A Pipeline and Metric for Validation of Personalized Human Body Models

Sukhraj Singh and Subodh Kumar

Computer Science Department, Indian Institute of Technology Delhi, New Delhi, India

Keywords: Shape Descriptors, Human Body Models, Anatomy, Mesh Validation.

Abstract: Advanced and personalized Human Body Models (HBM) are increasingly important in human centered industry design such as passive vehicular safety analysis, using finite element and other methods. Often accurate HBMs are painstakingly constructed for median human dimensions, and then modified and re-sized using personalization algorithms for various applications. Personalization algorithms rely on various anthropometric measurements, and sometimes manual intervention, to deform the median HBM. The quality of a personalized model is often defined in terms of local properties such as aspect ratio of finite elements produced. In some cases it is inferred by visual comparison with some ground truth model or by measuring the anthropometric errors with respect to known values. We seek to define the quality of deformation in anatomically suitable geometric terms, which can be automatically computed. To this end, we compare the deformed anatomical surface meshes with that of the median mesh in a shape descriptor space. Shape comparison and matching is a well studied area. The tools devised for the same are largely application dependent. We present pipeline and a metric for validating anatomical surface meshes. It is a problem that has not been extensively studied, even though general shape comparison and matching techniques abound. Our metric incorporates global and part based shape signatures. The main contribution of our work is to explore techniques suitable for comparison of anatomical meshes by non technical experts. We formulate a pipeline that needs minimal user intervention.

1 INTRODUCTION

With increasing use of Human Body Models (HBM) in human centered industry design there is ongoing effort to create more detailed advanced Finite Element based Human Body Models (FE-HBM). FE-HBMs are particularly important for injury prediction. In the last decade there has been effort to create highly detailed FE-HBMs for this purpose. Models like HU-MOS (Robin, 2001), THUMS (Iwamoto et al., 2002) and more recently GHBMC (Gayzik et al., 2011) are some of the most popular HBMs in the domain of injury biomechanics research.

Creation of a good quality FE-HBM from CT-scan data is a tedious and time consuming task, which can take many person-months. It usually requires manual intervention and quality control can be challenging. Due to these reasons FE-HBMs are generally available in a standard occupant pose for a median dimension of a given population. However, it is really important to predict the injury for subjects in non-standard pose, and of non-median shape and size. To solve this problem, deformation algorithms (Hwang

et al., 2016; Vezin and Verriest, 2005; CEESAR et al., 2014) are used to automatically resize or personalize a standard median template HBM within anatomical constraints. Thus, these algorithms deform an HBM known to be of a 'good quality' to a target HBM satisfying certain dimensional requirements. The target dimensions are in some cases derived from target X-ray or CT-Scan images or from anthropometric information (Cheng et al., 1996) derived from statistical shape analysis of a population. Deformation can be done on an individual anatomical volumetric mesh of an organ or at the skin of FE-HBM, which consequently deforms the internal anatomical meshes as well. Again, it is critical that deformation leads to a model of a 'good quality'.

Our goal is to validate, or measure the quality of, such a personalized FE-HBM. This could be done by directly defining the notion of 'good' in mathematical terms. The general way of computing the quality of a personalized FE-HBM in bio-mechanics community is by using distortion measures like scaled Jacobian, skewness or aspect ratio of the finite elements and comparing these values with the baseline

FE-HBM (Lalonde et al., 2013) and then manually evaluating simulation results. It is also done by measuring the errors in anthropometric measurements of the personalized HBM or that of palpable landmarks of internal organs with that of actual human (Poulard et al., 2012) or by visual comparison with the baseline HBM. An alternative or rather complementary way to assess quality is by comparing the geometric properties of a surface mesh extracted from an anatomical volumetric mesh with an accurate anatomically correct baseline surface mesh.

In computer graphics, 3D shapes are often represented as polygonal, or triangular, surface meshes. Comparison and matching of two shapes is fundamental in shape analysis. With the advancement in 3D capture technologies there has been a continuous expansion of shape databases. Consequently, significant advances in shape analysis have been made in the area of shape retrieval and shape matching. A large number of global shape signatures have been introduced so as to have efficient and correct shape retrieval (Iyer et al., 2005; Tangelder and Veltkamp, 2007).

The other important work in shape analysis is the analysis of the property of a mesh after it has undergone mesh processing algorithms like simplification, smoothing, watermarking, compression or deformation. These processed meshes are often compared with the original mesh using absolute measures like dihedral angles or other distortion measures. Metrics like Minkowski distance or Hausdorff distance are also used to find the difference between meshes (Aspert et al., 2002; Cignoni et al., 1998). More recently, metrics like MSDM2 by (Lavou, 2011) have been introduced which are human perception oriented distortion measures. However using such methods assumes that the two meshes are in the same coordinate space and fully aligned. There are other visualization based comparison approaches where Gaussian curvature or Mean curvature is used to color the mesh so as to facilitate visual comparison of the mesh (Zhou and Pang, 2001). This requires significant user intervention due to the visual inspection involved.

Further, many shape analysis methods depend on the kind of shape domains they are applied to, e.g., man-made shapes, natural shapes, CAD models, etc. Human anatomy is one such domain where these shape analysis methods have not been applied extensively. Although there has been research focused in the image domain for anatomical organs, 3D surface mesh analysis of human anatomical organs is not well explored.

In particular, in the anatomical context, some deformations are acceptable, even desirable, while others are not. For example, lengthening or fattening

of a Femur bone is likely due to the target dimension. On the other hand, a bulging lesser trochanter implies a poor deformation. Thus purely geometric distance between a median mesh and a personalized mesh would not suffice. Our research has lead us to the set of local and global geometric features, each already found in the literature, which together provide a usable metric to help guide the personalization algorithm.

