

DETC2014-34789

**DRAFT: TOWARD PATIENT-SPECIFIC COMPUTATIONAL STUDY OF AORTIC DISEASES : A POPULATION BASED SHAPE MODELING APPROACH**

**Kang Li**

Mechanical, Materials, and Aerospace Engineering  
Illinois Institute of Technology  
Chicago, Illinois 60616

**Xiaoping Qian \***

Computational Design & Manufacturing Lab  
Department of Mechanical Engineering  
University of Wisconsin-Madison  
Madison, Wisconsin 53706

**Caitlin Martin**

Tissue Mechanics Lab  
Department of Biomedical Engineering  
Georgia Institute of Technology  
Atlanta, GA 30332

**Wei Sun**

Tissue Mechanics Lab  
Department of Biomedical Engineering  
Georgia Institute of Technology  
Atlanta, GA 30332

**ABSTRACT**

*Patient-specific computational study of aortic disease provides a powerful means for diagnosis and pre-operative planning. However, creating patient-specific computational models can be time-consuming due to the fact that anatomical geometries extracted from clinical imaging data are often incomplete and noisy. This paper presents an approach for constructing statistical shape models (SSMs) for aortic surfaces with the eventual goal of mapping the mean aortic geometries to raw surface data obtained from the clinical images for each new patient so that patient-specific models can be automatically constructed.*

*The input aortic models in this study come in the form of triangle meshes generated from CT scans on 6 patients. Statistical models with modes that characterize the variation pattern are found after optimizing the group-wise correspondence across the aorta training set. We use the direct reparameterization approach to efficiently manipulate shape correspondence. We use*

*B-spline based differentiable shape representation for the training set and use the adjoint method for deriving analytical gradients in a gradient based approach for manipulating the shape correspondence to minimize the description length of the resulting SSM. Our numerical result shows that the evaluation measures of the optimized statistical model have been significantly enhanced.*

**Keywords:** Ascending aorta, Statistical Shape Model (SSM), Shape Correspondence, Direct Re-parameterization, Adjoint method, B-spline

**1 INTRODUCTION**

Aortic diseases, including aneurysm of the ascending aorta and sinuses as well as calcification of the aortic valve, are significant causes of morbidity and mortality. Aortic aneurysm can lead to dissection and rupture, two likely fatal events. Aneurysms deemed at risk are surgically repaired by removing the dilated portion of the vessel and removing it with a prosthetic graft. Aortic valve disease is also often treated with prostheses. Tradition-

---

\*Correspondence can be addressed to Dr. Xiaoping Qian at qian@engr.wisc.edu.

ally, the diseased native aortic valve was surgically removed and replaced with a prosthetic valve. More recently, aortic valve disease has also been treated via transcatheter aortic valve (TAV) replacement, where a prosthetic valve is deployed over the native valve leaflets. Biomechanics largely dictate the success of these various treatments. For instance, a TAV device which exerts excessive radial force may rupture the aortic root, whereas insufficient radial force may lead to device migration.

An understanding of the aortic biomechanics can offer scientific rationale to design better treatments for these conditions. Computational analyses, e.g. structural finite element (FE) and computational fluid dynamic (CFD) simulations, are particularly useful for improving prosthetic device design because numeric analysis allows for a fast and inexpensive way of analyzing conceptual designs and design optimization [1]. However, the accuracy of these simulations are highly dependent on the material properties, geometries, and boundary conditions prescribed, and for the human ascending aorta and aortic root, these parameters are not easily defined and they can vary greatly from patient to patient. In the past, many groups have used idealized aortic geometries to simplify their analyses, but this will detract from the simulation accuracy.

In the past few years, clinical diagnostic imaging modalities have advanced significantly. Today, multi-slice CT, MRI, and 3D echocardiography can offer high resolution images of vasculatures that were previously unavailable. It is now feasible to utilize such medical images to accurately reconstruct 3D geometries of arteries and build computational models to perform structural analysis of the ascending aorta and aortic root wall on a patient-specific level. Such analysis can be used for pre-operative planning to determine the proper prosthetic device, size and positioning for a particular patient. Currently, these type of decisions are made primarily based on the physician's intuition and experience.

The caveat is that the generation of patient-specific computational models can be time consuming. Often the anatomical geometries extracted from the clinical imaging data are not suitable for computational analysis, i.e. the surface data is too noisy or contains artifacts or holes, due to poor image resolution. As a result, significant post-processing of the 3D geometry data is often necessary. In order for simulation-based pre-operative planning to be realistic in the clinical setting, this process must be expedited.

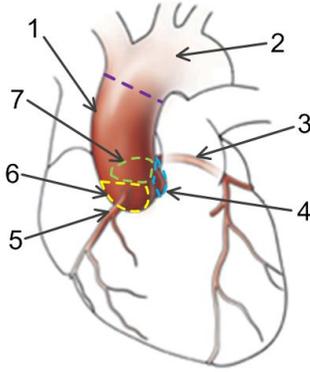
One possible solution to this problem is to create statistical shape models (SSMs) for the ascending aorta and aortic root. The mean ascending aorta and aortic root geometry could be mapped directly to the raw surface data obtained from the clinical images for each new patient, which will greatly reduce user input and consequently the time to complete the 3D geometry reconstruction process for the incoming patient. Aortic SSMs will also facilitate future probabilistic studies of the aortic biomechanics. While patient-specific analyses are essential for accu-

rate pre-operative planning, population-based probabilistic studies will be pivotal in the design of reliable valve and vessel prostheses and implantation techniques. The design of these devices should be robust to account for uncertainty in the tissue properties and anatomical geometries to avoid clinical adverse events and clarify patient selection criteria. Probabilistic computational analysis permits a rigorous quantification of various uncertainties and has been successfully applied to the design and analysis of a variety of engineering systems, including space vehicles and automobiles [2], and more recently, orthopedic implants [3–6]. In a probabilistic ascending aorta and aortic root computational model, the anatomical geometry will be defined as a random variable with shape variation defined from the aortic SSMs.

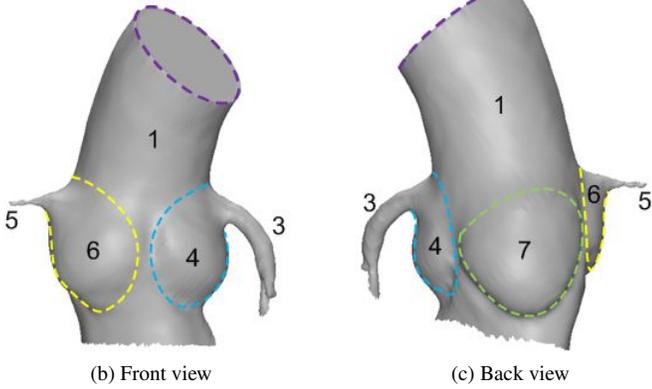
Statistical shape modeling is a powerful tool to capture the shape variation pattern across a group of shapes belong to a certain shape class [7]. SSM has seen many promising applications in a great variety of medical fields such as image analysis [8], image segmentation [9] [10], organ/bone shape reconstruction [11] [12], treatment tracking [13], patient-specific simulation [14] [15], diagnostics [16] [17] [18] and femoral allograft [19] and cam femoroacetabular impingement [20]. The utility of the statistical model relies on a sufficiently large training set data pool, and more importantly, a reasonably good correspondence across the entire training set. A good quality of the SSMs is usually obtained via manipulating correspondence across the shape populations in order to optimize some quality metric, e.g. the description length of the resulting SSM [21] [22]. The optimization is currently done via concatenation of multiple re-parametrization functions as proposed in [23], which is subject to undesirable computational efficiency. We propose a direct reparameterization scheme with better efficiency and use it in the statistical modeling for aorta in this paper. Our reparameterization function is represented by B-spline coefficients with diffeomorphic constraints to guarantee a valid deformation field in the parameter domain. The computation efficiency is further improved with a differentiable representation of training set shapes and the use of adjoint method for computing analytical gradients with respect to optimization variables.

This paper describes how we use our B-spline based reparameterization approach to build a statistical model from a given training set of 6 ascending aorta instances. The *ascending aorta* is the tube portion of the *aortic artery* starting from the upper base of left ventricle to the *aortic arch* where three branches originate from.

Figure 1(a) gives the anatomical overview of the location of the ascending aorta on heart in the anterior view of human body. The tube chunk shown consists of ascending aorta 1 (number legend) and aortic arch 2, separating roughly at the purple dotted section lines. At the lower part of ascending aorta, there are three sinuses and two outgoing arteries, namely *left coronary artery* 3 based on top of *left coronary sinus* 4; *right coronary artery* 5 based on top of *right coronary sinus* 6; non-coronary sinus 7



(a) Ascending aorta (anterior view)



(b) Front view

(c) Back view

**FIGURE 1:** Ascending aorta on heart with scanned data. Anatomical structure: 1) Ascending aorta (tube portion); 2) Aortic arch; 3) Left coronary artery; 4) Left coronary sinus; 5) Right coronary artery; 6) Right coronary sinus; 7) Non-coronary sinus.

(without no artery coming out). Figure 1(b)(c) are the front and back view of the isolated ascending aorta model in triangle mesh, where the sinuses and arteries are highlighted and numbered in consistency with Figure 1(a). The cut at the top side of the mesh corresponds roughly to the section line between ascending aorta and aortic arch, as denoted by the purple dotted line. The front view is in general different than the anterior view, and it is chosen so that the left and right coronary 4 and 6 sinuses face straight at us at the same time. The non-coronary sinus 7 then is located at the opposite side, as seen in the back view. The triangle mesh data of aorta shown in Figure 1(b)(c) are reconstructed from CT scanned images and are taken as the raw input for the work in this paper. The six shapes form the training set from which our statistical model is built.

