburaj et al. This method was implemented and tested. In addition, the method was extended to detect different geometric structures and axes.

**Material and methods:** As the proposed method is based on curvature values of surface meshes, segmented computer tomography scans were used to obtain triangular meshes of 15 knees. A semi-automatic multi-step process was implemented to detect bony landmarks with the shape of a peak, based on curvature values and adjacency relationships. The implemented algorithm was extended to detect further geometrical shapes of the surface such as edges and valleys. This enabled the computation of the farthest points of the tibial plateau and the patella as well as the cylindrical axis and the path of the trochlear groove. In addition, the shaft axes of the femur and tibia were calculated. This algorithm was applied to all 15 specimens. The results obtained for the bony landmarks were compared with the landmarks labeled by an experienced orthopedic surgeon on 7 of those 15 specimens.

**Results:** Manual intervention was necessary for all specimens to allow the algorithm to detect bony landmarks on the surface. However, the algorithm was only able to detect all landmarks of 5 femura and 6 tibias. The variability in location of bony landmarks of the femur and tibia, compared to the landmarks labeled by the surgeon, were found to be in range of 1.90 to 6.96 mm. The computation of the farthest points of the patella as well as tibial plateau, the trochlear groove, and the cylindrical axis delivered adequate results, required manual interventions for roughly half of all specimens. The computation of the shaft axis of the femur and tibia did not require any interventions to get results.

**Discussion:** The methods for detecting landmarks or axes which were based on larger regions or an overall contour of the bone were relatively stable. In contrast, using curvature values for locating small peak landmarks required partly manual interventions and this strategy was not successful for all specimens. The landmarks found were in good agreement with the literature. In conclusion, semi-automatic detecting and labeling of anatomical landmarks and axes could be achieved with the implemented algorithms.

# Zusammenfassung

**Motivation:** Anatomische Landmarken sind für viele verschiedene Anwedungen in der Medizin von großer Bedeutung. Am Kniegelenk werden sie zur Orientierung während Operationen und um Achsen zu definieren verwendet. Deshalb ist die Kenntnis ihrer genauen Position sehr wichtig. Vorhandene Methoden haben die Nachteile, dass sie entweder sehr zeitaufwendig, oder sehr komplex sind. Deshalb hat Subburaj et al. 2009 einen Ansatz präsentiert der hügelförmige Landmarken voll automatisch basierend auf Krümmungswerten auffindet. Dieser Ansatz wurde implementiert und getestet. Zusätzlich wurde diese Methodik erweitert um auch andere geometrische Strukturen und Achsen auffinden zu können.

**Material und Methoden:** Da der präsentierte Ansatz auf Krümmungswerten von Oberflächenmeshes beruht, wurden segmentierten computer-tomographische scans verwendet um Dreieckmeshes von 15 Knien zu erzeugen. Ein semiautomatischer, mehrstufiger Prozess wurde implementiert um, basierend auf Krümmungswerten und Nachbarschaftsbeziehungen, knochige Landmarken in der Form eines Hügels detektieren zu können. Der implementierte Algorithmus wurde erweitert, um weitere geometrische Strukturen wie Kanten und Täler auffinden zu können. Dies macht es möglich die Punkte mit dem größten Abstand bei der Patella und des Tibiaplateaus, sowie die zylindrische Achse und den Pfad der Trochlear Groove am Femur aufzufinden. Zusätzlich wurden die Schaftachsen von Femur und Tibia berechnet. Diese Algorithmen wurden auf alle 15 Knie angewandt. Die erhaltenen Ergebnisse wurden mit markierten Landmarken eines erfahrenen orthopedischen Chirurgen für 7 von diesen 15 Knien verglichen.

**Ergebnisse:** Manuelle Eingriffe waren notwendig um knochige Landmarken detektieren zu können. Dennoch war der Algorithmus nur in der Lage an 5 Femuren und 5 Tibien alle Landmarken aufzufinde. Die Schwankungen der Position der knochigen Landmarken von Femur und Tibia, zwischen Punkten die der Chirurg markiert hat und dem Algorithmus waren zwischen 1.90 und 6.96 mm. Für die Berechnung der Punkte mit dem größten Abstand von der Patella und dem Tibiaplateau, sowie der zylindrischen Achse und der "trochlear groove" waren manuelle Eingriffe für in etwa die Hälfte aller Proben notwendig. Die Berechnung der Schaftachsen von Femur und Tibia benötigte keine Interventionen um Ergebnisse zu erhalten.

**Diskussion:** Abschließend lässt sich sagen, dass das Detektieren von Landmarken und Achsen basierend auf größeren Regionen oder überregionalen Konturen gut funktioniert. Das Auffinden von kleinen, lokalen Strukturen der Oberfläche, wie Erhebungen, hingegen funktionierte nicht zufriedenstellend und verlangte manuelles eingreifen. Die aufgefundenen Regionen waren jedoch übereinstimmend mit der Literatur. Schlussfolgend lässt sich sagen, dass halbautomatisches Detektieren und Kennzeichnen von anatomischen Landmarken und Achsen mit den implementierten Algorithmen erreicht werden konnte.

### Acknowledgements

I want to thank my supervisor Dr. Andrea Lorenz for the support and her commitment during this thesis. Furthermore, I want to thank Prof. Dr. Dieter H. Pahr for sharing his expertise concerning this subject and his inputs relating to this thesis.

Concluding I want to thank some friends of mine, Emin Kocbay, Dominic Rauscher and Markus Bardach, as well as my whole family and Hannah Bresenhuber who supported me in very different ways. Each of them helped me, consciously or unconsciously, during this thesis in their own way.

# Contents

# Acronyms

| | |
|---|---|
| A | Anterior direction |
| $\boldsymbol{A}$ | 1. matrix used to compute the center point $\boldsymbol{p}_0$ of a spherical fit |
| AT | Adductor magnus Tubercle |
| $\alpha_l$ | Angle between two successive edges, that share a vertex $\boldsymbol{v}_k$ |
| $\alpha_{\mathrm{r}}$ | Rotation angle corresponding to $\boldsymbol{e}_{\mathrm{r}}$ |
| $\beta_j$ | Dihedral angle at an edge $\boldsymbol{e}_j$ |
| $\boldsymbol{B}$ | 2. matrix used to compute the center point $\boldsymbol{p}_0$ of a spherical fit |
| $C$ | Any curve |
| $c$ | A closed regular planar curve |
| $c_\rho$ | Circular arc segment with radius $\rho$ |
| $c_{\mathrm{r}}$ | $\cos(\alpha_{\mathrm{rot}})$ |
| CBA | Curvature Based Algorithm |
| CGAL | Computational Geometry Algorithm Library |
| D | Distal direction |
| $d$ | Euclidean distance |
| $\delta_m$ | Weighting factors |
| $\Delta_{\mathrm{window},x}, \Delta_{\mathrm{window},y}, \Delta_{\mathrm{window},z}$ | Tolerance range |
| $\boldsymbol{e}_j$ | Edges of element $t_i$ |
| $\boldsymbol{e}_{\mathrm{r}}$ | Rotation axis |
| $e_{\mathrm{r},x}, e_{\mathrm{r},y}, e_{\mathrm{r},z}$ | Vector elements of rotation axis $\boldsymbol{e}_{\mathrm{r}}$ |
| $\mathcal{G}$ | Gauss map |
| $\mathcal{G}_p$ | Gauss map of any point $\boldsymbol{p}_i$ |
| $g(\gamma_0, \gamma_i)$ | Error function of linear regression |
| $\gamma_0$ and $\gamma_1$ | Parameters of linear regression |
| $H$ | Mean curvature |
| $H_p$ | Mean curvature of any point $\boldsymbol{p}_i$ |
| $^{\mathrm{L}}H$ | Limit of mean curvature |
| $^{\mathrm{L}}H_{\mathrm{Tg}}$ | Individual limit of mean curvature for detecting the trochlear groove |
| $^{\mathrm{C}}H_i$ | Computed alternate mean curvature |
| $^{\mathrm{E}}H_i$ | Gaussian curvature of element $t_i$ |
| $^{\mathrm{R}}H_m$ | Gaussian curvature of ring $m$ |
| $^{\mathrm{V}}\bar{H}_k$ | Integral mean curvature of vertex $\boldsymbol{v}_k$ |
| $^{\mathrm{V}}H_k$ | Mean curvature of vertex $\boldsymbol{v}_k$ |
| HU | Houndsfield Unit |
| $K$ | Gaussian curvature |
| $\bar{K}$ | Integral Gaussian curvature |
| $K_p$ | Gaussian curvature of any point $\boldsymbol{p}_i$ |
| $^{\mathrm{L}}K$ | Limit of Gaussian curvature |

| | |
|---|---|
| ${}^{\mathrm{L}}K_{\mathrm{Ft}}$ | Individual limit of Gaussian curvature for farthest points detection of the tibial plateau |
| ${}^{\mathrm{L}}K_{\mathrm{Fp}}$ | Individual limit of Gaussian curvature for detecting the farthest points of the patella |
| ${}^{\mathrm{L}}K_{\mathrm{Tg}}$ | Individual limit of Gaussian curvature for detecting the trochlear groove |
| ${}^{\mathrm{L}}K_{\mathrm{Ca}}$ | Individual limit of Gaussian curvature for detecting the cylindrical axis |
| $\bar{K}_{k,i}$ | Gaussian curvature located at the vertex $\boldsymbol{v}_k$ and the triangle $t_i$ |
| ${}^{\mathrm{C}}K_i$ | Computed alternate Gaussian curvature |
| ${}^{\mathrm{E}}K_i$ | Gaussian curvature of element $t_i$ |
| ${}^{\mathrm{R}}K_m$ | Gaussian curvature of ring $m$ |
| ${}^{\mathrm{V}}\bar{K}_k$ | Integral gaussian curvature of vertex $\boldsymbol{v}_k$ |
| ${}^{\mathrm{V}}K_k$ | Gaussian curvature of vertex $\boldsymbol{v}_k$ |
| $\kappa_1$ and $\kappa_2$ | Eigenvalues of Weingarten map $\mathcal{W}_p$ |
| L | Lateral direction |
| LE | Lateral epicondyle |
| LIT | Lateral intercondylar tubercle |
| LP | Lateral peak |
| M | Medial drection |
| ME | Medial epicondyle |
| MIT | Medial intercondylar tubercle |
| MP | Medial peak |
| $\boldsymbol{n}$ | Surface normal vector of element |
| $n_{\mathrm{df}}$ | Dot product factor of farthtest points of patella |
| ${}^{\mathrm{E}}n_k$ | Number of adjancent elements of vertex $\boldsymbol{v}_k$ |
| $n_{\mathrm{R}}$ | Region limit that defines the number of elements of a peak |
| $n_{\mathrm{p}}$ | Window size parameter of tibial plateau |
| $n_{\mathrm{s}}$ | Percentage parameter for computing a shaft axis |
| $n_{\mathrm{W}}$ | Window size of tolerance range |
| P | Proximal direction |
| $\boldsymbol{p}_i$ | Any point of surface $\mathcal{S}$ |
| $\boldsymbol{p}_{\mathrm{center}}$ | Center point of a landmark region |
| $\boldsymbol{p}_0$ | Center point of a sphere |
| $p_{\mathrm{center},x}, p_{\mathrm{center},y}, p_{\mathrm{center},z}$ | Coordinates of center point of landmark region |
| $p_{0,x}, p_{0,y}, p_{0,z}$ | Coordinates of center point of a sphere |
| $p_{i,x}, p_{i,y}, p_{i,z}$ | Coordinates of any point $\boldsymbol{p}_i$ |
| $\phi_1, \phi_2, \phi_3$ | Interior angles of an element $t_i$ |
| $\boldsymbol{\psi}_1$ and $\boldsymbol{\psi}_2$ | Corresponding eigenvectors to $\kappa_1$ and $\kappa_2$ |
| Px | Proximal direction |
| $R$ | Possible landmark region |
| $r$ | Radius of a sphere |
| $R_{\mathrm{o}}$ | Other possible landmark region, that meet given requirements |

| | |
|---|---|
| $R^3$ | Euclidean space |
| $\rho$ | Radius of circular arc segment |
| $\mathcal{S}$ | Any surface |
| $s_\mathrm{r}$ | $\sin(\alpha_\mathrm{rot})$ |
| $S^2$ | Unit sphere |
| $s^2$ | Variance |
| $s$ | Standard deviation |
| $t_i$ | Any element of a triangular mesh |
| $\tau$ | Arc length of regular planar curve $c$ |
| TKA | Total knee arthroplasty |
| TT | Tibial tuberosity |
| $\boldsymbol{u}$ | Initial vector of rotation |
| $\boldsymbol{v}_k$ | Vertex of element $t_i$ |
| $\bar{v}$ | Mean value of Euclidean norm of scattered data points |
| $v_{\mathrm{CBA},x}$,$v_{\mathrm{CBA},y}$, $v_{\mathrm{CBA},z}$ | Coordinates of vertex labeled by the CBA |
| $v_{\mathrm{Surgeon},x}$,$v_{\mathrm{Surgeon},y}$, $v_{\mathrm{Surgeon},z}$ | Coordinates of vertex labeled by the Surgeon |
| $\mathcal{W}$ | Weingarten map |
| $\mathcal{W}_p$ | Weingarten map of any point $\boldsymbol{p}_i$ |
| $\boldsymbol{w}$ | Goal vector of rotation |
| $y_v$ and $y_{v+1}$ | Connecten points of circular arc $c_\rho$ and edges |
| 3D | 3-Dimensional |
| 2D | 2-Dimensional |

# Chapter 1

# Introduction

## 1.1 Motivation

Anatomical landmarks are of great importance in many medical fields and knowledge of their position is essential for various applications. Mosby's medical dictionary [1] defines the terminus landmark as "a readily recognizable anatomical structure used as a point of reference in establishing the location of another structure or in determining certain measurements" and a bony landmark as "a groove or prominence on a bone that serves as a guide to the location of other body structures". This means a bony landmark shows a characteristic geometric shape, like a peak, a pit, etc.

The importance of anatomical landmarks on the entire skeleton of the human body is manifold. For example, they can be used to label attachment points of ligaments or muscles, or to define mechanical joint axes [2, 3]. Consequently, landmarks are used for intra-operative navigation, which is why they are very important for operation planning [4, 5, 6, 7], for designing and adapting prosthesis and implants, and for measuring characteristic distances on bones [8, 1]. Especially, on the knee joint bony landmarks are for positioning prosthetic components, for performance of a computer assisted knee arthroplasty (TKA) and to define a reference system during motion [9, 10, 11]. In addition, variability in the location of landmarks has the potential to affect the joint coordinate systems and reported kinematic descriptions.[12].

Landmarks are used in different medical fields for different applications. Therefore, for many bones and joints, including the human knee joint, it is important to know their respective position. Finding such landmarks often requires manual intervention of experienced staff and is time consuming. For this reason the objective of this diploma thesis is to implement and to extend an existing automatic algorithm to detect and label such characteristic points.

## 1.2 The human knee joint

The human knee is a highly complex bicondylar joint and the largest joint in the human body. It consists of three bones the femur, tibia, and patella (figure 1.1). The femur, and tibia form the femorotibial joint. The femur and patella form the femoropatellar joint. Both joints are enclosed by one joint capsule, which is why the per se two joints are understood as the one knee joint. The fibula, the bone parallel to the tibia, is not part of the actual knee joint, but forms an autonomous joint, the tibiofibular joint, with the tibia [13, 14]. The two condyles of femur and tibia are round prominences. The epicondyles are eminences of the condyles. The part of the femur which is part of the knee joint is called distal end, and the part of the tibia which is part of the knee joint is called proximal end (figure 1.2) [3].

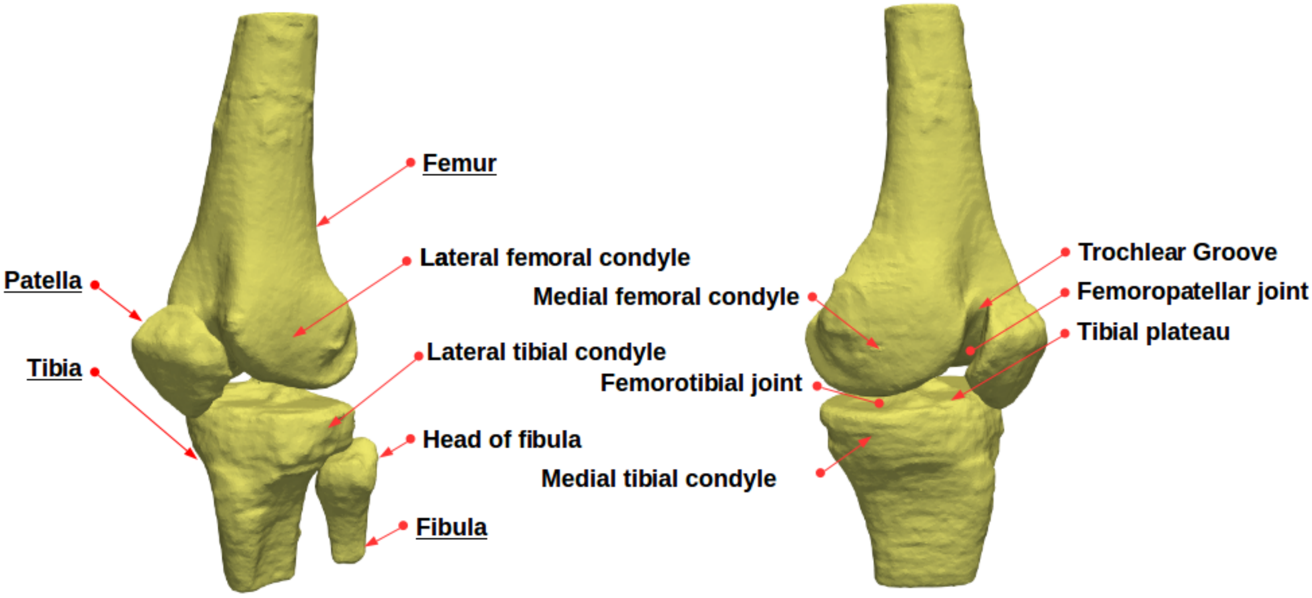


Figure 1.1: Bones forming the human knee joint.

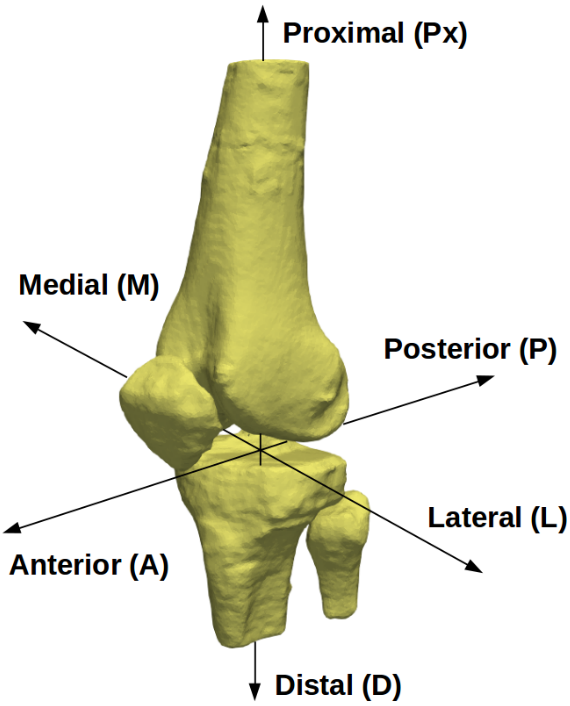


Figure 1.2: Anatomical directional terms shown on a left knee joint.

The femur and tibia are connected on the medial side directly through the medial collateral ligament and on the lateral side through the lateral collateral ligament (figure 1.3). Proximally those ligaments are attached to the epicondyles of the femur and distally to the medial condyle of the tibia and the fibular head. The cruciate ligaments are arranged like the letter X and connect the femur and the tibia as well. These ligaments keep the articular surfaces of the femur and the tibia in contact and provide stability to the knee joint, during motion. The patellar tendon connects the patella with the tuberosity of the tibia. The tibia tuberosity is considered to be the insertion of the quadriceps, which is directly connected to the proximal end of the patella and the femur, and is mainly responsible for extending the leg [3]. The hamstrings participates in the flexion of the knee joint, inserting on the tibia and fibula. The adductors are a muscle group that is attached to the femur and tibia. Their insertion occurs, among others, on the adductor magnus tubercle, whereas this muscle is mainly responsible for movements of the hip. The gastrocnemius is located for the most part at the posterior side of the lower leg, whereas the origin is on the medial and lateral condyle of the femur. It participates in the flexion of the knee joint as well (figure 1.3) [13].

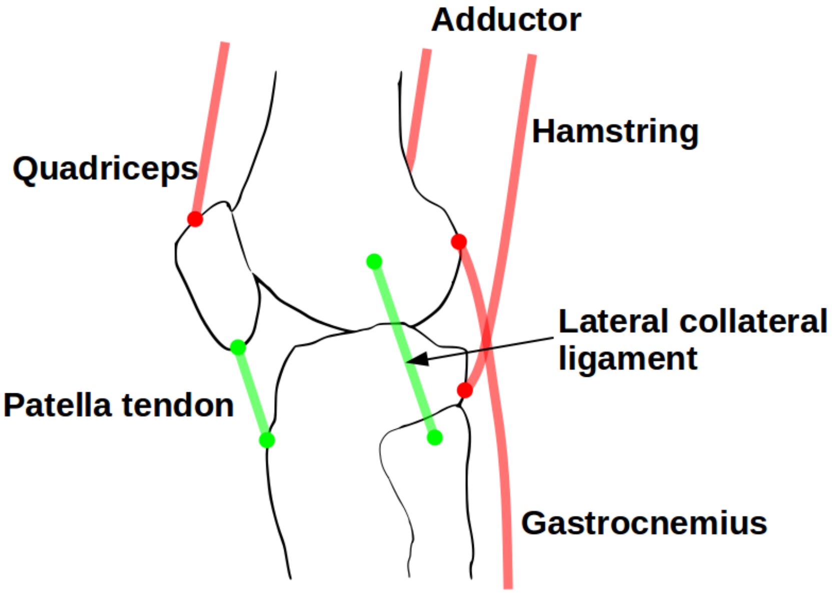


Figure 1.3: Simplified drawing of muscles and ligaments of the human knee joint.

The movement of the human knee joint is highly complex [3]. The maximum flexion of the human knee is about 150 degrees, as the center moves up and down [13]. The movement is rather a combination of rolling and sliding of the individual joint bones [15]. When knee flexion reaches its maximum, there is a secondary movement, the obligatory terminal rotation, where the tibia rotates internally by approximately 5 degrees [3].

Knowing and understanding the importance of landmarks in general and the complexity of the human knee leads to the question of how they interlink. Therefore, the next chapter investigates landmarks on the human knee.

## 1.3 Landmarks and axes at the human knee

At the human knee joint there are several anatomical landmarks with different meaning and appearance. In this work, only a selection of all landmarks on the knee are of interest and therefore listed below. They were chosen because they can all be found through a fundamentally consistent characteristic geometric structure. For example, all listed anatomical landmarks correspond to a peak.

### 1.3.1 Femur

Five different bony landmarks at the femur are of interest regarding this study, (table 1.1), which are depicted in figure 1.4.

| Landmark | Function[3, 14, 16, 17, 18] |
|---|---|
| Adductor magnus tubercle | • Insertion of adductor magnus |
| Medial epicondyle | • Attachment point of medial collateral ligament |
| Lateral epicondyle | • Attachment point of lateral collateral ligament |
| Medial peak | • Most anterior point of medial condyle |
| Lateral peak | • Most anterior point of lateral condyle |

Table 1.1: Bony landmarks at the distal end of the femur and their function (figure 1.4)

In addition, the characteristic structure of the trochlear groove is of interest (table 1.2).

| Characteristic structure | Function |
|---|---|
| Trochlear groove | • Concave notch in which patella moves along the femur during flexion and extension [19] |

Table 1.2: Characteristic structure at the distal and of the femur

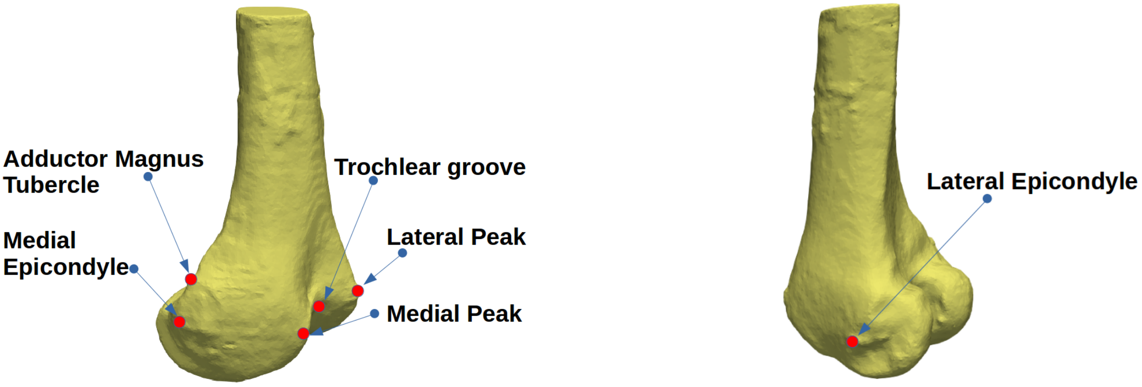


Figure 1.4: Femoral bony landmarks: Adductor magnus tubercle, lateral peak, medial peak, medial epicondyle, lateral epicondyle

Three different axes at the femur are of interest regarding this study (table 1.3), which are depicted in figure 1.5.

| Axis | Function[20, 21, 22, 23] |
|---|---|
| Cylindrical axis | • Axis of cylinder registered in posterior medial an lateral condyles [20] |
| Shaft axis | • Femoral proximal shaft axis [23]. The actual definition depends on what the axis is used for. |
| Epicondylar axis | • Connecting line of medial and lateral cpicondyles [20] |

Table 1.3: Axes of the distal end of the femur and their characteristics (figure 1.5)

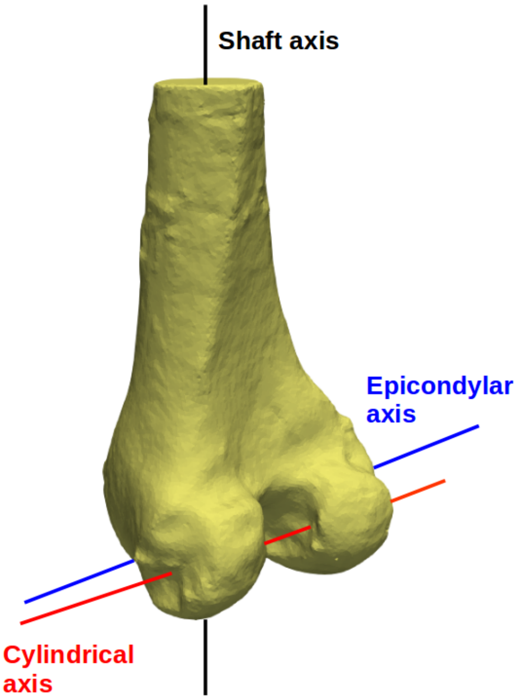


Figure 1.5: Shaft axis, cylindrical axis, epicondylar axis of the femur of the left knee joint.

### 1.3.2 Tibia

Concerning this study, there are also five different bony landmarks of interest at the tibia (table 1.4), which are depicted in figure 1.6.

| Landmark | Function[3, 14, 24, 25] |
|---|---|
| Medial intercondylar tubercle | • Medial peak of eminence intercondylaris |
| Lateral intercondylar tubercle | • Lateral peak of eminence intercondylaris |
| Medial peak | • Most medial point of tibial plateau |
| Lateral peak | • Most lateral point of tibial plateau |
| Tibial tuberosity | • Attachment of patellar tendon (considered insertion of quadriceps) |

Table 1.4: Bony landmarks at the proximal end of the tibia and their function (figure 1.6)

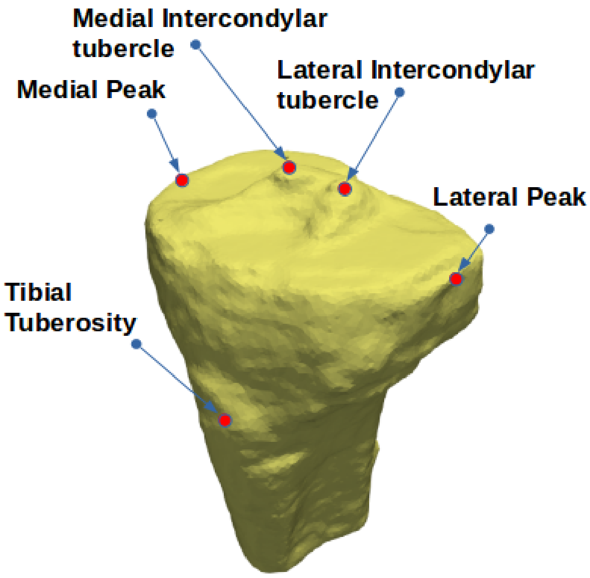


Figure 1.6: Tibial bony landmarks: Tibial tuberosity, lateral peak, medial peak, lateral intercondylar tubercle, medial intercondylar tubercle

One axis at the tibia is of interest for this study (table 1.5), which is depicted in figure 1.7.

| Axis | Function[26] |
|---|---|
| Shaft axis | • Axis located in the shaft of the Tibia [26]. The actual definition depends on what the axis is used for. |

Table 1.5: Axis of the distal end of the tibia and their characteristics (figure 1.7)

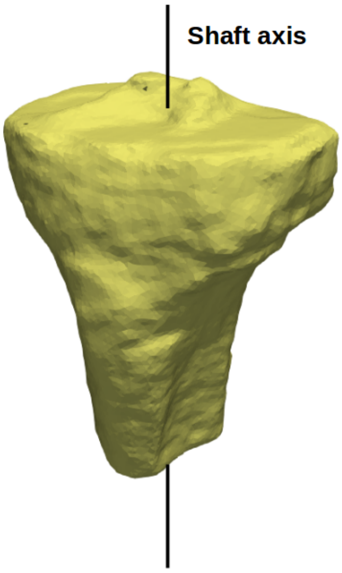


Figure 1.7: Shaft axis of the tibia of a left knee

The farthest points of the tibial plateau are defined by the largest distance between two points. This is another approach of finding the lateral and medial peaks of the tibial plateau, which are explained in table 1.4. (figure 1.8).