The main contribution of our work is a 3D anatomical surface mesh comparison pipeline and a quality metric – two scalar values that provide a measure of similarity between a template mesh and a deformed candidate mesh. Again, the candidate mesh is not expected to 'match' the template mesh in the sense that shape matching algorithms depend on. Rather, it is likely to be non-linearly deformed. Our metric focuses on the fact that 'good' meshes contain all anatomical parts in the right configuration and of the right relative dimensions. In order to compute such a metric, we

- Incorporate various geometric feature based signatures for anatomical meshes validation.
- Implement a part-based mesh validation pipeline, which considers the local part based features, and relational features between adjacent parts.
- Model the mesh validation problem as a graph matching problem.
- Coin a resultant validation metric, which considers the local, relational and global geometric features.
- Test the devised pipeline and metric for 3D anatomical surface meshes.

The remainder of the paper is organized as follows: in Section 2 we present the various processing blocks of the pipeline, Section 3 provides details about the segmentation algorithm used to find parts, Section 4 provides detail about various shape signatures used and also the formulation of graph. Further, Section 5 provides details about the algorithm used for finding correspondence with an optional segment merge step detailed in Section 6 and Section 7 details metric computation. Finally, in Section 8 we discuss the results of the pipeline and Section 9 concludes with limitations and future scope of work.

2 THE PIPELINE

In our work we have mainly used 3D surface meshes of human skeletal system, primarily of the lower limb region. On the basis of shape, anatomists broadly classify the bones as a) Long bones b) Short bones c) Irregular bones and d) Flat bones (Gray, 2001).

A particular bone may further be classified into parts based on its topography such as shaft, fovea, fossa, neck, head, condyles, etc. These are essentially anatomical terms whose analogy would be parts like, cylindrical, depression, protruding, spherical, rough, etc. However, a surface mesh, by construction, may not be able to represent all such features of the bone, it depends on the resolution of the mesh and the purpose for which a mesh is constructed.

We envisage to incorporate geometric features that cater to the above classification of bony anatomy. Features based on Eigen values help us gain an insight into the overall shape of a bone, the shape diameter function (Gal et al., 2007) caters to variability in shape, and shape index values based signature (Koenderink and van Doorn, 1992) considers detailed surface characteristics of a bone. The work of (Lavou et al., 2013) also tries to find an optimal metric by linearly combining various metrics with weights predicted using machine learning techniques applied on LIRIS Masking Database. They have used different geometrical properties viz. dihedral angles, max and min curvature. Also the database used for learning did not have anatomical meshes and the resultant metric is a global visual distortion metric. The details of the signatures used in current research are explained further in Section 4.

Further, the work of (Biederman, 1987) on recognition by parts (RBC) suggests that humans tend to recognize an object based on its parts, and this has been exploited in various part based shape retrieval algorithms (Theologou et al., 2014). The work by (Jiang et al., 2013) also follows a part based matching approach, using a signature based on eigenfunctions of Leplace-Beltrami operator for CAD models. In our present work, we also take this notion into consideration and develop a metric that focuses on part based signatures and relationships between parts as well. The steps involved in metric computation are explained next.

Let us consider two polygonal surface meshes P_A and P_B, where P_A is the template mesh known to be anatomically correct and P_B is the candidate mesh to be validated. Figure 1 shows the main processing blocks of pipeline and metric computation, the same are described with some details as follows:

- Mesh Segmentation of the given two meshes, P_A and P_B.

- Attributed Relational Graph(ARG) creation from the segmented meshes.

- Computation of cost matrix between the two ARGs

- Computation of the segment to segment matching,

i.e., permutation matrix, using Hungarian node assignment algorithm.

- An Optional segment merging step, in case there are unmatched segments.

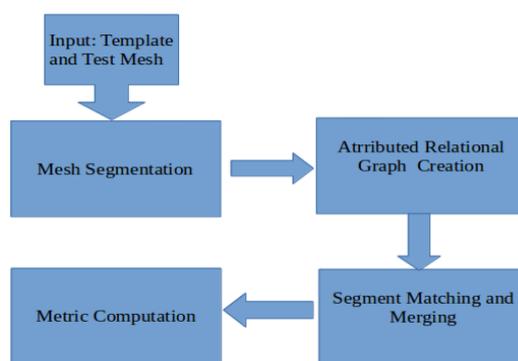- Finally, computation of global, part-based and resultant mesh validation metric.

Figure 1: Pipeline Block Diagram.

The following sections provide the detail of the above mentioned pipeline blocks.

3 MESH SEGMENTATION

In order to compute a part-based metric, the first step is to compute these parts. We have chosen standard segmentation to try to decompose a mesh into parts as most anatomically salient parts indeed are geometrically also distinct from its neighbors. Although segmentation is not our end goal and it is often sufficient for the two meshes P_A and P_B to undergo similar segmentation, we have found the Shape Diameter Function (SDF) based segmentation algorithm (Shapira et al., 2008) does well to align segments with anatomical parts. The Princeton benchmark on 3D mesh segmentation (Chen et al., 2009) ranks SDF based segmentation algorithm as third in its evaluation metric. Further, the SDF based segmentation takes about 10x less time than the top two. The algorithm is largely automatic: it does not need user intervention to decide segment count. In fact, the default parameters values mentioned by the authors generally work.

SDF is a volume based shape function and is invariant to rigid transformation and largely oblivious to pose changes. It is defined as a scalar field on the mesh surface and it gives a measure of the diameter of the object's volume in the neighborhood of each point on the surface. It involves computation of a local diameter at each vertex of a mesh. The histogram

of the SDF values is processed using Gaussian Mixture Models and Expectation Maximization algorithm to create consistent mesh partitions.