The remainder of this paper is organized as follow. Section 2 first describes the basics for statistical shape modeling and elaborate mathematically our direct reparameterization scheme for correspondence optimization. Section 3 discusses how the input raw aorta data is pre-processed; Section 4 shows the optimized

shape correspondence and the SSM with our algorithm; Section 5 gives the conclusion and future work.

## 2 A direct reparameterization approach to statistical shape modeling

This section describes the technical procedure of statistical shape modeling based on the proposed direct reparameterization for shape correspondence manipulation. Figure 2 shows the overall flowchart of the proposed statistical modeling framework, which consists of: 1) The statistical analysis via the Principal Component Analysis (PCA) that extracts the shape variational pattern underlying the training set as SSM, namely several eigenmodes on top of a mean shape; 2) A novel correspondence manipulation scheme driven by the direct reparameterization functions.

### 2.1 Statistical Shape Modeling

Since the PCA takes a matrix as input whose columns are the shape vectors of physical coordinates representing each training set instance, it naturally follows that the shape vectors length for all instances must be identical. This in turn requires that the same number of points, called the *landmarks* [24], to represent those instances. This landmark based representation is also referred to as the *Point Distribution Model* (PDM) [25]. Regardless of the geometric form of the training set, landmarks are constrained to be on the boundary of a shape instance of the given training set, and they form a point-based representation that approximates the original shape [23].

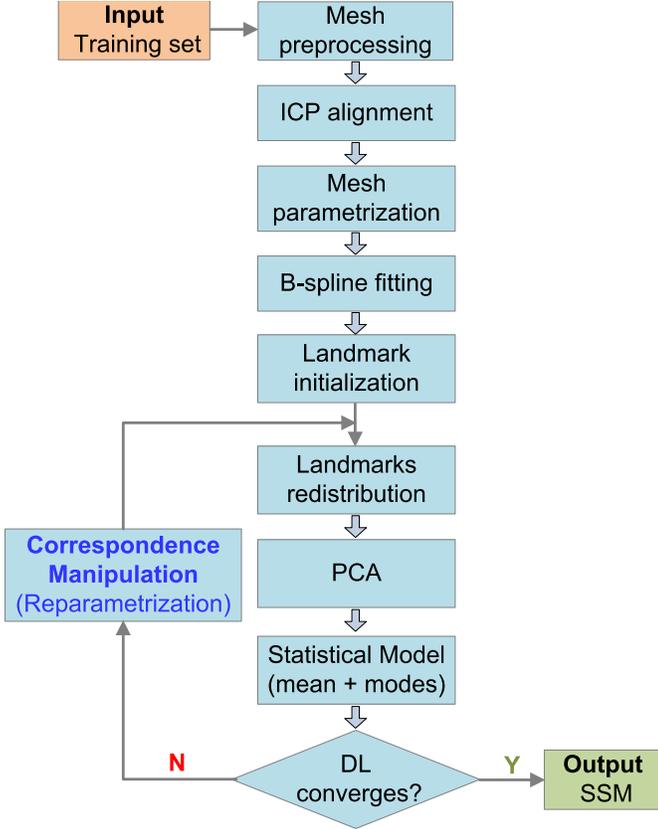
Suppose a training set  $\{\mathbb{T}_i\}$  ( $i = 1, 2, \dots, n_S$ ) comprises  $n_S$  shape instances and each is represented by  $n_P$  landmarks. The  $j$ -th landmark of the  $i$ -th instance is  $\mathbf{x}_i^{(j)} = [x^{(j)}, y^{(j)}, z^{(j)}]^T \in \mathbb{R}^3$ . Owing to the correspondence assumption of landmarks, all the  $n_S$  landmarks  $\{\mathbf{x}_i^{(j)}\}$  ( $i = 1, 2, \dots, n_S$ ) with label  $j$  should correspond across all instances. The landmark representation of each instance is usually written into a concatenation of  $n_P$  landmarks ordered by labels as a *shape vector* expression:

$$\mathbf{x}_i \doteq [x^{(1)}, y^{(1)}, z^{(1)}, x^{(2)}, y^{(2)}, z^{(2)}, \dots, x^{(n_P)}, y^{(n_P)}, z^{(n_P)}]^T$$

All the  $n_S$  shape vectors  $\{\mathbf{x}_i\}$  could be concatenated into a  $3n_P \times n_S$  shape vector matrix:

$$\mathbf{X}^S \doteq [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{n_S}] \quad (1)$$

The landmark represented shapes are then group-wise aligned by the *Generalized Procrustes Analysis* (GPA) [26]; this is done by performing the pair-wise Procrustes Analysis (PA) iteratively between each shape and the mean; the PA is used to



**FIGURE 2:** Flowchart of the statistical shape modeling for aorta training set

bring a shape vector  $\mathbf{x}$  to a fixed shape vector  $\mathbf{y}$  by similarity transformation as below

$$\min_{\mathbf{t}, s, \mathbf{R}} \|\mathbf{y} - s\mathbf{R}(\mathbf{x} - \mathbf{t})\|^2 \doteq \sum_{j=1}^{np} \left\| \mathbf{y}^{(j)} - s\mathbf{R}(\mathbf{x}^{(j)} - \mathbf{t}) \right\|^2 \quad (2)$$

where shape irrelevant factors including translation  $\mathbf{t}$ , scaling  $s$  and rotation  $\mathbf{R}$  are removed. This alignment operation is denoted by an alignment operator.

$$\mathbf{X}^A = \mathcal{A}(\mathbf{X}^S) \quad (3)$$

The training shapes  $\mathbf{S}(\mathbf{u})$  are continuous, we thus have the continuous representation of the covariance matrix entries for shape  $\mu$  and shape  $\nu$

$$\mathbf{E}_{\mu\nu} = \frac{1}{(n_S - 1)A} \int [\mathbf{S}_\mu(u) - \bar{\mathbf{S}}(u)] \cdot [\mathbf{S}_\nu(u) - \bar{\mathbf{S}}(u)] dA(u) \quad (4)$$

where  $\mathbf{S}(\cdot)$  is the vector-valued shape function that defines the continuous representation of the  $i$ -th shape by mapping the parameter space to the physical space.  $\bar{\mathbf{S}}(u)$  is the mean shape and  $A$  is the surface area of the mean shape. For numerical implementation, the continuous covariance matrix is obtained via discretization through a set of discrete landmarks is

$$\mathbf{E}_{\mu\nu} = \frac{1}{(n_S - 1)n_P} \sum_{i=1}^{n_S} (\mathbf{x}_i - \bar{\mathbf{x}})_\mu (\mathbf{x}_i - \bar{\mathbf{x}})_\nu. \quad (5)$$

This could be written simply in a matrix form [21]

$$\mathbf{E} = \frac{1}{(n_S - 1)n_P} \mathbf{X}_c^T \mathbf{X}_c \quad (6)$$

where  $\mathbf{X}_c$  is defined by

$$\mathbf{X}_c \doteq [\mathbf{x}_1^A - \bar{\mathbf{x}}, \mathbf{x}_2^A - \bar{\mathbf{x}}, \dots, \mathbf{x}_{n_S}^A - \bar{\mathbf{x}}] \quad (7)$$

and the *mean shape vector* of SSM is

$$\bar{\mathbf{x}} = \frac{1}{n_S} \sum_{i=1}^{n_S} \mathbf{x}_i^A \quad (8)$$

The *Principal Component Analysis* (PCA) [27] is then conducted to extract the principal modes of shape variability via the eigenvalue decomposition of the covariance matrix

$$\mathbf{E}\mathbf{v}_m = \lambda_m \mathbf{v}_m \quad (m = 1, 2, \dots, n_S - 1) \quad (9)$$

where  $\mathbf{v}_m$  is the  $m$ -th eigenvector and  $\lambda_m$  the corresponding eigenvalue. The mean shape  $\bar{\mathbf{x}}$ , modes  $\{\mathbf{v}_m\}$  and eigenvalues  $\lambda = \{\lambda_m\}$  constitutes the SSM.

## 2.2 Minimum description length based quality evaluation

A quality SSM can be obtained when a good group-wise correspondence across the training set is achieved. One approach to achieve good quality SSM is through optimization where an objective function characterizing the quality of the associated SSM is optimized. The selection of such quality measure has been extensively studied [25] [21] when the group-wise correspondence is to be optimized simultaneously. One current widely recognized measure is the *Description Length* (DL), a concept borrowed from the *Minimum Description Length* (MDL) principle in information theory. The MDL principle regards the statistical model comprising the mean and modes as a message

that transmits the entire training set, and the DL is the length of the message. According to the Occam's razor, the simplest message is the best, and the SSM with the minimum description length value is the best and is associated with optimal correspondence.

In our optimization formulation, DL is thus taken as the objective function. It was originally derived by Davies in [21] and elaborated in [23]. A simplified version presented in [28] defined as below is used in our correspondence optimization:

$$f \doteq \sum_{m=1}^{n_S-1} L_m \quad (10)$$

where each mode's contribution is

$$L_m = \begin{cases} 1 + \log(\lambda_m/\lambda_{\text{cut}}) & \lambda_m \geq \lambda_{\text{cut}}, \\ \lambda_m/\lambda_{\text{cut}} & \text{otherwise.} \end{cases}$$

The threshold  $\lambda_{\text{cut}}$  is determined by landmark resolution and shape scale

$$\lambda_{\text{cut}} = \frac{2l_{\text{min}}}{r_{\text{max}}}, \quad (11)$$

where  $l_{\text{min}}$  is the smallest edge length in the landmark-based representation and  $r_{\text{max}}$  is the radius of largest circumscribing sphere over training set shapes.

### 2.3 Direct reparameterization for correspondence manipulation

In general, the optimization typically starts with an initial state of correspondence as shown in Figure 2(b), and the subsequent iterations will seek to improve this correspondence until a minimal DL objective value is reached. The improvement of correspondence entails a correspondence manipulation scheme by redistributing the landmarks on training set instances. An easy way to manipulate correspondence by moving landmarks around is to first establish a one-to-one mapping from the physical domain where the training set shapes lie in a parameter domain, and then redistribute landmarks by moving the landmarks parameter points in the parameter domain. The search for such mapping is known as the parameterization. The curve parametrization is mostly straightforward, and surface mesh parametrization has seen tremendous literature [29].

