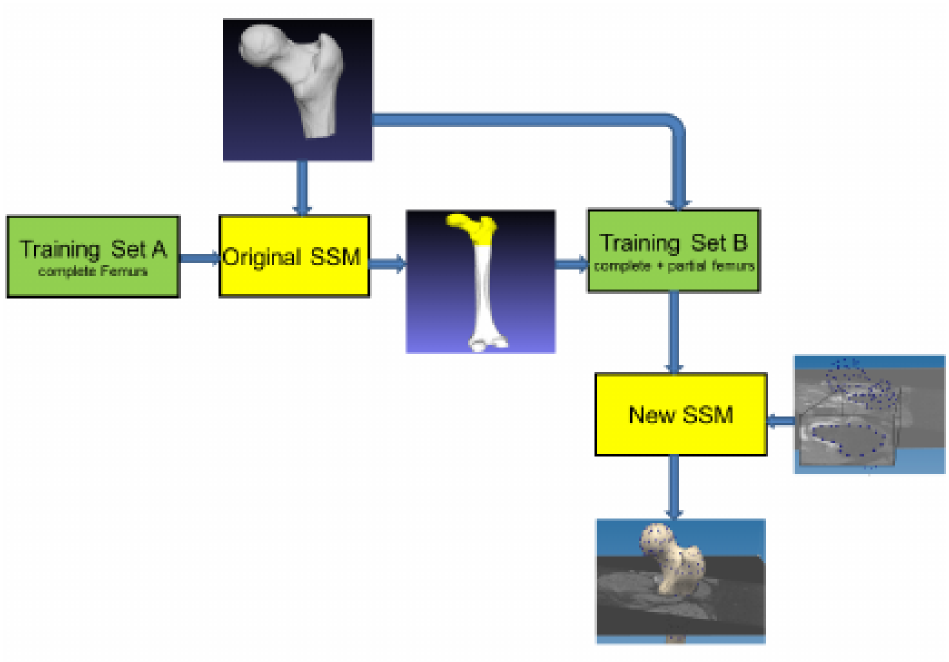
Automated Expansion of Statistical Shape Model Training Set for Femur

Master's Thesis



Hongjie Li

Automated Expansion of Statistical Shape Model Training Set for Femur

THESIS

submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

in

COMPUTER SCIENCE

by

Hongjie Li
born in Hebei, P.R. China

Computer Graphics and Visualization Group
Department of Intelligent Systems
Faculty EEMCS, Delft University of Technology
Delft, the Netherlands
`www.ewi.tudelft.nl`

Clinical Graphics
Molengraafsingel 12-14
Delft, the Netherlands
`www.clinicalgraphics.com`

Automated Expansion of Statistical Shape Model Training Set for Femur

Author: Hongjie Li
Student id: 4426002
Email: hongjie.lee888@gmail.com

Abstract

Active Shape Model (ASM) uses Statistical Shape Model (SSM) to fit the manually labeled landmark point in Magnetic Resonance Image (MRI) scan and segment the femur from the MRI scan. The SSM is trained from a training set containing 275 complete femur meshes. In general, with more femur meshes in the training set, there would be more shape variance information in SSM, and the segmentation result would be more accurate. Currently, it is hard to get complete femur meshes. However, 2000 partial femur meshes are available.

A method to incorporate these partial femur meshes to the SSM training set is proposed. This method includes building point to point correspondence between partial femur and complete femur, filling missing values to perform Principal Component Analysis(PCA) and designing the mechanism to utilize the content in partial femurs. Before incorporation, a rejection criteria is set to reject the unsatisfactory partial femur meshes that cannot contribute to the SSM.

A evaluation method is designed and performed to validate the incorporation method. The evaluation results show the accuracy of outliers of SSM fit can be improved by incorporating partial femurs to the SSM training set.

Thesis Committee:

Chair:	Prof. Dr. E. Eisemann, Faculty EEMCS, TU Delft
University supervisor:	Dr. A. Vilanova, Faculty EEMCS, TU Delft
Committee Member:	I. Flipse, Clinical Graphics B.V.
Committee Member:	Prof. Dr. E. Jansen, Faculty EEMCS, TU Delft

Preface

Thanks Dr. Anna Vilanova for her valuable guidance and suggestion during the whole procedure of this research. Also thanks her feedback in composing this thesis.
Thanks Ivo Flipse and Wynand Winterbach for providing the key resource and constructive guidance in executing the experiment in Clinical Graphics.
Their help is essential to the completion of this work.

Hongjie Li
Delft, the Netherlands
August 23, 2015

Contents

List of figures

Chapter 1

Introduction

Computer assisted analysis for medical imaging data, is a rapidly growing field for disease diagnosis and surgery planning. Computer Tomography (**CT**) and Magnetic Resonance Image (**MRI**) are two widely used medical image acquisition modalities. Despite the 3D nature of the acquired data, the data is commonly inspected by looking at the 2D slices one by one(see Fig 1.1). However, a 3D visualization of the interesting structures provides more information about the real 3D geometric structure. Image segmentation is usually an important step needed to achieve these 3D visualizations. Segmentations are methods that extract the shape of interest from imaging data. In the case of medical images, we want to segment tissues of interest, e.g, bones from 3D medical scans.

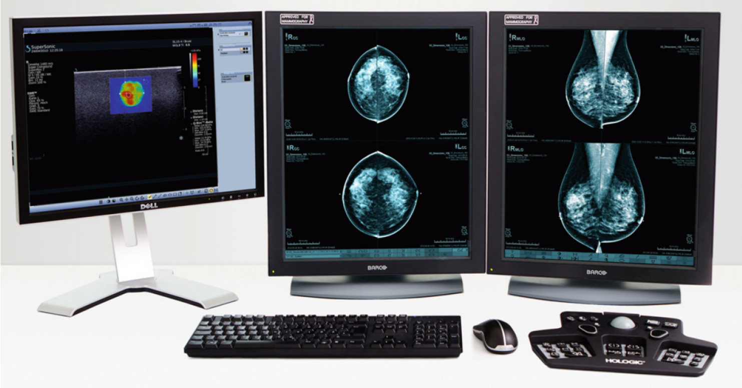
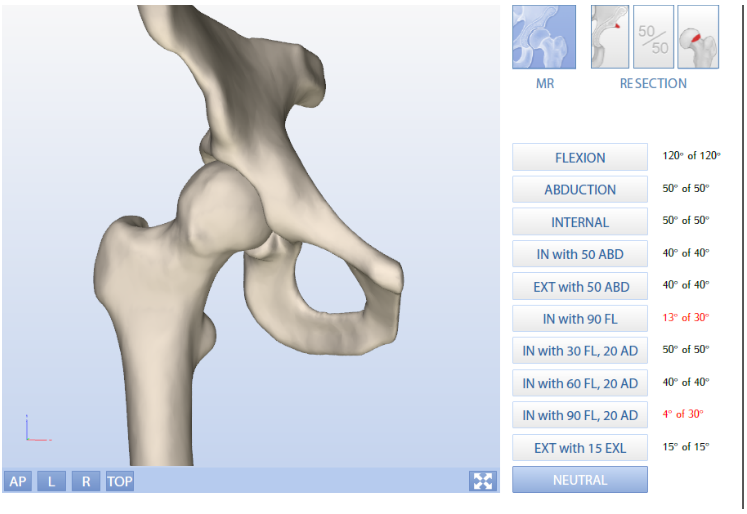


Figure 1.1: The workstation that clinicians use to read the CT or MRI scans. The clinicians can read the 3D scans slice by slice in different orientation, they can also compare the slice using different monitors. (Image Source, Preim and Botha [2013])

Meanwhile, surgeries on hip such as Total Hip Arthroplasty (THA) are very common. The segmentation and 3D visualization of the femur (see Fig 1.2) can be used in diagnosis and surgery planning procedure to perform analysis on the shape of hip and simulation on the movement of femur. With this information, the clinicians can deliver better treatment.



FLEXION 120° of 120°
ABDUCTION 50° of 50°
INTERNAL 50° of 50°
IN with 50 ABD 40° of 40°
EXT with 50 ABD 40° of 40°
IN with 90 FL 13° of 30°
IN with 30 FL, 20 AD 50° of 50°
IN with 60 FL, 20 AD 40° of 40°
IN with 90 FL, 20 AD 4° of 30°
EXT with 15 EXL 15° of 15°
NEUTRAL

Figure 1.2: An example of 3D visualization technique for hip bones. The clinicians can change the view angle and perform motion simulation. (Image Source: Report from Clinical Graphics)

There are several approaches to perform image segmentation, such as thresholding, clustering, edge detection, region growing and model based methods (see Szeliski [2010]). To segment femur from CT scans, thresholding is an effective method. In CT scans, different tissues have relatively fixed values of Hounsfield (HU). The HU values of bone are higher than other tissues, by selecting a proper HU value threshold, we can usually segment the femur from CT scans.

The radiation of CT scanning is harmful to body, so sometimes the clinicians choose to use MRI which is radiation-free to acquire the medical image. For the MRI scans, same tissues may have completely different intensity values within the same dataset, so thresholding does not work properly, and some other solutions are required. All femurs have a common shape structure, while there are some variances between individuals. Model based segmentation has shown to be a promising solution when the known shape information is introduced in the segmentation process. One widely used model based segmentation technique is Active Shape Model (ASM). ASM uses existing shapes to build Statistical Shape Model (**SSM**), and uses the trained SSM to keep the overall shape of the target object and to allow variance within a plausible degree.

The SSM needs a training set consisting of different plausible shapes used to extract the general model of the shape variance. With more shapes in the training set, the SSM contains more plausible shape variance that allows a more accurate fit to the individual bones.

Currently Clinical Graphic Inc has a SSM comprised of 275 3D complete femur meshes (see Fig 1.3 Right). The SSM is used to fit the femurs in MRI scans and perform segmentation. It would be desirable to add more femurs into the training set to improve the segmentation accuracy. However, in common practice the complete femur is not scanned, therefore, most data sets do not contain the complete femur, but a part of it. This data cannot be used at the moment for the SSM given the constraint of having the complete femur. Clinical Graphics has around 2000 3D partial femur meshes (Fig 1.3 Left). Therefore, we want to explore the possibility to add the partial femurs to the existing SSM training set.

It is unclear whether incorporating the partial femurs will improve the segmentation accuracy. We hypothesize that if we incorporate partial femurs to SSM training set, we can have more plausible shape variance so as to improve SSM fit accuracy. We expect it can improve the fit accuracy when the current complete femur SSM cannot fit the mesh properly.