We use the implementation provided by the CGAL library package (Yaz and Loriot, 2016) with default values of 25 rays, cone angle of 120 degrees, 5 clusters and 0.26 as smoothing factor. We further build upon the facet-id map provided as output of the algorithm, to create separate polygon segment meshes. These segment meshes are later used for creating shape signatures in order to get the part-based comparison metric.

We use the following notation. S_A and S_B are the segmented polygonal meshes such that

$$S_A = \bigcup_{i=1}^{n} S_{ai}, \; S_{ai} \text{ is the } i^{th} \text{segment of mesh } P_A$$

$$S_B = \bigcup_{j=1}^{m} S_{bj}, \; S_{bj} \text{ is the } j^{th} \text{segment of mesh } P_B$$

Where, n and m are number of segments of P_A and P_B, respectively.

4 ATTRIBUTED RELATIONAL GRAPH CREATION

The segmented meshes S_A and S_B are used to create two adjacency graphs. We create a node corresponding to a segment and an edge for every pair of adjacent segments of the segmented mesh. We associate properties a.k.a. attributes to the nodes and the edges based on these segments, such a graph is known as Attributed Relational Graph (ARG). The Boost Graph Library (Siek et al., 2001) has been extensively used for the implementation of ARG. Let $G_A = (V_a, E_a)$ be the ARG for the segmented mesh S_A and $G_B = (V_b, E_b)$ be the ARG for the segmented mesh S_B.

The node attributes are the signatures based on geometric shape descriptors of a mesh segment and the edge attributes are the relational properties between two adjacent mesh segments. The following subsections provide details of the various geometric properties and thus signatures used to create these attributes. We will use the terms node and segment interchangeably henceforth.

4.1 Node Attributes

Geometrical properties of segments that are assigned to the nodes are termed as node attributes. The fol-

lowing subsection provide details of the various node attributes used in our implementation.

4.1.1 SDF Signature

We create SDF signature of a segment, which is a log normalized histogram of SDF values as described in (Shapira et al., 2008).

$$nsdf = \frac{log\left(\frac{sdf(f)-min(sdf)}{max(sdf)-min(sdf)} * \eta + 1\right)}{log(\eta + 1)}$$

Where $sdf : F \to R$ is the SDF value for each facet f and η is a normalizing parameter, which is set to 4. The SDF signature is pose-oblivious and captures the volumetric variation of a 3D shape. We use 64 bins to create the histogram which is same as the original bin count used by the authors in (Gal et al., 2007). Let us denote, V_{ai}^{sdf} and V_{bj}^{sdf}, as the SDF signature for the ith and the jth node of G_A and G_B, respectively.

4.1.2 Shape Index Signature

We create a Shape Index (SI) (Koenderink and van Doorn, 1992) signature of a segment. which is a histogram of the SI values. If K_1 and K_2 are the principal curvatures for a patch around a vertex of mesh, the SI value for that patch is defined as:

$$si = \frac{1}{2} + \frac{1}{\pi} \arctan \frac{K_2 + K_1}{K_2 - K_1} \text{ , where } K_1 \leqslant K_2$$

The principal curvatures K_1 and K_2 are determined using the second order differential properties of a local representation of the manifold fitted using a jet, a truncated Taylor expansion (Cazals and Pouget, 2005). We use the CGAL library package by the same authors (Pouget and Cazals, 2016) for estimating local differential properties of a patch around a mesh vertex via jet fitting.

The formula for determining SI values is changed slightly from the original expression so that it is positive always. For all patches except the planar patch, for which it is indeterminate, $si \in [0, 1]$. si is a scalar that provides information about the shape of a patch, e.g., whether it is a cup, cap, ridge, rut or saddle. Likewise sdf signature, we use 64 bin histogram here as well to create SI signature for a mesh segment. Let, V_{ai}^{si} and V_{bj}^{si}, be the SI signature for the ith and the jth node of G_A and G_B, respectively.

4.1.3 Area Ratio

We compute the ratio of the area of a segment to the area of the complete mesh. If Ar_S is the surface area of a mesh segment and Ar_P the surface area of the

complete mesh, Area ratio accounts for the relative size of each part.

$$\alpha = \frac{Ar_S}{Ar_P}$$

is defined as the area ratio of that segment. The same is associated with the corresponding node of ARG. Let, V_{ai}^{α} and V_{bj}^{α}, be the area ratio for the ith and the jth node of G_A and G_B, respectively.

4.1.4 Eigen Value Based Shape Descriptor

If λ_1, λ_2 and λ_3 are the three eigenvalues of a segment s.t. $\lambda_1 > \lambda_2 > \lambda_3$ then the eigenvalues based shape descriptor can be defined as in (Sidi et al., 2011):

$$e_1 = \frac{\lambda_1 - \lambda_2}{\sum_{i=1}^{3} \lambda_i} \quad e_2 = \frac{2(\lambda_2 - \lambda_3)}{\sum_{i=1}^{3} \lambda_i} \quad e_3 = \frac{3\lambda_3}{\sum_{i=1}^{3} \lambda_i}$$

We compute the eigenvalues based shape descriptor as a vector having components $e = <e_1, e_2, e_3>$ of each segment and assign it as an attribute to the corresponding node of the ARG. We use CGAL library Principal Component Analysis package (Alliez et al., 2016) to compute the eigenvalues of a mesh segment. Let, V_{ai}^{e} and V_{bj}^{e}, be the shape descriptor for i^{th} and j^{th} node of G_A and G_B, respectively.