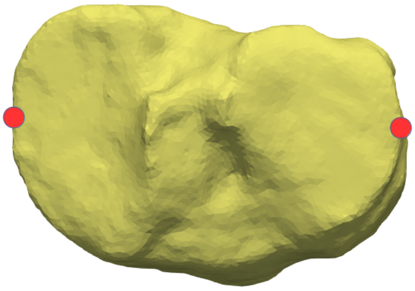


Figure 1.8: The farthest points of the tibia are defined by the distance between two points located at the edge of the tibial plateau.

### 1.3.3 Patella

Considering the farthest points the most proximal, most distal, most medial and most lateral of the anterior side of the patella are of interest, regarding this study (figure 1.9). These are determined by the largest distance between four points of the outer contour of the bone. The lines connecting two of those points must be orthogonal to each other.

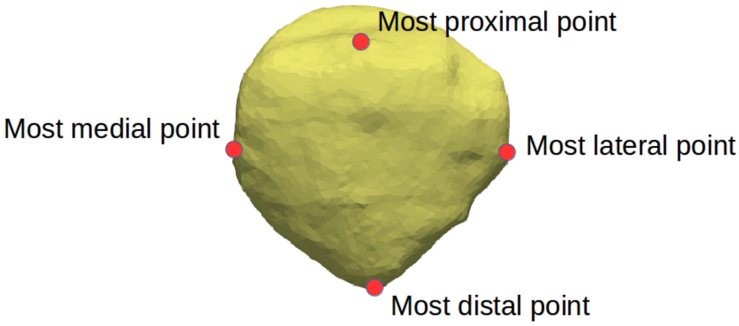


Figure 1.9: The farthest points of the patella are defined by the largest distance between two points of the outer contour of the bone.

## 1.4 State of the art of detecting landmarks

Currently, many different approaches for locating bony landmarks are known, which in principle can be assigned to two different methods. Either landmarks are determined on the actual body part, or alternatively based on images or models of the bone. However, there are also procedures that use both methods. Van Sint Jan [27] published a color atlas of skeletal landmarks and defined an approach for detecting them on a patient by palpation. Based on 3D foot and leg scans, Liu et al. [29] presented a method to extract landmarks by using principal curvatures of the skin. These are used to describe certain shapes of the body surface that are connected to bony landmarks. Those approaches are based on the actual body part, whereat the approach of Liu et al. can be seen as an approach that relies on actual body parts and images, as well. Approaches based on images and models can be broken down into methods that require only 2D or 3D medical images, and methods that need 3D models in the form of a mesh. In 2000, Griffin et al. [16] used magnetic resonance imaging to define landmarks in single pictures by manual examination, to show the variability of certain measurements of the distal femur. In 2005, Wörz [31] used similar images for a fully automatic approach to find three different predefined structures on the surface, by fitting shapes onto defined regions of interest. In 2013, Baek et al. [32] used surface meshes of the femur to create a mean model with labeled points, registered with another mesh to automatically find landmarks. To create such mean models, a rather extensive database is required. Using femoral surface meshes as well, in 2015 Yang et al. [33] presented an approach that uses complex convolutional neural networks in combination with curvature values to label anatomical landmarks fully automatically. Forty samples were used to train the neural network and ten to test the algorithm.

Most of these methods have at least one of the following disadvantages, that their results tend to vary, hat they are time consuming, or that they are relatively complex. Furthermore, many of those approaches cannot be performed fully automatically and need manual intervention. In 2009, Subburaj et al. [24] presented a systematic approach that uses curvature values and adjacency relationship between landmarks to automatically identify landmarks of 3D models of the knee joint. The presented algorithm has the advantage that it does not vary and that it is supposed to work fully automatically, without using highly complex approaches as neural networks. However, the results of only three different specimens were shown. During the presentation of the approach within the paper, some important steps are not adequate explained, so that some points are not fully comprehensible.

Furthermore, Subburaj et al. [24] focus entirely on landmarks that have the form of a peak. Curvature values are expected to be serviceable in locating features that are not exclusively in that particular form. The adaptation of the proposed approach therefore seems to have the potential to automatically find additional characteristics through the use of curvature values.

## 1.5 Objective of this Thesis

The following scientific question sets the objectives of the diploma thesis:

> Can information stored in surface meshes, and curvature values be used to detect different landmarks and axes fully automatically at the human knee joint?

The aim of this diploma thesis is to implement and improve the Curvature Based Algorithm [24] and to test its robustness and validate the results, by comparing to a manual detection of an experienced surgeon. Furthermore, the curvatures are used and combined with other approaches to detect further characteristics and axes of the knee joint. The following subgoals can be defined.

- Create 3D models and meshes from CT-Images.
- Implement and improve Curvature Based Algorithm (CBA) [24] to detect described landmarks.
- Test robustness of CBA.
- Validate created results by comparing them to landmarks detected by an experienced orthopedic surgeon.
- Extend algorithm so it is able to detect:
    1. Farthest points of Tibia
    2. Farthest points of Patella
    3. Trochlear Groove of Femur
    4. Cylindrical axes of Femur
    5. Shaft axis of Femur and Tibia

# Chapter 2

# Material and Methods

As a first step, a segmentation process needs to be developed (figure 2.1). Then, the Curvature Based Algorithm (CBA) is implemented and extended, so that axes and landmarks can be localized. As a final step the received results have to be validated by comparing them to landmarks detected by an experienced orthopedic surgeon.

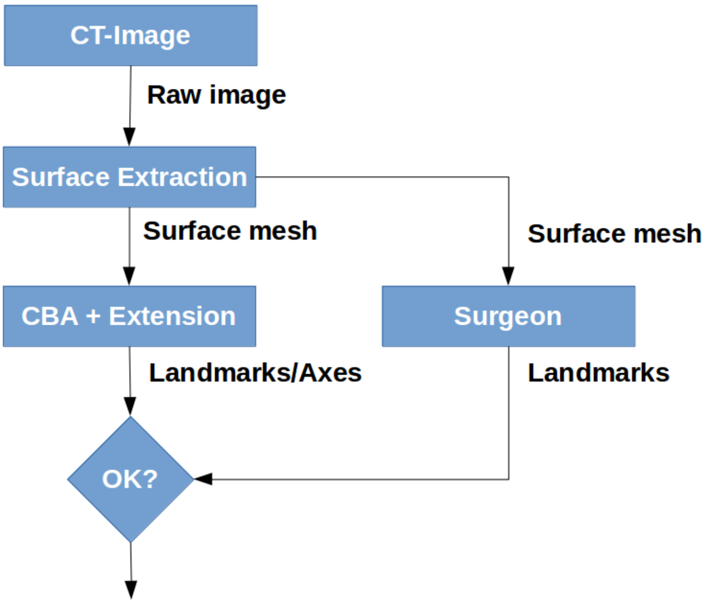


Figure 2.1: Logical flow of diploma thesis

## 2.1 Materials

The basis for creating the required meshes are 15 sets of CT-Images of left human cadaver knee specimens (figure 2.2). Those images were taken at the University Hospital Tübingen, in Tübingen and specially prepared for biomechanical experiments using a special knee simulator [34, 35]. Therefore, femur and tibia were cut approximately 15 cm from the joint line and the fibula was screwed to the tibia with cortical screws. Furthermore, the femur and tibia were fixed in aluminum cylinders and provided with screws as reference points. Those screws and other metallic objects cause strong artifacts within the CT-Images, making the contour of the bones difficult to see. The resolution of the images slightly differs for each specimen, but is about 0.24 mm, 0.24 mm, 0.6 mm. They are stored as dicom files, whereas each dicom file represents a single image. Gray values scaled in Hounsfield Unit (HU) are stored for each voxel [36], in these images.

Figure 2.2: CT-Image of a left knee joint with artifacts due to clips and screws of metal, stored in a dicom file.

In addition, after creating surface meshes an experienced orthopedic surgeon labeled anatomical landmarks (chapter 1.3) on 7 of these 15 specimens (figure 2.3). For one of those 7 specimens each landmark (chapter 1.3) was labeled eight times by the same orthopedic surgeon.

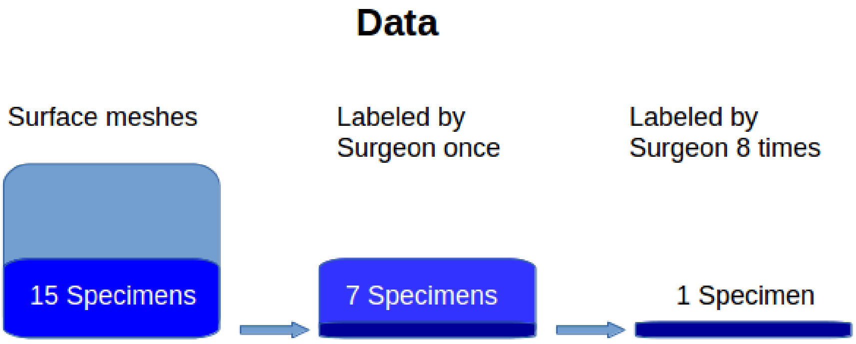


Figure 2.3: Overview existing data for diploma thesis.

## 2.2 Creating 3D models based on CT-Images

The whole segmentation process is subdivided into three main steps, whereas the final results are surface meshes (figure 2.4). The first step prepares the input data, for the actual segmentation. During the second step, the single bones are segmented, and in a final third step, 3D models are created.

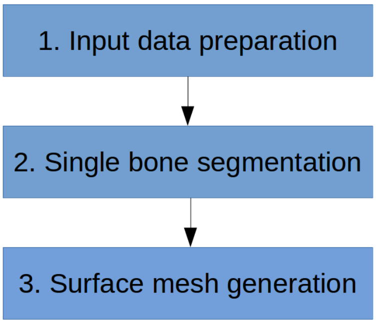


Figure 2.4: Process of creating surface meshes based on CT-Images.

For the implementation of the segmentation process the software medtool 4.2 (Dr. Pahr Ingenieurs e.U., Pfaffstätten, Austria) is used in each of the following steps. Furthermore the free and open source software 3D Slicer 4.8 is applied additionally in some steps. If the 3D Slicer software was used, this is mentioned within the step. This process is applied for each knee.

1. **Input data preparation**
   In the first main step, the original input data of the whole knee is prepared for the actual segmentation.

   1.1. **Dicom file convertion**
   Dicom files are converted into mhd and raw files for further calculation. The mhd file is a header file that contains basic information, whereas the actual image data is stored within the raw file.

   1.2. **Midplanes generation**
   Midplanes for all three planes (x-y, y-z, x-z) are created (figure 2.5). A midplane is an image taken in the geometric center of the image and used for visual inspection.

   1.3. **Image cropping**
   The image is cropped to save storage space, by keeping only the bones in the image. (The midplanes created in the previous step are required to define the cropping area.)

   1.4. **Voxel size refining**
   The voxel size is refined to a resolution of 0.6 mm, 0.6 mm, 0.6 mm so that it is the same for each direction and specimen.

   1.5. **Image scaling**
   The gray values are scaled between 0 and 255 in order to store them in one byte.

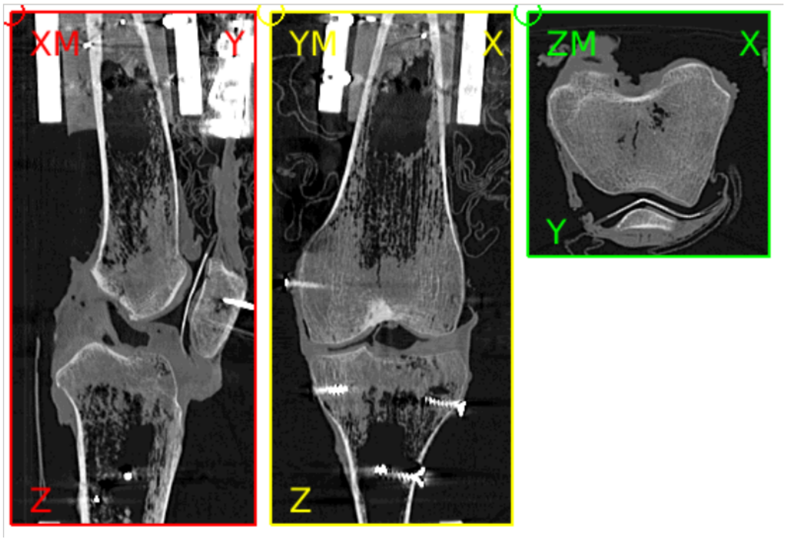


Figure 2.5: Midplanes created in step 1.5 Image scaling.

2. **Single bone segmentation**
   The following steps have to be executed for each of the four bones (femur, tibia, patella and fibula).

   2.1. **Presegmentation**
   The presegmentation is done by using the 3D Slicer Software. First, 5-10 images are manually segmented in each direction (x, y, z). Then, the command "fill between slices" is used to create a complete rough presegmented mask (figure 2.6). This method fills the skipped slices by interpolating between segmented slices. A mask is a binary image containing only 1 and 0. Based on this mask an image is created with gray values between 0 and 255, whereby all voxels outside the mask have the gray value 0.

Figure 2.6: Rough presegmented mask of a femur.

2.2. **Thresholding**
The created image is thresholded by defining a gray value limit, where only the voxels with a higher gray value remain. Therefore, the threshold is determined within an iterative process for each bone. The limit must be visually selected so that the contour of the bone is maintained as much as possible (figure 2.7). To prevent the bone in the thresholded image from being hollow after filling up holes, an end cap is attached to the cut end of the bone.

Figure 2.7: Tresholded image.

2.3. **Hole and cavity closing**
A morphological filter (table 2.1) is computed to close possible holes and cavities of the image (figure 2.7). Therefore, the bone specific threshold, determined in step 2.2 is used. Afterwards, the end cap attached in the previous step is removed again.

| **Parameter** | **Adjustment** |
|---|---|
| Threshold | Bone specific of step 2.2 |
| valid | 5 |
| type | out |
| Kernel | 1 |

Table 2.1: Used settings for the fill filter.

2.4. **Final Mask generation**
The image received of step 2.3 us used to generate a final mask, by using the 3D Slicer Software. if the mask differs from the contour of the original bone, it must be manually edited by adding or removing parts by using the 3D Slicer Software. In order to prevent new small holes and branches that might have been created during this fine segmentation process (figure 2.8), two morphological filters (table 2.2 & 2.3) are applied. By performing an opening filter (dilation + erosion) and a closing filter (erosion + dilatation), the surface is smoothed depending on the shape of the kernel and the chosen

radius. Both morphological filters use a spherical kernel (shape = 2) with a radius of one voxel. Finally, a smoothed final mask is obtained. (figure 2.9).

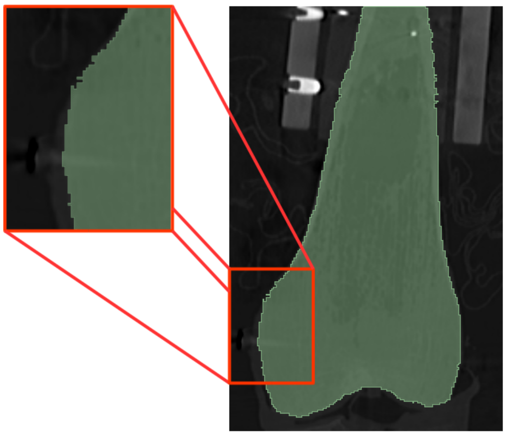


Figure 2.8: Final mask before applying the final morphological filters.

1. Filter - Open:

| Parameter | Adjustment |
|---|---|
| Radius | 1 |
| type | o |
| shape | 2 |
| threshold | 1 |

Table 2.2: Settings of the morphological opening filter.

2. Filter - Close:

| Parameter | Adjustment |
|---|---|
| Radius | 1 |
| type | c |
| shape | 2 |
| threshold | 1 |

Table 2.3: Settings of the morphological closing filter.

The received final mask (figure 2.9) can be used to create a 3D model in form of a mesh.

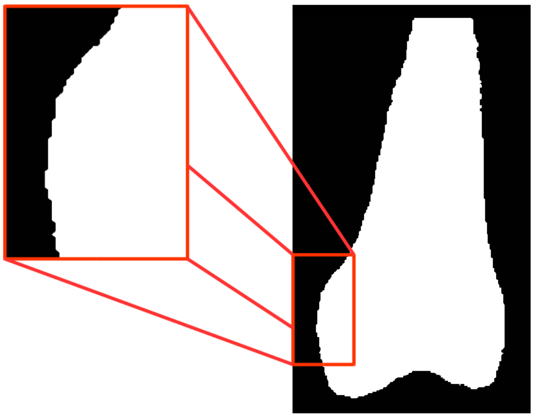


Figure 2.9: Final binary mask of a femur.

3. **Final mesh generation**
   A triangular surface mesh is an approximation of a surface, consisting of nodes and elements, with each element having three nodes. The size of those finite elements, used for discretize a structure, defines the number of points after the simplification and therefore, how much the original object is approximated. By creating a surface mesh, the number of nodes belonging to the surface, are reduced from an infinite amount to a finite one. This allows to compute differential geometry on every point of a surface and therefore forms the basis for the subsequent algorithms. In the differential geometry of meshes, nodes are often referred to as vertices.

   3.1. **Mesh generation**
   A surface mesh is computed (figure 2.10) using the Computational Geometry Open Source Algorithm Library (CGAL) included in medtool 4.2, based on the final mask of each bone. By creating 2D surface meshes, a complete 3D mesh is generated first, and in a second step all elements and nodes which are not part of the surface are deleted. After testing different settings, the following parameters for the CGAL3D mesher were used (table 2.4).

| **Parameter** | **Adjustment** |
|---|---|
| Cell size | 1 |
| Facet distance | 0.6 |
| edge size | 0.5 |

Table 2.4: Settings of the CGAL3D mesher for all bones.

The mesh is stored as inp file, containing the coordinates of the nodes and how the nodes are connected with the elements. Such inp files can be used as input for finite element simulations using ABAQUS of Dassault Systems.

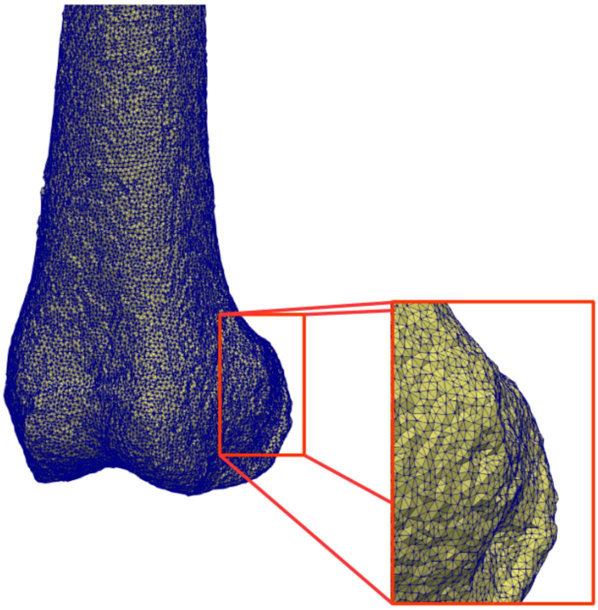


Figure 2.10: Mesh of specimen 15 without smooting

### 3.2. Surface smoothing

To obtain a clean surface without artifacts, Taubin smoothing filters [40], included in medtool 4.2, are used. Local, sharp-edged bumps of the surface mesh are often generated by the segmentation process and not part of the original surface. A Taubin smoothing filter is a linear low-pass filter that removes high curvature variations.

In particular, concerning the smoothing filter, there is a risk that the original contour will be deformed, which means a distortion of the surface curvature. This would be very problematic because the following algorithms are based on surface curvature values. Therefore, a parameter study was performed to understand the effects of all input parameters of the Taubin smoothing filter and to find the best settings (table 2.5).

| **Parameter** | **Adjustment** |
|---|---|
| Iteration steps | 50 |
| Scaling factor $\lambda$ | 0.2 |
| Pass-band frequency | 0.35 $k_{PB}$ |

Table 2.5: Settings of Taubin smoothing filter for all meshes.

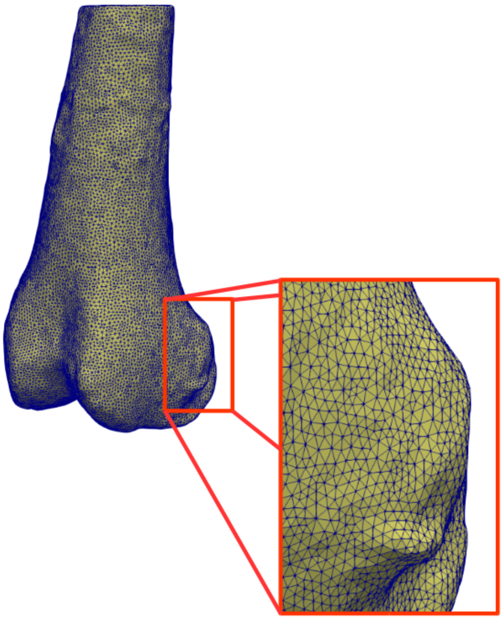


Figure 2.11: Final mesh of specimen 15 with smoothed surface.

The described segmentation and meshing process was applied to the scans of all 15 available knees.

## 2.3 Curvature Based Algorithm (CBA) for landmark detection

The basic idea of locating anatomical landmarks based on curvature values and adjacency relationships between landmarks was published by K. Subburaj et al. in 2009 [24]. The use of the mathematical signs of curvatures provides the ability to detect different geometric shapes and structures on the surfaces (figure 2.12). Applying this approach to the mesh of a bone results in a list of significant regions, all of which have the geometric potential to be an actual landmark. For labeling anatomical landmarks an iterative process is applied using relative positions of those significant regions and the information of relative position of landmarks [24].

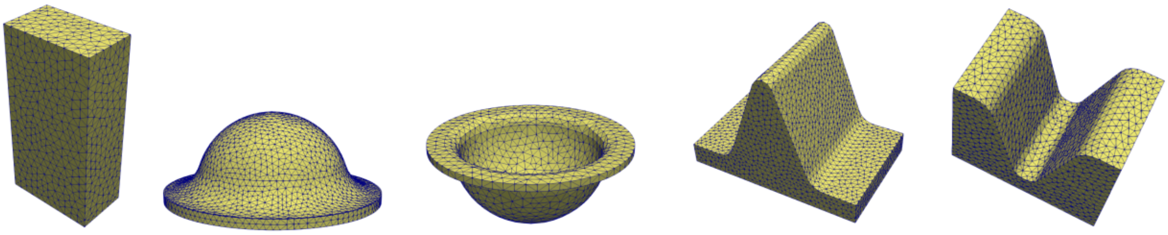


Figure 2.12: Geometric shapes identifiable by the mathematical sign of curvatures.

In general, the calculation of curvatures works for any desired surface, but the extraction and labeling process are specially designed for the knee joint.

### 2.3.1 Curvature computation

The computation of curvatures is the basis for the whole landmark detection process. Do Carmo [41] defines the curvature of a curve $C$ as the rate of change of the tangent line to $C$. For surfaces, this means that the curvature describes the rate of change of a surface to the respective tangent plane.

#### 2.3.1.1 Theoretical background

For each point $\boldsymbol{p}_i \in \mathcal{S}$ (figure 2.13), an associate unit normal vector $\boldsymbol{n}_i$ in Euclidean space $R^3$ exists. Using the Gauss map $\mathcal{G}$, a surface $\mathcal{S}$ and the corresponding normal vector $\boldsymbol{n}_i$ in euclidean space $R^3$ can be mapped to the unit sphere $S^2$ [41, 42].

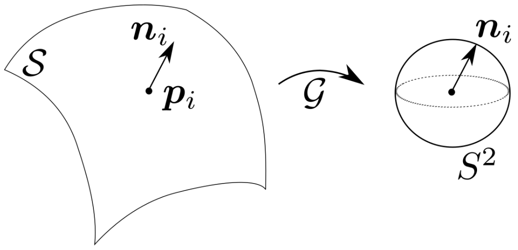


Figure 2.13: Gauss map $\mathcal{G}$ for a point $\boldsymbol{p}_i \in \mathcal{S}$ with its corresponding normal vector $\boldsymbol{n}_i$ (adapted from [42]).

The rate at which $\boldsymbol{n}_i$ changes across the surface $\mathcal{S}$ is measured by the derivative of $\mathcal{G}$ with respect to $\mathcal{S}$ and is known as the Weingarten map $\mathcal{W}$ [42]. Let $\mathcal{G}_p$ be the Gauss Map and $\mathcal{W}_p$ the Weingarten map of point $\boldsymbol{p}_i$.

The Gaussian curvature $K_p$ and the mean curvature $H_p$ of $\boldsymbol{p}_i \in \mathcal{S}$ are defined by:

$$K_p = \det(\mathcal{W}_p), \tag{2.1}$$

and

$$H_p = \frac{1}{2}\,\mathrm{trace}(\mathcal{W}_p). \tag{2.2}$$

The Eigenvalues of $\mathcal{W}_p$, $\kappa_1$ and $\kappa_2$, are called principal curvatures and $\boldsymbol{\psi}_1$ and $\boldsymbol{\psi}_2$, which are the corresponding eigenvectors, are called principal vectors. The principal curvatures $\kappa_1$ and $\kappa_2$ for a point $\boldsymbol{p}_i \in \mathcal{S}$ are the maximum and minimum values of the curvature [42]. The principal vectors are used as a basis for the Weingarten map

$$\mathcal{W}_p(\boldsymbol{\psi}_1, \boldsymbol{\psi}_2) = \begin{pmatrix} \kappa_1 & 0 \\ 0 & \kappa_2 \end{pmatrix}. \tag{2.3}$$

Considering equations 2.1 and 2.2, the Gaussian curvature $K_p$ and mean curvature $H_p$ can now be calculated using principal curvatures

$$K_p = \kappa_1\,\kappa_2, \tag{2.4}$$

and

$$H_p = \frac{1}{2}(\kappa_1 + \kappa_2). \tag{2.5}$$

#### 2.3.1.2 Curvatures of a mesh

Considering a triangular mesh the problem with computing curvatures of a surface mesh becomes obvious. The discretized surface consists of vertices which are part of elements representing the surface. These elements as such are completely flat and therefore without any curvature. To calculate curvature values anyways, the Gauss-Bonnet Theorem is used [41, 43]. Looking at an element $t_i$ (figure 2.14) on any surface, the Gauss-Bonnet Theorem states that the difference between $\pi$ and the sum of the interior angles $\phi_1, \phi_2, \phi_3$ is equal to the integral of the Gaussian curvature $K$ over $t_i$

$$\sum_{i=1}^{3} \phi_i - \pi = \iint_{\mathrm{t}_i} K dA. \tag{2.6}$$

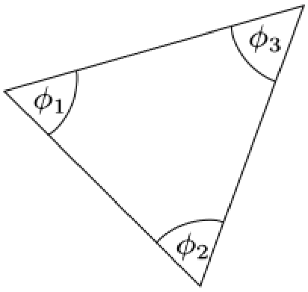


Figure 2.14: Triangle $t_i$ with its interior angles $\phi_1, \phi_2, \phi_3$ (adapted from [41]).

A triangular mesh consists of a set of elements $t_i$, each consisting of 3 edges $\boldsymbol{e}_j$ and 3 vertices $\boldsymbol{v}_k$ (figure 2.15) [43].