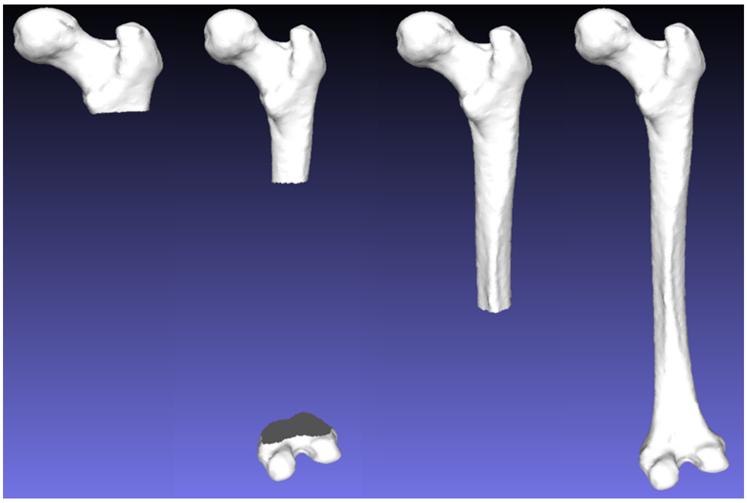


Figure 1.3: Four examples of 3D femur meshes. **From left to right:**, a partial femur with length 80mm, a partial femur with length 140mm and meshes on the bottom, a partial femur mesh with length 260mm, a complete 3D femur mesh with length around 420mm

The incorporation of partial femurs to the SSM training set is not a straightforward task. Since SSM are based on having the same identifiable landmark points in all structures, e.g, for the SSM of faces, all the shapes in SSM training set should have point to point correspondence at each position (See Fig 1.4).

The main issue with the partial data is that the bottom part or the middle part of the femur is missing, the partial femurs also differ in length (see Fig 1.3). This means the partial femurs have different amounts of missing values. To incorporate partial femurs to the training set, we need to solve three problems. First, partial femurs which are too short may not be able to provide any extra information or even provide misleading shape variance information, so we need to set a rejection criteria to reject the data sets with a too short partial femurs. The second problem is how to extract the points of interest from the partial femurs,

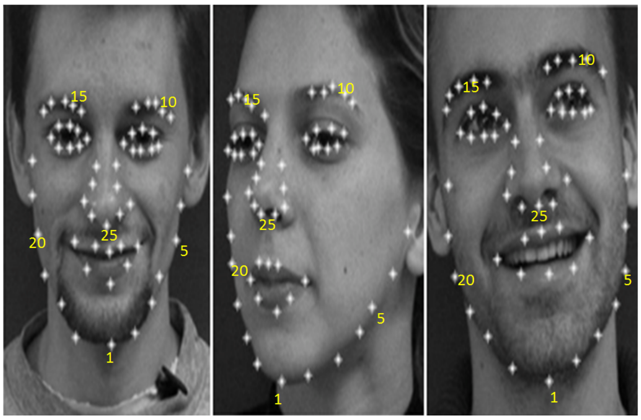


Figure 1.4: Three examples of faces shapes. The point should place in same order for all the points (Source: Improved active shape model and its application to facial feature extraction by Tang et al. [2013])

and build the point to point correspondence with the points that form the dimensions of the SSM. Finally, we have to study how to incorporate partial femurs to the training set which currently is comprised of complete 3D meshes. The procedure of incorporating partial femurs should not sacrifice the accuracy.

We propose a solution for all these steps together with an evaluation. A framework of incorporating partial femurs to the training set which consist of complete femurs is presented.

The contributions of this project are the following:

- The analysis and definition of rejection criteria for partial femurs with different length
- Two strategies to incorporate partial femurs to the SSM training set
- An evaluation of two partial femur incorporation strategies

The rest of this article is organized as follows. In Chapter 2, we introduce the background information and some previous work in this area. Rejection Criteria is described in Chapter 3. Building point to point correspondence and two strategies to incorporate partial femurs are described in Chapter 4. In Chapter 5, the evaluation method and results are described. In Chapter 6, we conclude the work of this master thesis project and describe the future work.

Chapter 2

Background

In this chapter, we introduce some information which is necessary to understand the contribution of this thesis. In Section 2.1, we introduce the basic hip anatomy and the accuracy requirements for hip surgery guidance. In Section 2.2, we introduce the training process of Statistical Shape Model (SSM). In Section 2.3, we introduce image registration algorithm, which is necessary to build the point to point correspondence. And in Section 2.4, we give an overview of our method of this thesis project.

2.1 Anatomy and Accuracy Requirements

Figure 2.1 left shows the bones which are related to the hip surgery. In the surgery like Total Hip Arthroplasty(THA), the clinicians replace the Femoral head and Femoral neck with an artificial hip joint (Fig 2.1 right). To get enough information to diagnose or plan the surgery on hip joint, clinicians usually need CT/MRI scans of the top part of femur. Sometimes, clinicians also want to scan the knee part, so that they can have a better understanding of the orientation of femur, e.g, the angle between femur and acetabulum. This is the reason why we have bottom (knee) parts in some partial femur meshes.

Generally, clinicians need an accurate 3D visualization and movement simulation of the top part of femur, the part above the Trochanter Minor (Fig 2.2). Based on this domain requirement, we need to provide accurate segmentation result on the top part above Trochanter Minor. According to our measurement, the length above Trochanter Minor is around 80mm. Currently, there is no standard requirement for the accuracy of segmentation result, our goal is to see whether we can make the current segmentation result better, especially when the current complete femur SSM cannot provide a satisfactory fit result.

2.2 Training Process of Statistical Shape Model

In this section, we describe the process of building Statistical Shape Model, one component of Active Shape Models (ASM) algorithm that is relevant for this thesis. We introduce the information that is necessary to understand our contribution. To get complete overview of

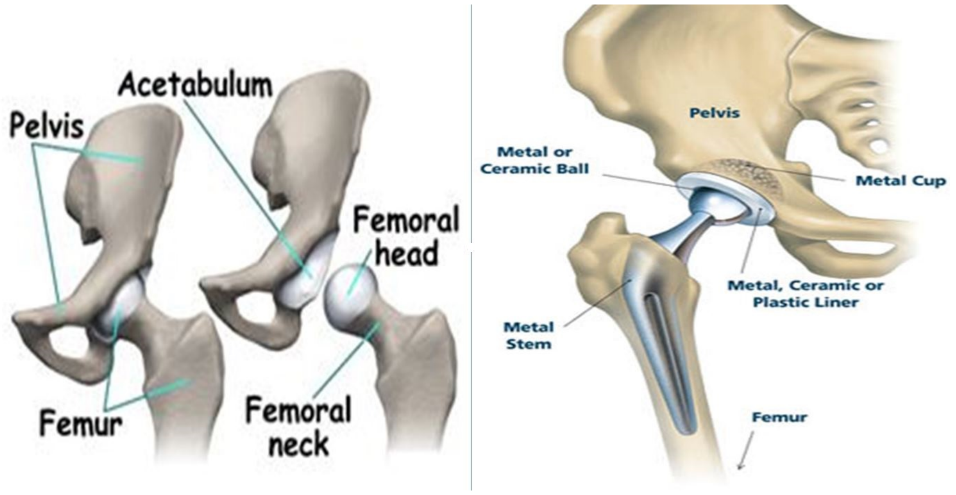


Figure 2.1: Anatomy related to hip surgery. **Left**: the related bone structures for the hip surgery. **Right:** illustration of THA results. (Images from Web)

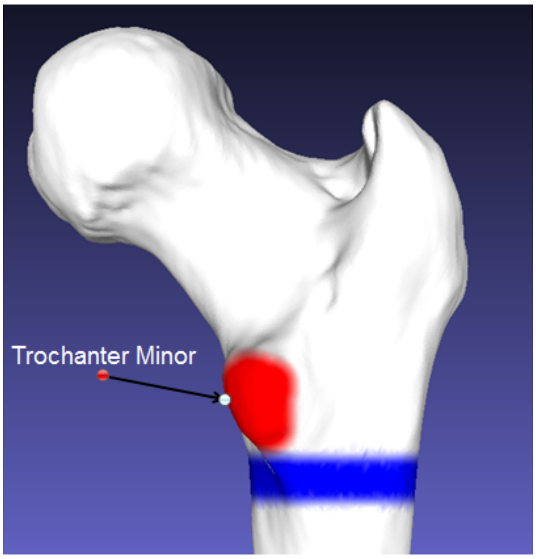


Figure 2.2: Accuracy Requirement of ASM Segmentation, the femur above Trochanter Minor (Red) or above the blue line should be accurate.

Clinical Graphics' implementation of ASM, see Appendix A.

Statistical Shape Model is represented by a set of landmark points which describe the general shape of one object. The training set of SSM contains 2D counters or 3D meshes with landmark points placed at characteristic positions. Each landmark point represents a particular part of the object or its boundary. The landmark points are placed on high curvature or junctions, or some intermediate positions which can help to make the boundary more precisely [Cootes et al., 2000]. One example of femur and how the landmark points should be placed are shown in Figure 2.3. The landmark points should correspond to each other. In other words, the same point index should represent the same landmark point in different shapes, as shown in Figure 1.4.

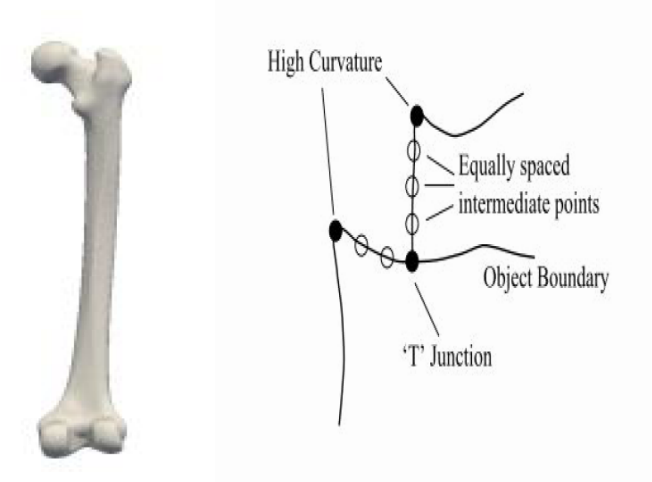


Figure 2.3: Illustration of femur and how landmark points are placed. **Left:** An example of femur. **Right:** Illustration of positions of landmark points on High Curvature, Junction and Intermediate positions (Right image from [Cootes et al., 2000])

The landmark points in each dataset define a shape. We turn them into a shape vector. For 3-D shape, the n landmark points from one mesh are represented as a $3n$ elements vector $\mathbf{v_i} = (x_{i1}, y_{i1}, z_{i1}, x_{i2}, y_{i2}, z_{i2}, ...x_{in}, y_{in}, z_{in})$, where (x_{ij}, y_{ij}, z_{ij}) represent the same landmark point j for each training shape i.