4.1.5 Node Degree

The node degree is the number of edges connected to a node. Although this attribute is assigned to a node, in strict sense it is a relational attribute. Let, V_{ai}^{deg} and V_{bj}^{deg} be the node degree for the ith and the jth node of G_A and G_B, respectively.

4.2 Edge Attribute

We compute a tuple that represents the angles between the two largest eigenvectors (the ones with the largest eigenvalues), respectively, of a segment with its adjacent segment. If u_{1s} is the largest and u_{2s} the next largest eigenvector of a segment S_A and v_{1s} and v_{2s} are similarly the two largest eigenvectors of an adjacent segment S_B, the tuple of eigenvector angles (in radians) is defined as.

$$\beta_e = [u_{1s} - v_{1s}, u_{2s} - v_{2s}]$$

This tuple is a relational feature that describes the relative orientation of an adjacent segment to the given segment. It is computed for all the edges incident on an ARG node. Let, V_{ai}^{β} and V_{bj}^{β}, be the eigenvector angle difference for the ith and the jth node of G_A and G_B, respectively.

5 SEGMENT MATCHING

In this section we describe how to find for each segment of the candidate mesh, its corresponding segment in the template mesh. Finding corresponding segments between segmented meshes S_A and S_B is a two step process. First, a cost matrix C^{nm} is computed, where n and m are the segment counts in S_A and S_B, respectively. Each element of the cost matrix is some form of distance between the two nodes in their signature space. Second, this cost matrix is used to obtain a node assignment using Harold Kuhn's well-known Hungarian Method for solving Optimal Assignment Problems. The permutation matrix thus computed is a boolean matrix with $1's$ for the matched nodes and $0's$ for others. The following subsections provide further details.

5.1 Cost Matrix Computation

Referring to the notations proposed in previous section, the cost matrix for segmented mesh S_A and S_B can be computed as follows:

loop: For all vertices a_i of G_A
    loop: For all vertices b_j of G_B
$$c_{ij}^{sdf} = EMD(V_{ai}^{sdf}, V_{bj}^{sdf})$$
$$c_{ij}^{si} = EMD(V_{ai}^{si}, V_{bj}^{si})$$
$$c_{ij}^{\alpha} = L^1(V_{ai}^{\alpha}, V_{bj}^{\alpha})$$
$$c_{ij}^{e} = L^2(V_{ai}^{e}, V_{bj}^{e})$$
$$c_{ij}^{deg} = L^1(V_{ai}^{deg}, V_{bj}^{deg})$$
$$c_{ij}^{\beta} = L^1(V_{ai}^{\beta} - V_{bj}^{\beta})$$
$$C_{ij} = L^2(c_{ij}^{sdf}, c_{ij}^{si}, c_{ij}^{e}, c_{ij}^{\alpha}, c_{ij}^{deg}, c_{ij}^{\beta})$$
    endloop
endloop

Where, EMD is the earth movers distance (Rubner et al., 1998). We use EMD to compute distance between the signatures and absolute difference in case of scalars. Each of the distances are normalized over its range during the computation of the cost matrix Table 1 gives the description of the notations used in the pseudocode above.

5.2 Permutation Matrix Computation

We use libhungarian library (Gerkey, 2004) to solve the node assignment problem. The cost matrix C^{nm} described in section 5.1 is used for this computation. We multiply Each element of the cost matrix with integer 10, to just scale it, since the input to a libhungarian is supposed to be only integral values. The Hungarian Algorithm produces a sparse permutation

Table 1: Notations used for the various cost/distance.

Notation	Description
c_{ij}^{sdf}	EMD between SDF signatures
c_{ij}^{si}	EMD between SI signatures
c_{ij}^{α}	Absolute difference between area ratio
c_{ij}^{e}	Euclidean distance between shape descriptors
c_{ij}^{deg}	Absolute difference between degrees
L^1	L1 norm
L^2	L2 norm

$n \times m$ sized matrix $\mathcal{P}^{nm}$, where $\mathcal{P}_{ij} = 1$ if i and j are matching nodes.

6 SEGMENT MERGING

After completing the node assignment explained in Section 5, some nodes of graph G_A or G_B may remain un-assigned. This happens when the number of nodes in the two graphs are unequal. This could be due to imperfect segmentation or some defect in the candidate mesh. We consider the initial matching as tentative in this case and iterate in an attempt to improve the segmentation. In particular, we study the impact on matching if we merge two adjacent segments. An improvement in matching is strongly suggestive of imperfect segmentation and we finalize the merger. The algorithm is listed in Algorithm 1. It may happen that *dist_merge* is greater than *dist_original* in all tentative merge cases, which results in ARGs having unequal nodes even after the merge step. In that case we use distance of the unmatched segment signatures from zero valued signatures and consider that as an added penalty in the metric computation.

G_M is the resultant graph after the merge step. We reapply the segment matching algorithm listed in Section 5 between G_M and the other graph. This again results in a cost matrix and a permutation matrix. We now use these two matrices to compute the part-based metric as explained in following section.

7 METRIC COMPUTATION

The result of the segment matching step (see Sections 5 and 6) is the cost matrix $\mathcal{C}^{nm}$ and the permutation matrix $\mathcal{P}^{nm}$. The cost matrix $\mathcal{C}^{nm}$ is a collation of the distances of the segments in signature space. These

---

Algorithm 1: Segment Merging algorithm.