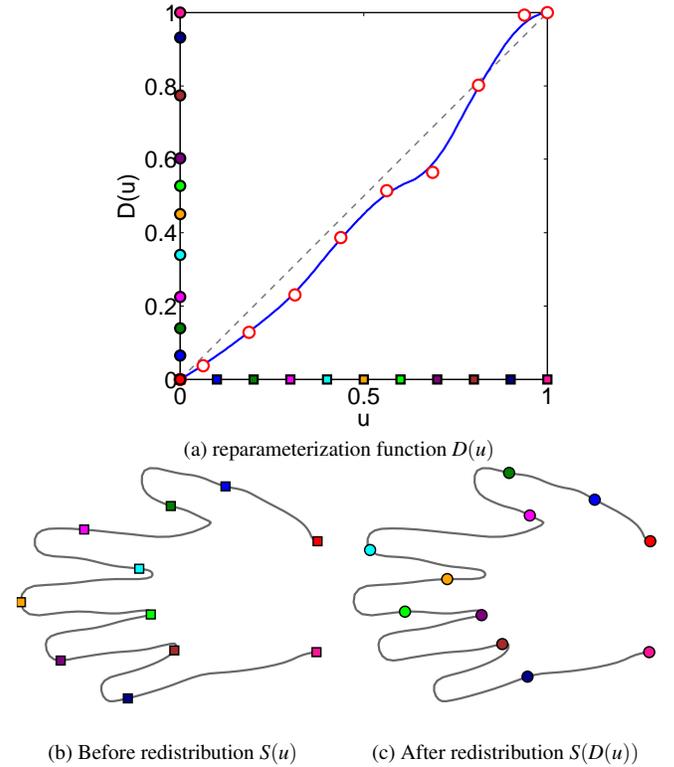
With a parametrization  $\mathbf{S}$  that maps a parameter point  $\mathbf{u}$  to a point  $\mathbf{S}(\mathbf{u})$ , the landmark redistribution of landmarks could be achieved by relocating the landmark parameter  $\mathbf{u}$  by a *reparameterization function*  $\mathbf{D}(\mathbf{u})$  so that the original landmark  $\mathbf{S}(\mathbf{u})$  is redistributed to a new position  $\mathbf{S}(\mathbf{D}(\mathbf{u}))$ .

**2.3.1 Direct reparameterization for curve** For the curve case, taking the hand as an example, Figure 3(a) is the second row of Figure 2(c), and it shows the reparameterization function  $D(u)$  for the Shape 2. After applying the reparameterization function to the initial landmarks in Figure 3(b) ( same as the second row in Figure 2(b)), the landmarks are redistributed to the state in Figure 3(c) (same as the second row in Figure 2(d)).

Mathematically,  $D(u)$  is directly represented by a single B-spline function with  $n_b$  coefficients

$$D(u) = \sum_{i=0}^{n_b-1} B_{i,p}(u)b_i, \quad 0 \leq u \leq 1; \quad (12)$$

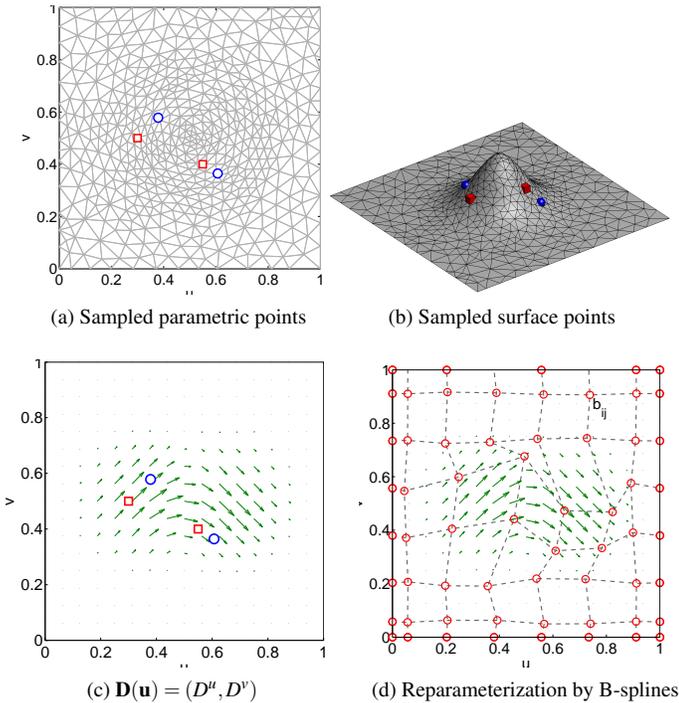
where  $B_{i,p}$  is the B-spline basis function [30] of degree  $p$  associated with the  $i$ -th B-spline coefficient  $b_i$  recursively defined on a non-decreasing knot vector  $\Xi = \{\bar{u}_0, \bar{u}_1, \dots, \bar{u}_{n+p+1}\}$ .



**FIGURE 3:** Reparameterization function for a curve and landmarks before and after reparameterization

**2.3.2 Direct reparameterization for surface** A reparameterization of a 3D surface is illustrated in Figure 4

where cubes and spheres respectively represent sampled points before and after the reparameterization. Since a 3D surface  $\mathbf{S}(\mathbf{u}) = (x(\mathbf{u}), y(\mathbf{u}), z(\mathbf{u}))$  is mapped to a 2D parametric domain, i.e.  $\mathbf{u} = (u, v)$ , the reparameterization  $\mathbf{D}(\mathbf{u})$  for 3D surfaces  $\mathbf{S}(\mathbf{u})$  have two components in  $u$  and  $v$  directions, i.e.  $\mathbf{D}(\mathbf{u}) = (D^u(u, v), D^v(u, v))$ , as shown in Figure 4. These two components effectively form a reparameterization vector field (Figure 4e).



**FIGURE 4:** Reparameterization vector field  $\mathbf{D}(\mathbf{u})$  and sample redistribution on a 3D surface: cubes and spheres respectively represent points before and after reparameterization.

It's proposed to use a single B-spline function to directly represent the reparameterization of  $u$  and  $v$  component of the parametric domain, as shown in Figure 4(f) where  $8 \times 8$  B-spline coefficients  $b_i$  (red circles) are used to represent the reparameterization field  $\mathbf{D}(\mathbf{u})$ . In general, the reparameterization  $\mathbf{D}(\mathbf{u}) = (D^u(u, v), D^v(u, v))$  in the square planar parameter do-

main is defined as

$$\begin{aligned} D^u(u, v) &= \sum_{i=0}^{n_{b1}-1} \sum_{j=0}^{n_{b2}-1} B_{i,p}(u) B_{j,q}(v) b_{i,j}^u \\ D^v(u, v) &= \sum_{i=0}^{n_{b1}-1} \sum_{j=0}^{n_{b2}-1} B_{i,p}(u) B_{j,q}(v) b_{i,j}^v \end{aligned} \quad (13)$$

$$0 \leq u, v \leq 1;$$

where  $B_{i,p}$  and  $B_{j,q}$  are the B-spline basis functions (??) of degree  $p$  and  $q$  associated with the  $(i, j)$ -th B-spline coefficient 2-tuple  $\mathbf{b}_{i,j} = (b_{i,j}^u, b_{i,j}^v)$ ; the coefficient number along the  $u$ - and  $v$ -direction are  $n_{b1}$  and  $n_{b2}$  respectively. They are respectively defined on two sets of non-decreasing knot vector  $\Xi_1$  and  $\Xi_2$ .

## 2.4 Optimization formulation

With the above B-spline representation of reparameterization functions  $\mathbf{D}(\mathbf{u})$ , we thus have the following optimization formulation for using B-spline based reparameterization for manipulating shape correspondence:

$$\min_{\mathbf{b}} f(\mathbf{b}) = \sum_{\lambda_i \geq \lambda_{\text{cut}}} \left[ 1 + \log \frac{\lambda_k(\mathbf{b})}{\lambda_{\text{cut}}} \right] + \sum_{\lambda_k < \lambda_{\text{cut}}} \frac{\lambda_k(\mathbf{b})}{\lambda_{\text{cut}}} \quad (14a)$$

$$\text{s.t. } [\mathbf{C}^T(\mathbf{b})\mathbf{C}(\mathbf{b})] \mathbf{v}_k(\mathbf{b}) = \lambda_i(\mathbf{b}) \mathbf{v}_k(\mathbf{b}) \quad (14b)$$

$$\mathbf{v}_k^T(\mathbf{b}) \mathbf{v}_k(\mathbf{b}) = 1, \quad k = 1, \dots, n_S \quad (14c)$$

$$\mathbf{g}(\mathbf{b}) < 0 \quad (14d)$$

In this formulation,  $\mathbf{b}$  is the set of optimization variables and represents the collection of interior B-spline coefficient tuples  $\mathbf{b}$  for  $n_S - 1$  shapes, where one shape from the training set is selected as a reference. The objective function  $f(\mathbf{b})$  is the simplified description length, which is a function of eigenvalues computed from (14b) and (14c). The matrix  $\mathbf{C}$  is related to the covariance matrix  $\mathbf{E}$  by  $\mathbf{E} = \mathbf{C}^T \mathbf{C}$  with

$$\mathbf{C} = \frac{\mathbf{X}_c}{\sqrt{(n_S - 1)n_P}}.$$

The constraint (14d) represents the diffeomorphic conditions for curves and surfaces, each of which is a function of optimization variables  $\mathbf{b}$ . The diffeomorphic conditions guarantees the bijectivity after applying reparameterization function for correspondence manipulation; it's straight forward for curve case, and for the surfaces it's fulfilled by enforcing the Jacobian positivity constraints.