Because the shapes from training set may have different positions and orientations, aligning them into the same coordinate frame is necessary. We can translate each shape's center of gravity to origin and rotate them to one coordinate frame using iterative closest point(ICP) algorithm [Besl and McKay, 1992]. After alignment, we put all m aligned shape vectors $\mathbf{v_i}$ together to build one $3n \times m$ shape matrix $X = (\mathbf{v_1}|\mathbf{v_2}|...|\mathbf{v_m})$. In the shape matrix, each column represents a shape in the training set.

By analyzing the mean shape $\overline{\mathbf{v}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{v_i}$ and the variance among the shape vectors, we can build a SSM which can represent the general shape and variance. The SSM can generate new examples which are similar to those shapes in the training set. To make the computation more efficient, reducing data dimensionality from $3n$ to a smaller value is desired. Principal Component Analysis (PCA) is an effective approach to achieve this goal. PCA is a statistical procedure that try to find the projections that better preserve the variance of the data. To get a complete introduction of PCA, see Appendix B. Performing PCA on the shape matrix X we can compute the main axes of shapes, and approximate original shapes using a model with less than $3n$ parameters using equation $\mathbf{v} = \overline{\mathbf{v}} + \mathbf{Pb}$. $\overline{\mathbf{v}}$ is the mean vector of training vectors, which represent the mean shape of training set. $\mathbf{P} = (\mathbf{p_1}|\mathbf{p_2}|...|\mathbf{p_t})$ contains the first t eigenvectors of the covariance matrix of X and $\mathbf{b}$ is a t dimensional vector given by $\mathbf{b} = \mathbf{P}^T(\mathbf{v} - \overline{\mathbf{v}})$. The b_i is also called shape parameter.

By varying the elements of $\mathbf{b}$, the shape of $\mathbf{v}$ can be varied. The variance degree of i^{th} parameter b_i is given by its corresponding eigenvalue λ_i from PCA. The larger λ_i means larger variance in corresponding principal component. Applying constraints to shape parameter b_i within plausible degree (e.g, $\pm 5\lambda_i$), can make sure the shapes are plausible. Figure 2.4 shows the effect of varying the first three shape parameters b_i of femur mesh between $\pm 5\lambda_i$. We can see that the first mode encodes the size of femur. The second mode encodes the femur's orientation. The third mode encodes the thickness of femur.

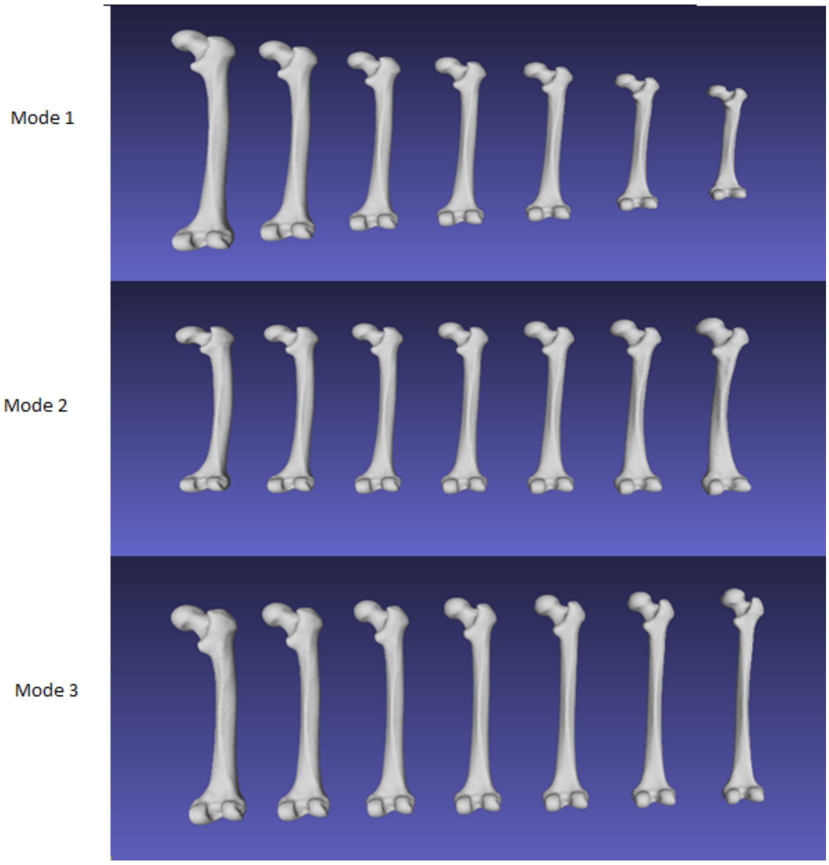


Figure 2.4: Effect of Varying the first three femur model shape parameters in turn between $\pm 5s.d.$. The shape parameters from left to right $-5, -3, -1, 0, 1, 3, 5$

2.3 Mesh Registration

Registration is the determination of a geometrical transformation that aligns points in one view of an object with corresponding points in another view of that object or another object [Fitzpatrick et al., 2000]. The term ”view” can be three-dimensional meshes or two-dimensional images. From an operation view, the inputs of registration are the two images to be registered, the output is a geometrical transformation, which is merely a mathematical mapping from points in one view to points in the second.

There are different methods of registration, e.g, rigid transformation, nonrigid transformation and some point based methods. Since there are deformations between the femurs of individuals, the elastic registration (also known as nonrigid registration) is preferred. We think the nonrigid registration methods are acceptable solutions to us.

There are lots of research on nonrigid registration. One of the most important registration methods for 3D meshes is Iterative Closest Point(ICP) algorithm proposed by Besl and McKay [1992]. The ICP algorithm requires only a procedure to find the closest point on a geometric entity to a given point. The ICP algorithm uses rotations and translations with a certain level deformation to minimize the mean-square distance between points from two images.

The ICP algorithm is applicable for two complete meshes. But in practice, there are needs to register partial object surface to its corresponding complete surface. Xiao et al. [2005] proposed a method for 3D partial surface registration that utilizes both regional surface properties and shape rigidity constraints to align a partial object surface to its corresponding complete surface.

Li et al. [2008] created a new algorithm that registers partial surfaces to complete scans and builds the point to point correspondence. They applied confidence weights to measure the reliability of each correspondence and identify the non-overlapping areas. Their method maximizes the regions of overlap and the spatial coherence of deformation while minimizing registration error.

In our case, we use the registration algorithm call nonrigidICP created by Dr. E Audenaert [2015], one partner of Clinical Graphics. The nonrigidICP algorithm registers partial mesh to its corresponding complete mesh. It uses the ICP algorithm to perform alignment and uses radial basis functions to perform the deformation. It also uses the weight confidence method from Li et al. [2008] to perform the local deformation.

2.4 Method Overview

Our methods mainly focus on adding partial femurs to the training set of SSM with the goal to improve the segmentation accuracy. A diagram of pipeline is shown in Figure 2.5.

As described in the Chapter 1, the partial femurs differ in the length. A femur which is too short may not be able to provide accurate shape variance information. So, we need to evaluate the ASM fit results for partial femurs with different lengths. Also, some partial femur meshes have bottom (knee) part, we want to see whether this information can improve the ASM fit result. We assume that if the partial femur contains enough information, we can get similar segmentation result to the result we get when the complete femur is used. If partial femurs are too short to provide enough shape information to accurately reconstruct

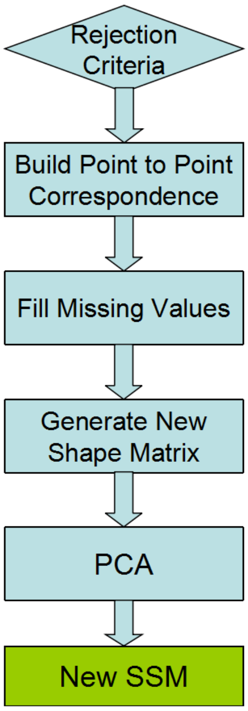

Rejection Criteria
Build Point to Point Correspondence
Fill Missing Values
Generate New Shape Matrix
PCA
New SSM

Figure 2.5: The pipeline of our method

femur, we assume they are not good candidates to be added to the training set and we should reject them. We set one experiment to validate our hypothesis and define the rejection criteria.

The shapes of femurs in SSM training set are represented by ordered landmark points. To utilize the content in partial femur meshes, we need to extract the shape of interest from the meshes, in other words, build the point to point correspondence between the partial femur points and the complete femur landmark points. After building the point to point correspondence, we can turn the partial femurs into shape vectors with some missing values.

After getting shape vectors of partial femurs, the next step is to perform PCA on the new SSM training set containing shape vectors with missing values and generate a new SSM. There is the need to have a method which is able to deal with missing values. We present two strategies to achieve this goal.

In the following 2 sections, we give detailed description of each step.

Chapter 3

Rejection Criteria

The partial femurs vary considerably in length, some meshes also have bottom (knee) part, as shown in Figure 3.1. We assume that if the partial femur is long enough, it can give a good estimation of the full bone, or at least a good estimation on the top part of femur that require accurate segmentation result for our purpose. However, a femur mesh which is too short may be not able to provide enough shape variance information to estimate the rest of the relevant parts of the femurs. To select the satisfactory partial femurs, we need a rejection criteria. In the following three sections, the method of setting rejection criteria are described in detail.

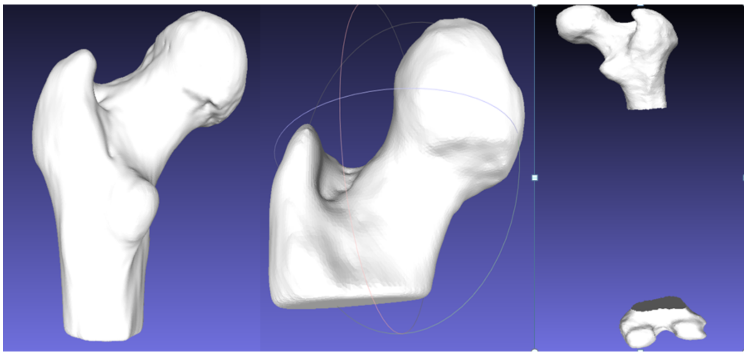


Figure 3.1: Three examples of existing partial femur meshes. **Left:** The majority of partial femurs. **Middle:** a femur which is too short. **Right:** a partial femur with bottom(knee) part

3.1 Method of Setting the Rejection Criteria

Based on the segmentation accuracy requirement above Trochanter Minor, we want to incorporate real shape variance information to the training set instead of the estimated result using SSM fit. For any partial mesh that does not have complete Trochanter Minor, or its length is shorter than 80mm, we reject it.