An edge $\boldsymbol{e}_j$ is defined by two vertices

$$\boldsymbol{e}_j = \boldsymbol{v}_{k+1} - \boldsymbol{v}_{\boldsymbol{k}}. \tag{2.7}$$

The angle $\alpha_l$ between two successive edges, which share the vertex $\boldsymbol{v}_{\boldsymbol{k}}$, is define as

$$\alpha_l = \angle(\boldsymbol{e}_j, \boldsymbol{e}_{j+1}). \tag{2.8}$$

For the triangle $t_i = \triangle(\boldsymbol{v}_k, \boldsymbol{v}_{k+1}, \boldsymbol{v}_{k+2})$ a surface normal vector $\boldsymbol{n}_{\boldsymbol{i}}$ can be defined as

$$\boldsymbol{n}_i = \frac{\boldsymbol{e}_j \times \boldsymbol{e}_{j+1}}{||\boldsymbol{e}_j \times \boldsymbol{e}_{j+1}||}. \tag{2.9}$$

The dihedral angle $\beta_j$ at an edge $\boldsymbol{e}_j$ which is the angle between the surface normals of the adjacent elements

$$\beta_j = \angle(\boldsymbol{n}_i, \boldsymbol{n}_{i+1}). \tag{2.10}$$

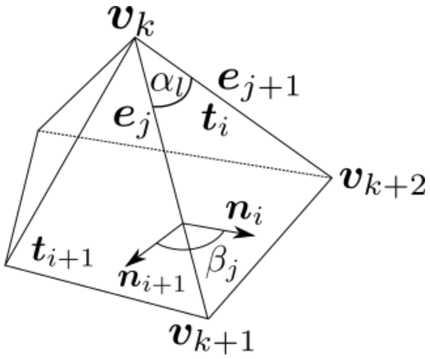


Figure 2.15: Vertex $\boldsymbol{v}_k$ and adjacent elements on a small part of a surface (adapted from [43])

For the general calculation of the integral Gaussian curvature $\bar{K}$, we consider the curvature $\kappa$ of a closed regular planar curve $c$, parameterized with respect to its arc length $\tau$ [44]. If the curve is simple, then the integral curvature $\bar{K}$ of $c$ becomes equal to

$$\bar{K} = \int_c \kappa(\tau) d\tau = 2\pi. \tag{2.11}$$

Based on this consideration the sharp corner at each vertex $\boldsymbol{v}_k$ is now replaced by a circular arc segment with radius $\rho$. The arc segment joins the adjacent edges tangentially (figure 2.16) to obtain a smooth, closed and simple curve $c_\rho$ [43].

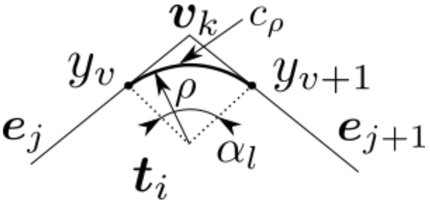


Figure 2.16: Sharp edge of $\boldsymbol{v}_k$ with insert arc segment $c_\rho$ (Adapted from [44])

Now, the integral Gaussian curvature $\bar{K}_{k,i}$ located at the vertex $\boldsymbol{v}_k$ and the triangle $t_i$ can be calculated as

$$\bar{K}_{k,i} = \int_{y_{v-1}}^{y_v} \kappa(\tau) d\tau = \pi - \alpha_i. \tag{2.12}$$

Where $y_v$ and $y_{v+1}$ mark the points where the circular arc touches the edges of the elements. Considering all elements, the vertex $\boldsymbol{v}_k$ is part of, leads to the total integral Gaussian curvature ${}^{\mathrm{V}}\bar{K}_k$ of this vertex

$${}^{\mathrm{V}}\bar{K}_k = 2\pi - \sum_{i=1}^{{}^{\mathrm{E}}n_k} \alpha_l. \tag{2.13}$$

Where $\alpha_l$ is the angle between two connected edges $\boldsymbol{e}_j$ and $\boldsymbol{e}_{j+1}$ that share the vertex $\boldsymbol{v}_k$ (figure 2.15). ${}^{\mathrm{E}}n_k$ is the number of adjacent elements to the vertex $\boldsymbol{v}_k$ [44, 43]. To receive the actual Gaussian curvature ${}^{\mathrm{V}}K_k$ of a vertex $\boldsymbol{v}_k$ from the integral curvature ${}^{\mathrm{V}}\bar{K}_k$ we assume the curvatures to be uniformly distributed around the considered vertex [45]. For this purpose, a scaling factor is used that divides the curvature equally between all adjacent elements [24, 46, 47]

$${}^{\mathrm{V}}K_k = \frac{{}^{\mathrm{V}}\bar{K}_k}{\frac{1}{3}{}^{\mathrm{E}}n_k} = \frac{2\pi - \sum_{i=1}^{{}^{\mathrm{E}}n_k} \alpha_l}{\frac{1}{3}{}^{\mathrm{E}}n_k}. \tag{2.14}$$

The same approach can be used to compute the total integral mean curvature ${}^{\mathrm{V}}\bar{H}_k$ of the vertex $\boldsymbol{v}_k$ [45, 47, 24]

$${}^{\mathrm{V}}\bar{H}_k = \frac{1}{4} \sum_{j=1}^{{}^{\mathrm{E}}n_k} ||\boldsymbol{e}_j||\,|\beta_j|. \tag{2.15}$$

Where $\beta_j$ is the dihedral angle associated to an edge (figure 2.15) and $||\boldsymbol{e}_j||$ is the length of an edge $\boldsymbol{e}_j$. To receive the actual mean curvature value from the integral mean curvature

${}^{\mathrm{V}}\bar{H}_k$ of a vertex, the same assumption as already explained for the Gaussian curvature is used [45, 47]

$$ {}^{\mathrm{V}}H_k = \frac{{}^{\mathrm{V}}\bar{H}_k}{\frac{1}{3}\,{}^{\mathrm{E}}n_k} = \frac{\frac{1}{4}\sum_{j=1}^{{}^{\mathrm{E}}n_k} ||\boldsymbol{e}_j||\,|\beta_j|}{\frac{1}{3}\,{}^{\mathrm{E}}n_k}. \tag{2.16} $$

For the curvature based algorithm, the integral curvatures would be sufficient, since the algorithm requires only the mathematical sign to distinguish between different geometric structures (table 2.6). To keep the programming algorithm as general as possible, the computation of the actual curvature was implemented instead of the integral curvature.

Based on curvature values, 5 different geometric shapes can be distinguished (table 2.6).

| | ${}^{\mathrm{V}}K_k$<0 | ${}^{\mathrm{V}}K_k$=0 | ${}^{\mathrm{V}}K_k$>0 |
|---|---|---|---|
| ${}^{\mathrm{V}}H_k$<0 | Ridge | Ridge | Peak |
| ${}^{\mathrm{V}}H_k$=0 | - | Flat | - |
| ${}^{\mathrm{V}}H_k$>0 | Valley | Valley | Pit |

Table 2.6: Different surface types based on ht mathematical sign of curvatures [24].

To distinguish between these structures, only the mathematical sign is needed.

### 2.3.1.3 Algorithm for curvature computation

The computation of curvature values to determine shapes on surfaces is subdivided into three steps (figure 2.17). First, the entire mesh is loaded. Subsequently, each vertex is considered one after the other. For each vertex, the coordinate system is rotated and the curvature values for each vertex ${}^{\mathrm{V}}K_k$ and ${}^{\mathrm{V}}H_k$ are computed.

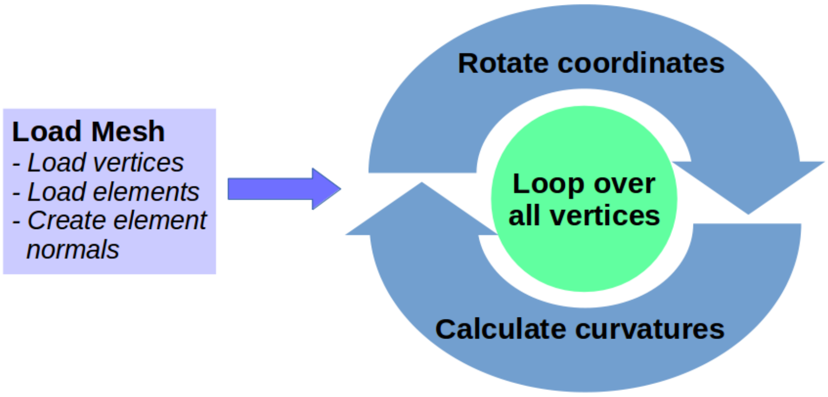


Figure 2.17: Major steps for computing curvature values

1. **Load Mesh:**
   As a first step, the algorithm loads the mesh, generated in chapter 2.2. During this loading process, surface normal vectors $\boldsymbol{n}_i$ are determined for each element $t_i$. In

addition, a normal vector for each vertex $\boldsymbol{v}_k$ is computed, considering all surface normal vectors $\boldsymbol{n}_i$ of adjacent elements

$$\boldsymbol{n}_k = \frac{1}{{}^{\mathrm{E}}n_k} \sum_{i=1}^{{}^{\mathrm{E}}n_k} \boldsymbol{n}_i. \tag{2.17}$$

${}^{\mathrm{E}}n_k$ defines the number of elements connected to $\boldsymbol{v}_k$.

2. **Rotate Coordinates:**
   For each vertex $\boldsymbol{v}_k$, the coordinate system is rotated such that the z-axis is parallel to the vertex normal vector $\boldsymbol{n}_k$ (figure 2.18). This rotated coordinate system is needed to define the mathematical sign of the mean curvature ${}^{\mathrm{V}}H_k$ later on.

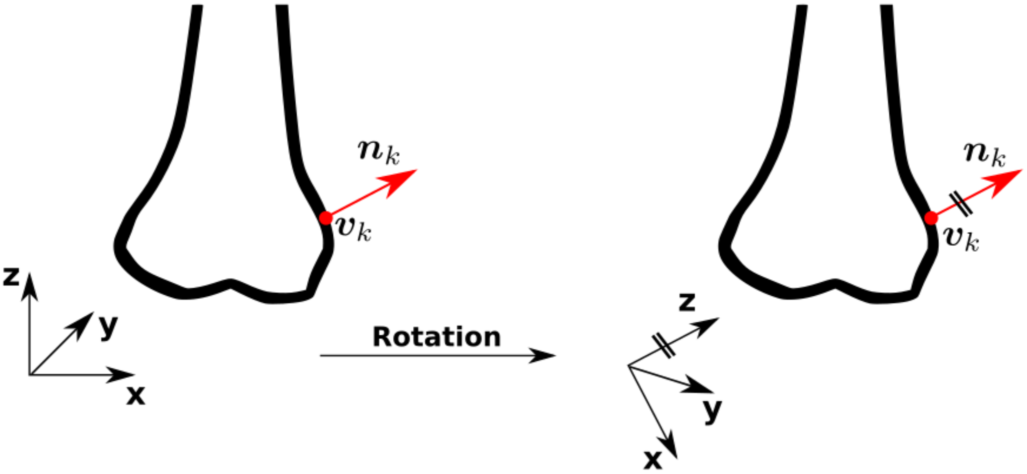


Figure 2.18: Rotation of the coordinate system to align z-axis with the vertex normal vector $\boldsymbol{n}_k$.

Rotation of vectors:
In order to rotate an initial vector $\boldsymbol{u}$ so it becomes parallel to a goal vector $\boldsymbol{w}$, a suitable rotation axis $\boldsymbol{e}_{\mathrm{r}}$ and a respective rotation angle $\alpha_{\mathrm{r}}$ must be found. Following Euler's rotation theorem, any displacement in a three dimensional space is equivalent to one single rotation. This means that instead of rotating a body around the x-, y-, and z-axis, only one rotation around an arbitrary axis is valid [48].

To calculate the rotation axis, the normalized cross product between $\boldsymbol{u}$ and $\boldsymbol{w}$ is used

$$\boldsymbol{e}_{\mathrm{r}} = \frac{\boldsymbol{u} \times \boldsymbol{w}}{||\boldsymbol{u} \times \boldsymbol{w}||}. \tag{2.18}$$

As a next step, the angle $\alpha_{\mathrm{r}}$ between the initial vector $\boldsymbol{u}$ and the goal vector $\boldsymbol{w}$ is calculated by using the definition of the cross product:

$$\left|\boldsymbol{u} \times \boldsymbol{w}\right| = \left|\boldsymbol{u}\right| \left|\boldsymbol{w}\right| \sin(\alpha_{\mathrm{r}}). \tag{2.19}$$

To obtain vector $\boldsymbol{u}$ parallel to $\boldsymbol{w}$, the initial vector has to be rotated around $\boldsymbol{e}_\mathrm{r}$ by an angle of $\alpha_\mathrm{r}$. Rotating around an arbitrary axis requires the following rotation matrix [51]

$$\boldsymbol{R}_\mathrm{arbitary}(\alpha_\mathrm{r}) = \begin{bmatrix} e^2_{\mathrm{r},x}(1-c_\mathrm{r})+c_\mathrm{r} & e_{\mathrm{r},x}\,e_{\mathrm{r},y}(1-c_\mathrm{r})-s_\mathrm{r}\,e_{\mathrm{r},z} & e_{\mathrm{r},x}\,e_{\mathrm{r},z}(1-c_\mathrm{r})+s_\mathrm{r}\,e_{\mathrm{r},y} \\ e_{\mathrm{r},x}\,e_{\mathrm{r},y}(1-c_\mathrm{r})+s_\mathrm{r}\,e_{\mathrm{r},z} & e^2_{\mathrm{r},y}\,(1-c_\mathrm{r})+c_\mathrm{r} & e_{\mathrm{r},y}\,e_{\mathrm{r},z}(1-c_\mathrm{r})-s_\mathrm{r}\,e_{\mathrm{r},x} \\ e_{\mathrm{r},x}\,e_{\mathrm{r},z}(1-c_\mathrm{r})-s_\mathrm{r}\,e_{\mathrm{r},y} & e_{\mathrm{r},y}\,e_{\mathrm{r},z}(1-c_\mathrm{r})-s_\mathrm{r}\,e_{\mathrm{r},x} & e^2_{\mathrm{r},z}\,(1-c_\mathrm{r})+c_\mathrm{r} \end{bmatrix}. \tag{2.20}$$

With $c_\mathrm{r} = \cos(\alpha_\mathrm{r})$ , $s_\mathrm{r} = \sin(\alpha_\mathrm{r})$ and $e_{\mathrm{r},x}, e_{\mathrm{r},y}, e_{\mathrm{r},z}$ are vector elements of the rotation axis $\boldsymbol{e}_\mathrm{r}$.

3. **Computing vertex curvatures:**
   The computation of the Gaussian curvature ${}^\mathrm{V}K_k$ and the mean curvature ${}^\mathrm{V}H_k$ for each vertex $\boldsymbol{v}_k$ is done with respect to equation 2.14 and 2.16. To detect different surface structures, the mathematical signs of curvatures are needed. To determine the mathematical sign of the mean curvatures, the coordinate system is rotated to align the z-axis with the vertex normal vector $\boldsymbol{n}_k$ of the considered vertex $\boldsymbol{v}_k$ (figure 2.19).

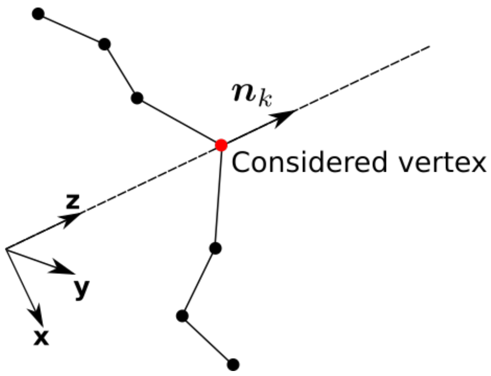


Figure 2.19: Rotated coordinate system to determine the mathematical sign of ${}^\mathrm{V}H_k$.

Comparing the z coordinates of the considered vertex $\boldsymbol{v}_k$ and all adjacent vertices, their relative location can be determined (figure 2.19). If the z coordinate of an adjacent vertex is lower than the z coordinate of the considered vertex $\boldsymbol{v}_k$, the dihedral angle $\beta_j$ is negative (figure 2.16). If the z coordinate of an adjacent vertex is higher than the z coordinate of the considered vertex $\boldsymbol{v}_k$, the dihedral angle $\beta_j$ is positive. The mathematical sign must be determined for all directly adjacent vertices. With respect to equation 2.16 the signed angles $\beta_j$ are then used to determine the sign of the mean curvature ${}^\mathrm{V}H_k$.

#### 2.3.1.4 Validation of computed curvatures

In order to validate the implemented algorithm for computing curvatures 5 different test objects (figure 2.12) with known curvatures were computed. The obtained results of those computations were adequate. In addition, the curvatures of a final test object, featuring peaks of different sizes were calculated (figure 2.20).

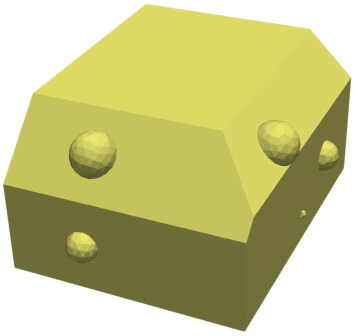
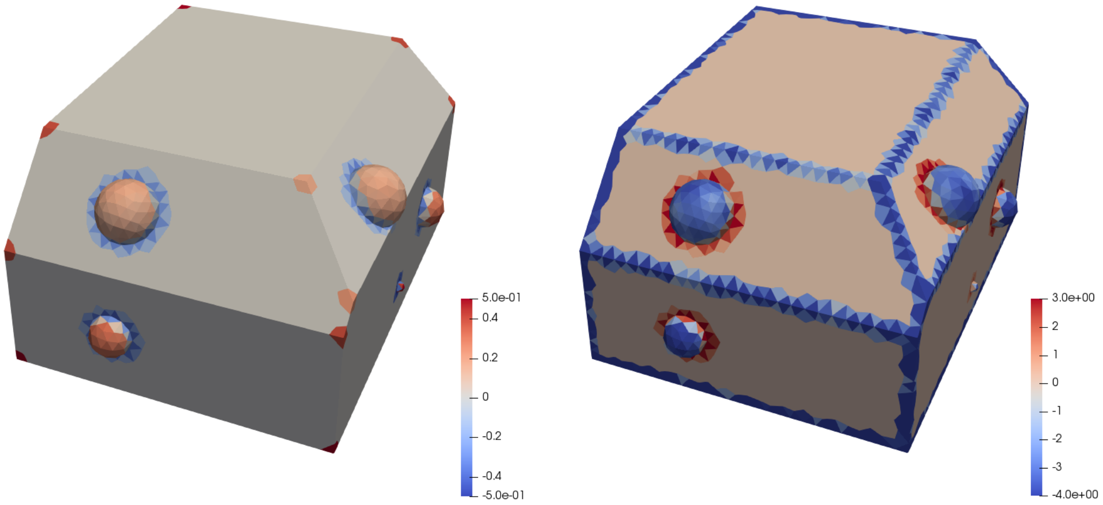


Figure 2.20: Final test object for validation of computed curvatures

At each peak of the final test object, the Gaussian curvature had to be positive and the mean curvature negative. In addition, the Gaussian curvature had to be zero at the edges and positive in the sharp corners. The mean curvature had to negative at the edges and in the corners.



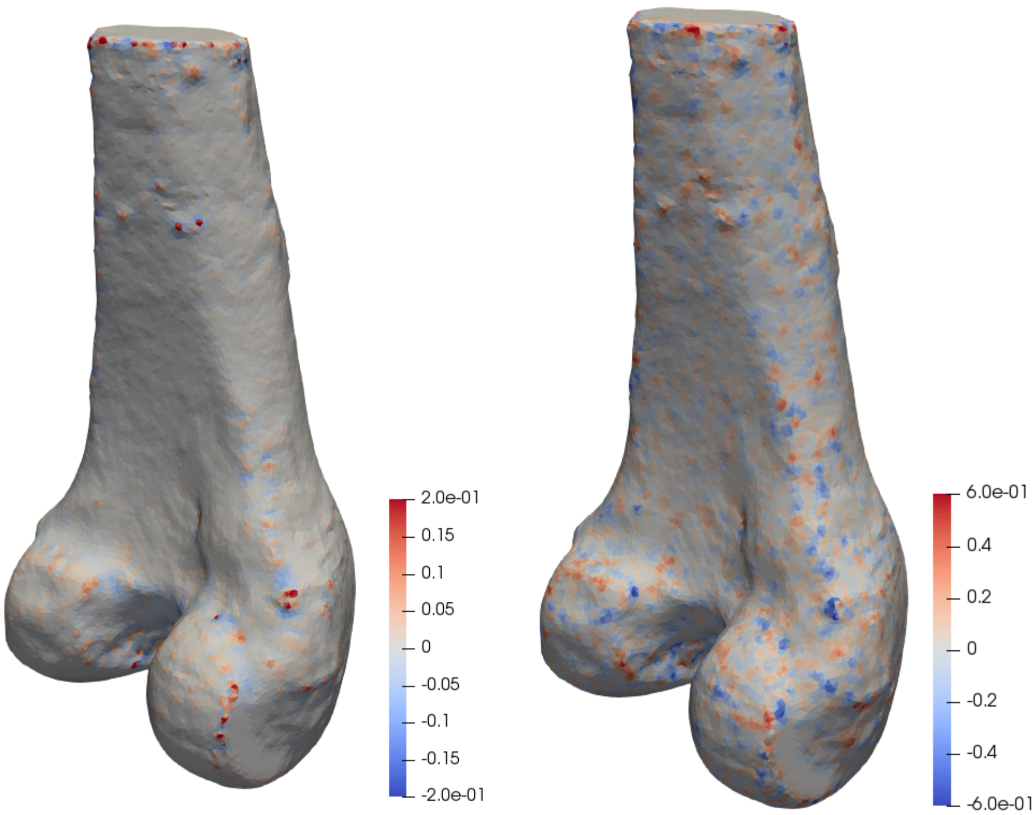
(a) Gaussian curvature

(b) Mean curvature

Figure 2.21: Values of Gaussian and mean curvature of the final test object

The calculated curvatures corresponded to the stated expectations.

**Bone - Femur:**
Gaussian curvature and mean curvature often change on the rough surface of a bone (figure 2.22).



(a) Gaussian curvature

(b) Mean curvature

Figure 2.22: Gaussian and mean curvature on a femur with 44766 elements. View from posterior-medial side.

### 2.3.2 Landmark regions extraction

As the following part of the Curvature Based Algorithm was not elaborated upon in the published paper of Subburaj et al. [24], the following approach might differ from the original algorithm.

To determine landmarks based on curvature values, regions with characteristic properties have to be extracted. The region extraction process can be subdivided into two major steps. First, curvature values of each element, with respect to the elements surroundings, have to be calculated. Afterwards, elements that meet specific requirements are grouped into regions (figure 2.23).

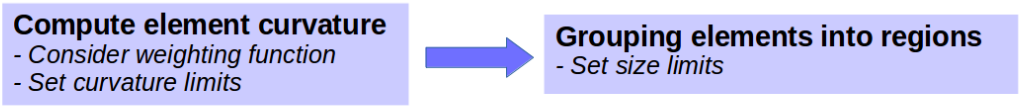


Figure 2.23: Major steps of the region extraction process.

1. **Compute element curvature**

   As explained in chapter 2.3.1, curvature values are calculated for each vertex. In order to find geometric shapes, however, the curvature values ${}^{\mathrm{E}}K_i$ and ${}^{\mathrm{E}}H_i$ of element $t_i$ are needed. For this, the curvature values of the vertices $\boldsymbol{v}_k$ of an element $t_i$ are summed up

   $${}^{\mathrm{E}}K_i = \sum_{k=1}^{3} {}^{\mathrm{V}}K_k, \tag{2.21}$$

   and

   $${}^{\mathrm{E}}H_i = \sum_{k=1}^{3} {}^{\mathrm{V}}H_k. \tag{2.22}$$

   Summing up the curvature values, ideally levels out small local pits on the surface. If a vertex of an element has a different sign due to a small local artifact on the surface, the sign of the element's curvatures ideally do not change after the summation. In such cases, small surface artifacts will have no appreciable effect, since the CBA needs only the mathematical sign to distinguish between the geometric structures (table 2.6).

   For the detection of the behavior of a larger area, the curvatures of vertices neighboring an element $t_i$ must be included. The more neighboring vertices are considered, the more the computed alternate curvature corresponds to a larger area. A ring defines the number of vertices around a considered element $t_i$. Using one ring means that all vertices directly connected to the considered element $t_i$ are included. By two rings, all vertices connected to vertices of ring one are considered, and so on (figure 2.24). For detecting anatomical landmarks three rings are used.

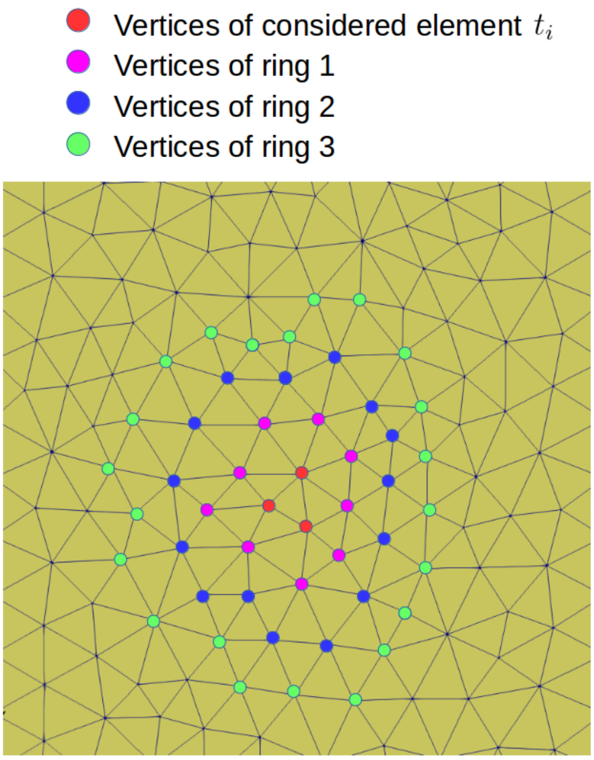


Figure 2.24: Using rings for labeling adjacent vertices.

Depending if a vertex belongs to the element $t_i$ or to one of the rings it is weighted. The curvature values of the vertices of the element $t_i$, are weighted with $\delta_0 = 1$. The weights corresponding to the three rings are referred to as $\delta_1, \delta_2, \delta_3$. Being the number of rings for $m \in [1, 2, 3]$ and the number of vertices in the respective ring for $j \in [1, 2, ..., n_j]$

$$^{\mathrm{R}}K_m = \delta_m \sum_{j=1}^{n_j} {}^{\mathrm{V}}K_j, \tag{2.23}$$

and

$$^{\mathrm{R}}H_m = \delta_m \sum_{j=1}^{n_j} {}^{\mathrm{V}}H_j. \tag{2.24}$$

After weighting the curvatures of the element $t_i$ and the three individual rings, the values are summed up

$$^{\mathrm{C}}K_i = {}^{\mathrm{E}}K_i + \sum_{m=1}^{3} {}^{\mathrm{R}}K_m, \tag{2.25}$$

and

$$^{\mathrm{C}}H_i = {}^{\mathrm{E}}H_i + \sum_{m=1}^{3} {}^{\mathrm{R}}H_m. \tag{2.26}$$

$^{\mathrm{C}}K_i$ and $^{\mathrm{C}}K_i$ represents the curvature values of one element, by considering the curvatures of the surrounding vertices as well.

To determine, if an element is part of a peak, the weighted and summed up curvature values ${}^{\mathrm{C}}K_i$ and ${}^{\mathrm{C}}H_i$ are compared with predefined limits (table 2.6). The limits that define the geometrical shape of a peak are

$$ {}^{\mathrm{C}}K_i > 0, \tag{2.27} $$

and

$$ {}^{\mathrm{C}}H_i < 0. \tag{2.28} $$

Using these limits only delivers information about whether an element $t_i$ is part of a peak, and no information about the size of the peak. Since the surface of a bone is rough, many elements meet these fundamental requirements. Consequently, specific limits have to be defined in order to find peaks of a certain size.

The use of specific curvature limits defines how much a peak has to be curved in order to be taken into account. Therefore, the parameters ${}^{\mathrm{L}}K$ and ${}^{\mathrm{L}}H$ are defined

$$ {}^{\mathrm{C}}K_i > {}^{\mathrm{L}}K, \tag{2.29} $$

and

$$ {}^{\mathrm{C}}H_i < {}^{\mathrm{L}}H. \tag{2.30} $$

Elements whose calculated curvature values ${}^{\mathrm{C}}K_i$ and ${}^{\mathrm{C}}H_i$ correspond to the limits form the basis for forming regions.

2. **Grouping elements into regions**
   All elements that correspond to the limits (equation 2.29 and 2.30) and are directly connected to each other, are grouped together as one region. To determine of how many elements a peak must consist off, the factor region limit $n_{\mathrm{R}}$ is defined. The region limit $n_{\mathrm{R}}$ defines a minimum number of elements from which a region must at least exist, to sort out too small structures.

Finally, a number of regions which contain potential landmarks are received. It should be noted that the subsequent labeling process can only work stably with a certain number of potential landmark regions. A hard limit on how many regions can be processed at maximum can not be defined. Not only the number, but also the location of regions and the number of interesting landmarks is crucial. The more landmarks are to be found, the more potential regions can be processed. For the 5 successive landmarks, at least 25 regions can be processed, depending on their location.

### 2.3.3 Landmark labeling

To label anatomical landmarks from the potential regions, adjacency relationships of defined landmarks are used, as presented by Subburay et al. [24]. First, tolerance ranges have to be defined for each computed landmark region, in order to use adjacency relationships for labeling landmark regions. Finally specific points have to be marked as the landmark point of the labeled landmark region.

The labeling process can be divided into three different steps, which symbolize the logical work-flow during computation (figure 2.25).