### 3 Training set data preprocessing

#### 3.1 Mesh preprocessing

The raw input ascending aorta of 6 shape instances are shown in Figure 5 with distinct colors. They all contain the ascending aorta tube portion as the major part, but vary in the detailed features for sinuses and particularly for coronary arteries. Since the outgoing arteries geometries differ greatly from one another, they will be excluded from major tube part and we are focusing on the sinus part and the tube portion. In order for the data to be processable by our proposed algorithm, the raw training set must undergo a series of preprocessing procedures that resolve issues such as incomplete data, noise and smoothness etc.

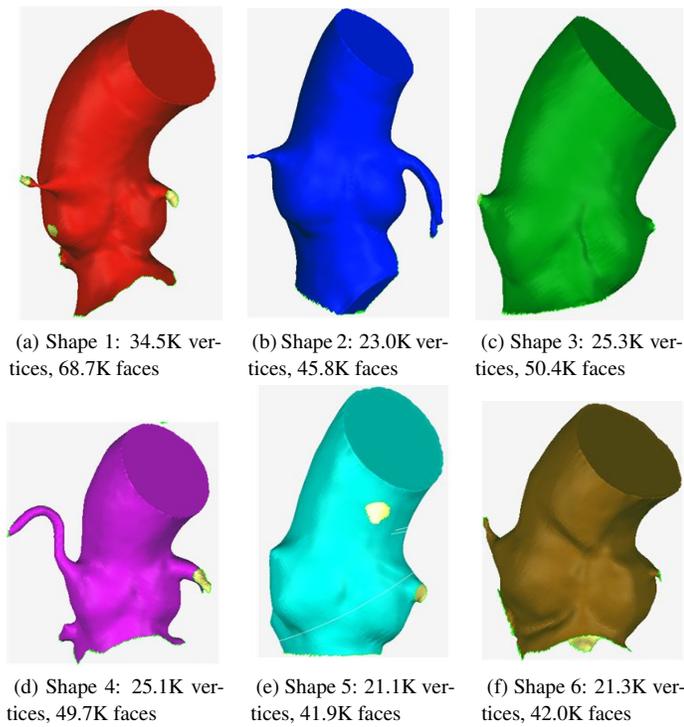


FIGURE 5: Raw triangle meshes of aorta data in front view

**3.1.1 Hole filling** Incomplete data is a common issue from reconstructed mesh from CT scans. This usually comes in the form of holes, which can be grouped into two types depending on the requirements of hole filling: 1) direct filling; 2) hole flattening and filling. Type 1 is usually minor data loss and has mild curvature variation in the vicinity; Type 2 is usually associated with a cut-off artery stemming from the sinus; in this case, direct filling of hole cannot recover the geometry at the cut-off location around the thin artery tube. The little influence of the

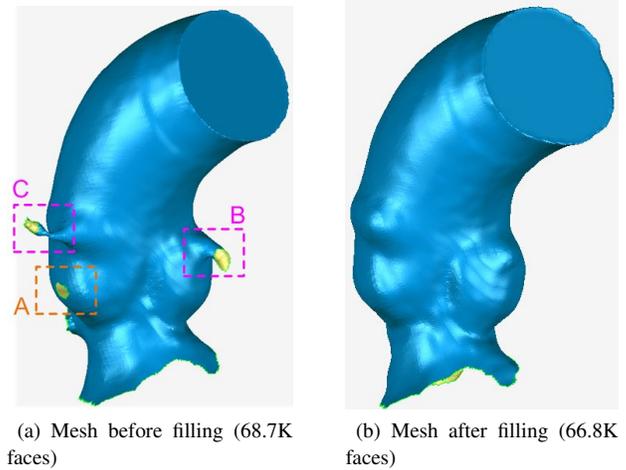


FIGURE 6: Mesh hole filling and hole types. Type 1: direct filling (hole A); type 2: flattening and filling (hole B and C)

artery tube also make it reasonable to remove the artery feature and seal the flattened hole off before doing any direct hole filling.

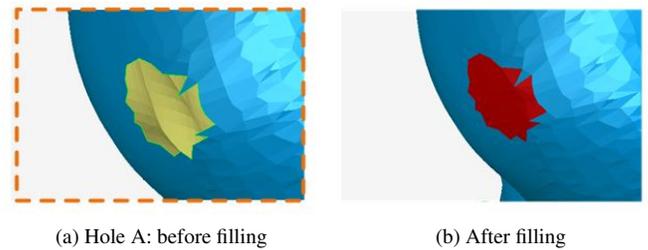


FIGURE 7: Direct hole filling for type 1 (hole A)

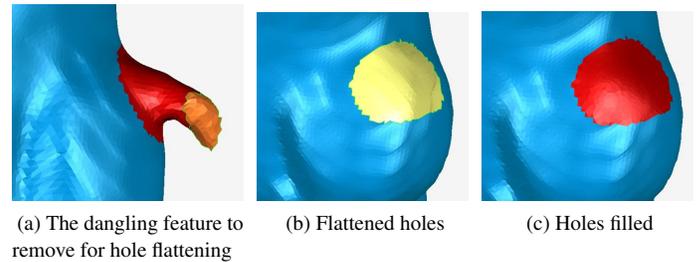
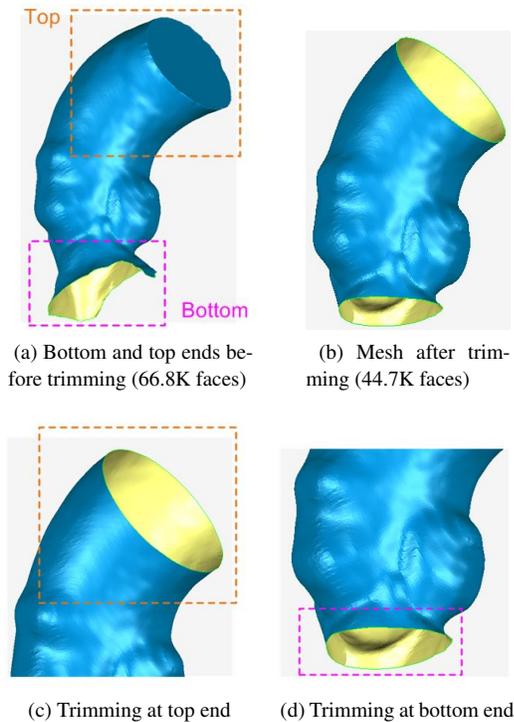


FIGURE 8: Flattening and filling for type 2 (hole B)

Taking Shape 1 as an example as seen in Figure 6(a), hole A located on the right coronary sinus belongs to Type 1, which can be directly filled by mesh completion algorithm as shown

in Figure 7. The other two holes B (left coronary artery) and C (right coronary artery) are of Type 2, where a feature removing step is necessary before mesh completion algorithm is applied. Figure 8(a) shows the feature portion around hole B expected to be eliminated, and a flattened hole in Figure 8(b) follows, and in the end the mesh completion based on curvature constraint would result in the filled hole show in Figure 8(c). Similarly hole C could be flattened and filled in the same way, and the aorta mesh after the hole filling procedure is shown in Figure 6(b).

**3.1.2 End trimming** The proposed algorithm currently can only handle training set geometry topologically equivalent to a disc. The geometry in Figure 6(b) after the hole filling is complete satisfies such criterion but wasted a rather large area at the top end since the planar cut between ascending portion and aortic arch carries no variation of the aortic tube surface. Therefore it is advisable to trim the top end to an open end and map the remaining cylindrical part onto a square domain. The top end trimming occurs as shown in Figure 9(c) compared to the original state in Figure 9(a).

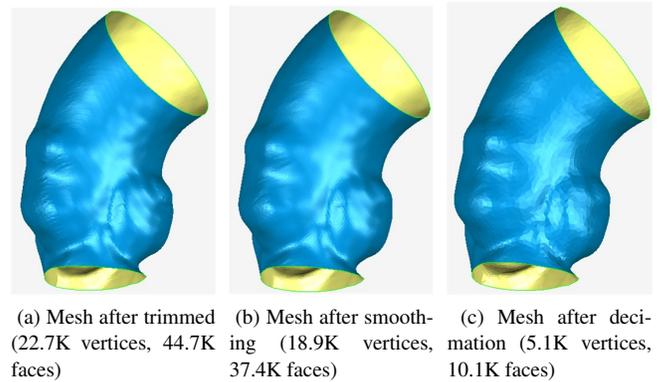


**FIGURE 9:** Trim mesh at top and bottom

For the bottom part, the factor of noise and data separation all make the bottom portion unreliable to faithfully represent the lower part of the ascending aorta. The irregular mesh bound-

aries observed for Shape 1, 2, 4 and 6 in Figure 5(a)(b)(d)(f) have confirmed this need for bottom end trimming. Additionally, when the bottom portion lack data to be trimmed as is the case for Shape 3 in Figure 5(c), then mesh boundary must first be extended to create enough data to be trimmed based on the bordering mesh's curvature information. The mesh after bottom end trimming would look like Figure 9(d) where the mesh boundary becomes much more regular and smooth. After the top and bottom trimming, the aorta mesh becomes the one in Figure 9(b) and the number of faces drops from 66.8K to 44.7K.

It's also desirable that the manual trimming position of all the 6 shapes are at approximately the corresponding location across the entire training set, considering the fact that the correspondence of the boundaries are assumed to be fixed and only the interior correspondence is optimized in our approach.