After setting this basic requirement for the femur length, we need to validate our hypothesis that given a partial femur, we can get a good estimation of the top part of femur or even better a good estimation of the complete femur. To validate this hypothesis, we need to figure out two questions. First, since the partial femurs vary in length, we need to figure out how do the lengths of partial femurs affect the accuracy of ASM segmentation. Second, since some partial femurs have meshes on the bottom (knee) part we need to evaluate whether the partial mesh on the bottom (knee) part will affect the segmentation result. We set two separate experiments to answer these two questions.

There are 275 complete femurs in SSM training set. We randomly selected complete femurs from SSM training set and cut each one in different length, range from 80mm to the complete bone (around 420mm), as shown in Figure 3.2. A complete femur generates 17 partial femurs with different lengths. Then we train a new SSM without these selected femurs, and use the new SSM to perform segmentation on the selected partial femurs. Since the SSM fitting is a time consuming job (4 minutes per case), to save computation time, we select 13 femurs from SSM training set to perform this experiment.

In real case, the partial femur meshes are not as well structured as the complete meshes in the training set. In general, a partial femur mesh has around 40000 vertices. However, there are 21096 vertices in the complete femur meshes from the existing SSM training set (around 10000 vertices on the top part). To make the experiment more realistic, we perform remeshing on the partial femurs to generate more vertices (around 40000) and introduce random displacement for the newly generated vertices.

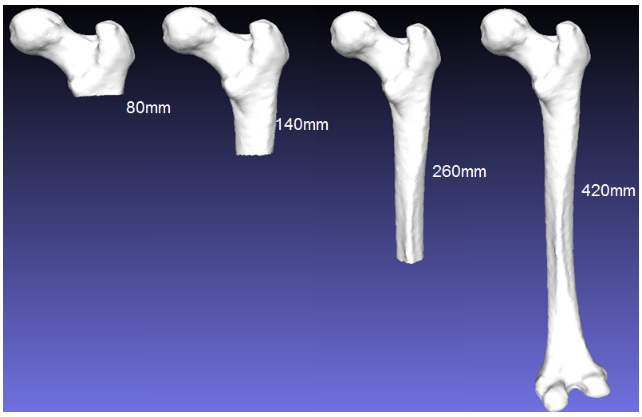


Figure 3.2: The femurs in different length.

To validate whether the bottom (knee) part can improve the segmentation accuracy, we use the same method. We also cut 13 femurs in different lengths, the only difference is we keep the bottom part (see Fig 3.3). Then we train a new SSM without these selected femurs, and use the new trained SSM to perform segmentation on these partial femurs.

Since we have the complete femur meshes, we can use them as ground truths. After getting the segmentation result, we compare it with the known complete femur mesh, by calculating the root mean square error (RMSE) on the top part above Trochanter Minor (Fig 2.2). The calculation of RMSE requires 3 steps. First, since the vertices in shape vectors are

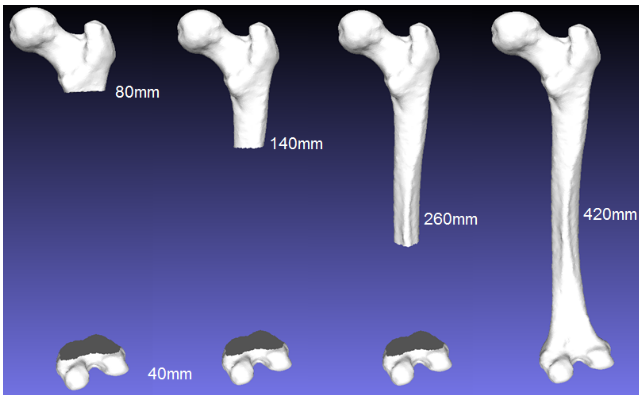


Figure 3.3: Cut one femur in different lengths, bottom (knee) part is also included

placed in order, we can get the index of vertices above the Trochanter Minor in the training set. Second, since the segmentation result and the ground truth are represented by shape vectors, we can get the coordinate of each vertex using the vertices' indexes. Finally, the points from same positions in different shapes have the same indexes, we can calculate the Euclidean distance of each corresponding point and get the RMSE. The whole evaluation procedure is shown in Figure 3.4.

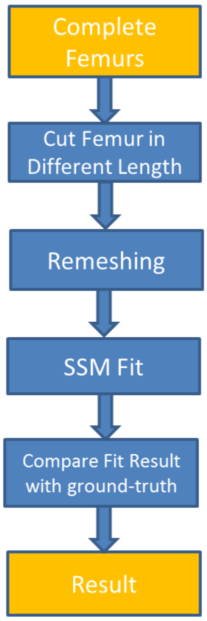

Complete Femurs
Cut Femur in Different Length
Remeshing
SSM Fit
Compare Fit Result with ground-truth
Result

Figure 3.4: The procedure of setting rejection criteria

3.2 Evaluation Results

Figure 3.5 shows the SSM fit result for partial femurs without bottom part using RMSE for the selected 13 femurs. As shown in the result, as long as the partial femur longer than 80mm, we can have a similar result on the top part, almost same as the complete femur is used.

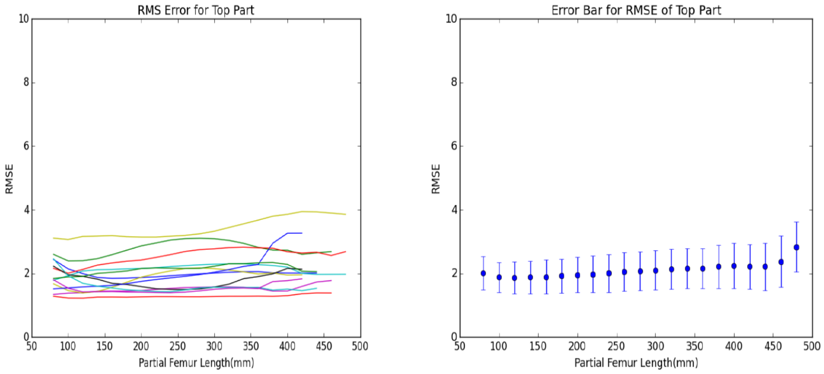


Figure 3.5: The RMSE on the top part of femurs for the selected partial femurs in different length. **Left:** The RMSE on the top part of the femurs, each color represents one femur, in horizontal line, we increase the lengths of femurs. **Right:** The mean value of RMSE with standard deviation for all the partial femurs in same length.

Figure 3.6 shows the SSM fit result for partial femurs that contain bottom part. Compare this result and result on Figure 3.5, we can see the bottom part of femur has no significant improvement on the accuracy of the top part.

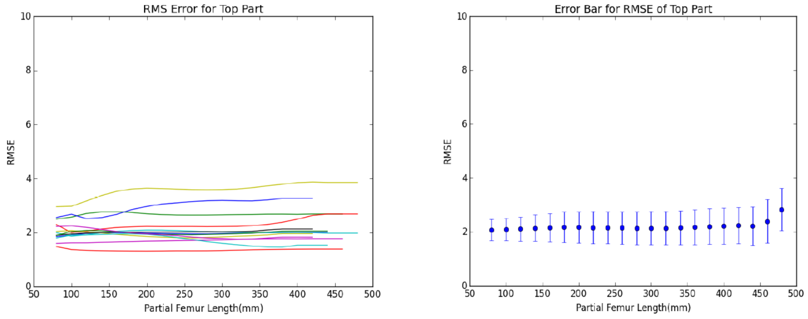


Figure 3.6: The RMSE on the top part of femurs for the selected partial femurs in different length, the partial femurs contain bottom part of femurs. **Left:** The RMSE on the top part of femurs. **Right:** The mean value of RMSE with standard deviation for all the partial femurs in same length.

To see the influence of length and bottom part of the femur on the segmentation accuracy of complete femur, we also calculate the RMSE on the complete femur instead of just the top part for the two experiments shown above. The results of partial femurs without bottom part are shown in Figure 3.7. The results of partial femurs with bottom part are shown in Figure 3.8. As we can see from the two results, without bottom part, the RMSE reduces significantly when we increase the length of partial femurs. However, if we have the bottom part, we almost have a stable RMSE like compete femurs are used, the length of femurs do not influence the RMSE for the whole bone. As long as we have a partial femur which contains bottom part, we can get the same result as we get with complete femur.

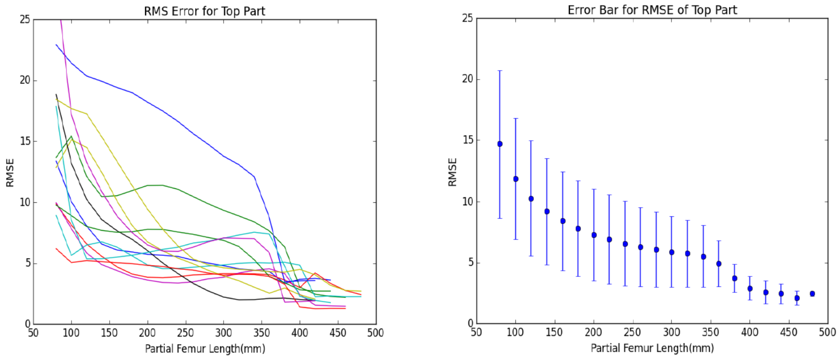


Figure 3.7: RMSE for the whole femur between the ASM segmentation result and the ground truth femur. **Left:** The RMSE on the top part of femurs. **Right:** The mean value of RMSE with standard deviation for all the partial femurs in same length.

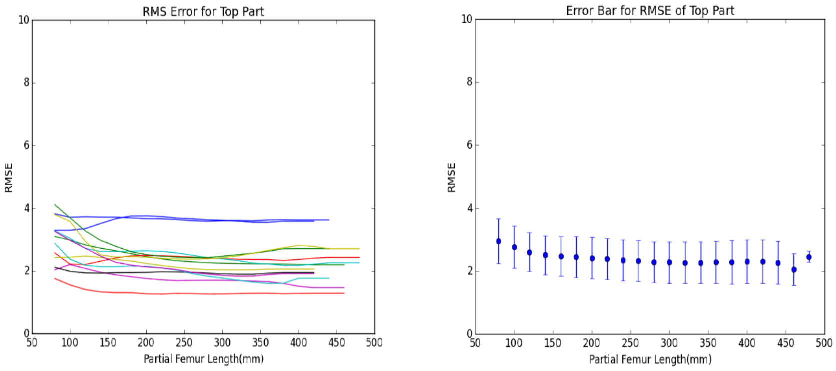


Figure 3.8: RMSE for the whole femur between the ASM segmentation result and the ground truth femur, bottom part of femur contained.**Left:** The RMSE on the top part of femurs. **Right:** The mean value of RMSE with standard deviation for all the partial femurs in same length.