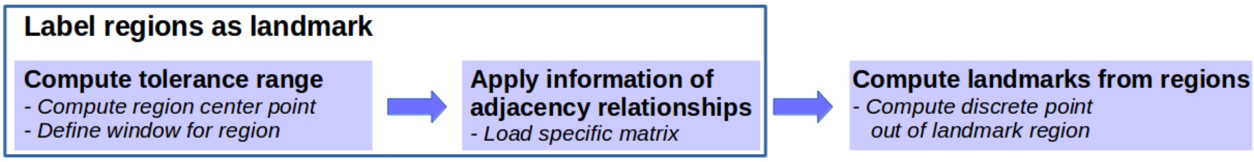


Figure 2.25: Logical work-flow for the landmark labeling process.

1. **Label regions as landmarks**
   The adjacency matrices differ slightly to the ones published in the paper [24], because of the different approach used for the landmark region extraction. Adjacency matrices contain the relative position of each individual landmark, considering each other landmark, expressed in 3 directions (figure 1.2). If a landmark is located in proximal direction (positive z-direction) of another landmark, the corresponding entry for this direction is Px. For example (figure 2.26), with the region of ME as basis, the region of AT is located in proximal direction and the corresponding entry in the matrix (table 2.8 line 2, column 1) for the proximal/distal direction is Px. If a landmark is located in the same level as another landmark, the corresponding entry in the adjacency matrix for that direction is 0. The current basis is defined by the line of the table that represents the adjacency matrix and the landmark that is considered is defined by the column. For example, looking at line 3 and column 1 of table 2.8 shows, that with LE as a basis AT is located in medial and proximal direction, while sharing a level for the anterior/posterior direction (0 entry) (figure 1.4). So, the individual matrix entries express the relative position of one landmark with another landmark as basis for each of the three different directions: P/A for the x-direction, M/L for the y-direction and Px/D for the z-direction. The use of 5 landmarks thus creates a 5x5 matrix.

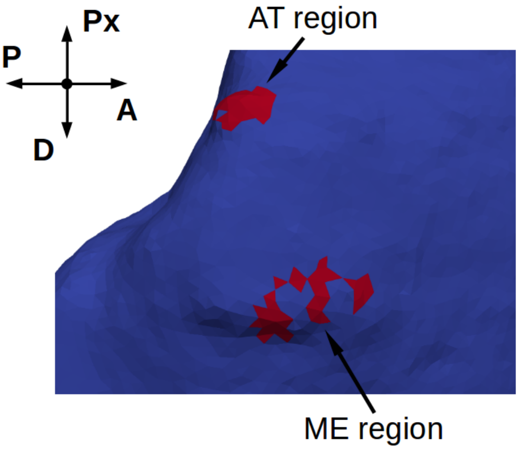


Figure 2.26: Example for the relative distances stored in adjacency matrices.

**Femur:**

The abbreviations of the landmarks used in the adjacency matrix of the femur (table 2.8) are explained in table 2.7

| **Acronym** | **Landmark** |
|---|---|
| AT | Adductor Magnus Tubercle |
| ME | Medial Epicondyle |
| LE | Lateral Epicondyle |
| MP | Medial Peak |
| LP | Lateral Peak |

Table 2.7: Abbreviations of femoral landmarks used in the adjacency matrix in table 1.1

| | **AT** | **ME** | **LE** | **MP** | **LP** |
|---|---|---|---|---|---|
| **AT** | - | 0,0,D | 0,L,D | A,L,D | A,L,D |
| **ME** | 0,0,Px | - | 0,L,0 | A,L,0 | A,L,0 |
| **LE** | 0,M,Px | 0,M,0 | - | A,M,0 | A,M,0 |
| **MP** | P,M,Px | P,M,0 | P,L,0 | - | 0,L,0 |
| **LP** | P,M,Px | P,M,0 | P,L,0 | 0,M,0 | - |

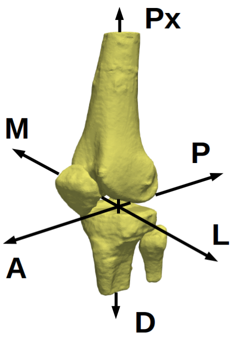


Table 2.8: Adjacency matrix of the left femur used for the labeling process.

**Tibia:**

The abbreviations of the landmarks used in the adjacency matrix of the tibia (table 2.8) are explained in table 2.7

| **Acronym** | **Landmark** |
|---|---|
| TT | Tibial Tuberosity |
| MIT | Medial Intercondylar tubercle |
| LIT | Lateral Intercondylar tubercle |
| MP | Medial Peak |
| LP | Lateral Peak |

Table 2.9: Abbreviations of tibial landmarks used in the adjacency matrix in table 2.10.

|  | **TT** | **MP** | **LP** | **MIT** | **LIT** |
|---|---|---|---|---|---|
| **TT** | - | P,M,Px | P,L,Px | P,0,Px | P,0,Px |
| **MP** | A,L,D | - | 0,L,0 | 0,L,0 | 0,L,0 |
| **LP** | A,M,D | 0,M,0 | - | 0,M,0 | 0,M,0 |
| **MIT** | A,0,D | 0,M,0 | 0,L,0 | - | 0,L,0 |
| **LIT** | A,0,D | 0,M,0 | 0,L,0 | 0,M,0 | - |

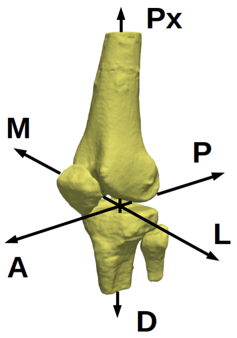


Table 2.10: Adjacency matrix of the left tibia used for the labeling process

Because landmarks vary in their location, it is difficult to define generally valid relative positions. Therefore, tolerance ranges are defined to make more robust statements about whether landmarks are considered to be at the same level (0 entry) in the adjacency matrices. In order to define such windows, the geometric center of a region $\boldsymbol{p}_{\text{center}}$ is calculated first. Then, a window is opened in each direction (figure 2.32), defining the tolerance range, based on this center. For this, the factor $n_{\text{W}}$ is added to the coordinates of the center and subtracted to set the boundaries of the tolerance range (equation 2.31).

$$\begin{aligned} \text{x direction:} \quad & \Delta_{\text{window},x} = p_{\text{center},x} \pm n_{\text{W}} \\ \text{y direction:} \quad & \Delta_{\text{window},y} = p_{\text{center},y} \pm n_{\text{W}} \\ \text{z direction:} \quad & \Delta_{\text{window},z} = p_{\text{center},z} \pm n_{\text{W}} \end{aligned} \tag{2.31}$$

The same relative positions can be used for each specimen if such tolerances are applied (figure 2.27).

For example, looking at the second entry of the first row of table 2.8 determines that, with AT as a basis, ME is located at the same level in posterior/anterior and medial/lateral direction, while it is located distally (figure 2.27). This information is only generally correct if tolerance ranges are used.

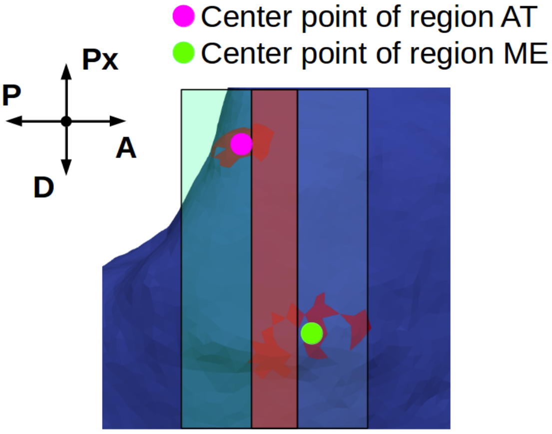


Figure 2.27: Center points of AT and ME and their tolerance range for the posterior/anterior direction.

Apply adjacency matrix

The adjacency matrix is applied, using the defined tolerance ranges. During the application process, each region $R$ of the possible landmark regions is considered for each row in the adjacency matrix once. This means each region $R$ is once used as reference system for each landmark. If there are other regions that meet all requirements stored in this row of the matrix, the other regions $R_o$ are assigned to the corresponding landmarks. For example, let's consider the region AT the current basis (figure 2.27). According to the adjacency matrix, there has to be at least one region which shares a level with AT in anterior/posterior- and medial/lateral direction and which is located distally. All regions that meet these requirements are assigned as ME. After considering each region $R$ as basis for each row of the adjacency matrix, all regions $R_o$ assigned to a landmark are compared. If a certain region was labeled as a certain landmark every time, this region is assigned to the landmark. Consequently, it is possible that two different regions are addressed as the same landmark.

2. **Compute landmarks from regions**

After using the adjacency matrices, some regions are labeled as specific landmark regions. In order to define a certain point from these regions as landmark, four different methods are implemented. Those method are based on finding,

(a) the largest Gaussian curvature ${}^{\mathrm{V}}K_k$ of a vertex.

(b) the largest negative mean curvature ${}^{\mathrm{V}}H_k$ of a vertex.

(c) the largest magnitude of the product of the Gaussian curvature ${}^{\mathrm{V}}K_k$ and the mean curvature ${}^{\mathrm{V}}H_k$.

(d) the geometric center.

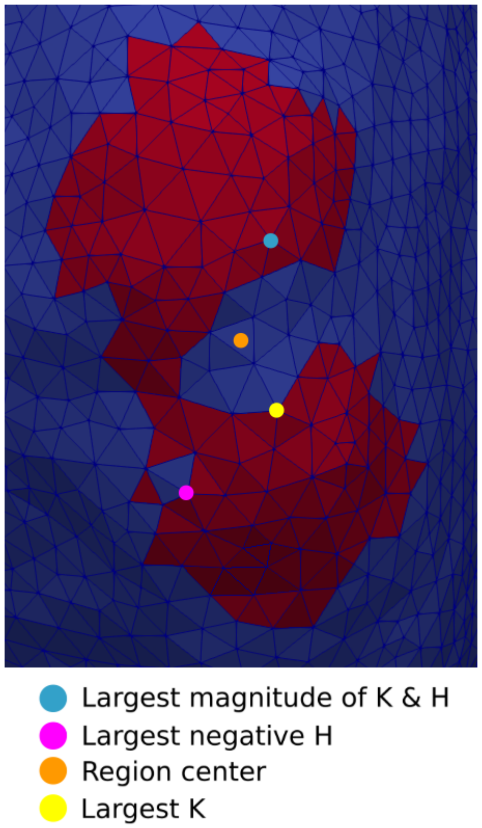


Figure 2.28: Points labeled by different methods in the region of the lateral peak.

### 2.3.4 Parameter adjustment and robustness

The implemented Curvature Based Algorithm possesses 6 parameters, that have a significant impact on the obtained results:

- Curvature limits ${}^{\mathrm{L}}K$ and ${}^{\mathrm{L}}H$
- Region limit $n_{\mathrm{R}}$
- Window size $n_{\mathrm{W}}$
- Weighting factors $\delta_m$
- Discrete point method

The influence of these parameters was to be investigated during the application of the algorithm.

1. <u>Curvature limits ${}^{\mathrm{L}}K$ and ${}^{\mathrm{L}}H$</u>
   The curvature limits ${}^{\mathrm{L}}K$ and ${}^{\mathrm{L}}H$ interact with each other. If a peak on a surface has to be found, ${}^{\mathrm{L}}K$ and ${}^{\mathrm{L}}H$ define together how strongly this peak has to curve, in order to be recognized as such by the algorithm. Therefore, these two limits must always be changed together.

   ${}^{\mathrm{L}}K$ defines a minimum value and ${}^{\mathrm{L}}H$ a negative maximum value for a peak of the surface. Therefore, the computed alternate Gaussian curvature ${}^{\mathrm{C}}K_i$ of an element must be higher and the computed alternate mean curvature ${}^{\mathrm{C}}H_i$ must be negative and lower than the limit, for further processing.

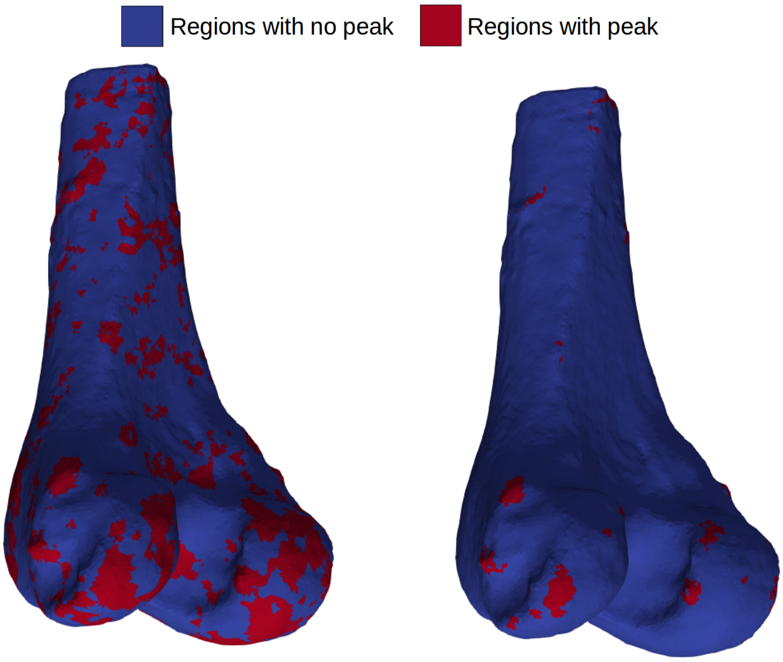
$$ {}^{\mathrm{C}}K_i > {}^{\mathrm{L}}K, \tag{2.32}$$

and

$$ {}^{\mathrm{C}}H_i < {}^{\mathrm{L}}H. \tag{2.33}$$

For other geometrical shapes ${}^{\mathrm{L}}K$ and ${}^{\mathrm{L}}H$ have to be chosen differently (table 2.6).

A general statement on how exactly this limits have to be set can not be found, as this depends on the specific bone surface. However, it can be noted that if too small peaks are found, the limits must be increased (figure 2.29). If large peaks are not recognized they must be decreased.



(a) Without limitation (b) With limitation

Figure 2.29: Comparison of peak finding process with different curvature limitations. (a) has no limitation for both curvatures. Therefore a vast number of elements fulfill the requirement of a peak. For (b), the following limitations were used: ${}^{\mathrm{C}}K_i > 0.35$ and ${}^{\mathrm{C}}H_i < 0.55$. Consequently, the number of shown regions decreases, and only peaks with distinctive curvature values remain. Those regions can be processed further on. View from posterior-medial side.

Using the appropriate limits is crucial to obtain a number of potential regions that can be further processed.

2. Region limit $n_{\mathrm{R}}$
   The region limit $n_{\mathrm{R}}$ defines, how many elements a peak must at least consist of. The curvature limits thus define how strongly a peak has to curve and the region limit $n_{\mathrm{R}}$ how large it must be, to be processed (figure 2.30).

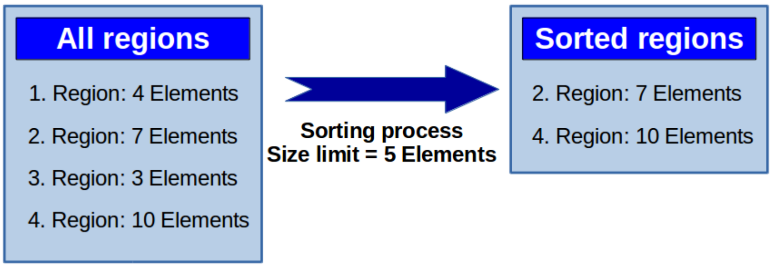
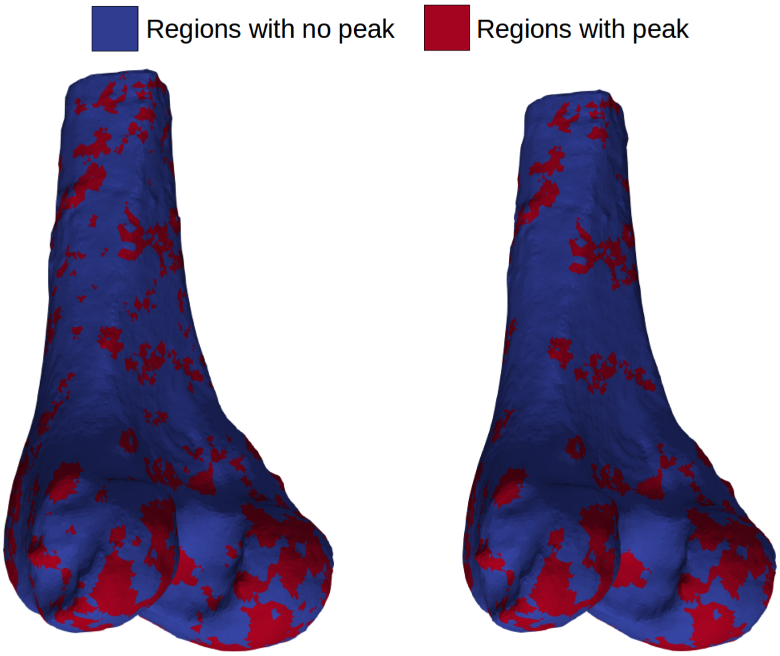


Figure 2.30: Sorting process with a region limit $n_{\mathrm{R}}$ of 5 elements. Therefore only regions that consist of at least 6 elements, are being processed.

This region limit $n_{\mathrm{R}}$ provides the opportunity to eliminate peaks that are too small to be a landmark (figure 2.31).



(a) Without region size limitation

(b) With region size limitation

Figure 2.31: Comparison of peak finding process with different region limits. (a) has no region limit $n_{\mathrm{R}}$ at all. (b) has a region limit $n_{\mathrm{R}} = 50$. Consequently many small regions are eliminated and only regions consisting of at least 51 elements are shown. This computation was done with specimen 15, ${}^{\mathrm{L}}K = 0.3$ and ${}^{\mathrm{L}}H = 0.45$. View from posterior-medial side.

Thus, the region limit also reduces the number of potential regions that are further processed.

3. Window size $n_{\mathrm{W}}$
   The window size parameter $n_{\mathrm{W}}$ defines the size of the computed tolerance range, needed for the labeling process. If the size of the calculated tolerance range changes

greatly, the relative locations in the adjacency matrices are no longer correct (chapter 2.3.3). The relative distances in the matrices were determined for a given window size $n_{\mathrm{W}}$. Therefore, changing this size usually causes the labeling process to stop working. The change of the window size $n_{\mathrm{W}}$ can improve the labeling process in rare cases, if the landmarks are at very atypical places.

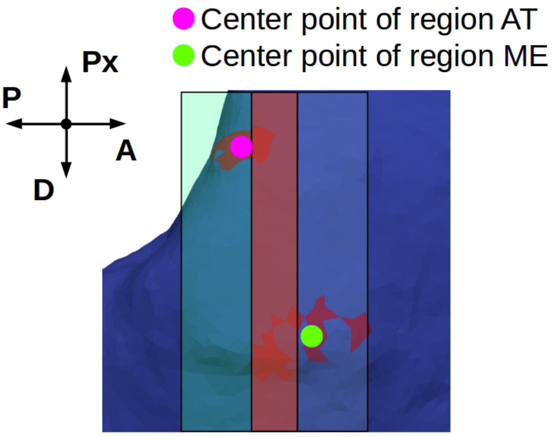


Figure 2.32: Center points of two potential regions (AT and ME) of a left femur with computed tolerance range. They share a level in anterior/posterior direction

For example, when tolerance ranges are used, the Landmarks AT and ME share a plane (figure 2.32) in anterior/posterior direction. As the position of the original coordinate system changes, the tolerance ranges ensure that AT and ME still share a level in anterior/posterior direction (figure 2.33).

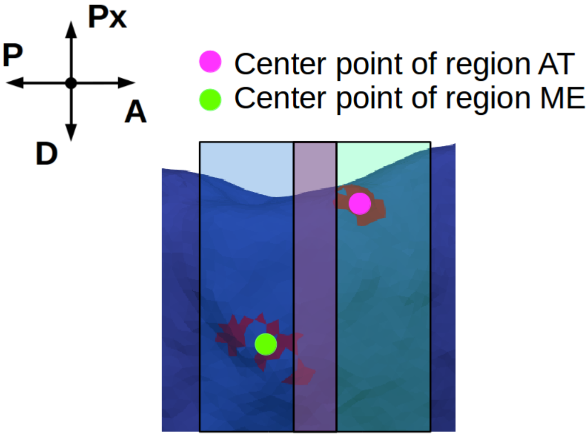
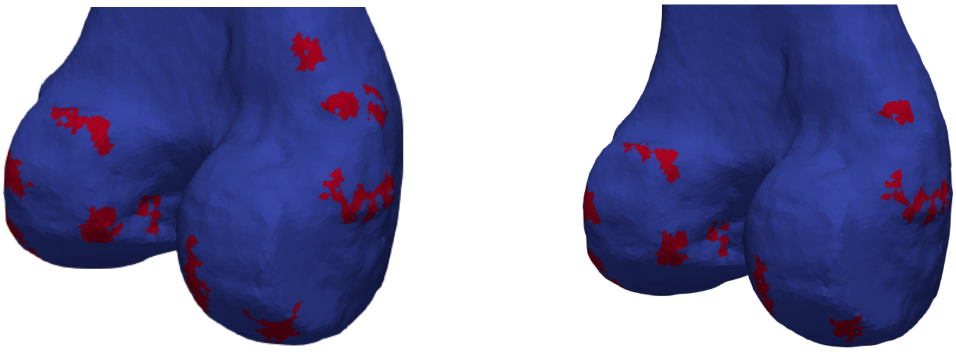


Figure 2.33: Position of Landmarks (AT and ME) with a different coordinate system. They still share a level in anterior/posterior direction.

4. Weighting factors $\delta_m$

   Weighting factors $\delta_1$, $\delta_2$, $\delta_3$ are used to include adjacent elements, whereat the curvature of the considered element $t_i$ is weighted with $\delta_0 = 1$. If these factors are changed the way in which the environment of an element is taken into account changes. The reduction of the weighting factors therefore means that less the global area but the local curvature is taken into account (figure 2.34).



(a) $\delta_1$=$\delta_2$=$\delta_3$ = 1

(b) $\delta_1$=$\delta_2$=$\delta_3$ = 0.8

Figure 2.34: Comparison of different weighing factors on specimen 15. (a) has the following weighting factors $\delta_1$=$\delta_2$=$\delta_3 = 1$ and (b) has $\delta_1$=$\delta_2$=$\delta_3 = 0.8$ as weighting factors. The computation was done with, ${}^{\mathrm{L}}K = 0.2$, ${}^{\mathrm{L}}H = 0.3$ and the region limit $n_{\mathrm{R}} = 30$. View from posterior-medial side.

Because the elements of the meshes used are small, high weighting factors for all algorithm have been used to calculate alternate curvature values (${}^{\mathrm{C}}K_i$ and ${}^{\mathrm{C}}H_i$) for considering curvatures of a larger area.

| Application | Default $\delta_1$, $\delta_2$, $\delta_3$ |
|---|---|
| All implemented algorithm | 0.95, 0.9, 0.9 |

5. Discrete point method
   Different points in a region are labeled as landmark, depending on the method used (figure 2.28). The location of the points received by those different methods vary, and therefore deliver different results for different specimens. Applying a validation process to the results obtained with each method showed no general statement is possible, about which method is appropriate.

If the CBA does not provide adequate results because single peaks can not be found, or because too many peaks are found, interventions can be made. This can be checked in the automatically created vtk file. If too weak or too strong peaks are detected, the ${}^{\mathrm{L}}K$ and ${}^{\mathrm{L}}H$ limits have to be adjusted. If too small or too large regions are detected, the region limit $n_{\mathrm{R}}$ can be changed. The way in which the respective parameters have to be changed for the individual cases is shown in table 2.11.

|  | Parameter |  |  |
|---|---|---|---|
| **Problem** | ${}^{\mathrm{L}}K$ | ${}^{\mathrm{L}}H$ | region limit $n_{\mathrm{R}}$ |
| Too weak peaks | ↑ | ↑ | - |
| Only too strong peaks | ↓ | ↓ | - |
| Too small regions | - | - | ↑ |
| Only to big regions | - | - | ↓ |

Table 2.11: Possibilites for changing parameters of the CBA.

However, it often makes sense to adjust all parameters at the same time, according to the knowledge gained from the vtk file.

### 2.3.5 Operator error and accuracy

The manually labeled points of an experienced orthopedic surgeon rests upon years of experience and know-how and, therefore, are seen as the gold standard [32, 33].

#### 2.3.5.1 Operator error

To define the operator error of labeled landmarks the variance and standard deviation is computed [55]. First the Euclidean norm oft the vertex $\boldsymbol{v}_k$ labeled by the surgeon is calculated as

$$v_k = \sqrt{v_{k,x}^2 + v_{k,y}^2 + v_{k,z}^2}. \tag{2.34}$$

The mean value of the Euclidean norm of $n_k$ data points can be expressed by

$$\bar{v} = \frac{1}{n_k} \sum_{k=1}^{n_k} v_k. \tag{2.35}$$

The variance $s^2$ is defined as follows

$$s^2 = \frac{1}{n_k - 1} \sum_{k=1}^{n_k} (v_k - \bar{v})^2. \tag{2.36}$$

Based on the variance the standard deviation can be determined, which has the advantage that the standard deviation has the same dimension and unit as the original data.

$$s = \sqrt{s^2} \tag{2.37}$$

#### 2.3.5.2 Accuracy

To determine the accuracy of landmarks labeled by the CBA, the Euclidean distances between those points and the manual labeling are caculated.

$$d = \sqrt{(v_{\mathrm{CBA},x} - v_{\mathrm{Surgeon},x})^2 + (v_{\mathrm{CBA},y} - v_{\mathrm{Surgeon},y})^2 + (v_{\mathrm{CBA},z} - v_{\mathrm{Surgeon},z})^2} \tag{2.38}$$

## 2.4 Extending Curvature Based Algorithm

To extend the Curvature Based Algorithm, other geometric structureswere detected by using curvature values. Those different geometric shapes were the basis for labeling additional characteristics and axes.

### 2.4.1 Mathematical theory

To compute additional characteristics and axes the two following fitting methods are applied to detected geometric structures.

#### 2.4.1.1 Spherical fit

In analytic geometry, a sphere is defined by the center point $\boldsymbol{p}_0$ and the radius $r$ (figure 2.35).

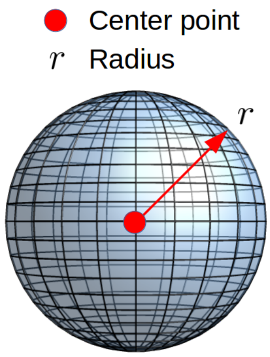


Figure 2.35: Geometric object sphere with center point $\boldsymbol{p}_0$ and the radius $r$.

If a spherical surface is described in $R^3$ by $n$ points defined as $\boldsymbol{p_i} = [p_{i,\mathrm{x}}, p_{i,\mathrm{y}}, p_{i,\mathrm{z}}]^T$, than the radius $r$ can be written as

$$(p_{i,x} - p_{0,x})^2 + (p_{i,y} - p_{0,y})^2 + (p_{i,z} - p_{0,z})^2 = r^2 \tag{2.39}$$

On a perfect sphere each surface point has exactly the same distance to the center point, which is the radius $r$. To fit a sphere into scattered points equation 2.39 is used [56]. If the center point and the radius as well are unknown, exactly four points $\boldsymbol{p}_{i=1...4}$ are needed to create an equation system that can be solved and that delivers the unknown parameters $p_{0,x}, p_{0,y}, p_{0,z}$ and $r$. For fitting a sphere into more than 4 data points the equation system is more difficult[57] and minimizes the summed square errors of the data points. Instead of exactly 4, now $n$ different points $\boldsymbol{p}_{i=1...n}$, where $n > 4$ have to be considered. To be able to calculate the center point $\boldsymbol{p}_0$ and the radius $r$ for this fit, first the mean values of the data points are calculated.

$$\bar{p}_x = \frac{1}{n}\sum_{i=1}^{n} p_{i,x} \qquad \bar{p}_y = \frac{1}{n}\sum_{i=1}^{n} p_{i,y} \qquad \bar{p}_z = \frac{1}{n}\sum_{i=1}^{n} p_{i,z} \tag{2.40}$$

In a next step two matrices are defined.

$$\boldsymbol{A} = 2 \cdot \begin{bmatrix} \sum_{i=1}^{n} \frac{p_{i,x}\cdot(p_{i,x}-\bar{p_x})}{n} & \sum_{i=1}^{n} \frac{p_{i,x}\cdot(p_{i,y}-\bar{p_y})}{n} & \sum_{i=1}^{n} \frac{p_{i,x}\cdot(p_{i,z}-\bar{p_z})}{n} \\ \sum_{i=1}^{n} \frac{p_{i,y}\cdot(p_{i,x}-\bar{p_x})}{n} & \sum_{i=1}^{n} \frac{p_{i,y}\cdot(p_{i,y}-\bar{p_y})}{n} & \sum_{i=1}^{n} \frac{p_{i,y}\cdot(p_{i,z}-\bar{p_z})}{n} \\ \sum_{i=1}^{n} \frac{p_{i,z}\cdot(p_{i,x}-\bar{p_x})}{n} & \sum_{i=1}^{n} \frac{p_{i,z}\cdot(p_{i,y}-\bar{p_y})}{n} & \sum_{i=1}^{n} \frac{p_{i,z}\cdot(p_{i,z}-\bar{p_z})}{n} \end{bmatrix} \tag{2.41}$$

$$\boldsymbol{B} = \begin{bmatrix} \sum_{i=1}^{n} \frac{(p_{i,x}^2+p_{i,y}^2+p_{i,z}^2)\cdot(p_{i,x}-\bar{p_x})}{n} \\ \sum_{i=1}^{n} \frac{(p_{i,x}^2+p_{i,y}^2+p_{i,z}^2)\cdot(p_{i,y}-\bar{p_y})}{n} \\ \sum_{i=1}^{n} \frac{(p_{i,x}^2+p_{i,y}^2+p_{i,z}^2)\cdot(p_{i,z}-\bar{p_z})}{n} \end{bmatrix} \tag{2.42}$$

Solving the following equation system leads to the center point.

$$\boldsymbol{p}_0 = \begin{pmatrix} p_{0,x} \\ p_{0,y} \\ p_{0,z} \end{pmatrix} = (\boldsymbol{A}^T \cdot \boldsymbol{A})^{-1} \cdot \boldsymbol{A}^T \cdot \boldsymbol{B} \tag{2.43}$$

In the last step, the radius $r$ is calculated

$$r = \sqrt{\frac{\sum_{i=1}^{n}((p_{i,x}-p_{0,x})^2 + (p_{i,y}-p_{0,y})^2 + (p_{i,z}-p_{0,z})^2)}{n}} \tag{2.44}$$

#### 2.4.1.2 Linear regression

The basic linear regression model [59, 58] can be stated as follows (figure 2.36)

$$p_{i,y} = \underbrace{\gamma_0 + \gamma_1 \cdot p_{i,x}}\, . \tag{2.45}$$

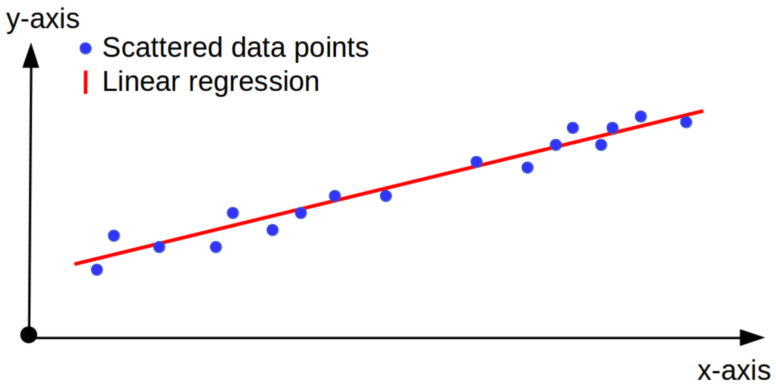


Figure 2.36: Example for a linear regression and the line which is fitted into originally scattered data points.