**FIGURE 10:** Mesh smoothing and decimation

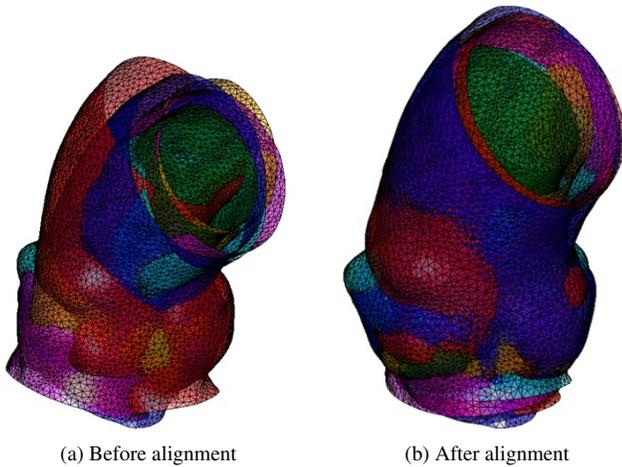
**3.1.3 Smoothing and decimation** The illustrative data obtained till trimming is shown in Figure 10(a). To further reduce the noise factor, we apply a smoothing filter on the mesh based on local mesh curvature to obtain the smoothed mesh in Figure 10(b) where the number of faces drops from 44.7K to 37.4K. For a more compact representation, the smoothed mesh went on to be decimated from 37.3K to 10.1K.

During the entire mesh pre-processing procedures consisting of hole filling, end trimming, smoothing and decimation, the number of vertices and triangle faces are generally changing. Table 1 keeps track of this information as each procedure progresses.

**TABLE 1:** Vertex and triangle numbers during SSM preprocessing.

Vertex# (Face#)	Raw	Filled	Trimmed	Smoothed	Decimated
Shape 1	34569 (68717)	33531 (66838)	22774 (44734)	18927 (37405)	5116 (10099)
Shape 2	23049 (45828)	20739 (41300)	16814 (32871)	13994 (27571)	5162 (10199)
Shape 3	25345 (50441)	25939 (51766)	20945 (41002)	17324 (34146)	5185 (10243)
Shape 4	25102 (49714)	22861 (45558)	18823 (36745)	15547 (30591)	5117 (10094)
Shape 5	21097 (41940)	21431 (42764)	17882 (30540)	14402 (28409)	5179 (10227)
Shape 6	21271 (42047)	20903 (41663)	17242 (33651)	14436 (28469)	5188 (10248)

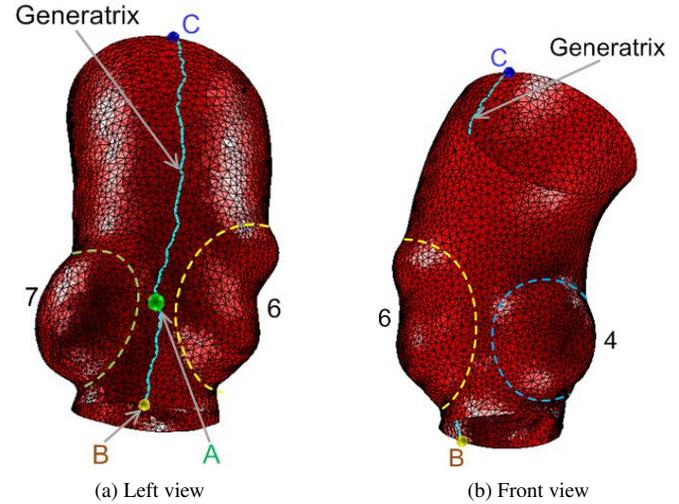
### 3.2 Training set B-spline fitting



**FIGURE 11:** 6 meshes before and after one-time alignment with ICP algorithm

**3.2.1 One-time mesh alignment** Since in the actual correspondence optimization, the alignment process is discarded for additional speed gain, and this requires that the shapes should already be aligned before they are fed into the optimization algorithm. We choose to perform a one-time alignment of the triangle meshes. The number of vertices of all aorta meshes after the mesh preprocessing is not identical across the trainings set as suggested by the last column in Table 1, the Iterative Closest Point (ICP) algorithm is employed to align these meshes. Before alignment, the 6 triangle meshes are superimposed and shown in Figure 11(a). The ICP aligned meshes of training set are shown

in Figure 11(b), where the 6 meshes are brought to a much better alignment state.



**FIGURE 12:** Generatrix determination. Number legend: 4). Left coronary sinus; 6). Right coronary sinus; 7) Non-coronary sinus

**3.2.2 Generatrix determination** Recall that our SSM algorithm requires a square domain, whereas the current triangle mesh is topologically equivalent to a cylinder. One simple way to obtain this is to cut the mesh along a line and unfold and map it onto the square domain, and this cutting line is just the “generatrix” to be determined. Since it’s better to leave the area between the left and right coronary sinus, namely 4 and 6, intact from whatever influence brought by the cutting line. The cutting line is placed on the back side between the right coronary sinus 6 and the non-coronary sinus 7 as shown in Figure 12(a).

Taking Shape 1 as an example, the first step is to manually specify vertex A, that’s supposed to lie at a valley point between 6 and 7. Then the Dijkstra’s algorithm is utilized to find a vertex at the bottom and top boundaries that has the shortest geodesic distance along mesh edges to A. Comparison within boundary vertices at bottom and top side gives vertex B and C. Connecting A with both B and C lead to the generatrix line BAC. Figure 12(b) shows where the generatrix line is located in the front view. Repeating these steps give the generatrix lines for the remaining 5 shapes as displayed in Figure 13.

**3.2.3 Mesh parametrization** After the generatrix is available, it’s safe to cut along it and the generatrix edge line on mesh would serve as the image mapped to two opposite sides of the square parameter domain. The generatrix and bottom and top

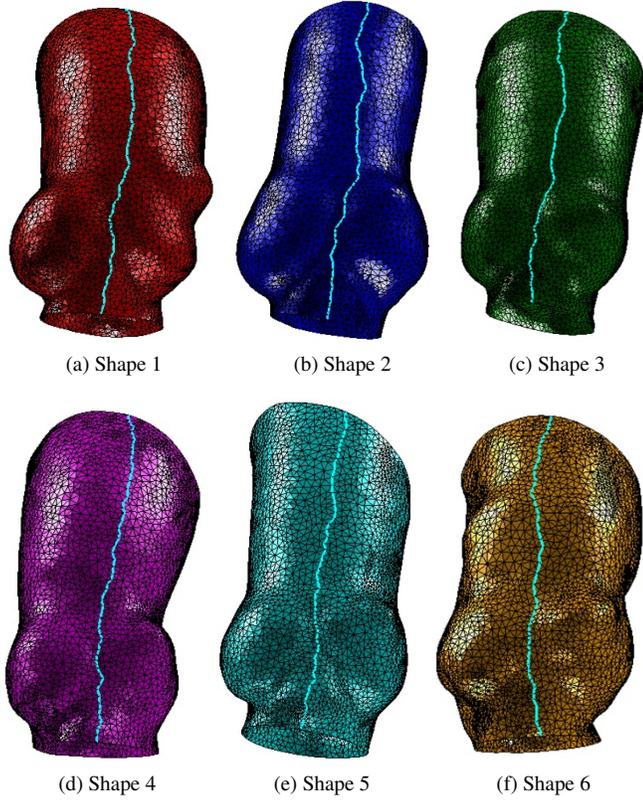


FIGURE 13: Generatrix on all the 6 shapes in left view

boundaries are highlighted in green in Figure 14(a). The two vertices of the generatrix at bottom and top would serve as the four corner vertices of the square domain as shown in Figure 14(b).

Suppose each raw shape instance  $\mathbb{T}D_i$  is represented by a triangulated mesh consisting of vertex list  $\mathcal{V}_p = \{\mathbf{p}_j\} (j = 1, \dots, n_v)$  with an associated triangle list  $\mathcal{T} = \{\mathbf{T}_k\} (k = 1, \dots, n_t)$ , where the  $j$ -th vertex is  $\mathbf{p}_j = [x_j, y_j, z_j]$ , and the  $k$ -th triangle is of vertex index set  $\mathbf{T}_k = [\tau_{k1}, \tau_{k2}, \tau_{k3}]$ , and  $n_v$  and  $n_t$  are the number of vertices and triangles respectively. The mesh parametrization procedure contains two steps, namely 1). the initial parametrization; 2) parametrization improvement by minimizing distortion.

**Initial parametrization** Mesh parametrization seeks to find a mapping of vertices between physical domain and the parameter domain namely  $\mathcal{V}_U = \{\mathbf{u}_j\} = \{u_j, v_j\}$ . If a square parameter domain is chosen, the  $n_b$  boundary vertices parameters set  $U_b = (\mathbf{u}_{n_e+1}, \dots, \mathbf{u}_{n_v})$  on the four sides could be either manually determined by know correspondence, or by the correspondence manipulation for 2D case, where  $n_e = n_v - n_b$  is the number of interior vertices. The standard approach to obtain a fold-over free vertices parameter with attached triangulation is to adopt the simple idea that the edges of the triangle mesh are spring connected at vertices [29]. In this spring model, the minimum spring energy

state is reached when each interior parameter point  $\mathbf{u}_j$  is an affine combination of its neighbors, i.e.

$$\begin{aligned} \mathbf{u}_j &= \sum_{k \in N_j} w_{jk} \mathbf{u}_k \\ \text{s.t. } \sum w_{jk} &= 1 \end{aligned} \quad (15)$$

where  $w_{jk}$  is the normalized weight coefficients of the neighbor set  $N_j$  for the  $j$ -th parameter point. Separating interior and boundary vertices gives

$$\mathbf{u}_j - \sum_{k \in N_j, k \leq n_e} w_{jk} \mathbf{u}_k = \sum_{k \in N_j, k > n_e} w_{jk} \mathbf{u}_k \quad (16)$$

this reduces to solving two linear systems

$$AU = \bar{U} \quad \text{and} \quad AV = \bar{V} \quad (17)$$

where  $U$  and  $V$  are the interior parameters to solve and  $\bar{U}$  and  $\bar{V}$  are the boundary parameter conditions.  $n_e \times n_e$  weight coefficient matrix  $A = (a_{ij})_{i,j=1,\dots,n_e}$  has elements

$$a_{ij} = \begin{cases} 1 & \text{if } j = k \\ -w_{jk} & \text{if } k \in N_j \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

**Parameterization distortion minimization** There exist several options for assigning weight coefficients including constant (mesh geometry irrelevant) [31], and other geometry-aware barycentric coordinates such as *Wachspress*, *Discrete harmonic* and *Mean value* coordinates. These weighting options all lead to a valid parametrization but in general the mesh distortion introduced is rather high. Therefore, a mesh distortion reduction must follow the initial parametrization. The simple method we employ comes from [32] that minimizes a quantity called “stretch” which effectively measures the mesh distortion.