Based on the evaluation results above, we can see that the most ideal case is we have partial femurs that contain Trochanter Minor on the top and knee part on the bottom. However, we only have quite limited number of partial femurs which have bottom part, and we are interested just on the top part above Trochanter Minor. Adding the bottom(knee) part will not help much diagnosis and surgery guidance, so we do not require the partial femur to contain the bottom part. Based on the experiment and analysis, we set the rejection criteria: **As long as the partial femur has a top part that contains Trochanter Minor, we accept it. In most cases, it also means the partial femur is longer than 80mm.**

Chapter 4

Incorporate Partial Femurs to SSM Training Set

After getting satisfactory partial femurs, we need to turn them from meshes into shape matrix in the training set. Then perform PCA on shape matrix to get the new SSM, as described in Section 2.4. To achieve this goal, we need to solve two problems. First, we need to extract the shapes of interest from the 3D meshes, in other words, we need to find the points from partial meshes that are corresponding to the landmark points in the SSM training set. Second, we need to fill the missing values in the shape vectors of partial femur because the current PCA method in Clinical Graphics requires a complete matrix. In the following sections, we describe the two steps to incorporate satisfactory partial femurs into the training set and get new SSM.

4.1 Build Point to Point Correspondence

As we discussed above, after getting satisfactory femurs, we need to extract the shapes of interest from the meshes. To achieve this goal, we need to build the point to point correspondence between the partial meshes and the complete meshes in the SSM training set. As shown in Figure 4.1, we need to extract the point of interest, then we can build shape vectors with missing values $\mathbf{v_i'} = (x_{i1}, y_{i1}, z_{i1}, x_{i2}, y_{i2}, z_{i2}, ..., nan, nan, nan, ..., x_{in}, y_{in}, z_{in})$. The *nan* means missing values in the vector.

Since each shape of interest has 21096 vertices, it is impossible to label the landmark points and build point to point correspondence manually. One effective approach to achieve this goal automatically is the non-rigid registration. We register partial femurs into one complete femur from the SSM training set, then get the point to point correspondence. For the partial registration, we use the nonrigidICP algorithm [Audenaert, 2015]. As described in Section 2.3, the nonridigICP algorithm registers the partial mesh to the complete mesh and return the corresponding position in the complete mesh for each vertex from partial mesh. These corresponding positions can help us build the point to point correspondence. For convenience, we call the partial femur to be registered as source mesh, and the complete

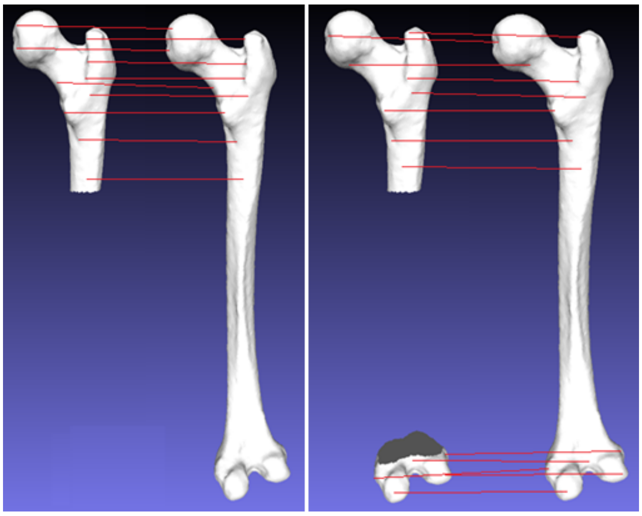


Figure 4.1: Illustration of point to point correspondence. **Left**: The point to point correspondence between partial femur without knee and complete mesh. **Right**: The point to point correspondence between partial femur with knee and complete mesh.

femur as target mesh. In Protocol 1, the nonrigidICP algorithm is described briefly. The detail procedure of nonrigidICP algorithm is described in Appendix C.

Protocol 1: nonrigidICP

1. **Evaluating the source mesh and pre-processing on the target mesh**
2. **Use ICP algorithm to find the initial alignment of source mesh**
3. **Perform deformation on the source meshes using radial basis functions**
4. **Perform local deformation on the source mesh to find the best mesh which is closest to the target mesh**

Unlike the partial femurs in setting rejection criteria, here we do not have the corresponding complete femurs to the partial femurs. So we need to find a proper complete femur mesh as the target mesh. One common choice is selecting one complete mesh from the training set, and register all partial meshes into it. In our case, since we have the SSM training set contains 275 complete meshes, we can utilize the content to get a better result. We use the existing SSM fit on the partial meshes, get the fit result which is a complete femur mesh. Then we can use nonrigidICP algorithm to register the partial meshes to the SSM fit result, so that we get the positions of vertices from partial mesh in the complete result. By searching the nearest neighbor in the registered partial mesh for each landmark point in the complete mesh, we build the point to point correspondence. Using the index of nearest neighbor for each landmark point, we can find the real coordinate for each point in the source mesh. The steps are described in Protocol 2. Figure 4.2 shows the effect of nonrigidICP algorithm. As we can see, the partial femur is aligned and deformed to fit the complete mesh.

Protocol 2: Procedure to get shape vectors from partial mesh

1. **Perform SSM Fitting on the partial femur, get the fit result as target mesh (T)**
2. **Register partial femur as source mesh (S) to T using nonrigidICP algorithm, get the registered source mesh $S^{'}$**
3. **For the vertices from $S^{'}$, search the nearest landmark points P from T, so that we get the index of landmark point index I**
4. ***For the landmark points P in step 3, search their nearest points $P^{'}$ in the $S^{'}$, get their indexes $I^{'}$***
5. **Using $I^{'}$, we can get the real coordinate of vertices from S and their positions in the complete meshes are recorded in I**
6. **Turn the real vertices into shape vectors $v^{'}$ with missing values**

4.2 Fill Missing Values in Shape Vector

After getting the l shape vectors v_i', we add them to the training set X, and get a new shape matrix $X' = (\mathbf{v_1}|\mathbf{v_2}|...|\mathbf{v_m}|\mathbf{v_1'}|\mathbf{v_2'}|...|\mathbf{v_l'})$, same as the original training set, each column represents a shape, the first m vectors v_i represent the complete shapes, and the remaining l vectors v_i' represent the newly added partial meshes.

Since the current PCA method in Clinical Graphics requires a complete matrix, we need a strategy to deal with the missing values when performing PCA. One approach is filling the missing values. We come up with two strategies. The first one is generating the missing values using the mean values. The second one is using the SSM fit result to fill the missing values. We discuss the two strategies in more detail in the following two sections.

4.2.1 Method 1: Filling missing values using mean values

In the real SSM fitting scenario in Clinical Graphics, 20 eigenvectors can represent 62% of the shape variance in the SSM training set, and give good estimation in most cases. So we

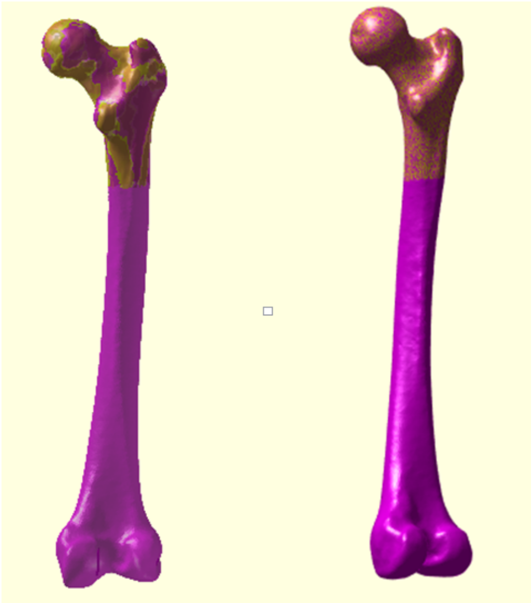


Figure 4.2: Illustration of partial registration. **Left**: the partial femur and complete femur before registration. **Right**: The registered partial femur and the complete femur.

assume that shape variance of femurs are highly correlated, using the mean values to fill the missing values can be a good choice. In this method, we use the shape vectors of partial femurs on the top part. For the missing values v'_{ij} in shape vector, we use the the mean values of their corresponding values in the m complete femurs. We calculate the missing value v'_{ij} using equation

$$v'_{ij} = \frac{1}{m}\sum_{i=1}^{m} v_{ij} \tag{4.1}$$

The v_{ij} means j^{th} parameters from the i^{th} complete training shape vector.

Using this method, all the partial femurs have the same values for the missing part, we do not introduce any invalid variation to the training set. However, there is one risk that by adding shape vectors to the top part and mean values to the other parts, we add more shape variance on top part without corresponding shape variance in bottom part. It is possible that we change the direction of eigenvectors. One example is shown in Figure 4.3. As we can see, when adding variance in x axis, and using the mean value in y axis, the directions of principal components change a lot, together with the eigenvalues (length of the arrows in the Figure).

To analyze the effect of adding mean values to the training set, we calculate the dot product of eigenvectors of new SSM training set and the eigenvectors of original SSM training set. Since the dot product $v_i \cdot v'_i = |v_i||v'_i|cos(\theta)$ and v_i and v'_i are normalized vectors, the dot product represent the difference between the 2 eigenvectors, 1 means the eigenvectors are exactly same, 0 means eigenvectors are orthogonal, the result are shown in Figure 4.4

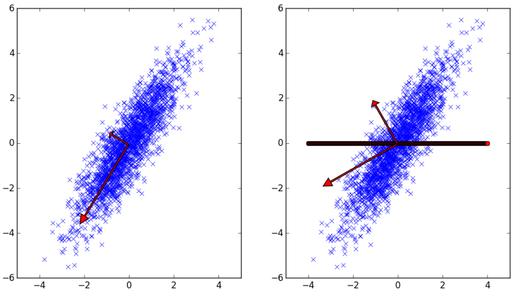


Figure 4.3: Examples of side effect of using the mean value fill missing values in 2D case. **Left:** The original 2000 observations and principal components. **Right:** The principal components after adding another 1500 observation with same value in y axis

Left. Except the first eigenvectors, all the other eigenvectors has dot product much smaller than 1, which means we indeed changed the eigenvectors by filling the missing values with mean values.