Where $\gamma_0$ and $\gamma_1$ are unknown parameters of the linear function. To find the parameters $\gamma_0$ and $\gamma_1$ we introduce the method of least squares. Therefore, the error function $g(\gamma_0, \gamma_1)$ which defines the error between $m$ data points $p_{i,f}$ and the linear fit [58], is minimized

$$g(\gamma_0, \gamma_i) = \sum_{i=1}^{m} (\gamma_0 + \gamma_i \cdot p_{i,x} - p_{i,f})^2. \tag{2.46}$$

The minimization of the error function is proposed with respect to $\gamma_0$ and $\gamma_1$. Therefore the partial derivatives from $g(\gamma_0, \gamma_1)$ to $\gamma_0$ and $\gamma_1$ have to disappear.[59]

$$\begin{aligned} \frac{\partial g}{\partial \gamma_0} &= 2\sum_{i=1}^{m}(\gamma_0 + \gamma_1 \cdot p_{i,x} - p_{i,f}) = 2\gamma_0 \sum_{i=1}^{m} 1 + 2\gamma_1 \sum_{i=1}^{m} p_{i,x} - 2\sum_{i=1}^{m} p_{i,f} = 0 \\ \frac{\partial g}{\partial \gamma_1} &= 2x_i \sum_{i=1}^{m}(\gamma_0 + \gamma_1 \cdot p_{i,x} - p_{i,f}) = 2\gamma_0 \sum_{i=1}^{m} p_{i,x} + 2\gamma_1 \sum_{i=1}^{m} p_{i,x}^2 - 2\sum_{i=1}^{m} p_{i,x}\, p_{i,f} = 0 \end{aligned} \tag{2.47}$$

Dividing both equations by 2 leads to the following linear equation system

$$\begin{pmatrix} \sum_{i=1}^{m} 1 & \sum_{i=1}^{m} p_{i,x} \\ \sum_{i=1}^{m} p_{i,x} & \sum_{i=1}^{m} p_{i,x}^2 \end{pmatrix} \cdot \begin{pmatrix} \gamma_0 \\ \gamma_1 \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{m} p_{i,f} \\ \sum_{i=1}^{m} p_{i,x}\, p_{i,f} \end{pmatrix} \tag{2.48}$$

The unknown parameters $\gamma_0$ and $\gamma_1$ are determined by solving this equation system [59, 58]. As input data for the linear regression the middle points of ten slices of the shaft of the tibia and femur are used 2.4.6.

### 2.4.2 Farthest points of the tibial plateau

The farthest points are those points of the contour of the tibial plateau with the greatest distance between each other. The use of this approach aims to improve the detection of the lateral and medial peak of the tibial plateau.

In order to detect those points, the contour of the tibial plateau has to be found using curvature values (figure 2.37). This contour corresponds to an edge. On an idealized sharp edge, the Gaussian curvature is equal to zero (figure 2.21). Because the edge of the contour of the tibial plateau is not an idealized sharp edge the Gaussian curvature must at least have a specific minimum value ${}^{\mathrm{L}}K_{\mathrm{Ft}}$, defined for detecting the contour of the tibial plateau. The mean curvature is neglected to detect the contour of the edge

$${}^{\mathrm{C}}K_i > {}^{\mathrm{L}}K_{\mathrm{Ft}}. \tag{2.49}$$

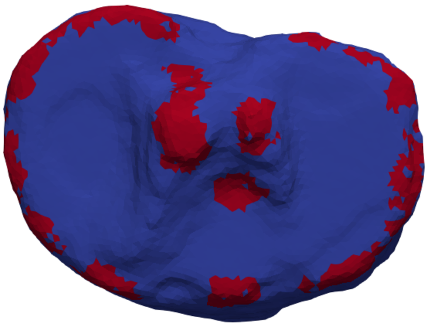


Figure 2.37: Contour of the tibial plateau. This computation was done with parameters shown in table 2.12. View from proximal side.

Using this curvature limit delivers the edges, but many other regions as well. Therefore, it must be additionally defined which of these found regions should be used to calculate the distances. Starting from the highest points of the intercondylar tubercle, a window in distal direction is defined, within which all regions are understood as regions of the edge. To define the size of this window, the parameter window size plateau $n_\mathrm{p}$ is used. For all vertices in the obtained regions, the distances to each other are calculated. The two vertices with the greatest distance are then labeled as farthest points of the tibial plateau (figure 2.38). This also includes regions that are on the intercondylar tubercles. However, since the largest distance between two vertices within these detected regions are calculated, this is irrelevant.

Figure 2.38: Detected farthest points of the tiblial plateau at the lateral and medial condyle of the tibia. View from proximal side.

If the window size plateau $n_\mathrm{p}$ is too big or too small, regions are considered that are not part of the edge. This will label wrong points. Based on experiences gathered during the application of the algorithm, the following default values are used:

| **Parameter** | **Value** |
|---|---|
| Number of elements $n_\mathrm{r}$ | 10 |
| Gaussian curvature limit ${}^{\mathrm{L}}K_{\mathrm{Ft}}$ | 0.1 |
| Window size plateau $n_\mathrm{p}$ | 18 |

Table 2.12: Default settings for computing farthest points of the Tibia.

#### 2.4.2.1 Validation of the farthest points at the tibial plateau

This approach is supposed to detect the lateral and medial peaks of the tibial plateau. Therefore, the received results are compared with the lateral and medial peaks of the original Curvature Based Algorithm (chapter 2.3) and with the points manually labeled by the surgeon. To validate the accuracy the euclidean distance is computed (equation 2.38).

### 2.4.3 Farthest points of patella

The furthest points of the patella are defined by four points on the anterior side with the greatest distance between them. The lines between two of these points must be rectangular. In order to detect those points the outer contour of the patella has to be found first by using curvature values. The edge to be found here is basically similar to the edge of the tibial plateau. Therefore the mean curvature is neglected and only the Gaussian curvature is considered

$$ {}^{\mathrm{C}}K_i > {}^{\mathrm{L}}K_{\mathrm{Fp}}. \tag{2.50} $$

Using the specific curvature limit ${}^{\mathrm{L}}K_{\mathrm{Fp}}$ for detecting the contour of the patella delivers the following regions (figure 2.39).

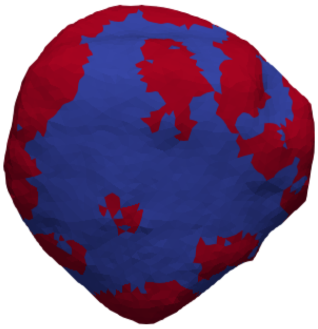


Figure 2.39: Detected outer contour of the patella which is used to compute the farthest points. Those computation was done with the parameters shown in table 2.13. View from anterior side.

Based on the vertices in the detected edge regions, the distances from each vertex to the other vertices are calculated to find the two vertices with the greatest distance. These vertices are used to define an axis. Next, the distances from each vertices that lie orthogonal to the previous computed axis are determined to identify the largest distance again. The dot product is used to determine the orthogonality of two lines. If two lines are exactly orthogonal to each other, then the dot product of the two is equal to zero. Because the connecting lines between the farthest points of the patella will not be exactly orthogonal, a dot product factor $n_{\mathrm{df}}$ is used that defines the maximum deviation. The dot product factor $n_{\mathrm{df}}$ states the maximal deviation of those lines.

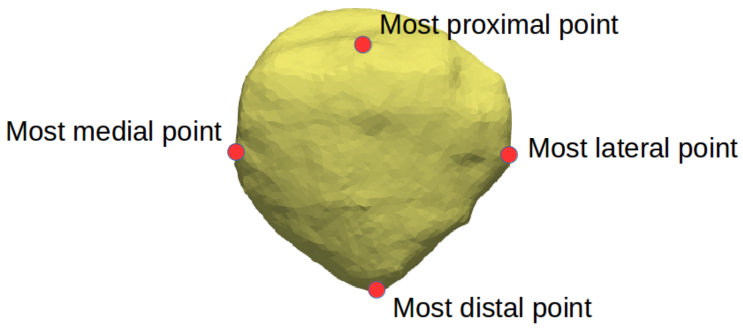


Figure 2.40: Farthest points of Patella. View from anterior side.

Based on experience gathered during the application of the algorithm, the following default values are used:

| Parameter | Value |
|---|---|
| Number of elements $n_\mathrm{r}$ | 10 |
| Gaussian curvature limit ${}^\mathrm{L}K_\mathrm{Fp}$ | 0.2 |
| Dot product factor $n_\mathrm{df}$ | 2 |

Table 2.13: Default settings for computing farthest points of the Patella

#### 2.4.3.1 Validation of the farthest points of the patella

For the farthest points of the patella manually labeled landmarks of an experienced orthopedic surgeon exist. For determining the accuracy the euclidean distances (equation 2.38) and for determining the operator error the standard deviations (equation 2.37) are computed.

### 2.4.4 Trochlear Groove

The trochlear groove is a concave notch of the femur in which the patella moves along during flexion and extension and is also called patella valley [19]. Curvature values are used, to detect the path of the bottom of the patella valley. Because the tochlear groove resembles a curved valley, the requirements for a valley (table 2.6) are modified. This modification is necessary as it is not an idealized valley with no curvature along the valley floor

$$ {}^\mathrm{L}K_\mathrm{Tg} < {}^\mathrm{C}K_i < 0, \tag{2.51}$$

and

$$ 0 \leq {}^\mathrm{C}H_i < {}^\mathrm{L}H_\mathrm{Tg}. \tag{2.52}$$

Using the specific curvature limits ${}^\mathrm{L}K_\mathrm{Tg}$ and ${}^\mathrm{L}H_\mathrm{Tg}$ for detecting the trochlear groove delivers the following region in the patella valley (figure 2.41).

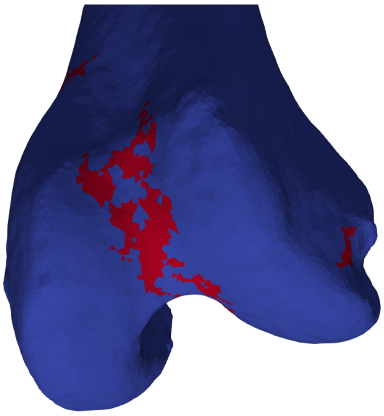


Figure 2.41: Patella valley detected using curvature values, with the following input parameters: region limit $n_\mathrm{r} = 100$, ${}^\mathrm{L}K_\mathrm{Tg} = 0.05$, ${}^\mathrm{L}H_\mathrm{Tg} = 0.55$ and weighting factors are $\delta_1 = \delta_2 = \delta_3 = 0.9$. The region has holes, because due to local artifacts single elements do not match the given requirements. View from distal side.

The region has holes, because due to local artifacts single elements of the patella valley do not match the given requirements. In order to consider each element of the trochlea groove for the computation, these holes have to be closed. The vertices of the detected region give us enough information to computed a spherical fit (figure 2.42).

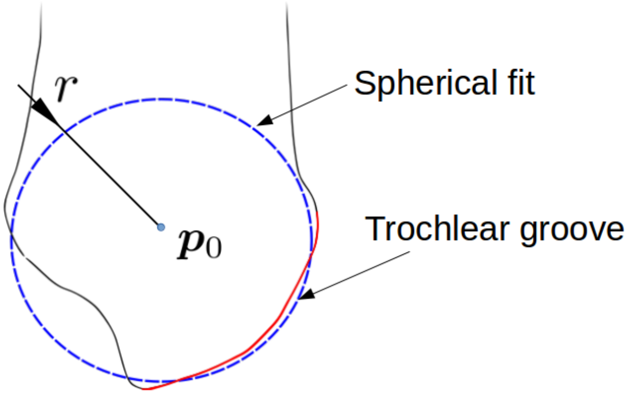


Figure 2.42: Contour of the midplane (posterior/anterior-proximal/distal) of the distal end of femur with a spherical fit of the trochlear groove.

The center point $\boldsymbol{p}_0$ and radius $r$ of this sphere is now used to define a valley area (table 2.14).

| | **lower area limit** | **upper area limit** |
|---|---|---|
| **x direction** | $p_{0,x} - 10$ | $p_{0,x} + 10$ |
| **y direction** | $p_{0,y} - r \cdot 1.2$ | $p_{0,y} + r \cdot 0.1$ |
| **z direction** | $p_{0,z} - r \cdot 1.2$ | $p_{0,z} + r \cdot 0.1$ |

Table 2.14: Area for detecting elements of the patella valley.

All elements within this defined valely area are considered for detecting the patella valley bottom (figure 2.43).

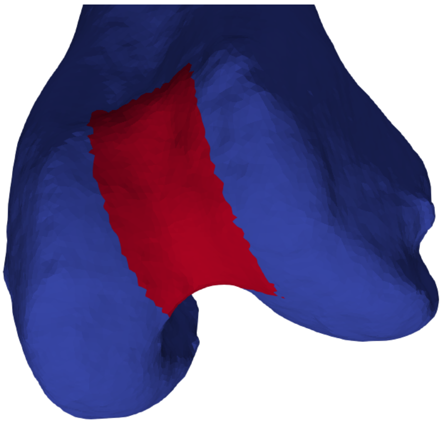


Figure 2.43: All elements which are identified as elements of the trochlear groove, based on the area limits. View from distal side.

To determine the path of the bottom of the trochlea groove, a starting point is selected. This is done using the point with the shortest distance to the midpoint $\boldsymbol{p}_0$ at the proximal edge of the defined area. Subsequently, the distance of each adjacent vertex to the center is calculated. The one vertex with the smallest distance is seen as the next vertex of the valley bottom. This is done until the path reaches the distal edge of the region.

This approach initially requires some vertices to find the actual path (figure 2.44).

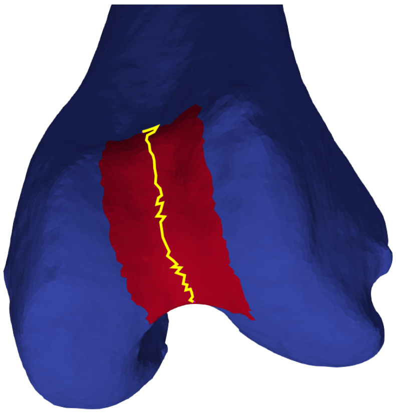


Figure 2.44: Path of the Trochlear Groove bottom. View from distal side.

Based on experiences obtained during application the algorithm the following default values are used:

| **Parameter** | **Value** |
|---|---|
| Number of elements $n_{\mathrm{r}}$ | 150 |
| Gaussian curvature limit ${}^{\mathrm{L}}K_{\mathrm{Tg}}$ | 0.01 |
| Mean curvature limit ${}^{\mathrm{L}}H_{\mathrm{Tg}}$ | 0.55 |

Table 2.15: Default settings for computing the Trochlear Groove of a left Femur

### 2.4.5 Cylindrical axis of the posterior femoral condyles

The cylindrical axis of the femoral condyles is the axis obtained by fitting a cylinder in the medial and lateral condyle [20]. In order to compute the cylindrical axis, the outer contour of the medial and lateral condyle has to be detected, using curvature values. For this purpose, contours similar to an edge must be found again. Since the edges are not idealized but curved, the mean curvature is again neglected and only the Gaussian curvature is considered by using a specific curvature limit ${}^{\mathrm{L}}K_{\mathrm{Ca}}$ for detecting the contour of the condyles

$$ {}^{\mathrm{C}}K_i > {}^{\mathrm{L}}K_{\mathrm{Ca}}. \tag{2.53}$$

The curves along the outer contour of the condyles are much larger than those of the edges of the tibia plateau, leading to a higher limit for the Gaussian curvature.

The vertices of these detected contour regions are used to compute a spherical fit for each condyles. The cylindrical axis is then defined as the connection of the center points $\boldsymbol{p}_0$ of those spheres (figure 2.45).

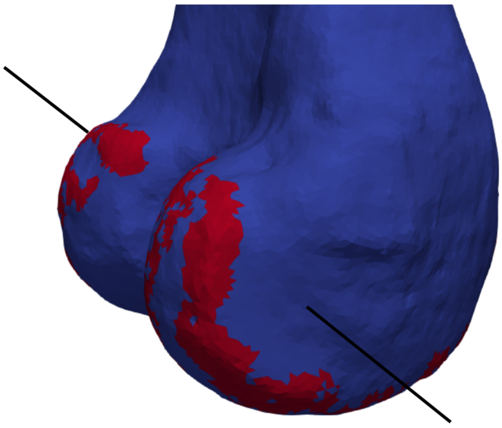


Figure 2.45: Cylindrical axis and the couture regions, which where detected using curvature values, with the following input parameters: region limit $n_{\mathrm{r}} = 250$ and ${}^{\mathrm{L}}K_{\mathrm{Ca}} = 0.2$. View from posterior-medial side.

Based on experiences obtained during application the algorithm the following default values are used:

| Parameter | Value |
|---|---|
| Number of elements $n_\mathrm{r}$ | 300 |
| Gaussian curvature limit ${}^\mathrm{L}K_\mathrm{Ca}$ | 0.2 |

Table 2.16: Default settings for computing the cylindrical axis of a left Femur

### 2.4.6 Shaft axis

The shaft axis of the femur and tibia are computed in the same way. First, the cylindrical part of the bone is sliced into 10 equally distributed slices of 2 mm thickness. In the next step, a spherical fit is computed to find the centers of each slice. Finally, linear regression is used to fit a line into those center points $\boldsymbol{p}_0$ (figure 2.46).

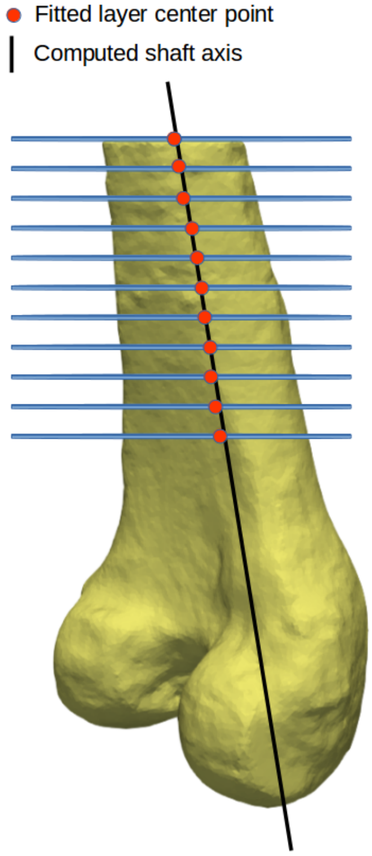


Figure 2.46: Slices of the cylindrical part of the femur with each slices center point, depicted as a red dot. The shaft axis was computed using linear regression on the fitted center points. View from posterior-medial side.

Computing a connection line between the center points $\boldsymbol{p}_0$ of two slices would be sufficient in order to compute a shaft axis. By using a linear regression and the center points of more then two slices, the stability of the computation increases. If a slice is deformed, taking into account a total of 10 slices, due to poor segmentation or deformed bone, this has no great influence on the computed axis.

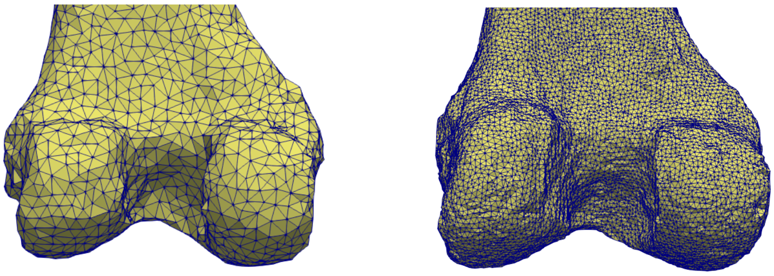
To account for varying length of the femur and tibia the parameter percentage $n_\mathrm{s}$ is defined. This parameter $n_\mathrm{s}$ determines how much bone is used for the computation starting at the proximal edge. $n_\mathrm{s} = 50\%$ of the proximal end of the femur and $n_\mathrm{s} = 40\%$ of the distal end of the tibia are used as default values based on experience obtained during the application of the algorithm.

# Chapter 3

# Results

## 3.1 Created 3D models from CT images

The determination of the appropriate mesh size was the first step of creating meshes.

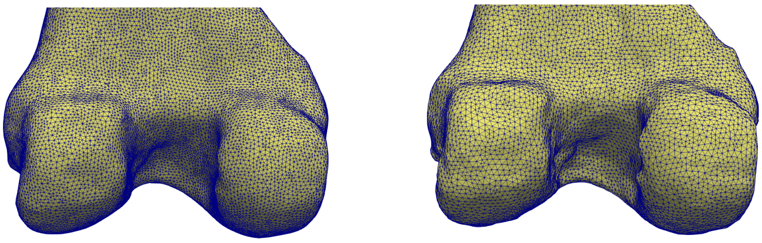


(a) Rough mesh (b) Fine mesh

Figure 3.1: Influence of the number of elements at the mesh of a femur. (a) 5313 elements (b) 44704 elements computed with final settings. View from posterior side.

If the element size of the mesh was chosen too large, the original surface could not be appropriately approximated. After testing different sizes the element size was determined (table 2.4) for all specimens (figure 3.1b).

Determining the appropriate Taubin smoothing settings was the second step.



(a) Strong smooth (b) Weak smooth

Figure 3.2: Comparison of two Taubin smoothing filters. (a) Too strong smoothing (b) Final smoothing filter settings. View from posterior side.

If the Taubin smoothing was too strong, the original contour was distorted. After testing different Taubin smoothing settings the settings were determined (table 2.5) for all specimens (figure 3.2b)

15 data sets (chapter 2.1) were generated from CT-images, with the developed segmentation process. Therefore, the femur, tibia, patella and fibula were segmented.

Depending on the quality of the CT-Scans the times for an entire segmentation process varied between two and six hours.

| **Step of procedure** | **Computation time** | **Manual time** |
|---|---|---|
| **1. Input data preparation** | | |
| 1.1 Dicom files converting | <1 Minute | - |
| 1.2 Midplanes generation | <1 Minute | - |
| 1.3 Mhd file cropping | 1 Minute | 1 Minute |
| 1.4 Voxel size refining | 20-30 Minutes | - |
| 1.5 Data scaling | <1 Minute | - |
| **2. Single bone segmentation** | | |
| 2.1 First mask creation | 5 Minutes | 15-25 Minutes |
| 2.2 Thresholding | 7 Minutes | 3 Minutes |
| 2.3 Holes and cavities closing | <5 Minutes | - |
| 2.4 Final Mask generation | 10 Minutes | 60 - 240 Minutes |
| **Final mesh generation** | | |
| 3.1 Mesh generation | <5 Minutes | - |
| 3.2 Surface smoothing | 10-20 Minutes | - |

Table 3.1: Time breakdown of each single step of the segmentation process. Especially the steps in which manual segmentation work is required have the highest risk to slow down the process.

It should be noted that especially in those areas of the bones where landmarks were supposed to be the manual segmentation was done very precisely.

## 3.2 Anatomical landmarks detection

Computing curvature values was a computational expensive process and took up to one hour. If the curvature values were stored, reloading them took only up to five minutes. All following computations and simulations were done with a computer with the following stats:

| **System** | |
|---|---|
| Processor | Intel(R) Core(TM) i5-6200U CPU with 2.30GHz |
| RAM | 8 GB DDR |
| Operating System | Ubuntu 16.4 |

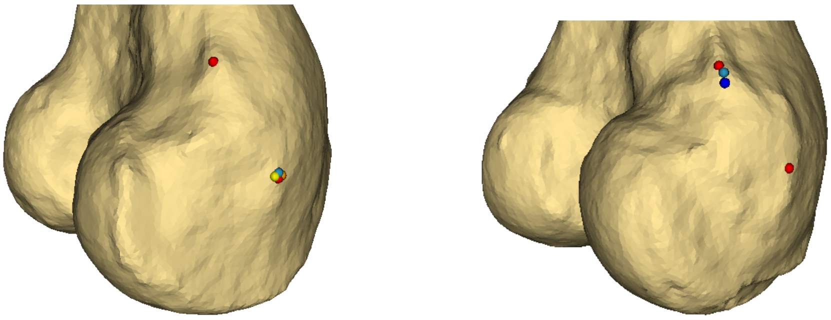
Table 3.2: Computer used for computations

Thus, the given calculation times are only plausible for a computer with this specification..

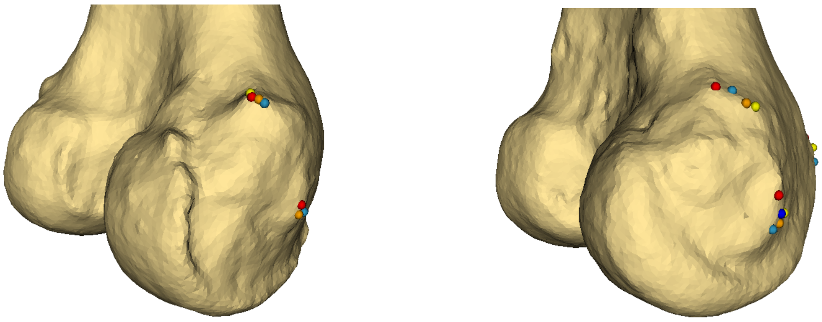
### 3.2.1 Femur

#### 3.2.1.1 Validation of labeling methods femur

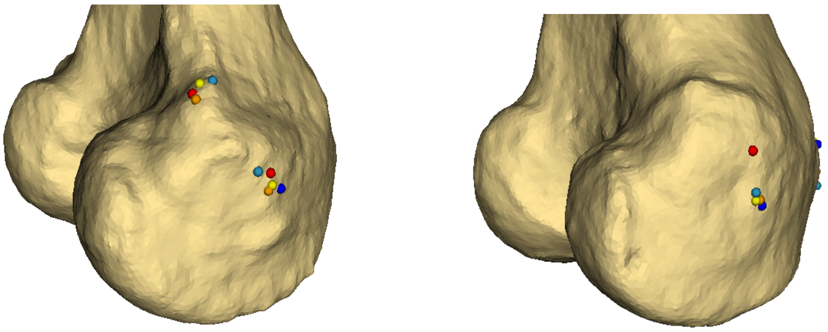
Four implemented labeling methods were used to compute results on seven different specimens (figures 3.3 and 3.4). In Table 3.3 the mean Euclidean distances, standard deviations (chapter 2.3.5) and maximum and minimum distance for each labeling method and each landmark are shown. To compute those quantities the Euclidean distances between points labeled by the algorithm and labeled by an experienced orthopedic surgeon are used.