Consider a mesh triangle  $T^P = (\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$  in physical domain and its corresponding triangle  $T^U = (\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)$ . This defines a one-to-one mapping from triangles in parameter domain to those in physical domain, namely,  $\mathcal{S} : T^U \rightarrow T^P$ . If the maximal and minimal eigenvalues of the metric tensor induced by  $\mathcal{S}$  is denoted by  $\Gamma(T)$  and  $\gamma(T)$ , the triangle stretch in the  $k$ -th triangle of the mapping or parametrization  $\mathcal{S}$  could be characterized by

$$\sigma(T_k) = \sqrt{\frac{\Gamma^2(T_k) + \gamma^2(T_k)}{2}} \quad (19)$$

Then the stretch of each vertex is defined by

$$\sigma(\mathbf{u}_j) = \sqrt{\frac{\sum A(T_k) \sigma^2(U_k)}{\sum A(T_k)}} \quad (20)$$

where  $A(T)$  is the area of triangle  $T$  and the sums taken over all triangles  $T_k$  that have  $\mathbf{p}_j$  as a vertex. Then the weights in (18) for the  $(h+1)$ -th iteration are updated according to the vertex stretch value at the current  $h$ -th iteration by

$$w_{jk}^{h+1} = \frac{w_{jk}^h}{\sigma(\mathbf{u}_j^h)} \quad (21)$$

The stopping criterion is based on the global stretch metric defined by

$$E^h = \sqrt{\frac{\sum_{k=1}^{n_t} A_k \sigma^2(T_k^h)}{\sum_{k=1}^{n_t} A_k}} \quad (22)$$

Iteration stops if  $E^{h+1} > E^h$ , meaning there is no room to relax the stretch and distortion. Finally  $\{\mathbf{u}_j^{h+1}\}$  will output as the optimized and the mesh parametrization  $\mathcal{S}$  is established. This approach could significantly reduce the mesh distortion introduced by parametrization.

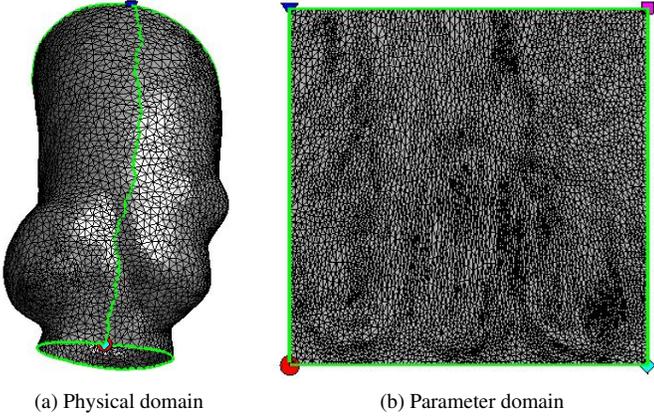


FIGURE 14: Mesh parametrization

**3.2.4 Regular sampling** With an established parametrization mapping the parameter domain to the physical domain, it's time to generate a regular grid by sampling at a

regularly spaced point in the parameter domain as shown in Figure 15(a).

Specifically for each shape,  $n_M$  sampling parameters  $\{\mathbf{m}_l\} (l = 1, \dots, n_M)$  will be regularly placed within the parameter domain  $[0, 1] \times [0, 1]$  where each sampling parameter point  $\mathbf{m}_l = (u_l, v_l)$  results in a sampled point  $\mathbf{q}_l$  on the physical mesh with the previously computed parametrization  $\mathcal{S}$  by barycentric interpolation

$$\mathbf{q}_l = \beta_{1,l} \mathbf{p}_1 + \beta_{2,l} \mathbf{p}_2 + \beta_{3,l} \mathbf{p}_3 \quad (23)$$

where the barycentric coordinates are determined by area fractions of the three sub-triangles formed by connected  $\mathbf{m}_l$  with  $\mathbf{u}_1$ ,  $\mathbf{u}_2$  and  $\mathbf{u}_3$  of the triangle in parameter domain by

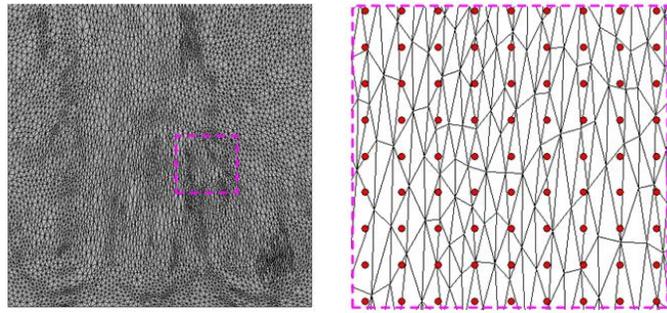
$$\begin{aligned} \beta_{1,l} &= \frac{A(\mathbf{m}_l, \mathbf{u}_2, \mathbf{u}_3)}{A(\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)} \\ \beta_{2,l} &= \frac{A(\mathbf{u}_1, \mathbf{m}_l, \mathbf{u}_3)}{A(\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)} \\ \beta_{3,l} &= \frac{A(\mathbf{u}_1, \mathbf{u}_3, \mathbf{m}_l)}{A(\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)} \\ \text{s.t. } &\beta_1 + \beta_2 + \beta_3 = 1 \end{aligned} \quad (24)$$

Here a regularly spaced grid of resolution  $51 \times 51$  are sampled as seen in the zoomed-in local view Figure 15(b). The sampling involves interpolation of barycentric coordinates and the sampled grid in the front and left view in Figure 15(c) and (d). The yellow and blue spheres are the bottom and top end vertices of the generatrix line.

**3.2.5 B-spline fitting** Base on the coordinates of the regular grid of  $50 \times 50$  data points just obtained, it's time to use the global approximation technique described in [30] and get a well-fitted B-spline surface as shown in Figure 16(a)(b) in the front and left view respectively. The B-spline resolution is set to be  $30 \times 30$  and the degree along  $u$ - and  $v$ -direction is both quadratic. The B-spline is re-visualized with knot lines shown in Figure 16(c)(d). The generatrix line can be also shown in Figure 16(d). And the B-spline training set of 6 instances are shown in Figure 17.

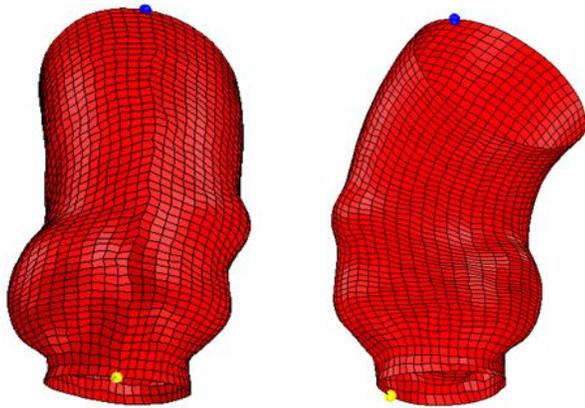
## 4 SSM results

The raw training set in triangle meshes have been processed into differentiable B-spline representation. The reparameterization function is governed by a by quadratic B-spline of coefficients resolution of  $12 \times 12$ , and reference shape is chosen to be Shape 1. The optimization is done by the Sequential Quadratic Programming (SQP) optimizer, and the convergence criterion is



(a) Regular sampling grid in parameter domain

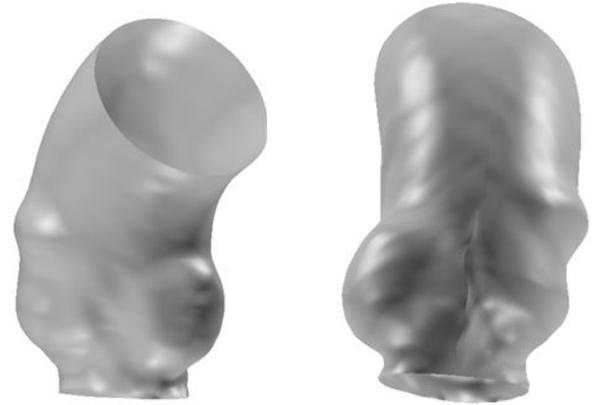
(b) Zoom-in view



(c) Sample grid in physical domain (left view)

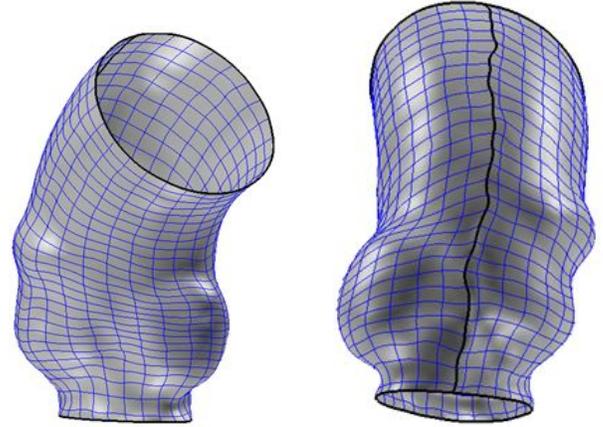
(d) Front view

**FIGURE 15:** Regular grid sampling



(a) Surface (front view)

(b) Surface (left view)



(c) With knot lines (front view)

(d) With knot lines (left view)

**FIGURE 16:** B-spline fitting

the relative change in DL objective function at the  $k$ -th iteration drops below a threshold  $\varepsilon$ , namely  $\left| \frac{DL^{(k)} - DL^{(k-1)}}{DL^{(0)}} \right| < \varepsilon$ , and here  $\varepsilon = 10^{-6}$ . The optimization took 439 iterations and 2293.9 seconds to converge, and the DL has dropped from 98.7 to 92.8.