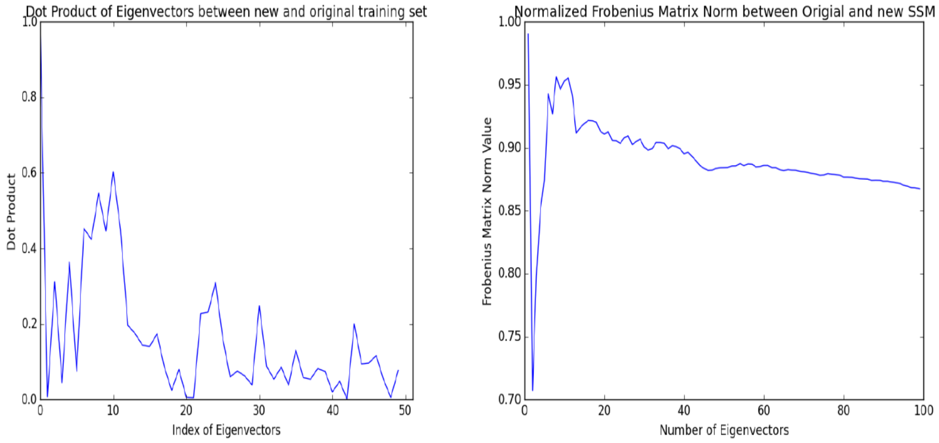


Figure 4.4: Analysis of Eigenvectors for filling missing values with mean values. **Left:** The dot product of eigenvectors from new SSM and original SSM. **Right:** The Frobenius matrix norm of new SSM and original SSM.

However, O'Donnell and Westin [2007] showed the eigenvectors basis can rotate when eigenvalues are near each other. In this case, even though we have different order of eigenvectors from two training sets, actually they describe the same subspace. In our case, it is possible that the eigenvectors are just switching their orders rather than generate new eigenvectors. One method to validate this hypothesis is the normalized Frobenius matrix norm

$$\frac{1}{N_E}||U^T V||_F \tag{4.2}$$

where **U** and **V** are two matrices containing N_E eigenvectors, one is from the original training set with complete femurs and another one is from the training set with partial femurs added. The Frobenius matrix norm is high if two sets of eigenvectors span similar subspaces, and the value is low otherwise [Fowlkes et al., 2004]. Figure 4.4 Right shows the result of using different numbers of eigenvectors from our original training set and new training set which fill missing values using the mean value, as we can see for the top 10 eigenvectors in the two SSM training set describe the same subspace. If we use more than 10 eigenvectors, we will have different subspaces, in other words, we will have different principal components from the original training set.

4.2.2 Method 2: Filling missing values using SSM Fit Result

Instead of filling the missing values using the mean values in original training set, we can also fill the missing values using the SSM fit result for each partial femur. The assumption of this method is that the SSM can ensure the missing shape variance in a plausible degree and similar to the original meshes in the training set. Another advantage of using SSM fit result to fill missing values is that we avoid changing the direction of principal component due to missing values, since we add shape variance to all the axes we have.

To analysis the effect of adding SSM fit results to the shape vector, we also calculate the dot product of eigenvectors and the Frobenius matrix norm of new SSM and original, as shown in Figure 4.5. As we can see, the dot product of eigenvectors is higher than the dot product shown in Figure 4.4, which means comparing to previous method, the eigenvectors of this method is closer to the original SSM training set. Also there is less switching of eigenvectors as expected, the Frobenius matrix norm (Fig 4.5 Right) provides this information. For the top 20 eigenvectors, they have Frobenius matrix norm near 1, which means the top 20 eigenvectors in the new SSM training set describe the same principal components as the original SSM training set, only the order switches.

4.3 Incorporate Partial Femur and Train new SSM

After filling the missing values of SSM training set, we can perform PCA on the new training set and generate a new SSM. However, we do not want to introduce bias when adding partial femurs. So, we keep two SSM models, training set A which consists of 275 complete femur meshes is used for generating the missing values when we have new data. Another training set B contains training set A plus the shape vectors we extract from partial femur meshes. Training set B is used for segmentation task. The mechanism is shown in Figure 4.6.

To incorporate partial femurs, we use the original SSM training set to build point to point correspondence, fill missing values then add new shape vectors to training set B. If we have new MRI scans, we use training set B To perform segmentation task.

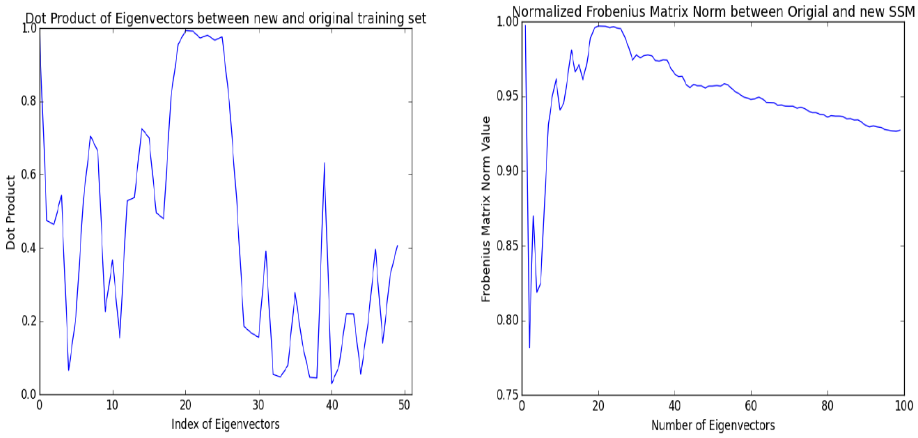


Figure 4.5: Analysis of Eigenvectors for filling missing values with SSM fit result. **Left:** The dot product of eigenvectors from new SSM and original SSM. **Right:** The Frobenius matrix norm of new SSM and original SSM.

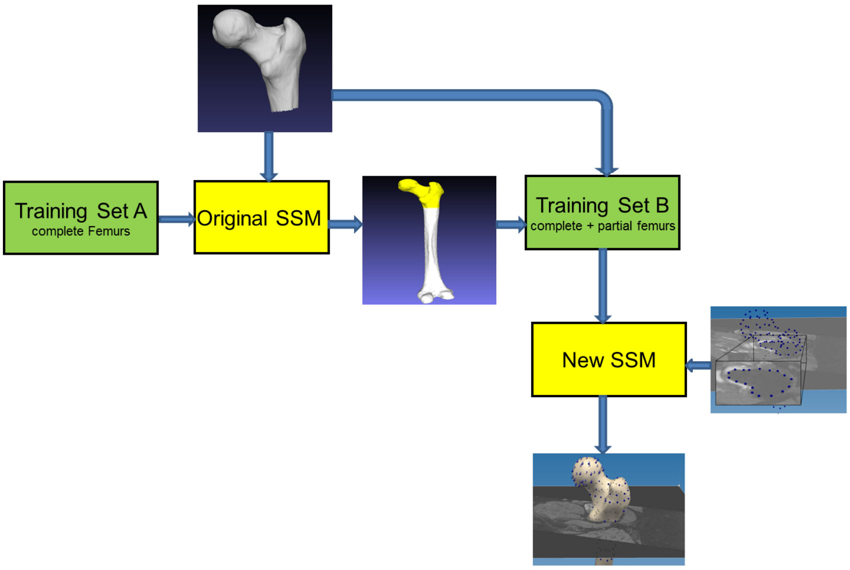


Figure 4.6: The mechanism of incorporating partial femurs

Chapter 5

Evaluation

We hypothesize that if we incorporate partial femurs to SSM training set, we can have more plausible shape variance so as to improve SSM fit accuracy. We expect it can reduce the RMSE, we also expect it can reduce the number of outliers.

A cross-validation method is conducted to test our hypothesis. We first present the evaluation method and then present the evaluation result.

5.1 Evaluation Method Design

After applying our rejection criteria, we get 300 partial femurs currently. Since we do not have a large amount of partial femurs, we want to make sure each partial femur is tested once. Cross-validation is a good choice to achieve this goal. Due to the computation time, leave-one-out method is not practical in this case. So we choose n-fold cross-validation methodology to evaluate our result. We first divide the 300 partial femurs into n subsets. Each time, one of the n subsets is used for testing and the other $k-1$ subsets are added to the SSM training set, as shown in Figure 5.1. After we get the error for each subset, we take the average of separate estimates error E_i as the true error estimate using equation 5.1, where K is the number of subsets in cross-validation.

$$E = \frac{1}{K}\sum_{i=1}^{K} E_i \tag{5.1}$$

We also want to evaluate whether incorporating more partial femurs will improve or degrade the SSM fit accuracy. So we use cross-validation with different number of subsets and compare their result. In our evaluation procedure, we set the number of subset to be 2, 3, 6 and 10, and evaluate their results respectively.

In the real scenario of Clinical Graphics, user select 400 landmark points manually or automatically and then use the SSM to fit the landmark points and achieve the segmentation. To make our evaluation more realistic, we also randomly select 400 landmark points from the partial meshes and then use the new trained SSM to fit the 400 landmark points and get the segmentation result. In real scenario, the partial femurs do not have the corresponding complete femurs to compare and evaluate the errors. So we choose another method to

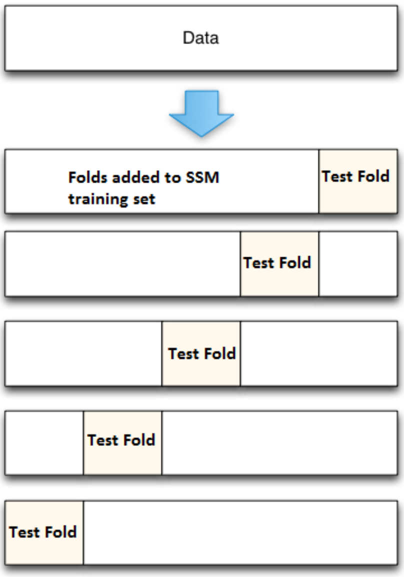


Figure 5.1: Illustration of a 5-fold cross-validation.

calculate the errors. When we get the SSM fit result for one partial femur, we search the nearest point for each vertex in the partial mesh and take their distance as the error for that vertex.

The RMSE is one criteria to evaluate the accuracy of SSM fit. However, it is possible that the fit result have a good performance in RMSE, but for some parts the errors are very high (see Fig 5.2). We also want to evaluate the performance in such case, so besides the RMSE to evaluate the general accuracy, we also evaluate whether the new SSM can reduce the number of vertices which has large error(e.g, larger than 2mm) in each partial femur.

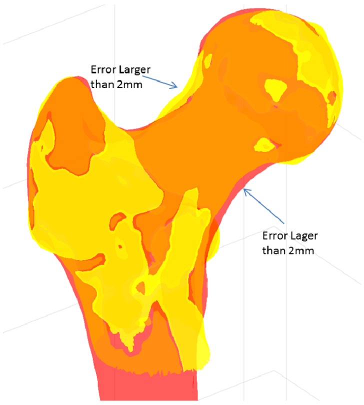


Figure 5.2: An example with low RMSE but for some parts the errors are larger than 2mm. Yellow one is the partial femur mesh. Red one is the SSM fit result

The procedure of evaluation is shown in Protocol 3.