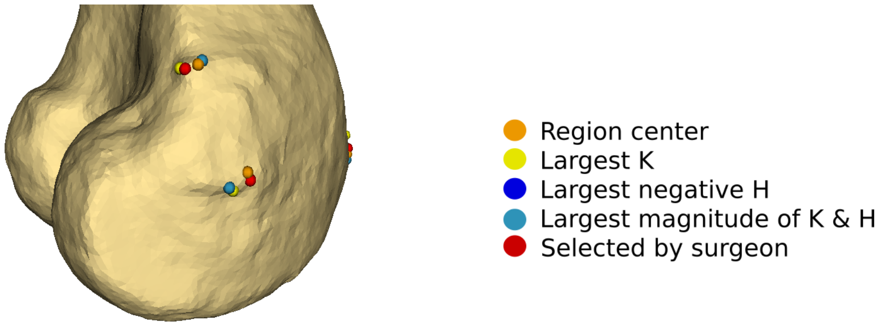
(a) Specimen 1 (b) Specimen 2



(c) Specimen 7 (d) Specimen 9

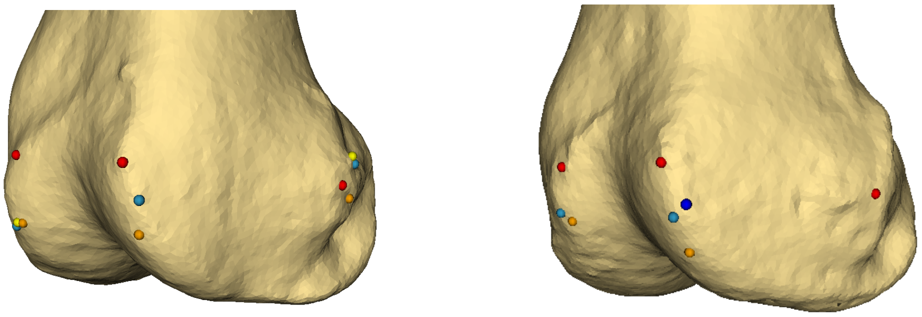


(e) Specimen 12 (f) Specimen 13

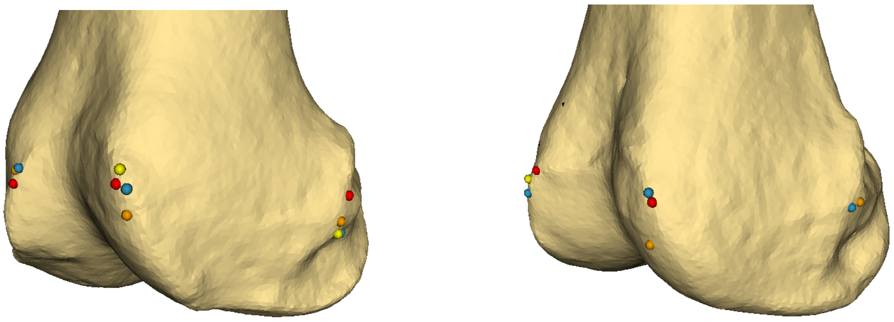


(g) Specimen 15

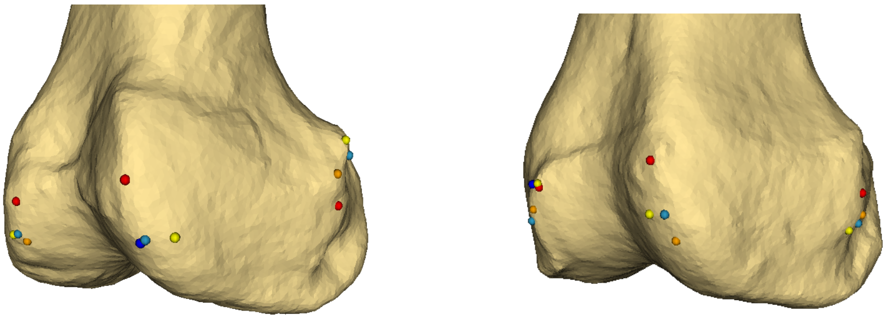
Figure 3.3: Selection of differently labeled landmarks on the femur, view from posterior-medial side.



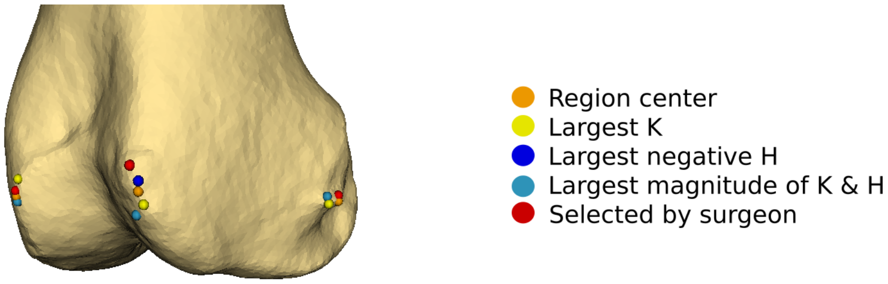
(a) Specimen 1 (b) Specimen 2



(c) Specimen 7 (d) Specimen 9



(e) Specimen 12 (f) Specimen 13



(g) Specimen 15

Figure 3.4: Selection of differently labeled landmarks on the femur, view from anterior-lateral side.

| **Method** | **Mean** [mm] | **Std** [mm] | **Max** [mm] | **Min** [mm] |
|---|---|---|---|---|
| Landmark AT | | | | |
| 1. center | 2.61 | 2.09 | 6.32 | 1.30 |
| 2. gauss | 2.95 | 3.09 | 8.39 | 1.03 |
| 3. mean | 3.36 | 0.67 | 4.32 | 2.66 |
| 4. both | 2.68 | 1.27 | 4.32 | 1.03 |
| Landmark ME | | | | |
| 1. center | 4.02 | 3.37 | 9.71 | 1.15 |
| 2. gauss | 4.02 | 3.37 | 9.71 | 1.15 |
| 3. mean | 4.28 | 2.62 | 8.0 | 1.03 |
| 4. both | 4.4 | 3.52 | 11.12 | 1.03 |
| Landmark LE | | | | |
| 1. center | 3.25 | 2.16 | 5.44 | 0.73 |
| 2. gauss | 5.17 | 2.65 | 8.53 | 2.35 |
| 3. mean | 5.09 | 1.91 | 7.44 | 2.69 |
| 4. both | 5.51 | 2.34 | 8.53 | 2.69 |
| Landmark MP | | | | |
| 1. center | 7.27 | 4.56 | 14.12 | 1.23 |
| 2. gauss | 6.0 | 4.74 | 14.12 | 0.87 |
| 3. mean | 7.71 | 4.37 | 15.34 | 2.26 |
| 4. both | 6.74 | 4.74 | 15.34 | 2.24 |
| Landmark LP | | | | |
| 1. center | 10.25 | 4.85 | 16.89 | 5.11 |
| 2. gauss | 7.87 | 4.63 | 14.6 | 1.87 |
| 3. mean | 7.58 | 4.19 | 12.35 | 1.87 |
| 4. both | 6.96 | 4.63 | 12.87 | 1.87 |

Table 3.3: Mean euclidean distance, standard deviation and maximal and minimal distances between four differently labeled points from the CBA (chapter 2.3.3) and the surgeon for all 5 landmarks (chapter 1.3) of 7 femura.

| **Method** | **Mean** [mm] | **Std** [mm] | **Max** [mm] | **Min** [mm] |
|---|---|---|---|---|
| 1. center | 5.48 | 2.21 | 16.89 | 0.73 |
| 2. gauss | 5,20 | 1.89 | 14.12 | 0.87 |
| 3. mean | 5,60 | 1.96 | 15.34 | 1.03 |
| 4. both | 5,28 | 1.77 | 15.34 | 1.03 |

Table 3.4: Computed mean distances, standard deviations and maximal and minimal distances for each labeling method of the femur.

Looking at the computed mean distances (table 3.4), indicated that no method could be generally favored for all femoral landmarks. Because of repeatability, for subsequent uses of the CBA applied the points with the highest magnitude of the product of both curvature values. This method was more repeatable, since the point with maximum curvature values within a region always remained the same.

#### 3.2.1.2 Operator error of surgeon

To determine the operator error of the landmarks labeled by an experienced orthopedic surgeon, first the average of 8 labeling procedures performed on one specimen was calculated (figure 2.3). Following the mean euclidean distance, standard deviation and maximal and minimal distance between (figure 3.5) the average and each single labeling was computed (table 3.5).

| **Femur - Specimen 2** | | | | |
|---|---|---|---|---|
| **Landmark** | **Mean** [mm] | **Std** [mm] | **Max** [mm] | **Min** [mm] |
| LE | 1.76 | 0.735 | 2.66 | 0.78 |
| ME | 3.78 | 3.69 | 7.17 | 0.707 |
| LP | 0.821 | 0.786 | 1.54 | 0.20 |
| MP | 1.51 | 1.95 | 3.65 | 0.41 |
| AT | 1.63 | 1.05 | 3.07 | 0.41 |

Table 3.5: Mean euclidean distance, standard deviation, and maximal and minimal distance between the average of 8 segmentations and each of those individual labelings performed by the surgeon on specimen 2 .

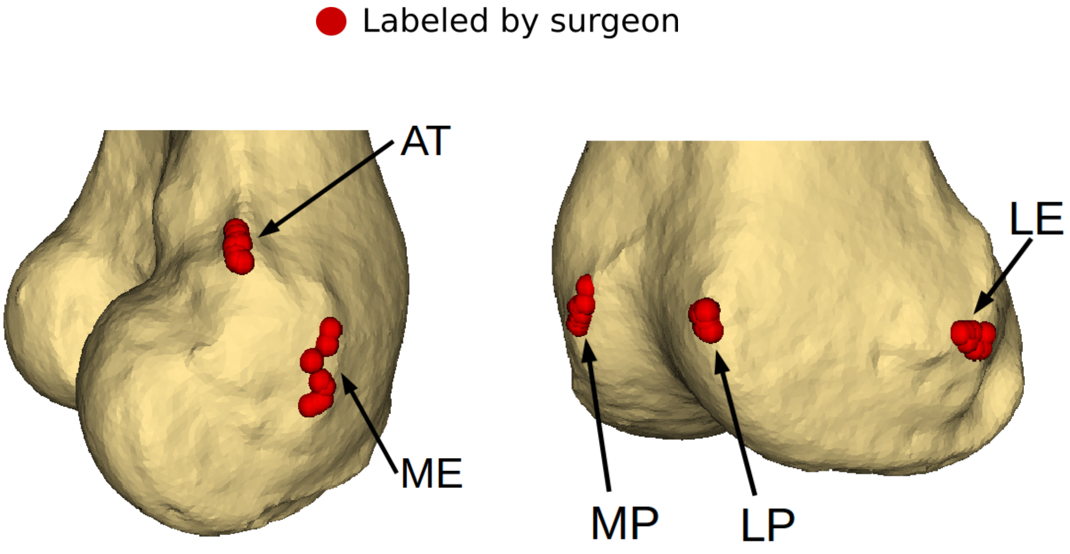
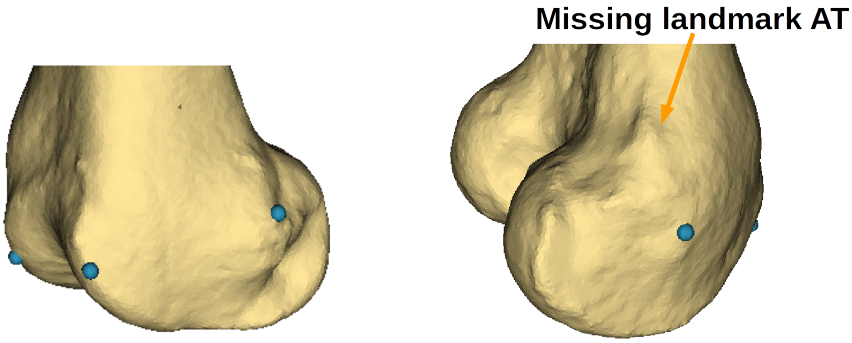


Figure 3.5: Scattering landmarks of the femur of specimen 2 labeled by an orthopedic surgeon. View from medial-posterior and lateral-anterior side.

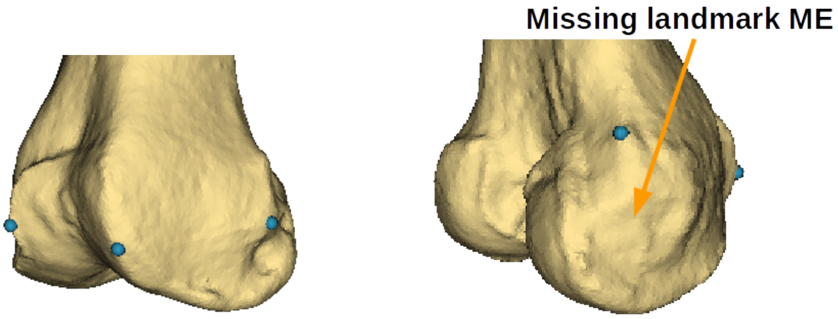
Looking at the standard deviation of the landmarks marked by the surgeon (table 3.5), showed that it was largest for the ME and smallest for the LE. Compared with the standard deviation of the medial epicondyle (table 3.3) labeled by the algorithm indicates that landmarks that the surgeon labeled with high variability, are hard to find for the algorithm as well.

#### 3.2.1.3 Numerical robustness of the CBA

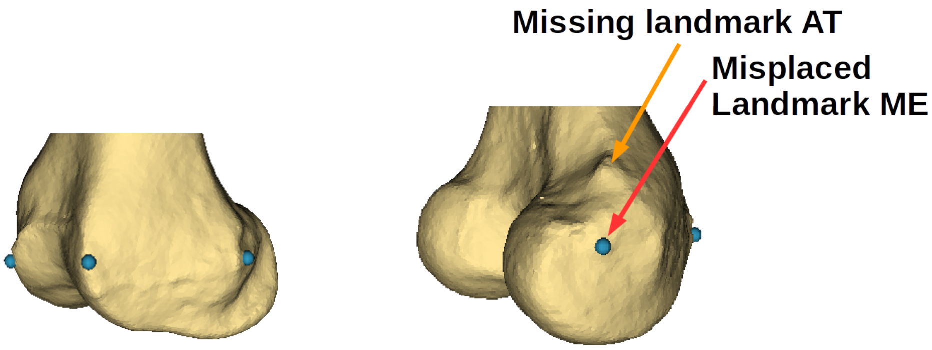
The Curvature Based Algorithm was not able to detect each landmark for each specimen (table 3.6). On four of fifteen specimens the algorithm was able to detect all landmarks. On 8 landmarks not all five and on three specimens no landmarks could be detected. There are two main reasons why the algorithm can not find landmarks. First, it is possible that other peaks of the surface have similar characteristics as a landmark and is located in similar directions so using adjacency relationships does not work and a landmark is not labeled (figure 3.6 (a)). This can also lead to a wrong point being labeled (figure 3.6 (c)). Second, it is possible that at the location were a landmark is supposed to be, no real peak exists (figure 3.6 (d)). Consequently, the parameter ${}^{\mathrm{L}}K$, ${}^{\mathrm{L}}H$, $n_r$, $n_w$ (chapter 2.3.4), which control the labeling process had to be adjusted for each specimen and the algorithm could nor detect landmarks fully automatically.



(a) Specimen 1



(b) Specimen 3



(c) Specimen 4

Figure 3.6: Selection of specimens where some landmarks could not be detected or wrong points were labeled as landmarks. View from anterior-lateral and posterior-medial side.

The created results were computed with the following parameters:

🟩 Found all landmarks
🟧 Found some landmarks
🟥 Found no landmarks

| Specimen | $^{L}K$ | $^{L}H$ | $n_r$ | $n_w$ | AT | ME | LE | MP | LP | Result |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.15 | 0.15 | 35 | 5 |  | x | x | x | x | 4/5 |
| 2 | 0.1 | 0.1 | 30 | 5 | x |  |  | x | x | 3/5 |
| 3 | 0.1 | 0.2 | 30 | 5 | x |  | x | x | x | 4/5 |
| 4 | 0.1 | 0.2 | 30 | 5 |  | x | x | x | x | 4/5 |
| 5 | 0.05 | 0.5 | 20 | 5 | x |  |  |  |  | 1/5 |
| 6 | 0.15 | 0.25 | 30 | 5 |  |  |  |  |  | 0/5 |
| 7 | 0.25 | 0.35 | 30 | 5 | x | x | x | x | x | 5/5 |
| 8 | 0.15 | 0.25 | 30 | 5 |  |  |  |  |  | 0/5 |
| 9 | 0.1 | 0.25 | 30 | 5 | x | x | x | x | x | 5/5 |
| 10 | 0.1 | 0.1 | 30 | 5 |  |  |  |  |  | 0/5 |
| 11 | 0.2 | 0.3 | 20 | 6 | x | x |  | x | x | 4/5 |
| 12 | 0.05 | 0.1 | 35 | 3 | x | x | x | x | x | 5/5 |
| 13 | 0.1 | 0.1 | 20 | 5 |  | x | x | x | x | 4/5 |
| 14 | 0.15 | 0.25 | 40 | 5 |  | x |  |  |  | 1/5 |
| 15 | 0.4 | 0.6 | 30 | 5 | x | x | x | x | x | 5/5 |
| $\sum$ |  |  |  |  | 8 | 9 | 8 | 10 | 10 |  |

Table 3.6: For labeling landmarks on the femur the parameter $^{L}K$, $^{L}H$, $n_r$, $n_w$ (chapter 2.3.4), which control the labeling process, had to be adjusted. AT, ME, LE, MP, LP (chapter 1.3) depict if a landmark was detected for a specimen.

Detecting anatomical landmarks on the femur took up to 8-12 minutes, if the curvature values were already computed (table 3.7).

| **Work step** | **Required time** |
|---|---|
| Loading mesh and creating surface normals | 5-7 minutes |
| Load curvature values | 1 minute |
| Grow potential regions | 1-2 minutes |
| Label landmark regions | 1-2 minute |
| Visualization of result | $<$ 1 minute |

Table 3.7: Required time for detecting anatomical landmarks on the left Femur

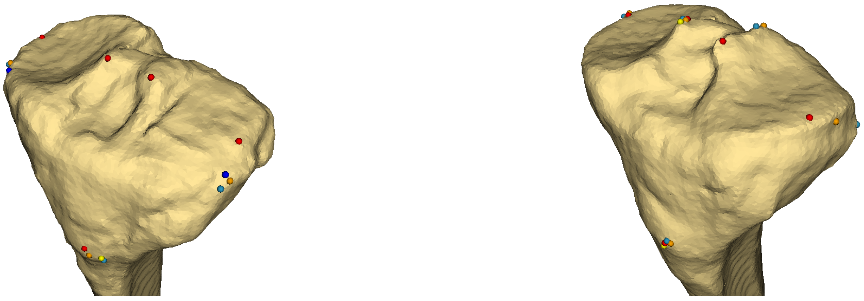
## 3.2.2 Tibia

### 3.2.2.1 Validation of labeling methods tibia

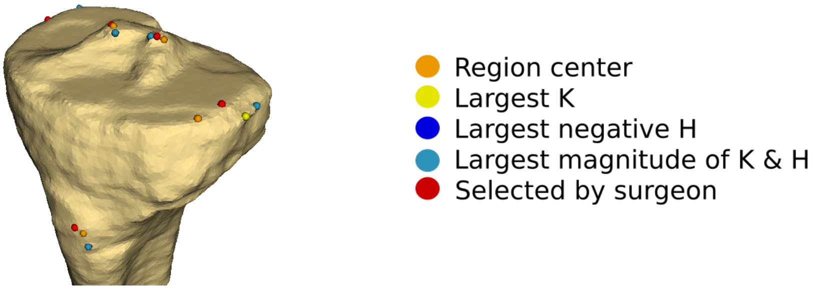
Four different labeling methods were used to compute results on tibias of seven different specimens (figure 3.7 and 3.8). In table (table 3.8) the mean Euclidean distances, standard deviations (chapter 2.3.5) and the maximal and minimal distance of each labeling method and each landmark are depicted. In order to compute distances and standard deviations landmarks labeled by an experienced orthopedic surgeon are used as gold standard.

(a) Specimen 1 (b) Specimen 2

(c) Specimen 7 (d) Specimen 8



(e) Specimen 12 (f) Specimen 13



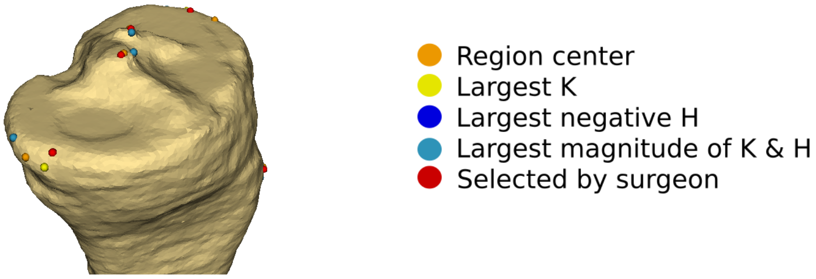
(g) Specimen 15

Figure 3.7: Selection of differently labeled landmarks on the tibia, view from anterior-lateral-proximal side.

(a) Specimen 1 (b) Specimen 2

(c) Specimen 7 (d) Specimen 8

(e) Specimen 12 (f) Specimen 13



(g) Specimen 14

Figure 3.8: Selection of differently labeled landmarks on the tibia, view from anterior-medial-proximal side.

| **Method** | **Mean** [mm] | **Std** [mm] | **Max** [mm] | **Min** [mm] |
|---|---|---|---|---|
| Landmark TT | | | | |
| 1. center | 3.14 | 1.27 | 5.1 | 1.64 |
| 2. gauss | 4.98 | 2.1 | 6.63 | 1.36 |
| 3. mean | 6.15 | 3.0 | 9.13 | 1.31 |
| 4. both | 6.67 | 2.97 | 9.8 | 1.31 |
| Landmark MIT | | | | |
| 1. center | 1.90 | 0.75 | 3.13 | 1.23 |
| 2. gauss | 3.56 | 0.54 | 4.46 | 3.03 |
| 3. mean | 3.43 | 1.36 | 5.00 | 1.26 |
| 4. both | 3.04 | 0.68 | 3.53 | 1.88 |
| Landmark LIT | | | | |
| 1. center | 5.13 | 6.74 | 15.19 | 0.82 |
| 2. gauss | 4.99 | 4.87 | 12.21 | 1.53 |
| 3. mean | 4.29 | 5.28 | 12.21 | 1.51 |
| 4. both | 4.63 | 5.12 | 12.21 | 1.51 |
| Landmark MP | | | | |
| 1. center | 7.58 | 5.92 | 16.62 | 2.3 |
| 2. gauss | 5.72 | 6.38 | 16.91 | 0.88 |
| 3. mean | 8.20 | 7.96 | 20.98 | 1.95 |
| 4. both | 5.94 | 6.19 | 16.91 | 1.95 |
| Landmark LP | | | | |
| 1. center | 6.57 | 2.59 | 8.15 | 2.7 |
| 2. gauss | 11.39 | 9.29 | 25.2 | 5.77 |
| 3. mean | 12.54 | 8.51 | 25.2 | 6.94 |
| 4. both | 11.68 | 9.06 | 25.2 | 6.15 |

Table 3.8: Mean Euclidean distance, standard deviation and maximal and minimal distances between four differently labeled points from the implemented algorithm (chapter 2.3.3) and the surgeon for all 5 landmarks (chapter 1.3) of 7 tibias.

| **Method** | **Mean** [mm] | **Std** [mm] | **Max** [mm] | **Min** [mm] |
|---|---|---|---|---|
| 1. center | 4.86 | 2.35 | 2.70 | 0.82 |
| 2. gauss | 6,07 | 3.04 | 5.77 | 0.88 |
| 3. mean | 6,92 | 3.64 | 6.94 | 1.26 |
| 4. both | 6,39 | 3.26 | 6.15 | 1.31 |

Table 3.9: Computed mean distances, standard deviations and maximal and minimal distances for each labeling method applied to tibias.

Looking at the computed mean Euclidean distances (table 3.9 and 3.8), indicates that no method could be generally favored for all tibial landmarks. Because the labeling process did not work stably, most of the following results were labeled manually out of a list of potential regions. During this process, the method used was adjusted specifically for each sample.

#### 3.2.2.2 Operator error of surgeon

To determine the operator error of landmarks of the tibia labeled by an orthopedic surgeon, the average of 8 different labelings (figure 2.3) on one specimen was calculated. Following the mean euclidean distance, standard deviation and maximal and minimal distance between the average and each of those individual labelings (table 3.10) were computed.

| **Tibia - Specimen 2** | | | | |
|---|---|---|---|---|
| **Landmark** | **Mean** [mm] | **Std** [mm] | **Max** [mm] | **Min** [mm] |
| LP | 2.14 | 1.33 | 3.73 | 0.94 |
| MP | 1.83 | 1.69 | 3.82 | 0.39 |
| LIT | 2.13 | 1.00 | 3.50 | 1.10 |
| MIT | 1.19 | 0.83 | 3.04 | 0.33 |
| TT | 1.75 | 1.08 | 2.65 | 0.31 |

Table 3.10: Mean euclidean distance, standard deviation, and maximal and minimal distance between the average of 8 labelings and each of those individual labelings of the tibia ofspecimen 2 performed by the surgeon.

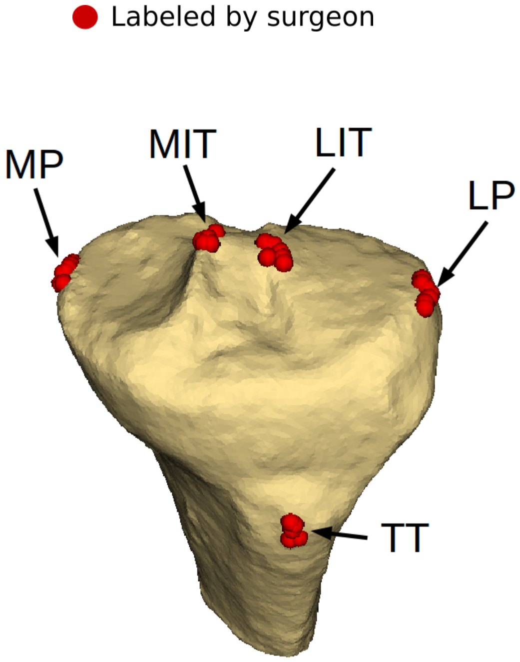
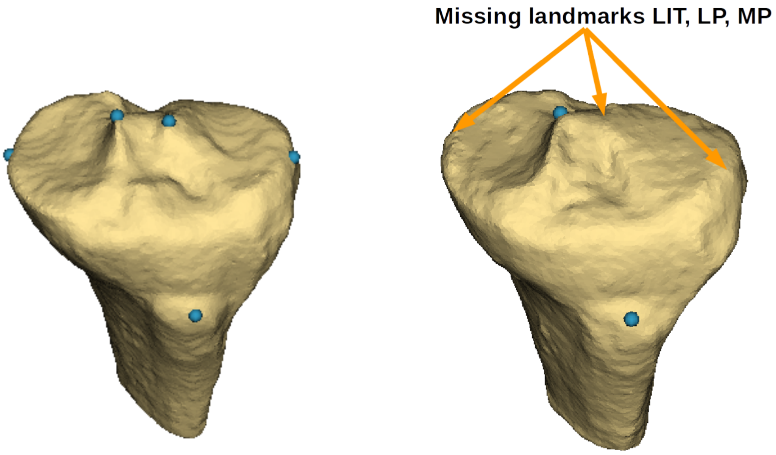


Figure 3.9: Scattering landmarks labeled by an experienced orthopedic surgeon of the tibia of specimen 2. View from anterior-proximal side.

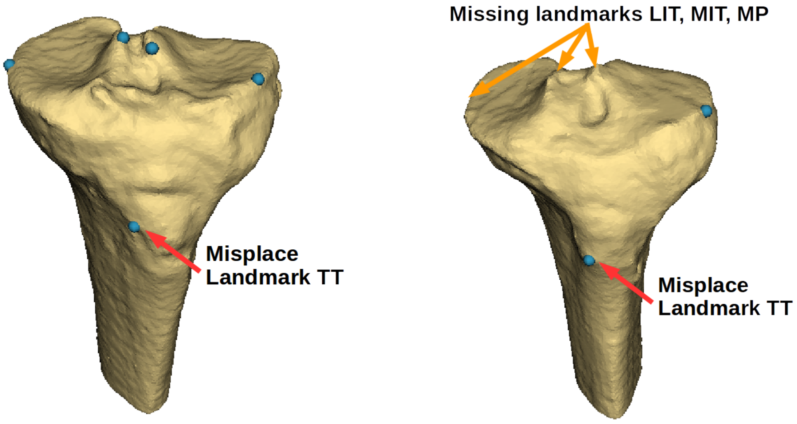
As with the femur, the points of the surgeon were scattered at the tibia, with the standard deviations being very similar for each point (table 3.10). Comparing with table 3.8 shows, that the standard deviations of the algorithm are much larger and vary between the different landmarks.