The history of aggregated constraint for all the non-reference Shapes 2,3,4,5,6 is shown in Figure 19(b). It's seen that at the ending iteration, some of the constraints are active, and a previously violated constraint will be rectified to valid to ensure the diffeomorphism and prevent self-intersection of reparameterization function.

The deformed grid under the optimized reparameterization function for Shape 2  $\mathbf{D}_2(\mathbf{u})$  is shown in Figure 19(a) and it's clearly seen that the interior B-spline control coefficients while the boundary B-spline coefficients are fixed. Figure 19(b) displays the corresponding Jacobian field.

The first two modes are shown in Figure 20 where  $\beta$  is the parameter value along mode directions and  $\lambda$  is the variation along mode direction. The parameter values is chosen at  $\beta = -3, 0, +3$  since the  $[-3, +3]$  range would cover 97.7% of the possible shape variation along a particular mode. It's seen

that after the correspondence optimization, the first two modes together account for 78.3% of the total variation of all possible variation patterns. Mode 1 and 2 have characterized the changing of diameter of the aorta tube and also the size changing at the left coronary sinus.

Lastly, we use the three statistical model evaluation measures [33], namely generalization, specificity and compactness error to analyze the effectiveness of our proposed algorithm.

The generalization error measure the ability of the SSM to extrapolate any valid instance that is not seen in training set but belongs to the shape class. The error is calculated by the *leave-one-out* test, where each instance is left out and the remaining  $n_S - 1$  shapes are used to generate statistical model and the left-out instance is reconstructed by projecting onto the eigenmode directions, and finally the error is identified as the difference between the left-out instance and its reconstruction. The specificity error measures the ability of SSM to only represent instance that belongs to the shape class. The computation goes as follows, a large number of randomly generated instances are obtained by

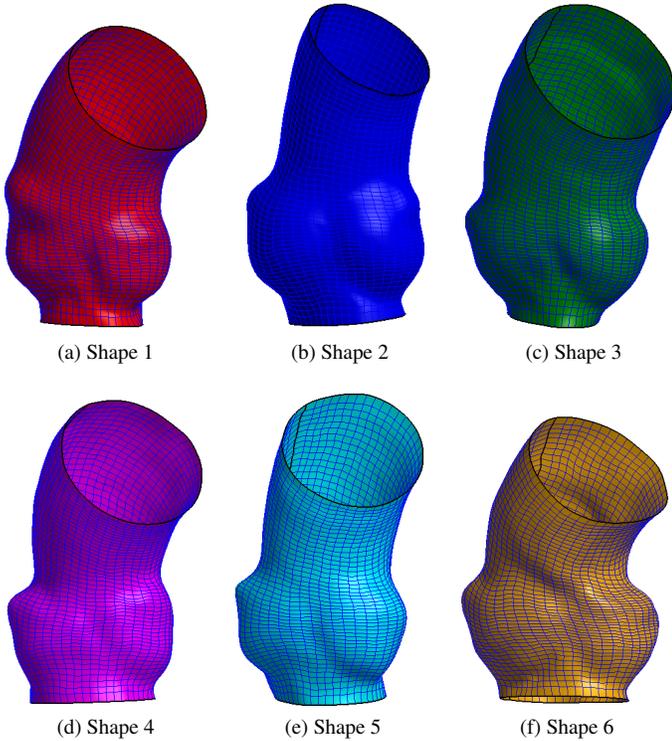


FIGURE 17: Training set B-splines

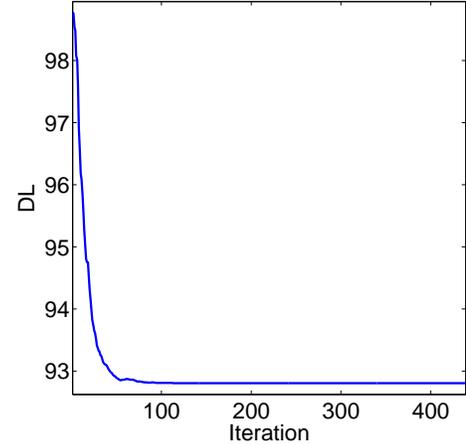
SSM and the error is identified as difference between the random instance and its closest training set instance. The compactness error is just the sum of eigenvalues of variations that measure how compact the SSM is in the shape space.

It's seen from Figure 21 that these three errors have all decreased from initial correspondence to the optimized correspondence.

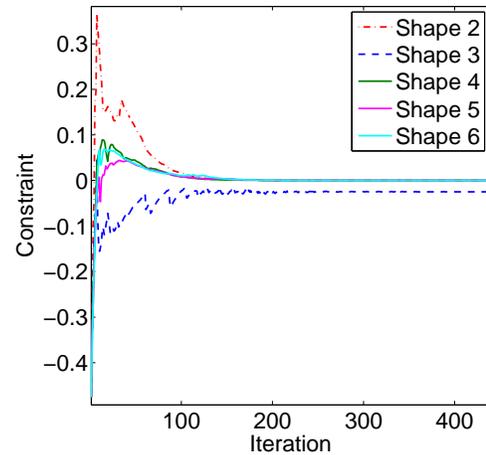
## 5 Conclusion

In this paper, we have built the Statistical Shape Model out of a training data set of six ascending aorta. The shape instances are preprocessed and fitted in B-spline to perform as the actual input training set instances. The search for a reasonably high-quality SSM is reduced to an correspondence optimization problem, and we propose a novel and efficient scheme for manipulating group-wise shape correspondence, i.e. the direct reparameterization driven by B-spline coefficients.

Although our proposed correspondence optimization algorithm currently concentrates on the shape of disk topology, the raw shapes are carefully preprocessed and the interested areas such as the coronary sinuses are kept to form a geometry with cylindrical topology. The generatrix is introduced to transform cylindrical topology shapes to disk topology shapes as valid input into optimization algorithm.



(a) Objective function DL

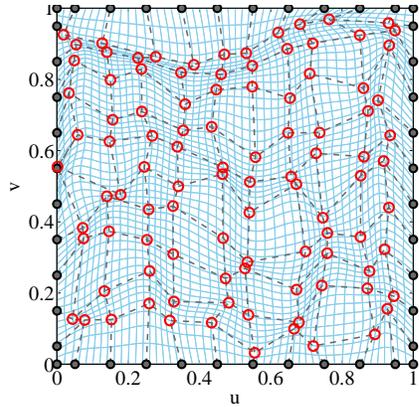


(b) Aggregated constraints

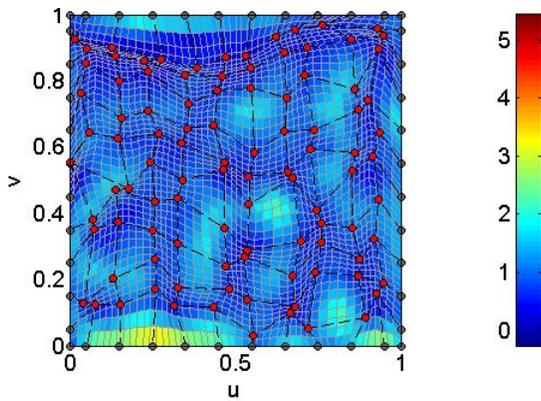
FIGURE 18: Optimization history

Reliable statistical model evaluation measures of generalization ability, specificity and compactness have been studied to show that correspondence is improved and simultaneously an optimized SSM has been achieved. The SSM then can be utilized for a good variety of downstream medical applications such patient-specific modeling and diagnosis where alternatives are either expensive or potentially hazardous.

Future work include acquiring more aorta shapes and form a larger shape population so as to gain an SSM that more reliably characterizes the underlying shape variation pattern of the aortic geometries. The resulting SSM will then be used to construct patient-specific computational models for studying aortic disease.