Protocol 3: Procedure of k-Folds Cross-Validation

1. **Split the 300 partial meshes into k folds**
2. **Repeat k times**
    - **Select one fold for test**
    - **Add other $k-1$ folds partial femur meshes to the training set and train the new SSM**
    - **For each partial femurs in the test fold, randomly select 400 vertex as landmark points and perform SSM fit using the new trained SSM**
    - **Find the nearest point in the segmentation result for each point in partial mesh, calculate their euclidean distance as errors**
    - **Calculate the RMSE and the number of points larger 2mm for each partial femur**

5.2 Evaluation Result

We first fit the partial femur using the original SSM, then fit the partial femur using the new trained SSM with different amount of partial femurs and compare their performance. Since we have two different methods to fill the missing values, we evaluate them separately in the following two subsections.

5.2.1 Result on filling missing values using mean values

We first calculate the distribution of errors for all the vertices in the partial femurs, the results are shown in Figure 5.3. As we can see, compared to the original SSM, the new SSM with partial femurs can reduce the amount of large errors (larger than 1mm) and increase the number of small errors (smaller than 1mm).

The RMSE comparison between original SSM and the SSM trained using different folds are explained in Fig 5.4. As we can see, for all the new SSM with partial femurs, the results are in general slightly better than the original one. With more folds for cross-validation, which means more partial femurs added to the training set, we can have better results. But we also see from Figure 5.4, using the mean value to fill the missing values in shape vector, we have larger number of outliers, and for some cases, the new SSM fit result has much larger RMSE than the original SSM.

For the count of vertices with error larger than 2mm, we get similar result, as shown in Figure 5.5 . Adding partial femurs to the training set can reduce the mean value, with more

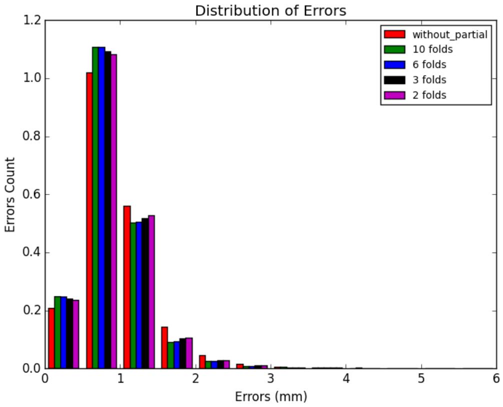


Figure 5.3: The distribution of errors for all the vertices from the partial femurs using different SSM.

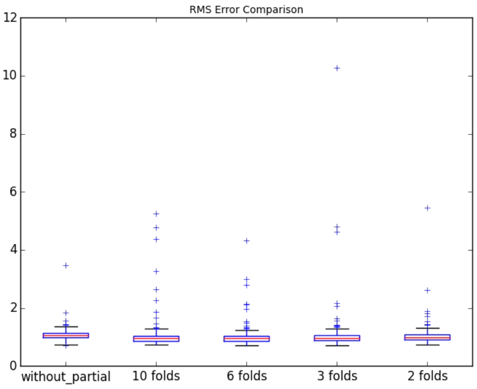


Figure 5.4: The comparison of RMSE using different SSM.

partial femurs, we can have better result. However, same as the RMSE, number of outliers increased.

5.2.2 Result on filling missing values based on SSM fit result

We use the same evaluation method and criteria to evaluate the performance of incorporating method based on SSM fit result. We calculate the distribution of errors for all the vertices in the partial femurs, the results are shown in Figure 5.6. As we can see, compared to the original SSM, the new SSM with partial femurs can reduce the amount of large errors(larger than 1mm) and increase the number of small errors(smaller than 1mm).

The RMSE comparison between original SSM and the SSM trained using different folds

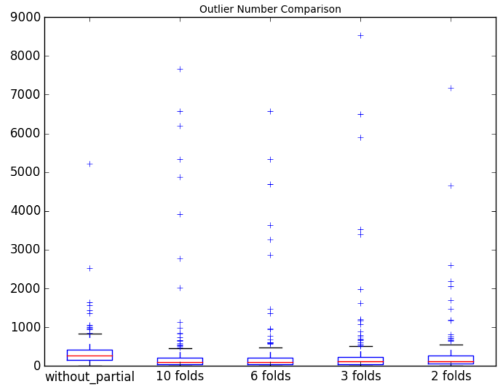


Figure 5.5: Comparison of number of vertices with error larger than 2mm.

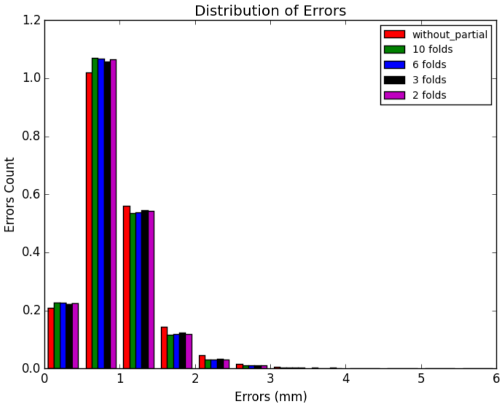


Figure 5.6: The distribution of errors for all the vertices from the partial femurs using different SSM.

are described in Fig 5.7. As we can see, for all the new SSM with partial femurs, the results are slightly better than the original one. With more partial femurs added to the training set, we can have better results. Unlike the incorporation method using mean value, Figure 5.7 shows that we reduce the amount of outliers and we also reduce the maximum RMSE (5mm).

The count of vertices with error larger than 2mm is shown in Figure 5.8 . As we can see, by adding more partial femurs to the training set, we also reduce the number of outliers in general.

As we can see from the evaluation result, filling the missing values with SSM fit result shows better performance than filling missing values with mean values. We also notice that with more partial femurs added into the SSM training set, we get lower RMSE and fewer outliers. We compare the fit result using original SSM and the new SSM filling missing

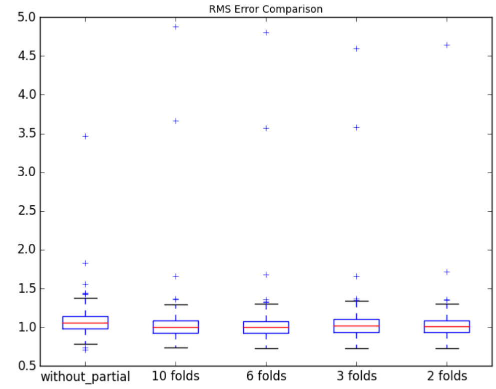


Figure 5.7: The comparison of RMSE using different SSM.

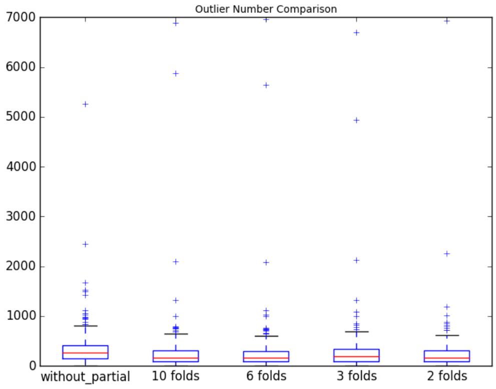


Figure 5.8: Comparison of number of vertices with error larger than 2mm.

values with SSM fit result. For the 300 partial femurs, the new SSM gives lower RMSE on 282 partial femurs. Using the original SSM, we get 56 outliers, using the new SSM, we get 24 outliers.

Chapter 6

Conclusions and future work

In this work, a method to incorporate partial femurs to SSM training set which contains complete femur meshes is presented. By adding partial femurs to the SSM training set, we hypothesized we can incorporate more reasonable shape variance to training set and improve the fit accuracy.

An evaluation of the fit accuracy for rejection criteria on partial femurs is performed. A partial femur mesh longer than 80mm can give same accuracy on the top part of femur as a complete femur is used. A partial femur longer than 80mm with bottom (knee) part gives same accuracy for the whole bone as using complete femur. Based on the evaluation result, we set the rejection criteria, as long as the femur is longer than 80mm, we accept it as training set candidate.

The method of building point to point correspondence is proposed. We use partial registration to register partial femur meshes to the complete bone and build the point to point correspondence, and turn the partial meshes into shape vectors with missing values.

We proposed two strategies to fill the missing values in the shape vectors of partial femurs. One is using the mean values in the existing complete femurs, the other one is using the SSM fit result to fill missing values. We analyzed their influence on the directions of principal components. Furthermore, we designed a mechanism to incorporating partial femurs to the training set.

We designed and executed one method to evaluate the performance for the new SSM training set with the two different strategies for filling the missing values. Both of the two methods can reduce the RMSE, and the method which uses SSM fit result to fill the missing values shows better performance in reducing the number of outliers and reducing the RMSE for most of the partial femurs.

In conclusion, the evaluation result shows that using our methods to incorporate partial femurs into the SSM training set, we can improve the fit accuracy, typically we can reduce the number of outliers. This evaluation supports our initial hypothesis that adding partial femurs to SSM training set can improve the fit accuracy.

For future work, we should find more partial femurs with bottom part and incorporate them into the SSM training set. As we can see from section 3.2, if we have partial femurs with bottom part, we can almost get the same accuracy as using complete femurs, so potentially adding accurate result for the whole bone.

Another suggestion would be trying to take the morphometric parameters (see Figure 6.1) into account to get better prediction of the SSM fit. According to the research of Blanc et al. [2012], if we take the Morphometric parameters and the SSM models together, using multi-variance regression, we can have a better prediction of the shape than just using SSM fit, this can be applied to fill the missing values in shape vector.

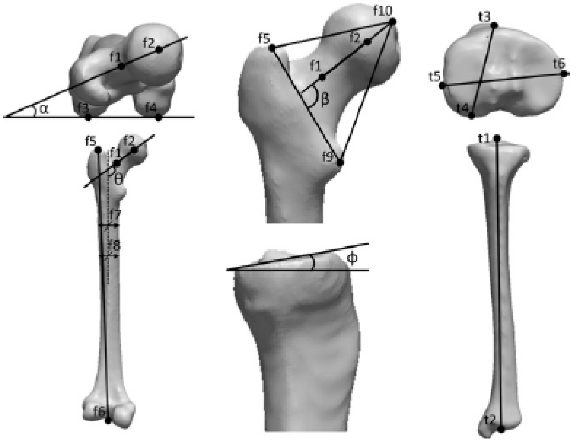


Figure 6.1: Some examples of morphometric parameters.

Bibliography

Emmanuel Audenaert. nonrigidICP. http://http://nl.mathworks.com/matlabcentral/fileexchange/41396-nonrigidicp/, 2015. [Online; accessed 19-July-2015].

Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.

Rémi Blanc, Christof Seiler, Gabor Székely, Lutz-Peter Nolte, and Mauricio Reyes. Statistical model based shape prediction from a combination of direct observations and various surrogates: application to orthopaedic research. *Medical image analysis*, 16(6):1156–1166, 2012.