1: **if** num_vertices(G_A) > num_vertices(G_B) **then**
2:     MERGESEGMENT(G_A,G_B);
3: **else**
4:     MERGESEGMENT(G_B,G_A);
5: **end if**
6: **function** MERGESEGMENT (G_X,G_Y)
7:     **for all** Unmatched vertices V_u in G_X **do**
8:         **for all** Neighbours V_{nbr} of V_u **do**
9:             **if** V_{nbr} is matched **then**
10:                 Merge V_{nbr} and V_u to get V_{merge}
11:                 Compute unary attributes of V_{merge}
12:                 Find distance dist_merge between V_{merge} and corresponding vertex of V_{nbr} in G_Y
13:                 **if** dist_merge < dist_original **then**
14:                     Keep the merge
15:                 **else**
16:                     Discard the merge
17:                 **end if**
18:             **end if**
19:         **end for**
20:     **end for**
21: **end function**
    ▷ Return the resultant graph after merging
return G_M

---

distances are EMD, Euclidean, Manhattan or Absolute difference. An element $\mathcal{C}_{ij}$ of matrix $\mathcal{C}^{nm}$ is the distance between segments i and j, respectively, of the two meshes. As explained in Section 4, these node attributes account for the segment shape and the edge attributes account for the relation between segments. The resultant metric is thus able to capture both the part based shape and inter part relational aspect of the meshes under comparison. Section 7.1 describes the computation of complete part based metric. Section 7.2 describes the computation of the global metric.

7.1 Part-Based Metric

We use the computed cost matrix $\mathcal{C}^{nm}$ (refer Section 5.1) and permutation matrix $\mathcal{P}^{nm}$ (refer Section 5.2) to compute the part based metric. We iterate through $\mathcal{P}^{nm}$ and check for matched nodes, and the corresponding cost is picked from $\mathcal{C}^{nm}$. The L^2 norm of all these costs is the computed part based metric. That is:

$$M_p = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{m} \mathcal{C}_{ij}^2 \mathcal{P}_{ij}}$$

7.2 Global Metric

In order to compute the global metric, we first compute the signature corresponding to the complete mesh. We compute the SDF signature for the meshes P_A and P_B and compute EMD between the two signatures. If H_A and H_B are the histograms of SDF values of the meshes P_A and P_B,

$$m_{sdf} = EMD(H_A, H_B)$$

We also compute the eigenvalues based shape descriptor of the meshes as described in Section 4.1.4 for mesh segments. If e_A and e_B are shape descriptors of meshes P_A and P_B then,

$$m_e = \sqrt{e_A^2 - e_B^2}$$

Lastly, we compute the SI values based global shape signature of the meshes P_A and P_B (see Section 4.1.2). Thus if SI_A and SI_B are the histograms of SI values of the meshes P_A and P_B,

$$m_{si} = EMD(SI_A, SI_B)$$

We tested with both Euclidean distance and EMD for histograms, and observed that EMD gives more reliable results in practice than does the Euclidean distance. This is due to the fact that EMD imparts a more global sense to the distance between the two histograms, whereas the Euclidean distance can be unduly influenced by individual bins. For eigenvalues based shape descriptor, which is not a histogram, we use the Euclidean distance.

The resulting global metric is defined as the L^2 norm of m_{sdf}, m_e and m_{si}:

$$M_g = \sqrt{m_{sdf}^2 + m_e^2 + m_{si}^2}$$

7.3 Overall Metric

The final metric is reported as the pair of the part based metric and the global metric, i.e.,

$$M_r = (M_p, M_g)$$

Since both M_p and M_g are based on metrics like EMD, Euclidean or Manhattan distances, the resultant metric does follow non-negativity and symmetry properties, but may not follow triangular inequality.

8 EXPERIMENTAL RESULTS

In section 8.1 we show an example validation for femur mesh having 16516 vertices and 33028 facets,

where the template femur mesh has been deformed at the Lesser Trochanter region as shown in Figure 2. We shall discuss two examples: Example 1, where the number of segments in the Template and the Test meshes are the same, and Example 2, where the number of segments of the Template and Test meshes are different. In the latter case, the mesh that has more segments will undergo automatic merging based on the algorithm as described in Section 6 The experiment was carried out on a system having Processor: Intel i7 CPU@2.40GHz and RAM: 8 GiB.

8.1 Example 1

Figure 5 shows the segmentation results of the template mesh and the deformed test mesh at the Lesser Trochanter region. It can be observed that for the given parameters of the segmentation algorithm, it is able to segment the femur into anatomically relevant regions like a) Head b) Shaft c) Condylar region and d) Greater trochanter. The results of the segmentation can change with the supplied parameters. In the current example the test mesh has the same number of segments as does the template mesh. The ARGs are constructed from the segmented meshes, Figure 3 and Figure 4 show the obtained cost and permutation matrices, respectively. Matching colors in Figure 5 represent the matched segments obtained in this step.

The cost and permutation matrices are used to compute the part-based metric and the global metric. The resultant metric is the pair of part-based and global metric as explained in section 7.

$$M_p = 13.5712 \tag{1}$$

$$M_g = 0.1307 \tag{2}$$

$$M_r = (13.5712, 0.1307) \tag{3}$$

where, M_p, M_g and M_r are the part-based, global and resultant metrics, respectively. where, M_p, M_g and M_r are the part-based, global and resultant metrics, respectively. The complete process took approximately 43 seconds.