(a) Deformed landmark grid under the optimized  $D_2(\mathbf{u})$

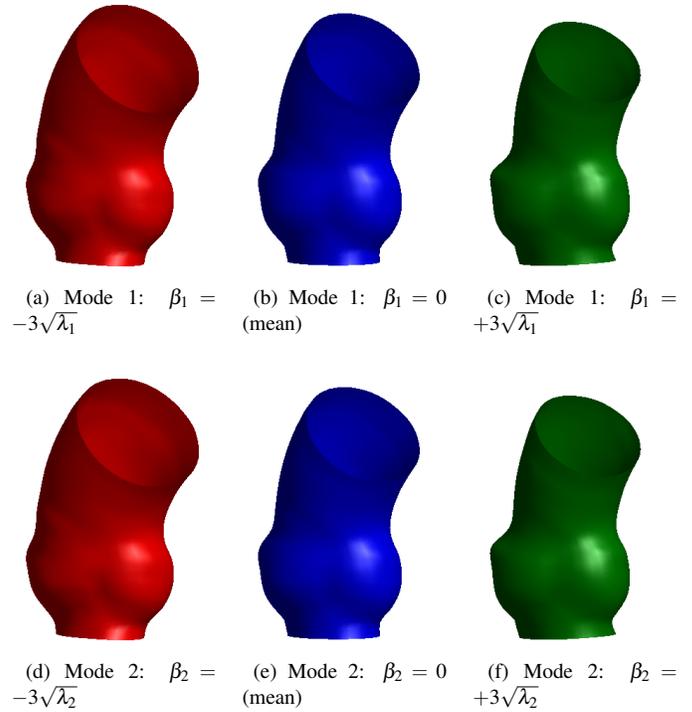


(b) Jacobian field with reparametrized parameter domain

**FIGURE 19:** Deformed grid under optimized reparametrization function  $D_2(\mathbf{u})$

## REFERENCES

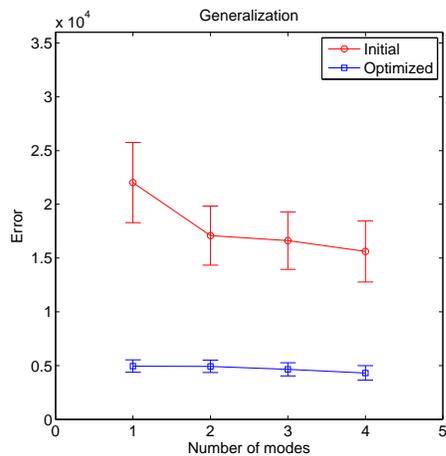
- [1] Sacks, M. S., Mirnajafi, A., Sun, W., and Schmidt, P., 2006. “Bioprosthetic heart valve heterograft biomaterials: structure, mechanical behavior and computational simulation”. *Expert review of medical devices*, **3**(6), pp. 817–834.
- [2] Thacker, B. H., Riha, D. S., Fitch, S. H., Huyse, L. J., and Pleming, J. B., 2006. “Probabilistic engineering analysis using the nessus software”. *Structural Safety*, **28**(1), pp. 83–107.
- [3] Baldwin, M. A., Laz, P. J., Stowe, J. Q., and Rullkoetter, P. J., 2009. “Efficient probabilistic representation of tibiofemoral soft tissue constraint”. *Computer Methods in Biomechanics and Biomedical Engineering*, **12**(6), pp. 651–659.
- [4] Laz, P. J., Pal, S., Fields, A., Petrella, A. J., and Rullkoetter, P. J., 2006. “Effects of knee simulator loading and alignment variability on predicted implant mechanics: a probabilistic study”. *Journal of orthopaedic research*, **24**(12),



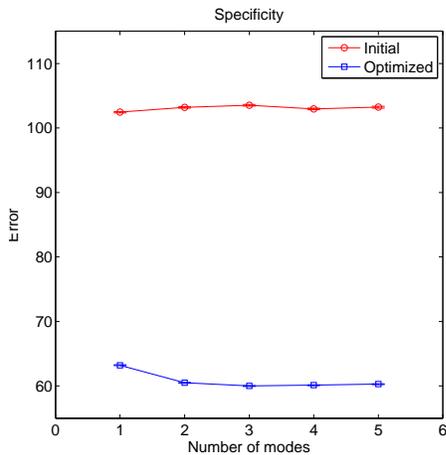
**FIGURE 20:** The first 2 modes of statistical model for distal femur data after optimization.  $\lambda_1 = 1.71(64.5\%)$ ,  $\lambda_2 = 0.37(13.8\%)$ .

pp. 2212–2221.

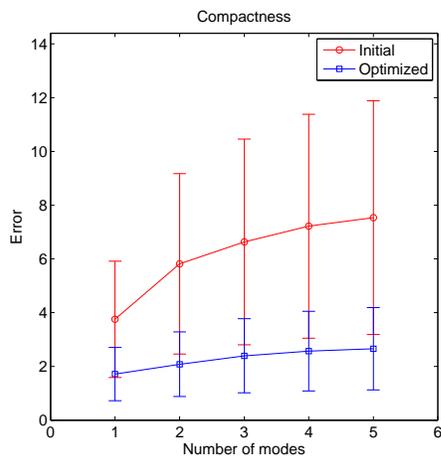
- [5] Laz, P. J., Pal, S., Halloran, J. P., Petrella, A. J., and Rullkoetter, P. J., 2006. “Probabilistic finite element prediction of knee wear simulator mechanics”. *Journal of biomechanics*, **39**(12), pp. 2303–2310.
- [6] Pal, S., Langenderfer, J. E., Stowe, J. Q., Laz, P. J., Petrella, A. J., and Rullkoetter, P. J., 2007. “Probabilistic modeling of knee muscle moment arms: effects of methods, origin–insertion, and kinematic variability”. *Annals of biomedical engineering*, **35**(9), pp. 1632–1642.
- [7] Cootes, T. F., Taylor, C. J., Cooper, D. H., Graham, J., et al., 1995. “Active shape models—their training and application”. *Computer vision and image understanding*, **61**(1), pp. 38–59.
- [8] Dryden, I., and Mardia, K., 1998. *Statistical analysis of shape*. Wiley.
- [9] Cootes, T. F., Hill, A., Taylor, C. J., and Haslam, J., 1994. “Use of active shape models for locating structures in medical images”. *Image and vision computing*, **12**(6), pp. 355–365.
- [10] Heimann, T., Meinzer, H.-P., et al., 2009. “Statistical shape models for 3d medical image segmentation: A review”. *Medical image analysis*, **13**(4), p. 543.



(a) Generalization error



(b) Specificity error



(c) Compactness error

FIGURE 21: SSM evaluation before and after optimization

- [11] Fleute, M., and Lavallée, S., 1998. “Building a complete surface model from sparse data using statistical shape models: Application to computer assisted knee surgery”. In *Medical Image Computing and Computer-Assisted Intervention MICCAI98*. Springer, pp. 879–887.
- [12] Frangi, A. F., Rueckert, D., Schnabel, J. A., and Niessen, W. J., 2002. “Automatic construction of multiple-object three-dimensional statistical shape models: Application to cardiac modeling”. *Medical Imaging, IEEE Transactions on*, **21**(9), pp. 1151–1166.
- [13] King, A., Blackall, J., Penney, G., and Hawkes, D., 2001. “Tracking liver motion using 3-d ultrasound and a surface based statistical shape model”. In *Mathematical Methods in Biomedical Image Analysis, 2001. MMBIA 2001. IEEE Workshop on*, IEEE, pp. 145–152.
- [14] Deligianni, F., Chung, A. J., and Yang, G.-Z., 2006. “Non-rigid 2-d/3-d registration for patient specific bronchoscopy simulation with statistical shape modeling: Phantom validation”. *Medical Imaging, IEEE Transactions on*, **25**(11), pp. 1462–1471.
- [15] Shi, Y., Qi, F., Xue, Z., Chen, L., Ito, K., Matsuo, H., and Shen, D., 2008. “Segmenting lung fields in serial chest radiographs using both population-based and patient-specific shape statistics”. *Medical Imaging, IEEE Transactions on*, **27**(4), pp. 481–494.
- [16] Allen, P. D., Graham, J., Farnell, D. J., Harrison, E. J., Jacobs, R., Nicopolou-Karayianni, K., Lindh, C., van der Stelt, P. F., Horner, K., and Devlin, H., 2007. “Detecting reduced bone mineral density from dental radiographs using statistical shape models”. *Information Technology in Biomedicine, IEEE Transactions on*, **11**(6), pp. 601–610.
- [17] Smyth, P. P., Taylor, C. J., and Adams, J. E., 1999. “Vertebral shape: Automatic measurement with active shape models<sup>1</sup>”. *Radiology*, **211**(2), pp. 571–578.
- [18] Devlin, H., Allen, P. D., Graham, J., Jacobs, R., Karayianni, K., Lindh, C., van der Stelt, P. F., Harrison, E., Adams, J., Pavitt, S., et al., 2007. “Automated osteoporosis risk assessment by dentists: a new pathway to diagnosis”. *Bone*, **40**(4), pp. 835–842.
- [19] Ritacco, L. E., Aponte-Tinao, L., Muscolo, D., de Quirós, F., and Nozomu, I., 2009. “Three-dimensional morphometric analysis of the distal femur: a validity method for allograft selection using a virtual bone bank.”. *Studies in health technology and informatics*, **160**(Pt 2), pp. 1287–1290.
- [20] Harris, M. D., Datar, M., Whitaker, R. T., Jurrus, E. R., Peters, C. L., and Anderson, A. E., 2013. “Statistical shape modeling of cam femoroacetabular impingement”. *Journal of Orthopaedic Research*, **31**(10), pp. 1620–1626.
- [21] Davies, R. H., Twining, C. J., Cootes, T. F., Waterton, J. C., and Taylor, C. J., 2002. “A minimum description length approach to statistical shape modeling”. *Medical Imaging, IEEE Transactions on*, **21**(5), pp. 525–537.

- [22] Grünwald, P. D., 2007. *The minimum description length principle*. The MIT Press.
- [23] Davies, R., Twining, C., and Taylor, C. J., 2008. *Statistical models of shape: Optimisation and evaluation*. Springer-Verlag London.
- [24] Hill, A., and Taylor, C. J., 1994. “Automatic landmark generation for point distribution models”. In Proc. British Machine Vision Conference, Vol. 2, pp. 429–438.
- [25] Cootes, T. F., Taylor, C. J., Cooper, D. H., and Graham, J., 1992. “Training models of shape from sets of examples”. In *BMVC92*. Springer, pp. 9–18.
- [26] Goodall, C., 1991. “Procrustes methods in the statistical analysis of shape”. *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 285–339.
- [27] Jolliffe, I., 2005. *Principal component analysis*. Wiley Online Library.
- [28] Thodberg, H. H., 2003. “Minimum description length shape and appearance models”. In *Information Processing in Medical Imaging*, Springer, pp. 51–62.
- [29] Sheffer, A., Praun, E., and Rose, K., 2006. “Mesh parameterization methods and their applications”. *Foundations and Trends® in Computer Graphics and Vision*, 2(2), pp. 105–171.
- [30] Piegl, L. A., and Tiller, W., 1997. *The NURBS book*. Springer Verlag.
- [31] Tutte, W. T., 1963. “How to draw a graph”. *Proc. London Math. Soc.*, 13(3), pp. 743–768.
- [32] Yoshizawa, S., Belyaev, A., and Seidel, H.-P., 2004. “A fast and simple stretch-minimizing mesh parameterization”. In *Shape Modeling Applications, 2004. Proceedings, IEEE*, pp. 200–208.
- [33] Davies, R. H., 2002. *Learning shape: optimal models for analysing natural variability*. University of Manchester.