Tim Cootes, ER Baldock, and J Graham. An introduction to active shape models. *Image processing and analysis*, pages 223–248, 2000.

J Michael Fitzpatrick, Derek LG Hill, and Calvin R Maurer Jr. Image registration. 2: 447–513, 2000.

Charless Fowlkes, Serge Belongie, Fan Chung, and Jitendra Malik. Spectral grouping using the nystrom method. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(2):214–225, 2004.

John C Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975.

Hao Li, Robert W Sumner, and Mark Pauly. Global correspondence optimization for non-rigid registration of depth scans. In *Computer graphics forum*, volume 27, pages 1421–1430. Wiley Online Library, 2008.

Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial & Applied Mathematics*, 11(2):431–441, 1963.

Neil Muller, Lourenço Magaia, and BM Herbst. Singular value decomposition, eigenfaces, and 3d reconstructions. *SIAM review*, 46(3):518–545, 2004.

Lauren J O'Donnell and Carl-Fredrik Westin. Automatic tractography segmentation using a high-dimensional white matter atlas. *Medical Imaging, IEEE Transactions on*, 26(11): 1562–1575, 2007.

Bernhard Preim and Charl P Botha. *Visual Computing for Medicine: Theory, Algorithms, and Applications*. Newnes, 2013.

Richard Szeliski. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

Dejun Tang, Weishi Zhang, Xiaolu Qu, and Dongqing Zhang. Improved active shape model and its application to facial feature extraction. *Journal of Electronic Imaging*, 22(3): 033023–033023, 2013.

Gaoyu Xiao, Sim Heng Ong, and Kelvin WC Foong. Efficient partial-surface registration for 3d objects. *Computer Vision and Image Understanding*, 98(2):271–293, 2005.

Appendix A

Active Shape Model from Clinical Graphics

In this section, we describe the Active Shape Model (ASM) implemented by Clinical Graphics on the femur. The ASM contain two main steps, first is training the Statistical Shape Model (SSM) that describe the shape and variance of femur from the training set, the second step is using the SSM to fit the new MRI scans of femur. The first step has been described in Section 2.2. Here we focus on the second step, how to use SSM to fit new MRI scans.

In a typical implementation of ASM, the algorithm search along the normal of surface to find the position which has the most similar intensity distribution to the current landmark point, as shown in Figure A.1. After finding the best positions for all the landmark points, ASM uses SSM as constraints to ensure the shape variance in a plausible degree Cootes et al. [2000].

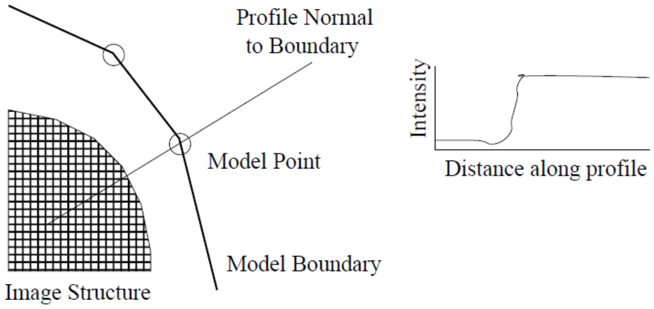


Figure A.1: Illustration of how does the typical ASM find the best match points. **Left:** Search along the normal of current boundary. **Right:** Calculate the difference of intensity distribution and find the minimal position (Image from [Cootes et. al 2000])

However, in practice, the intensity of MRI scans differs greatly from each other, the ASM may be not able to find the correct position for each landmark point. To ensure the accuracy, Clinical Graphics introduces human interaction as compensation. The users label some landmark points with some zones (e.g, Head, Nick, Epicondyle) on the MRI scan based on their domain knowledge. We take these labeled points which is called regression points as the correct landmark points. The number of manually regression points ranges from 50-100. Currently we are developing algorithm that can detect 400 landmark points, based on this requirement, we use 400 landmark points for evaluation in Chapter 5.

After getting the regression points, the ASM changes the shape parameters to find the model that best match the regression points. This procedure contains two steps. The first step is using the Iterative Closest Point (ICP) [Besl and McKay, 1992] algorithm to find the transformation matrix which can align the SSM and regression points to a same coordinate frame. During this procedure, only the first shape parameter b_1 is changed. As described in Section 2.2, the first parameter is corresponding to the size of femur. This procedure will repeat until a convergence is reached. After this step, the SSM are aligned to the same coordinate frame of regression points, with similar size to the regression points.

When the initial alignment is finished, the second iterative step of fitting starts. For each regression point, the algorithm try to find the nearest landmark points T in the SSM model. Then ASM changes the shape parameters $\mathbf{b}$ to find the best match the regression points and their corresponding landmark points in SSM, using the equation $\mathbf{T} = \overline{\mathbf{v}} + \mathbf{Pb}$. The shape vector $\mathbf{b}$ is updated using Levenberg-Marquardt algorithm [Marquardt, 1963] to find the vector $\mathbf{b}$ that minimize $T - \overline{\mathbf{v}}$. After find the shape parameter $\mathbf{b}$, we update the SSM models and repeat the whole procedure until the convergence is reached.

After these two steps, we can find the best fit result of regression points and get the segmentation result of MRI scan.

Appendix B

Principal Component Analysis

One important step in building Statistical Shape Model (SSM) is Principal Component Analysis (PCA). The PCA is a statistical method that convert a set of observations of possibly correlated variables into a set of linear uncorrelated variables using orthogonal transformation. These uncorrelated variables are called principle components. The principle components are the main axes of variance of the observations. One example of PCA in 2D case is shown in Figure B.1. As we can see, the PCA finds the directions (Red Arrows) which can preserve the largest variance, also each principal component is orthogonal to all the other principal components.

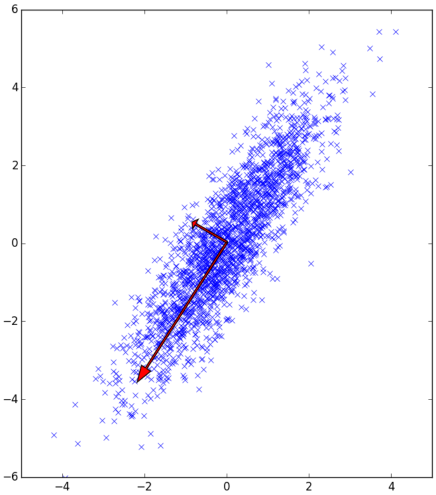


Figure B.1: Illustration of PCA.

Because the observations are possibly correlated, when they are projected into uncorrelated directions, it is possible to get lower dimensions comparing to the original observations. In this way, we can preserve the most of variance with a much lower dimension, thus we can save the computation time during SSM fitting procedure.

There are different methods to perform PCA. Clinical Graphics uses the Singular Value Decomposition (SVD) method. The SVD is of $m \times n$ matrix M is a factorization of the form $M = U\Sigma V^*$, where the U is an $m \times m$ matrix, V is an $n \times n$ matrix and Σ is an $m \times n$ diagonal matrix. The columns of U and V are called the left and right singular vectors of M. The right singular vectors V are equivalent to the eigenvectors of $M^T M$, and the diagonal entries $\Sigma_{i,i}$ of Σ are the singular values of M, it also the square root of eigenvalues λ_i of matrix $M^T M$. Using SVD we can perform PCA faster than the traditional way.

To get more information about SVD, please refer to Muller et al. [2004]

Appendix C

nonrigidICP Algorithm

In this chapter, we describe the nonrigidICP algorithm [Audenaert, 2015] in Detail. The nonrigidICP registers the source (partial) mesh S to the target (complete) mesh T. The nonrigidICP algorithm includes four main modules, pre-processing, rigidICP, general deformation and local deformation. We introduce them one by one.

The main task of pre-processing is assessing the quality of S and perform remeshing on T. The nonrigidICP algorithm first removes the nonsense faces, non-connected and duplicated vertices from both S and T. The second step is to calculate the mean μ and standard deviation λ of edges between connected vertices in S, so that we have knowledge about how the quality of S. The third step is to perform remeshing on T. If the edges in T are longer than 2μ, we subdivide them to generate more faces and edges. If the edges in T are shorter than $\mu - \lambda$, we use edge collapsing to remove them. After remeshing on T, the T and S have similar vertices distribution, so that we can search the nearest points for the vertices in S and perform the following operations.

The second step is rigidICP, which includes transformation, rotation and scaling to align S and T to a same coordinate frame. The rigidICP is described in Protocol 4.

Protocol 4: rigidICP

1. **For each vertex in S, find their nearest vertex in T, calculate their distance d**
2. **For each vertex in T, find their nearest vertex in S, calculate their distance d'**
3. **Calculate the mean μ_d and standard deviation λ_d of d we got in step 1**
4. **Using $\mu_d + 1.96\lambda_d$ as threshold, if the distance in d' is larger than the threshold, we take it as non-overlap part in T, and remove the corresponding vertices in T, get the partial target mesh T_p**
5. **Use Procrustes Analysis [Gower, 1975] to calculate the linear transformation Tr which can best transform S to T_p**
6. **Using the linear transformation Tr to transform S to new position which is closer to T**

7. **Repeat step 1-6 until converge**

After the rigidICP, the S and T are aligned together, as shown in Figure 4.2 Left. The third step is performing general deformation on S. The nonrigidICP algorithm uses the radial basis functions $\theta(r) = e^{-(\sigma\gamma)}$ to describe the surface of S. Then move the vertices in S into their corresponding positions calculated by radial basis functions. Then performing the rigidICP again to align S toward T. After repeating this procedure for 10-20 times, the deformed $S^{'}$ is well aligned and deformed like T.

The general deformation can give a smooth deformed $S^{'}$. However, in practice, the target mesh may not that smooth, their are some local difference between T and $S^{'}$, so we need a local deformation as the final step of nonrigidICP algorithm. The local deformation takes n nearest vertices of each vertex v_i in $S^{'}$, assign weight w_{ij} to each vertex based on their distance to v_i, the farther vertex get lower weight. These n vertices make one local mesh for deformation. For each vertex in the local mesh, calculate its vector **nv** to move towards its corresponding vertex in T, calculate the direction using equation $\mathbf{nv_i} = \frac{1}{n}\sum_{j=1}^{n} nv_{ij}w_{ij}$ and move v_i to the target position by $v_i + \mathbf{nv_i}$. The reason we use local deformation on local mesh instead of single vertex is to avoid the local holes in the T, refer to Li et al. [2008] for more detail. After 10-20 iterations of local deformation, we can get the registration result, the vertices' positions from source mesh on target mesh, as shown in Figure 4.2 Right.