#### 3.2.2.3 Numerical robustness of the CBA

The Curvature Based Algorithm was not able to detect all five landmarks for all specimens (table 3.11). On six of fifteen specimens all landmarks could be found. On seven specimens not all five and on two specimens no landmarks could be located. Depending on how strong or weak a peak was, the algorithm was unable to detect a certain landmark, because no appropriate limits ${}^{\mathrm{L}}K$ and ${}^{\mathrm{L}}H$ could be found. (figure 3.10 (b) and (d)). Especially because the peaks of the tibia were very different from each other, detecting all peaks in the right place was not possible for each specimen (figure 3.10 (c)). If all landmarks could be computed the landmarks were clearly recognizable peaks on the surface (figure 3.10 (a)). Manual interventions, by changing the parameters ${}^{\mathrm{L}}K$, ${}^{\mathrm{L}}H$, $n_r$, $n_w$ (chapter 2.3.4) that control the labeling algorithm, were necessary for all specimens to compute results



(a) Specimen 1 (b) Specimen 2



(c) Specimen 3 (d) Specimen 4

Figure 3.10: Selection of specimens where some landmarks could not be detected, or were located in a wrong place. View from anterior-proximal side.

The created results were computed with the following parameters:

🟩 Found all landmarks
🟧 Found some landmarks
🟥 Found no landmarks

| Specimen | $^{L}K$ | $^{L}H$ | $n_R$ | $n_W$ | Meth | TT | MP | LP | MIT | LIT | Result |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.1 | 0.15 | 50 | 7 | center | x | x | x | x | x | 5/5 |
| 2 | 0.1 | 0.15 | 20 | 5 | both | x |  |  | x |  | 2/5 |
| 3 | 0.05 | 0.15 | 20 | 5 | both | x | x | x | x | x | 5/5 |
| 4 | 0.1 | 0.2 | 20 | 5 | both |  |  |  |  |  | 0/5 |
| 5 | 0.15 | 0.25 | 20 | 5 | both | x | x |  |  |  | 2/5 |
| 6 | 0.15 | 0.25 | 10 | 5 | both | x |  | x | x | x | 4/5 |
| 7 | 0.1 | 0.18 | 10 | 5 | both | x | x | x | x | x | 5/5 |
| 8 | 0.1 | 0.2 | 70 | 5 | center | x | x | x | x | x | 5/5 |
| 9 | 0.05 | 0.05 | 5 | 5 | both |  |  |  |  |  | 0/5 |
| 10 | 0.15 | 0.2 | 50 | 5 | both | x | x | x | x |  | 4/5 |
| 11 | 0.1 | 0.15 | 50 | 5 | both | x | x |  | x | x | 4/5 |
| 12 | 0.15 | 0.2 | 50 | 5 | center | x | x | x |  |  | 3/5 |
| 13 | 0.1 | 0.15 | 50 | 7 | center | x | x | x | x | x | 5/5 |
| 14 | 0.1 | 0.15 | 20 | 7 | center | x |  | x | x | x | 4/5 |
| 15 | 0.1 | 0.15 | 50 | 7 | center | x | x | x | x | x | 5/5 |
| $\sum$ |  |  |  |  |  | 13 | 11 | 10 | 10 | 10 |  |

Table 3.11: For labeling landmarks on the tibia the parameters $^{L}K$, $^{L}H$, $n_r$, $n_w$ (chapter 2.3.4), which control the labeling process had to be adjusted. TT, MP, LP, MIT, LIT (chapter 1.3) show if the landmark was detected. Meth defines which labeling method (chapter 2.3.3) was used to label the landmark point in the landmark region.

For the tibia the automatic labeling process was turned off, because no adequate results could be computed. Therefore, manual landmark labeling from regions of interest depicted on the bone surface was performed. For this the manual labeling process, the following parameters computed adequate results for all specimens (table 3.12):

| **Parameter** | **Value** |
|---|---|
| Number of elements $n_r$ | 50 |
| Gaussian curvature limit $^{L}K$ | 0.1 |
| Mean curvature limit $^{L}H$ | 0.1 |

Table 3.12: Default settings for manual labeling of landmarks on the tibia

The whole landmark labeling process took up to 5-9 minutes, if the curvature values were already computed (table 3.13).

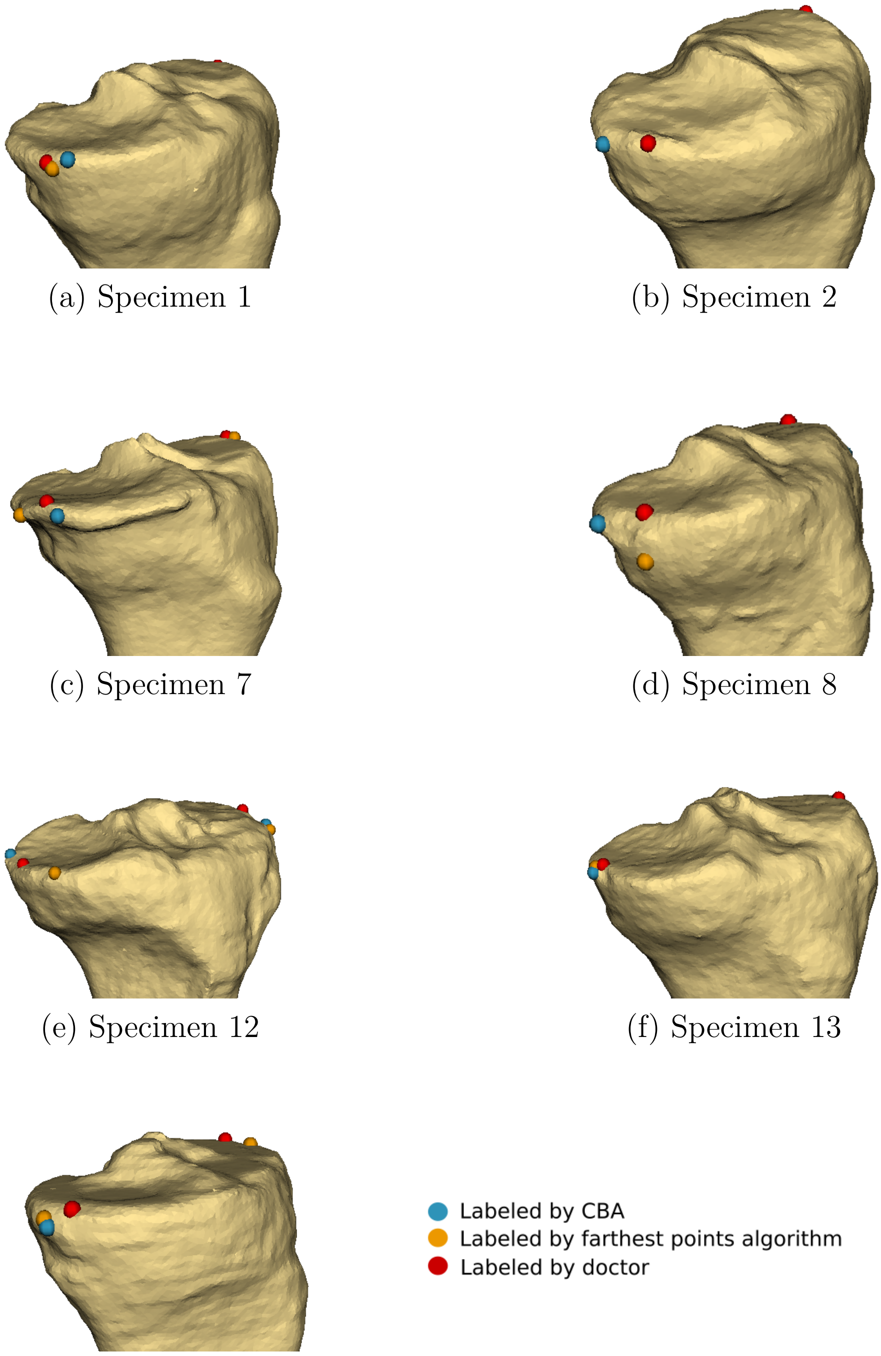
| **Work step** | **Required time** |
|---|---|
| Loading mesh and creating surface normals | 3-4 minutes |
| Load curvature values | 1 minute |
| Grow potential regions | 1-2 minutes |
| Label landmark regions | 1-2 minute |
| Visualization of result | < 1 minute |

Table 3.13: Required time for detecting anatomical landmarks on the left Tibia

## 3.3 Farthest points of tibial plateau

### 3.3.1 Validation of farthest points method

The algorithm for determining the farthest points of the tibial plateau was intended to be an alternative approach for the determination of the lateral and medial peaks of the tibial plateau. Therefore, the computed points were compared with the lateral and medial peaks labeled by the orthopedic surgeon and the CBA (figures 3.11 and 3.12). For one specimen, only the points from the farthest points algorithm and the surgeon were compared, because the CBA could not detect those landmarks. To compare the accuracy of both methods the Euclidean distances between the labeling of the surgeon and both algorithms were computed (table 3.14). The results of the CBA were computed with respect to table 3.6.



(a) Specimen 1
(b) Specimen 2
(c) Specimen 7
(d) Specimen 8
(e) Specimen 12
(f) Specimen 13
(g) Specimen 15

Figure 3.11: Selection of lateral and medial peaks of the tibial plateau labeled with different approaches, view from anterior-medial side.

(a) Specimen 1 (b) Specimen 2

(c) Specimen 7 (d) Specimen 8

(e) Specimen 12 (f) Specimen 13

(g) Specimen 15

Figure 3.12: Selection of lateral and medial peaks of the tibial plateau labeled with different approaches, view from anterior-lateral side.

Computed values for evaluating the accuracy of the computed points:

| | **Distance [mm]** | | | |
|---|---|---|---|---|
| | **CBA** | | **Farthest points** | |
| Specimen | LP | MP | LP | MP |
| 1 | 7.19 | 2.55 | 6.33 | 8.08 |
| 2 | | | 8.89 | 12.71 |
| 7 | 10.52 | 2.34 | 6.23 | 9.03 |
| 8 | 9.03 | 9.45 | 14.87 | 11.68 |
| 12 | 11.58 | 16.62 | 9.04 | 10.67 |
| 13 | 2.30 | 7.61 | 3.24 | 9.18 |
| 15 | 2.34 | 1.60 | 7.34 | 6.59 |

Table 3.14: Euclidean distances between landmarks labeled by the surgeon and by the CBA as well as the farthest points algorithm.

| **Algorithm** | **Mean [mm]** | **Std [mm]** | **Max [mm]** | **Min [mm]** |
|---|---|---|---|---|
| Landmark LP | | | | |
| CBA | 7.07 | 4.03 | 11.58 | 2.30 |
| Farthest point | 7.84 | 3.34 | 14.87 | 3.24 |
| Landmark MP | | | | |
| CBA | 7.53 | 5.81 | 16.62 | 1.60 |
| Farthest points | 9.20 | 1.97 | 12.71 | 6.59 |

Table 3.15: Mean distances, standard deviation and maximal and minimal distances of the lateral and medial peak labeled by the CBA as well as the farthest point algorithm, compared with landmarks labeled by the surgeon.

The mean distance and standard deviations (table 3.15) of the distances computed between points labeled by the CBA, the farthest points algorithm and points labeled by the surgeon (table 3.14), indicated that using the CBA to detect those points is generally more appropriate.

### 3.3.2 Numerical robustness of tibial farthest points computation

Computing the farthest points of the tibial plateau delivered results for all 15 specimens. Except for five specimens the default values could be used (table 3.16). The used default values were:

| **Parameter** | **Value** |
|---|---|
| Number of elements $n_\mathrm{r}$ | 10 |
| Gaussian curvature limit ${}^{\mathrm{L}}K_{\mathrm{Ft}}$ | 0.1 |
| Window size plateau $n_\mathrm{p}$ | 18 |

Table 3.16: Default settings for computing farthest points of the tibial plateau (chapter 2.4.2).

For five specimens the parameter window size plateau $n_\mathrm{p}$ (explained in chapter 2.4.2) had to be changed from $n_\mathrm{p} = 18$ (table 3.16) in order to compute a result (table 3.17).

| Specimen | Window size plateau $n_\mathrm{p}$ |
|---|---|
| 1 | 15 |
| 4 | 12 |
| 5 | 12 |
| 9 | 25 |
| 11 | 10 |
| 12 | 12 |

Table 3.17: From the default deviating settings for the farthest points of a left tibia.

If a wrong value for the window size plateau parameter $n_\mathrm{p}$ (chapter 2.4.2) was used the computed points were not part of the contour of the tibial plateau and therefore misplaced (figure 3.13). Consequently, the parameter $n_\mathrm{p}$ had to be changed to include all contour regions.

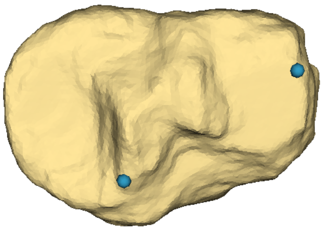


Figure 3.13: Wrong farthest points of the tibial plateau due to a to small value for the parameter window size plateau $n_\mathrm{p}$. View from proximal side.

Computing the farthest points of tibial plateau required 5-7 minutes, if the curvature values were already computed (figure 3.18).

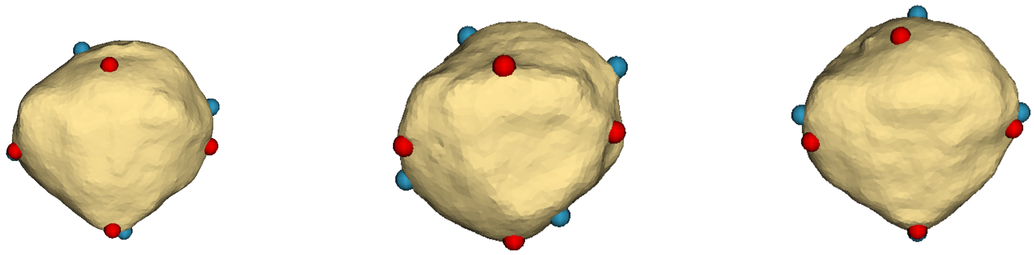
| Work step | Required time |
|---|---|
| Loading mesh and creating surface normals | 3-4 minutes |
| Load curvature values | 1 minute |
| Grow potential regions | 1 minutes |
| Define the farthest points | < 1 minute |

Table 3.18: Required time for detecting farthest points of condyles of the left Tibia.

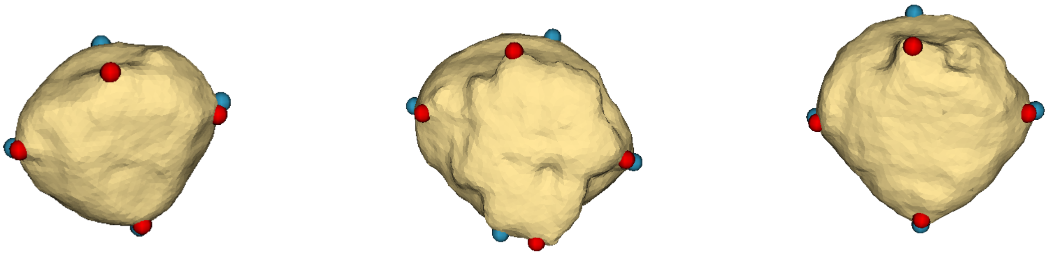
## 3.4 Farthest points of patella

### 3.4.1 Validation of farthest points method patella

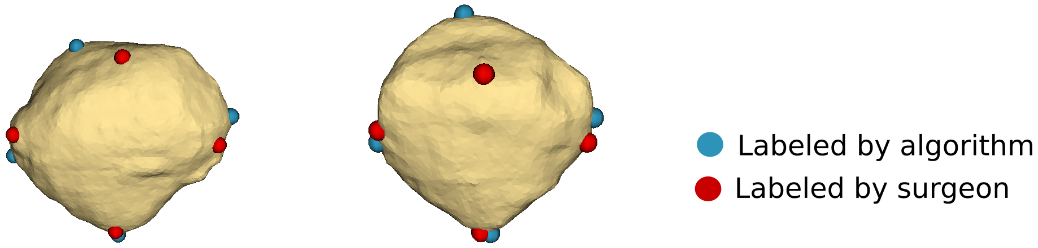
The algorithm determining the farthest points of the patella was supposed to detect the most proximal, distal, medial and lateral points of the anterior side of the patella. The lines between two of those points had to be orthogonal (figure 3.14). To determine the accuracy of these results, the mean euclidean distances between the landmarks of the farthest points algorithm and the surgeon were calculated (table 3.19).



(a) Specimen 1 (b) Specimen 2 (c) Specimen 7



(d) Specimen 8 (e) Specimen 9 (f) Specimen 12



(g) Specimen 13 (h) Specimen 15

Figure 3.14: Selection of specimens with farthest points of the patella labeled by the surgeon and the algorithm, view from anterior side.

Computed values for evaluating the accuracy of the computed points:

| | **Distance** [mm] | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **Patella - Specimen** | | | | | | | |
| **Landmarks** | **1** | **2** | **7** | **8** | **9** | **12** | **13** | **15** |
| Most proximal point | 11.20 | 4.00 | 9.68 | 14.91 | 16.18 | 16.41 | 14.53 | 17.53 |
| Most distal point | 5.26 | 2.22 | 2.80 | 1.38 | 7.97 | 3.02 | 3.76 | 2.05 |
| Most medial point | 4.09 | 3.00 | 5.44 | 2.73 | 8.74 | 2.30 | 10.56 | 3.68 |
| Most lateral point | 8.73 | 3.04 | 3.39 | 2.89 | 6.54 | 2.55 | 7.67 | 4.76 |

Table 3.19: Euclidean distance between farthest points of patella labeled by a surgeon and the algorithm

| **Landmark** | **Mean [mm]** | **Std [mm]** | **Max [mm]** | **Min [mm]** |
|---|---|---|---|---|
| Most proximal point | 13.05 | 4.53 | 17.53 | 4.00 |
| Most distal point | 3.56 | 2.14 | 7.97 | 2.05 |
| Most medial point | 5.07 | 3.03 | 10.56 | 2.30 |
| Most lateral point | 4.95 | 2.40 | 8.73 | 2.89 |

Table 3.20: Mean distances, standard deviation and maximal and minimal distances of the farthest points of the patella labeled by the CBA, compared with landmarks labeled by the surgeon.

The calculated mean Euclidean distances (table 3.20) indicated, that for the most proximal point of the anterior side a different point was labeled by the algorithm.

### 3.4.2 Operator error of surgeon

To determine the operator error of the landmarks labeled by an experienced orthopedic surgeon, the average of 8 different labelings was computed. Following the mean distance, the standard deviation and the maximal and minimal distance between the average and each single labeling was computed (table 3.21).

| **Patella - Specimen 2** | | | | |
|---|---|---|---|---|
| **Landmark** | **Mean** [mm] | **Std** [mm] | **Max** [mm] | **Min** [mm] |
| Most proximal point | 0.62 | 0.60 | 0.94 | 0.26 |
| Most distal point | 1.69 | 0.80 | 2.70 | 0.64 |
| Most medial point | 2.05 | 1.76 | 3.50 | 1.38 |
| Most lateral point | 2.35 | 1.04 | 3.47 | 1.03 |

Table 3.21: Mean euclidean distance, standard deviation, and maximal and minimal distance between the average of 8 labelings and each of those individual labelings of the patella of specimen 2 performed by the surgeon.

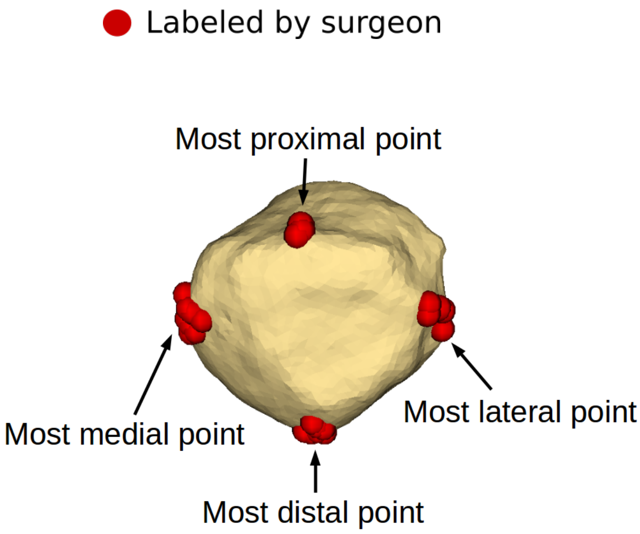


Figure 3.15: Scattering farthest points labeled by an experienced surgeon of patella of specimen 2 (table 3.21). View from anterior side.

A look at the standard deviations (table 3.21) showed that the points labeled by the surgeon fluctuated.

### 3.4.3 Numerical robustness of patellar farthest points computation

Computing the farthest points of the patella delivered results for all 15 specimens. Except for eight specimens the default values could be used (table 3.22):

| **Parameter** | **Value** |
|---|---|
| Number of elements $n_\mathrm{r}$ | 10 |
| Gaussian curvature limit ${}^{\mathrm{L}}K_{\mathrm{Fp}}$ | 0.2 |
| Dot product factor $n_\mathrm{df}$ | 2 |

Table 3.22: Default settings for computing farthest points of the patella (chapter 2.4.3).

For eight specimens the parameter dot product factor $n_\mathrm{df}$ (explained in chapter 2.4.3) had to be changed from $n_\mathrm{df} = 2$ (table 3.22) in order to compute a result (table 3.23).

| **Specimen** | **Dot product factor $n_{\mathrm{df}}$** |
|---|---|
| 1 | 0.5 |
| 2 | 0.5 |
| 3 | 0.5 |
| 4 | 5 |
| 6 | 10 |
| 10 | 5 |
| 12 | 5 |
| 14 | 15 |

Table 3.23: From the default deviating settings for the farthest points of a left Patella.

If a wrong dot product factor $n_{\mathrm{df}}$ (explained in chapter 2.4.3) was used, it may not be possible to actually label the farthest points, because the requirement of orthogonality prevented this (figure 3.16). Therefore, the dot product factor $n_{\mathrm{df}}$ had to be changed, to actually label the points with the largest distance.

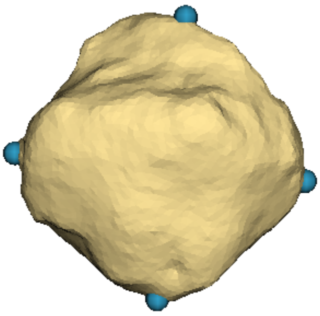


Figure 3.16: Wrong farthest points of the patella due to a too strict dot product factor $n_{\mathrm{df}}$. View from anterior side.

Computing farthest points of the patella was relatively fast, because the patella is a rather small bone. Hence, the computation of the farthest points took up 2-4 Minutes. (table 3.24)

| **Work step** | **Required time** |
|---|---|
| Loading mesh and creating surface normals | 1-2 minutes |
| Define first pair of farthest points | $<$ 1 minute |
| Define second pair of farthest points | $<$ 1 minutes |
| Visualize Results | $<$ 1 minute |

Table 3.24: Required time for detecting farthest points of the patella.

## 3.5 Trochlear groove detection

Computing the path of the trochlear groove delivered results for all 15 specimens. The results were computed with the following default values.

| **Parameter** | **Value** |
|---|---|
| Number of elements $n_{\mathrm{r}}$ | 150 |
| Gaussian curvature limit ${}^{\mathrm{L}}K_{\mathrm{Tg}}$ | 0.01 |
| Mean curvature limit ${}^{\mathrm{L}}H_{\mathrm{Tg}}$ | 0.55 |

Table 3.25: Default settings for computing the trochlear groove of a left femur (chapter 2.4.4).

Results of four specimens are shown in figure 3.17.

(a) Specimen 1 (b) Specimen 2

(c) Specimen 3 (d) Specimen 4

Figure 3.17: Selection of path of the trochlear groove, view from anterior side.

For four specimens, the mean curvature limit had to be changed from ${}^{\mathrm{L}}H_{\mathrm{Tg}} = 0.55$ in order to detect the patella valley and to create results (table 3.26).

| **Specimen** | ${}^{\mathrm{L}}H_{\mathrm{Tg}}$ |
|---|---|
| 1 | 0.77 |
| 3 | 1.05 |
| 8 | 0.75 |
| 14 | 0.35 |

Table 3.26: From the default deviating settings for the Trochlear Groove of a left Femur

If the ${}^{\mathrm{L}}H_{\mathrm{Tg}}$ was set incorrectly, it happened that not just the patella valley but a larger region, or a too small region was detect. In this case, the path of the patella valley was misplaced. If the detected region was too large, the ${}^{\mathrm{L}}H_{\mathrm{Tg}}$ had to be decreased. If the region was too little, the ${}^{\mathrm{L}}H_{\mathrm{Tg}}$ had to be increased.

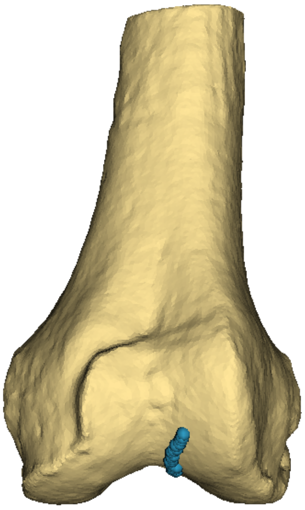


Figure 3.18: Wrong path of the trochlear groove due to a too big region because of a wrong mean curvature limit ${}^{\mathrm{L}}H_{\mathrm{Tg}}$. View from anterior side.

Detecting the path of the trochlear groove took up to 9 till 13 minutes, if the curvature values were already computed and stored (table 3.27).

| **Work step** | **Required time** |
|---|---|
| Loading mesh and creating surface normals | 5-7 minutes |
| Load curvature values | 1 minute |
| Grow potential regions | 1-3 minutes |
| Define Trochlear Groove region | < 1 minute |
| Detecte valley path of Trochlear Groove | 1-2 minutes |
| Visualization of result | <1 minute |

Table 3.27: Required time for detecting the trochlear groove of a left femur.

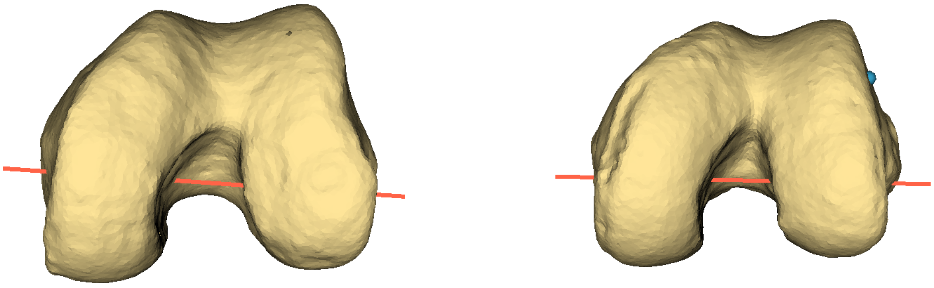
## 3.6 Cylindrical axis detection

Computing the cylindrical axis of the posterior femoral condyles delivered results for all 15 specimens. The results were computed with the following parameters (table 3.28):

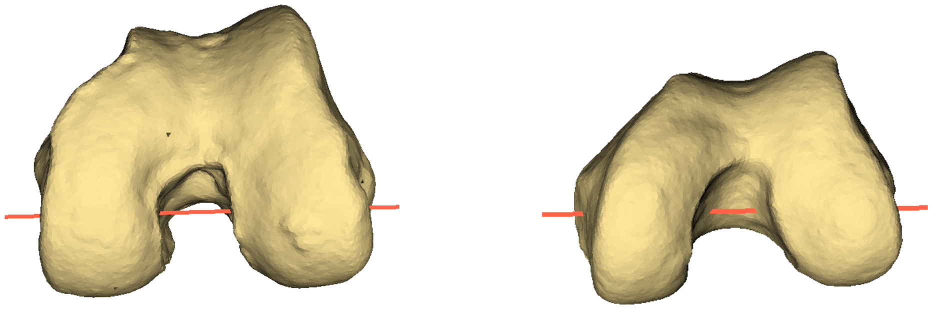
| **Parameter** | **Value** |
|---|---|
| Number of elements $n_{\mathrm{r}}$ | 300 |
| Guassian curvature limit ${}^{\mathrm{L}}K_{\mathrm{Ca}}$ | 0.2 |

Table 3.28: Default settings for computing the cylindrical axis of a left femur (chapter 2.4.5).

Results of four different specimens can be seen in figure 3.19.



(a) Specimen 1 (b) Specimen 2



(c) Specimen 3 (d) Specimen 4

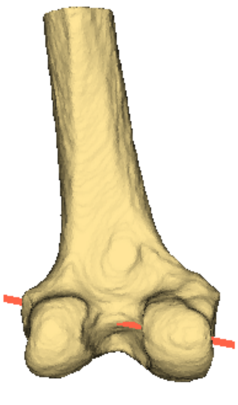
Figure 3.19: Selection of cylindrical axis of the posterior femoral condyles. View from distal direction.

For six specimens, the Gaussian curvature limit had to be changed from ${}^{\mathrm{L}}K_{\mathrm{Ca}} = 0.2$ to detect the contour of the condyles and in order to create results (table 3.29).

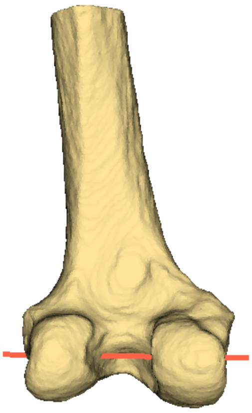
| **Specimen** | ${}^{\mathrm{L}}K_{\mathrm{Ca}}$ |
|---|---|
| 1 | 0.16 |
| 2 | 0.17 |
| 3 | 0.15 |
| 6 | 0.12 |
| 12 | 0.22 |
| 14 | 0.15 |

Table 3.29: Settings deviating from the default values for the cylindrical axis of the femur

If ${}^{\mathrm{L}}K_{\mathrm{Ca}}$ was set incorrectly, it happened that too little of the contour or the contour and adjacent regions were detected. In both cases, this resulted in a wrong axis (figure 3.17). If the detected region was too large, the ${}^{\mathrm{L}}K_{\mathrm{Ca}}$ had to be increased. If the region was too little, ${}^{\mathrm{L}}K_{\mathrm{Ca}}$ had to be decreased.