8.2 Example 2

Figure 10 shows the segmentation results of the the template mesh and test mesh deformed at the same location, but by a larger amount, as that in the mesh in Example 1. The segmentation algorithm, for the same parameters produces seven segments in this case. Thus the test mesh undergoes a merger step. Figure 6 and Figure 7 show the obtained cost matrix and permutation matrix respectively. The matched segments in Figure 10 are shown in the same color. Notice that there are two unmatched segments, colored
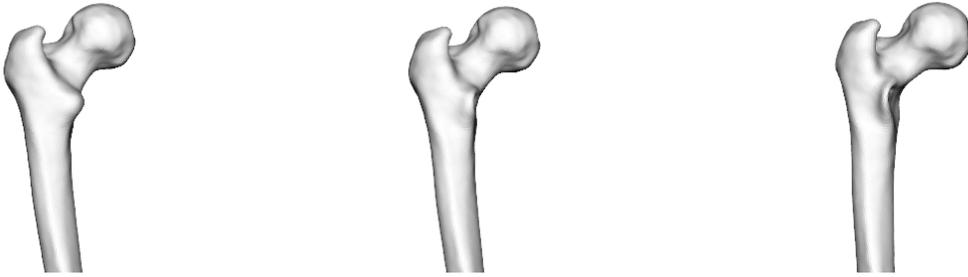
Figure 2: Femur Deformed at Lesser Trochanter: Template Mesh - Test Mesh 1 - Test Mesh 2.

$$\begin{bmatrix} 10 & 39 & 23 & 25 & 41 \\ 40 & 2 & 29 & 27 & 25 \\ 25 & 32 & 6 & 20 & 34 \\ 33 & 29 & 18 & 1 & 26 \\ 44 & 26 & 27 & 22 & 5 \end{bmatrix}$$

Figure 3: 5×5 cost matrix femur example 1.

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 4: 5×5 permutation matrix computed femur example 1.

pink and white. The complete process took approximately 42 seconds.

Figure 8 and Figure 9 show the obtained cost matrix and permutation matrix after merging. Figure 11 shows the matched segments finally obtained in this step. Notice that the two unmatched segments (from Figure 10) are merged with their neighbouring segments in this step.

The cost and permutation matrices are used to compute the part-based metric and the global metric. These are then joined to produce the resultant metric as explained in section 7.

$$M_p = 19.2116 \tag{4}$$

$$M_g = 0.1347 \tag{5}$$

$$M_r = (19.2116, 0.1347) \tag{6}$$

Clearly, in this case output global as well as part based metric has greater value as compared to the example 1 results, thus indicating a greater amount of deformation. The merge step with computation of metric took approximately 27 seconds.
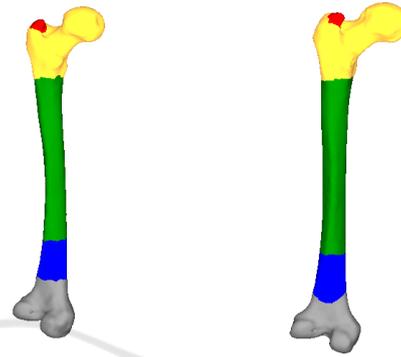

Figure 5: Corresponding segments of Template and Test Mesh 1.

$$\begin{bmatrix} 24 & 7 & 39 & 31 & 21 & 29 & 33 \\ 33 & 33 & 5 & 34 & 29 & 23 & 21 \\ 24 & 21 & 30 & 27 & 7 & 22 & 23 \\ 29 & 24 & 24 & 33 & 19 & 5 & 17 \\ 37 & 36 & 21 & 32 & 32 & 19 & 12 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 6: 7×7 cost matrix Femur case 2.

8.3 Survey Results

We conducted a survey to check the efficacy of the pipeline and the metric. The survey comprised images of five different sets of meshes. Each set had four meshes: one original template mesh and three deformed versions of it. Side-on snapshots of each mesh from four mutually perpendicular viewing an-

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Figure 7: 7×7 permutation matrix Femur example 2.

$$\begin{bmatrix} 7 & 45 & 20 & 27 & 36 \\ 37 & 11 & 31 & 24 & 20 \\ 24 & 40 & 8 & 21 & 27 \\ 28 & 32 & 20 & 5 & 21 \\ 40 & 28 & 32 & 20 & 8 \end{bmatrix}$$

Figure 8: 5×5 cost matrix femur example 2 after merge.

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 9: 5×5 permutation matrix femur example 2 after merge.

gles were shown for evaluation. The three deformed meshes in each set all had deformations at the same location but with varying magnitudes. One of the sets include deformed mesh output from a GEBOD based (CEESAR et al., 2014; Cheng et al., 1996) FE-Mesh personalization algorithm. A group of ten experts, which comprised bio-mechanic researchers and medical practitioners, were asked to compare each deformed mesh (on the basis of the four snapshots) to the template mesh and assign a rank from 1 to 3 indicating the degree of deformation in their judgement. Figure 12 shows the results of the ranking marked by the surveyors. There was significant agreement among them.

We also computed the part-based and global metrics of the same set of deformed test meshes (with respect to their respective templates). In Figure 13 and Figure 14, we show the computed global and part-based metrics, respectively.

We evaluate the metric against the survey. From the survey, the modal rank of each deformed mesh was taken as its anatomically standard rank. The rankings from the metric were obtained directly: a higher value of the metric implied a lower rank. Figure 15 shows the results of the comparison. Further, the surveyors were also asked to judge the anatomical correctness of each test mesh by comparing it to the corresponding template meshes. Specifically, the question was "Are the two bones anatomically the same?", to which they had to answer as a) Strongly agree b) Agree c) Disagree or d) Strongly disagree. We computed a cumulative-score for a particular answer as $\sum_{i=1}^{4} weight_i * mode_i$, where weights were assigned from $1 - 4$ with 1 for 'strongly agree' and so on. It was observed that the correlation between cumulative-score and part-based metric is significant with correlation coefficient = 0.724. The correlation coefficient between rankings and global-metric is 0.900 and between rankings and part-based metric
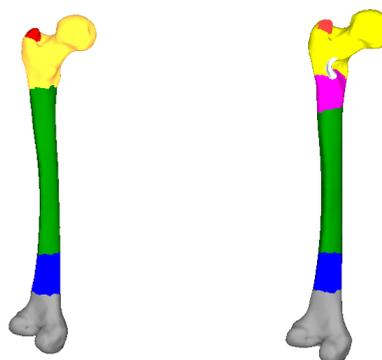