(a) Specimen 12 wrong cylindrical axis



(b) Specimen 12 correct cylindrical axis

Figure 3.17: Different values of ${}^{\mathrm{L}}K_{\mathrm{Ca}}$ deliver different regions, which are used to compute the cylindrical axis. (a) the default settings provided too big regions which therefore leads to a wrong cylindrical axis. (b) Changing ${}^{\mathrm{L}}K_{\mathrm{Ca}}$ in a way that the for computation used regions are more fitting delivers a adequate cylindrical axis. View from posterior side.

Detecting the cylindrical axis took up to 8 till 13 minutes, if the curvature values were already computed and stored (table 3.30).

| **Work step** | **Required time** |
|---|---|
| Loading mesh and creating surface normals | 5-7 minutes |
| Load curvature values | 1 minute |
| Grow potential regions | 1-3 minutes |
| Define the two condyle regions | $<$ 1 minute |
| Compute fit | $<$ 1 minute |
| Visualization of result | $<$ 1 minute |

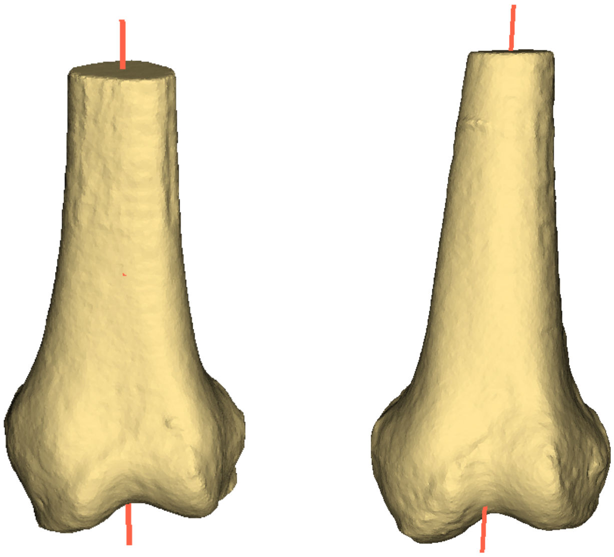
Table 3.30: Required time for the detecting cylindrical axis of the femur.

## 3.7 Shaft axis detection

### 3.7.1 Femur

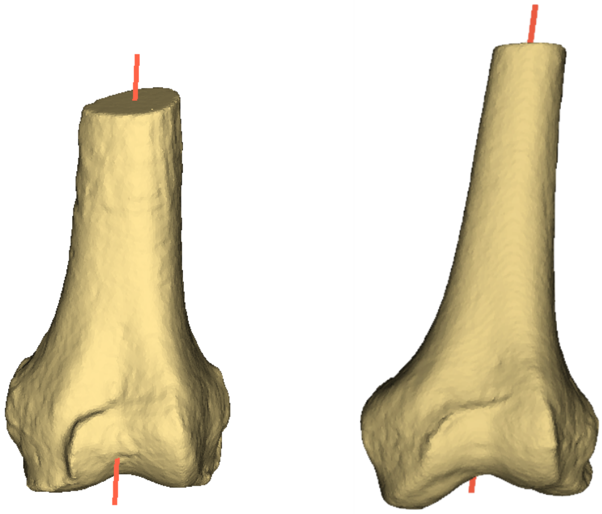
Computing the shaft axis of the femur delivered results for all 15 specimens. As default value of the computation a percentage $n_{\mathrm{s}} = 50\%$ (explained in chapter 3.7) was used.

The obtained results of four different specimens can be seen in figure 3.21.



(a) Specimen 1

(b) Specimen 2



(c) Specimen 3

(d) Specimen 5

Figure 3.21: Selection of specimens with shaft axis of the femur, view from anterior side.

For seven specimens the default value of percentage $n_s$ (explained in chapter 2.4.6) had to be changed, because the cylindrical part of the femur was longer or shorter than 50 % of the total bone length (table 3.31). If the length of the bone differed, the computed axis could be in another place then intended, when the shaft of the bone was curved. Therefore, the percentage $n_s$ parameter had to be changed.

| **Specimen** | **Percentage $n_\mathrm{s}$** |
|---|---|
| 4 | 75% |
| 5 | 65% |
| 8 | 75% |
| 10 | 70% |
| 11 | 55% |
| 12 | 70% |
| 14 | 75% |

Table 3.31: From the default deviating settings for the shaft axis of a left femur

The computation time for detecting the shaft axis was 5-8 minutes (table 3.32).

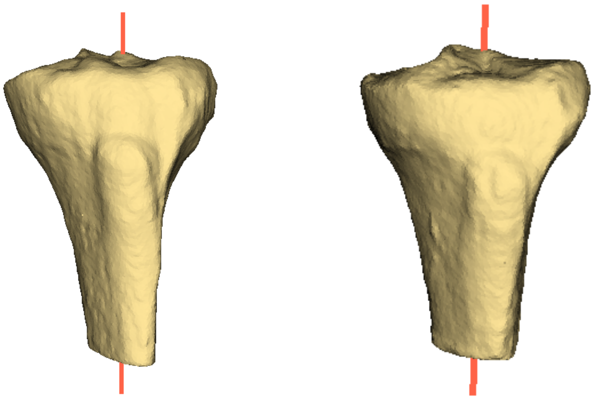
| **Work step** | **Required time** |
|---|---|
| Loading mesh and creating surface normals | 5-7 minutes |
| Define slices | $<$ 1 minutes |
| Compute fit through slices | $<$ 1 minutes |
| Visualization of result | $<$ 1 minute |

Table 3.32: Required time for detecting the shaft axis of a left femur.

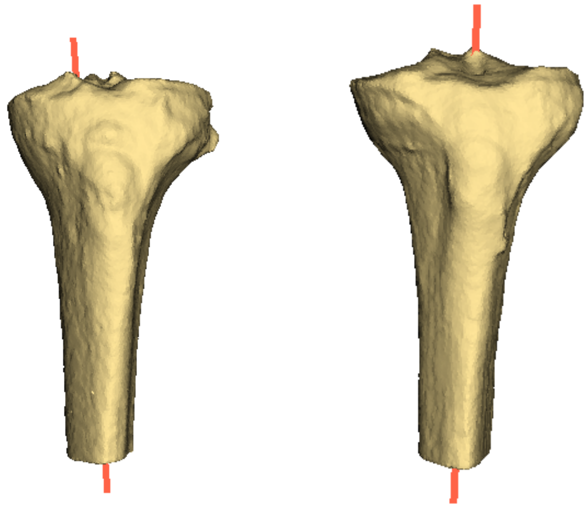
### 3.7.2 Tibia

Computing the shaft axis of the tibia delivered results for all 15 specimens. For the computation a default value for the percentage parameter $n_\mathrm{s} = 40$ % (explained in chapter 2.4.6) was used.

The computed results of four different specimens can be seen in figure 3.22.



(a) Specimen 1 (b) Specimen 2



(c) Specimen 3 (d) Specimen 4

Figure 3.22: Selection of specimens with shaft axis of the tibia, view from anterior side.

For five different specimens the percentage parameter $n_s$ had to be changed, because the length of the tibia differed (table 3.33).

| **Specimen** | **Percentage $n_s$** |
|---|---|
| 3 | 70% |
| 4 | 60% |
| 5 | 30% |
| 7 | 70% |
| 9 | 80% |

Table 3.33: From the default deviating settings for the shaft axis of a left tibia

The computation of the shaft axis of the tibia took around 4-6 minutes (table 3.34).

| **Work step** | **Required time** |
|---|---|
| Loading mesh and creating surface normals | 3-4 minutes |
| Define slices | $<$ 1 minutes |
| Compute fit through slices | $<$ 1 minutes |
| Visualization of result | $<$ 1 minute |

Table 3.34: Required time for detecting shaft axis of a left tibia.

# Chapter 4

# Discussion

Landmarks are important for different fields in medicine. An existing approach was implemented, tested, extended and applied to 15 different specimens to detect such landmarks nearly automatically. Appropriate parameter sets were found for these algorithms to extract anatomical landmarks, additional characteristics and axes. Not all computations delivered an adequate result for each specimen. Landmarks labeled by an experienced orthopedic surgeon are used as gold standard to evaluate computed results of anatomical landmarks.

The quality of the extracted surface meshes from CT-Images is very important as they are the basis for all subsequent computatoins. The mesh element size and the mesh smoothing filter settings are crucial for computing adequate meshes. Especially the Taubin smoothing filter involves the risk of distorting the curvatures of the surface. The surface smoothing has to be efficient, in a way that the original contour of the surface does not change, but local artifacts get eliminated.

Four different methods were implemented to label a particular point of a landmark region as the landmark. The labeling of the geometric center of the region, maximum Gaussian curvature, maximum negative mean curvature, and the largest product of the curvature values (chapter 2.3.3) was tested. The euclidean mean distances (explained in chapter 2.3.5) were found to be in range of 2.61 to 10.25 mm for the femur and 1.90 to 12.54 mm for the tibia. The results showed that none of the four methods can generally be described as most appropriate. What can be stated is that for different landmarks different methods are preferable.

The computed anatomical landmarks were compared to literature, showing good agreement concerning the distances to the gold standard for most of the landmarks. Computing the adductor magnus tubercle delivered a slightly better result with a mean distance of 2.61 mm which was nearly 36 % more accurate than published by Subburaj et al. [24]. In contrast the median distance of the lateral peak of the anterior femoral condyles with 6.96 mm had more than twice the mean distance as in the same paper. The lateral epicondyle had a mean distance of 5.51 mm which was at least 83 % more inaccurate then in literature [32, 33, 24].

The landmarks labeled by the algorithm are reproducible using the same parameters. In contrast the 8 labelings of the same landmarks on a specimen made by an experienced surgeon, shows that the labelings of the surgeon vary. The standard deviations, characterizing the operator error, of the positions of the landmarks varied in a range 0.735 to 3.69 mm and corresponds to the literature [62]. However, the mean distances between the landmarks labeled by the algorithms and the surgeon are larger than the deviations of the surgeon labeling the same points multiple times.

Automatic labeling of regions that represent bony landmarks using curvature values and adjacency matrices, did not work very stable and required manual interventions. For the automatic labeling not only the size of landmarks but their size in comparison with other natural bumps on the surface were crucial. If the algorithm detected all five anatomical landmark regions and, for example, 50 additional peak regions then the adjacency matrices did not contain enough information for automatic labeling. Furthermore, if the landmarks had varying sizes the algorithm could not be adjusted (chapter 2.3.4) to adequate detect all landmarks at the same time, which led to too much detected regions as well. Consequently, the individual parameters (explained in chapter 2.3.4) that control the labeling process had to adapted eventually and manual interventions were necessary. In general the automatic labeling process did not work stable for the femur and tibia, which made manual labeling necessary.

A different approach based on curvature values was implemented to label anatomical landmarks that often could not be found with the original CBA. The computation of the farthest points of the tibial plateau (chapter 2.4.2) was intended to detect the lateral and medial peaks (LP and MP). The mean distances between the lateral and medial peak computed with the farthest points algorithm were 7.84 mm and 9.2 mm, respectively and at least 22 % more inaccurate than the points labeled by CBA. The detected points of the tibial plateau were detected in a range defined by the window size plateau $n_\mathrm{p}$ (chapter 2.4.2). If $n_\mathrm{p}$ was set too large the detected points often were not on the edge of the plateau but offset in distal direction. Consequently, the computed results were distorted. However, since the lateral and medial peaks of the tibial plateau were often not recognized by the CBA automatically they could be detected by the farthest points algorithm instead in such cases to increase the stability of the labeling process (chapter 2.3.3).

The original Curvature Based Algorithm has been enhanced to detect and process different sized geometric shapes, such as edges and valleys. Those large-scaled geometric shapes were used for the computation of the farthest points of the patella (chapter 2.4.3), of the trochlear groove (chapter 2.4.4) and of the cylindrical axis (chapter 2.4.5). For these large areas with an overall stable curvature good working default parameters could be found, which led to a stable working algorithm. If surrounding areas of the detected geometric structure were curved in a similar way, the parameters that control the region detection (chapter 2.3.4) had to be adapted. This had to be done for 40 % of the specimens detecting the cylindrical axis and for 27% of the specimens detecting the patellar valley. No comparable literature for automatic detection of the trochlear groove and the cylindrical axis were found. However, looking at the movement of the patella during knee flexion, a comparable path as the computed path of the trochlear groove was reported [19]. For the cylindrical axis similar results were shown for manually labeling the contours of the condyles [20]. Computing the farthest points of the patella did not detect the most distal point of the anterior side, which is evident when looking at the mean distance of 13.05 mm of those point. The reason for this is, that the algorithm did not detect only the edges of the anterior side, and therefore computed maximum distances for the posterior side as well. It can be conclude that detecting large-scaled areas of the bone surface using curvature values works stable. However, the subsequent processes are an additional source of error and should be further enhanced. Consequently, the whole algorithm requires an optical check of the results.

Computing the shaft axes of femur and tibia based on information stored in the mesh worked stable. The shaft axis of the femur and the tibia was computed without using curvature values. Due to the fact that the length of the bones represented by the meshes are depending on the size in the original CT-Image the proportion of the shaft of the entire bone varies. To consider the length of the bone for computing the axis, the percentage parameter $n_\mathrm{p}$ was required (chapter 3.7). In general the shaft of the femur as curved. Therefore, different results were obtained for different parts of the bone considered. Consequently, it is a matter of definition and further application which axis is considered adequate. It can be stated that calculations which do not rest on curvature values are more stable.

One limitation was that it was not tested how much axes of the original coordinate system may vary without affecting the algorithms. The algorithms were implemented in a way that allows axes to vary in their position. How far the axes were allowed to vary has not been tested, but it was estimated that changing the axes by 20 degrees would not affect the algorithm. A limited deviation of the axes of the coordinate system could be assumed and was realistic because the original coordinate system was taken from the CT-Images and the knee of a patient is always placed in a similar position in the CT device [63]. Another limitation was that the manual labeling of anatomical landmarks was only done by one surgeon. To make more general statements, a larger database established from more surgeons would be required.

In conclusion, using curvature values to detect and label anatomical landmarks completely automatically turned out to be demanding and could not be done with the implemented algorithms without manual interaction. In many cases an intervention by the operator was necessary to receive results, which limits the use of the CBA. Detecting overall regions and contours of the shape of the bone using curvature values was much more stable than detecting little peaks. However, further developments of such algorithms are promising with respect to future clinical usage.

# Chapter 5

# Outlook

The following considerations can improve the implemented algorithms and make them work more stable.

More information in the adjacency matrices make the whole automatic labeling process more stable. To expand the adjacency matrices more landmarks have to be included into the algorithm. Therefore, points have to be defined that can be found on all specimens based on curvature values and based on other approaches and methods. Applying the landmark extraction process (chapter 2.3.2) with different parameter settings allows to include landmarks of different sizes and shapes. Consequently, more landmarks could be included into the adjacency matrix.

In order to enhance the quality of the results and the computation time, areas in which landmarks are located could be estimated by using statistic mean models [64]. A similar approach that predefines such areas could be a combination with neuronal networks [33]. If the landmark detection algorithm does not have to be used for the whole, but just for predefined parts of the surface, the region detection procedure would not have to sort such a high number of elements. This would lead to less regions that fulfill given limits and the whole labeling process would be much more stable and much faster.

# List of Figures

# List of Tables

# References

[1] Mosby. *Mosby's medical dictionary.* 9th ed. St. Louis: Elsevier, 2013. ISBN: 978-0-323-08541-0.

[2] T. Derek V. Cooke, Elizabeth A. Sled, and R. Allan Scudamore. “Frontal plane knee alignment: a call for standardized measurement.” In: *The Journal of Rheumatology* 34.9 (Sept. 1, 2007), pp. 1796–1801.

[3] Werner Platzer and Gerhard Spitzer. *Bewegungsapparat.* 10., überarb. und erg. Aufl. Taschenatlas der Anatomie in 3 Bänden ; Bd. 1. OCLC: 551847189. Stuttgart: Thieme, 2009. 467 pp.

[4] Sameh M. Amin et al. “Endoscopic Orientation of the Parasellar Region in Sphenoid Sinus with Ill-Defined Bony Landmarks: An Anatomic Study”. In: *Skull Base* 20.6 (Nov. 2010), pp. 421–428.

[5] R. Ewers et al. “Basic research and 12 years of clinical experience in computer-assisted navigation technology: a review”. In: *International Journal of Oral and Maxillofacial Surgery* 34.1 (Jan. 2005), pp. 1–8.

[6] Jyrki P. Mäkelä et al. “Three-dimensional integration of brain anatomy and function to facilitate intraoperative navigation around the sensorimotor strip”. In: *Human Brain Mapping* 12.3 (Feb. 6, 2001), pp. 180–192.

[7] Max Heiland, Christian R. Habermann, and Rainer Schmelzle. “Indications and limitations of intraoperative navigation in maxillofacial surgery”. In: *Journal of Oral and Maxillofacial Surgery* 62.9 (Sept. 1, 2004), pp. 1059–1063.

[8] Kwok-Chuen Wong et al. “Computer assisted pelvic tumor resection and reconstruction with a custom-made prosthesis using an innovative adaptation and its validation”. In: *Computer Aided Surgery* 12.4 (Jan. 1, 2007), pp. 225–232.

[9] A Cappozzo et al. “Position and orientation in space of bones during movement: anatomical frame definition and determination”. In: *Clinical Biomechanics* 10.4 (June 1995), pp. 171–178.

[10] Emil G. Haritinian and Ashvin L. Pimpalnerkar. “Computer Assisted Total Knee Arthroplasty: Does it Make a Difference?” In: *Mædica* 8.2 (June 2013), pp. 176–181.

[11] Marco Viceconti et al. “An automated method to position prosthetic components within multiple anatomical spaces”. In: *Computer Methods and Programs in Biomedicine* 70.2 (Feb. 1, 2003), pp. 121–127.

[12] Nicholas A. Morton et al. “Effect of variability in anatomical landmark location on knee kinematic description”. In: *Journal of Orthopaedic Research* 25.9 (Sept. 2007), pp. 1221–1230.

[13] Anne M. Gilroy et al. *Atlas of Anatomy.* 3rd ed. New York: Thieme, 2016.

[14] Michael Schünke, Erik Schulte, and Udo Schumacher. *Allgemeine Anatomie und Bewegungssystem - Prometheus - LernAtlas der Anatomie.* 2nd ed. Stuttgart, 2007.

[15] *Kniegelenk.* In: *Wikipedia.* Page Version ID: 177556447. May 19, 2018. URL: `https://de.wikipedia.org/w/index.php?title=Kniegelenk&oldid=177556447` (visited on 10/04/2018).

[16] Frankie M. Griffin et al. "Anatomy of the epicondyles of the distal femur: MRI analysis of normal knees". In: *The Journal of Arthroplasty* 15.3 (Apr. 1, 2000), pp. 354–359.

[17] Rebecca J. Nesbit et al. "Effects of Population Variability on Knee Loading During Simulated Human Gait". In: *Annals of Biomedical Engineering* 46.2 (2018), pp. 284–297.

[18] Pascal L. Poilvache et al. "Rotational Landmarks and Sizing of the Distal Femur in Total Knee Arthroplasty:" in: *Clinical Orthopaedics and Related Research* 331 (Oct. 1996), pp. 35–46.

[19] Farhad Iranpour et al. "Patellofemoral joint kinematics: The circular path of the patella around the trochlear axis". In: *Journal of Orthopaedic Research* 28.5 (May 1, 2010), pp. 589–594.

[20] Donald Eckhoff et al. "Difference Between the Epicondylar and Cylindrical Axis of the knee". In: *Clinical Orthopaedics and Related Research* (2007), pp. 238–244.

[21] Yuki Yoshioka, David Siu, and Derek T. v. Cooke. "The Anatomy and Functional Axes of the Femur". In: *The Journal of Bone and Joint Surgery* 69 (1987), pp. 873–880.

[22] Anne M. Hollister et al. "The Axes of Rotation of the Knee". In: *Clinical Orthopaedics and Related Research* 290 (May 1993), pp. 259–268.

[23] Maximilian J. Hartel et al. "Determination of Femoral Neck Angle and Torsion Angle Utilizing a Novel Three-Dimensional Modeling and Analytical Technology Based on CT Datasets". In: *PLOS ONE* 11.3 (Mar. 2, 2016). Ed. by Angel Alberich-Bayarri, e0149480.

[24] K. Subburaj, B. Ravi, and Manish Agarwal. "Automated identification of anatomical landmarks on 3D bone models reconstructed from CT scan images". In: *Computerized Medical Imaging and Graphics* 33.5 (July 2009), pp. 359–368.

[25] Chad Starkey, Sarah D. Brown, and Jeffry L. Ryan. *Examination of Orthopedic and Athletic Injuries.* 3rd ed. Philadelphia: F.A. Davis Company, 2010.

[26] Shuichi Matsuda et al. "Tibial shaft axis does not always serve as a correct coronal landmark in total knee arthroplasty for varus knees". In: *The Journal of Arthroplasty* 18.1 (Jan. 1, 2003), pp. 56–62.

[27] Serge van Sint Jan. *Color Atlas of Skeletal Landmark Definitions: Guidelines for Reproducible Manual and Virtual Palpations.* 1st ed. Churchill Livingstone, 2007.

[28] Emel Avci et al. "Lateral posterior fossa venous sinus relationships to surface landmarks". In: *Surgical Neurology* 59.5 (May 1, 2003), pp. 392–397.

[29] Xiang Liu, Wangdo Kim, and Burkhard Drerup. “3D characterization and localization of anatomical landmarks of the foot by FastSCAN”. In: *Real-Time Imaging*. Imaging in Bioinformatics: Part III 10.4 (Aug. 1, 2004), pp. 217–228.

[30] D. D. Maudgil et al. “Identifying homologous anatomical landmarks on reconstructed magnetic resonance images of the human cerebral cortical surface”. In: *Journal of Anatomy* 193 (Pt 4 Nov. 1998), p. 559.

[31] Stefan Wörz and Karl Rohr. “Localization of anatomical point landmarks in 3D medical images by fitting 3D parametric intensity models”. In: *Medical Image Analysis* 10.1 (Feb. 1, 2006), pp. 41–58.

[32] Seung-Yeob Baek et al. “Automated bone landmarks prediction on the femur using anatomical deformation technique”. In: *Computer-Aided Design*. Solid and Physical Modeling 2012 45.2 (Feb. 1, 2013), pp. 505–510.

[33] Dong Yang et al. “Automated anatomical landmark detection ondistal femur surface using convolutional neural network”. In: *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*. 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI 2015). Brooklyn, NY, USA: IEEE, Apr. 2015, pp. 17–21.

[34] Ulf G. Leichtle et al. “Increased patellofemoral pressure after TKA: an in vitro study”. In: *Knee surgery, sports traumatology, arthroscopy: official journal of the ESSKA* 22.3 (Mar. 2014), pp. 500–508.

[35] Ulf G. Leichtle et al. “Influence of Different Patellofemoral Design Variations Based on Genesis II Total Knee Endoprosthesis on Patellofemoral Pressure and Kinematics”. In: *Applied Bionics and Biomechanics* 2017 (2017), p. 5492383.

[36] Tahmineh Razi, Mahdi Niknami, and Fakhri Alavi Ghazani. “Relationship between Hounsfield Unit in CT Scan and Gray Scale in CBCT”. In: *Journal of Dental Research, Dental Clinics, Dental Prospects* 8.2 (2014), pp. 107–110.

[37] David C. Wymer. “CHAPTER 5 - Imaging”. In: *Comprehensive Clinical Nephrology (Fourth Edition)*. Ed. by Jürgen Floege, Richard J. Johnson, and John Feehally. Philadelphia: Mosby, Jan. 1, 2010, pp. 56–74.

[38] F. G. Rammersdorfer. “Einführung in die Finite Elemente Methoden”. Vienna, Sept. 24, 2014.

[39] *CGAL*. In: *Wikipedia*. Page Version ID: 839999950. May 7, 2018. URL: `https://en.wikipedia.org/w/index.php?title=CGAL&oldid=839999950` (visited on 10/13/2018).

[40] Gabriel Taubin. “Curve and Surface smoothing withput shrinkage”. In: *IEEE* 8 (1995), pp. 852–857.

[41] Manfredo P. Do Carmo. *Differential Geometry of Curves and Surfaces*. 2nd ed. Rio de Janeiro, Brazil: DOVER PUBLICATIONS, INC., 2017.

[42] Andrew Pressley. *Elementary differential geometry*. 2nd ed. London: Springer, 2010.

[43] Nira Dyn et al. “Optimizing 3D Triangulations Using Discrete Curvature Analysis”. In: *Mathematical Methods for Curves and Surfaces*. Nashville, TN: Vanderbilt University Press, 2001, pp. 135–149.

[44] R. van Damme and L. Alboul. “Mathematical Methods for Curves and Surfaces”. In: *Mathematical Methods for Curves and Surfaces*. Vanderbilt University Press, 1995, pp. 517–526.

[45] Sun-Jeong Kim, Chang-Hun Kim, and David Levin. “Surface simplification using a discrete curvature norm”. In: *Computers & Graphics* 26.5 (Oct. 1, 2002), pp. 657–663.

[46] Jean-Daniel Boissonnat and Monique Teillaud. *Effective Computational Geometry for Curves and Surfaces*. (Visited on 07/02/2018).

[47] Evgeni Magid, Octavian Soldea, and Ehud Rivlin. “A comparison of Gaussian and mean curvature estimation methods on triangular meshes of range image data”. In: *Computer Vision and Image Understanding* 107.3 (Sept. 1, 2007), pp. 139–159.

[48] *Rotation matrix*. Wikipedia. Page Version ID: 847088280. June 22, 2018. URL: `https://en.wikipedia.org/wiki/Rotation_matrix` (visited on 07/08/2018).

[49] Siegfried Bosch. *Lineare Algebra*. 3rd ed. Berlin Heidelberg: Springer, 2006.

[50] Werner Mack. *Mechanik 1*. 5th ed. Vienna: Vienna University of Technology.

[51] Ian Richard Cole. “Modelling CPV”. PhD thesis. Loughborough: Loughborough University, 2015.

[52] Georg N. Duda et al. “Variability of femoral muscle attachments”. In: *Journal of Biomechanics* 29.9 (Sept. 1, 1996), pp. 1185–1190.

[53] Cong-Bo Phan and Seungbum Koo. “Predicting anatomical landmarks and bone morphology of the femur using local region matching”. In: *International Journal of Computer Assisted Radiology and Surgery* 10.11 (Nov. 1, 2015), pp. 1711–1719.

[54] Karl Rohr. “Extraction of 3d anatomical point landmarks based on invariance principles”. In: *Pattern Recognition* 32.1 (Jan. 1999), pp. 3–15.

[55] Martin Kozek. “Stochastik”. Vienna, 2014.

[56] *Sphere*. Wikipedia. Page Version ID: 860830180. Sept. 23, 2018. URL: `https://en.wikipedia.org/wiki/Sphere` (visited on 09/30/2018).

[57] *JPE - Experts In Positioning*. JANSSEN PRECISION ENGINEERING. URL: `https://www.janssenprecisionengineering.com/` (visited on 10/01/2018).

[58] Michael H. Kutner et al. *Applied Linear Statistical Models*. 5th ed. New York: McGraw-Hill Irwin, 2005.

[59] Hendrik C. Kuhlmann. “Numerische Methoden der Ingenieurwissenschaften”. Vienna, 2014.

[60] D. R. Carter, M. C. H. Van der Meulen, and G. S. Beaupré. “Mechanical factors in bone growth and development”. In: *Bone*. Proceedings of the International Symposium on Physical Loading, Exercise, and Bone 18.1 (Jan. 1, 1996), S5–S10.

[61] Jon E. Wergedal et al. “Skeletal growth factor and other growth factors known to be present in bone matrix stimulate proliferation and protein synthesis in human bone cells”. In: *Journal of Bone and Mineral Research* 5.2 (Feb. 1, 1990), pp. 179–186.

[62] J. Victor et al. “How precise can bony landmarks be determined on a CT scan of the knee?” In: *The Knee* 16.5 (Oct. 2009), pp. 358–365.

[63] H. Ihara. “Double-contrast CT arthrography of the cartilage of the patellofemoral joint.” In: *Clinical orthopaedics and related research* 198 (Sept. 1985), pp. 50–55.

[64] David Liu and S. Kevin Zhou. “Anatomical Landmark Detection Using Nearest Neighbor Matching and Submodular Optimization”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012*. Ed. by Nicholas Ayache et al. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 393–401.

[65] A. Schulze and H. P. Scharf. “Satisfaction after total knee arthroplasty. Comparison of 1990-1999 with 2000-2012”. In: *Der Orthopade* 42.10 (Oct. 2013), pp. 858–865.