Figure 10: Corresponding segments of Template and Test Mesh 2 with two unmatched segments in color pink and white.
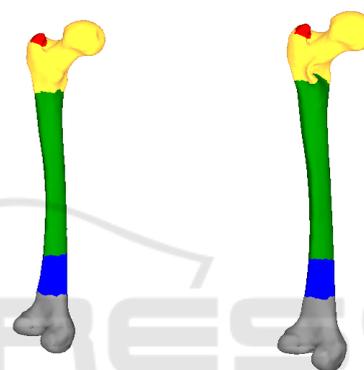

Figure 11: Corresponding segments of Template and Test Mesh 2 – Merged.

is 0.500.

Out of the five sets, the part-based metric disagrees with the standard ranking of the first mesh and the fifth mesh. For the first mesh, which is quite symmetric with similar shapes for opposite ends of meshes under comparison, matching failed, in part due to the segmentation. The sphere-like lateral condylar region was segmented for the template mesh but it matched the femoral head segment of the test mesh. These had similar geometric and relational features, and ended up in an inverse segment match. One way to improve this is to use different parameters for the segmentation algorithm and break symmetry. Since we wanted the same parameter for all meshes, we did not explore this further.

The fifth set had a torsion deformation at the femur condylar region. It was observed that the second mesh in this set had an equal number of segments to that of the template mesh. But the first mesh had an unequal number of segments. Even though the segment merging algorithm accounts for this discrepancy, this case results in significant variations in
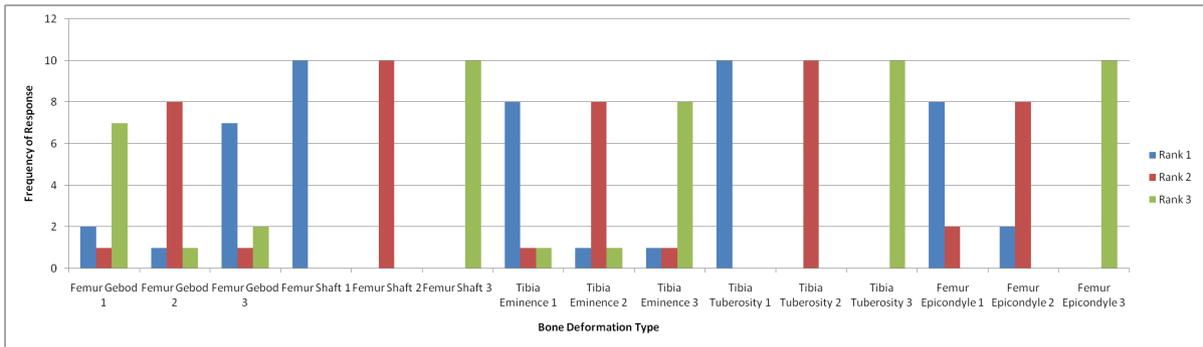
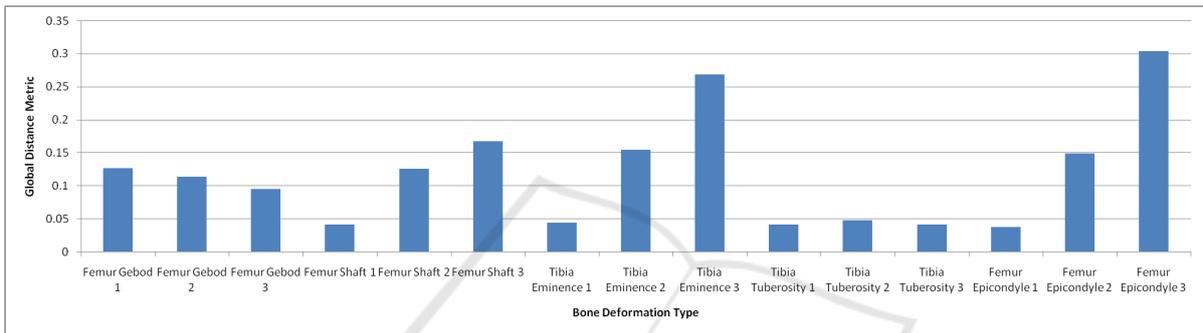Figure 12: Ranking of Bone Deformation based on a Survey.

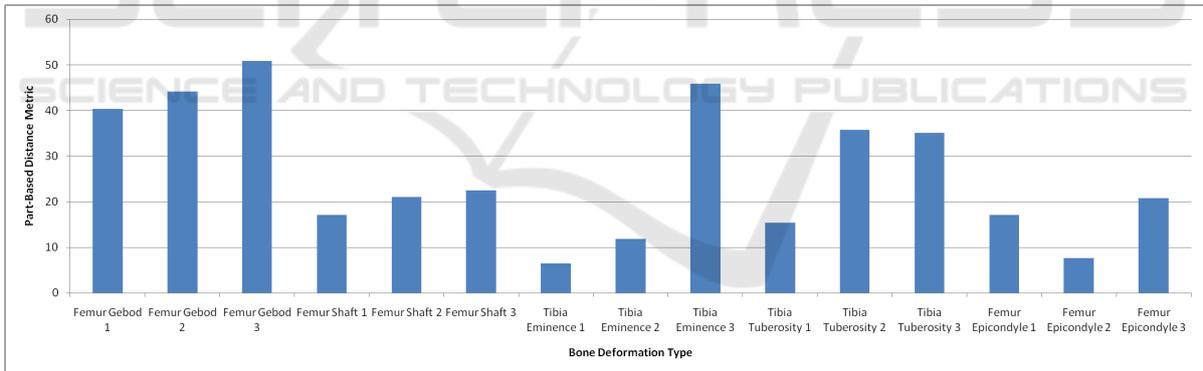Figure 13: Deformed Mesh Global Metric.

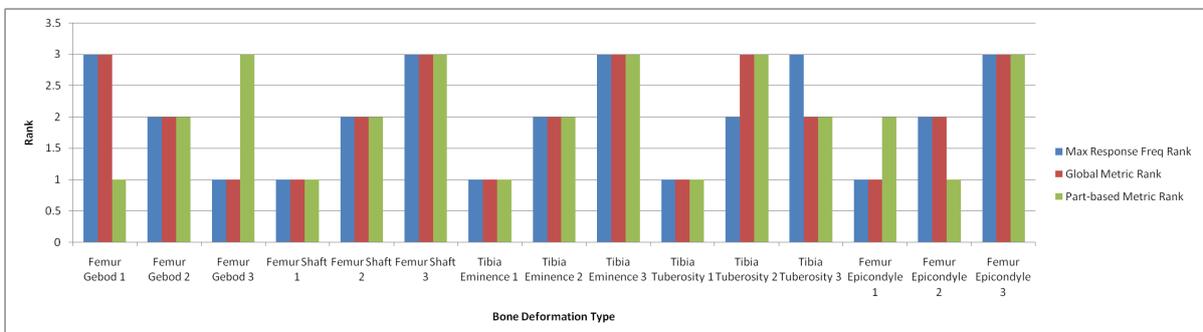Figure 14: Deformed Mesh Part-based Metric.

Figure 15: Comparison of Rankings based on a Survey, Global Metric and Part-Based Metric.

the segment boundaries, leading to deviations in the computed metric. Overall, three cases out of five had rankings identical to the survey's for the part-based metric and the other two cases had partly similar ranking, but in these case, the global metric had ranking almost identical to the survey's.

In particular, one might conclude that if both part-based and global metrics indicate that a test-mesh has a high rank, it is indeed of good anatomical quality.

9 CONCLUSION AND FUTURE WORK

In this paper, we present a pipeline for the validation of personalized/deformed anatomical meshes by comparing these with known correct meshes. The pipeline considers part-based, relational and global geometrical features and gives a resultant validation metric. The complete pipeline requires little user intervention and computes the validation metric in an automatic way. The efficacy of the validation metric depends on three important factors, a) segmentation algorithm b) the choice of geometric features or signatures and c) the distance metric. We observed that SDF based segmentation algorithm gives satisfactory mesh partitions in case of anatomical organs. However, the segmentation output varies with the change in parameters provided to the algorithm. Further investigation is needed to determine if a better segmentation into actual anatomical parts can improve the match metric further. In further work, one may provide some control to segment the template as a one time effort. The candidate mesh can then be automatically co-segmented with the template. In future, one can test for additional signatures, anatomical meshes of upper body region and improve on limitations of the metric.

ACKNOWLEDGEMENTS

REFERENCES

Alliez, P., Pion, S., and Gupta, A. (2016). Principal component analysis. In *CGAL User and Reference Manual*. CGAL Editorial Board, 4.9 edition.

Aspert, N., Santa-Cruz, D., and Ebrahimi, T. (2002). Mesh: measuring errors between surfaces using the hausdorff distance. In *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on*, volume 1, pages 705–708 vol.1.

Biederman, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94:115–147.

Cazals, F. and Pouget, M. (2005). Estimating differential quantities using polynomial fitting of osculating jets. *Computer Aided Geometric Design*, 22(2):121 – 146.

CEESAR, INRIA, U. et al. (2014). Position and Personalize Advanced Human Body Models for Injury Prediction. http://www.piper-project.eu/. [Online; accessed 18-October-2016].

Chen, X., Golovinskiy, A., and Funkhouser, T. (2009). A benchmark for 3d mesh segmentation. In *ACM SIGGRAPH 2009 Papers*, SIGGRAPH '09, pages 73:1–73:12, New York, NY, USA. ACM.

Cheng, H., Obergefell, L., and Rizer, A. (1996). The development of the gebod program. In *Biomedical Engineering Conference, 1996., Proceedings of the 1996 Fifteenth Southern*, pages 251–254.

Cignoni, P., Rocchini, C., and Scopigno, R. (1998). Metro: Measuring error on simplified surfaces. *Computer Graphics Forum*, 17(2):167–174.

Gal, R., Shamir, A., and Cohen-Or, D. (2007). Pose-oblivious shape signature. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):261–271.

Gayzik, F. S., Moreno, D. P., Vavalle, N. A., Rhyne, A. C., and Stitzel, J. D. (2011). Development of the global human body models consortium mid-sized male full body model. *Injury Biomechanics Research*, pages 39–12.

Gerkey, B. P. (2004). C Implementation of the Hungarian Method. http://robotics.stanford.edu/gerkey/tools/hungarian.html. [Online; accessed 14-July-2016].

Gray, H. (2001). Anatomy of the human body. Philadelphia: Lea & Febiger, 1918; Bartleby.com, 2000. http://www.bartleby.com/107/17.html. [Online; accessed 18-October-2016].

Hwang, E., Hallman, J., Klein, K., Rupp, J., Reed, M., and Hu, J. (2016). Rapid development of diverse human body models for crash simulations through mesh morphing. Technical report, SAE Technical Paper.

Iwamoto, M., Kisanuki, Y., Watanabe, I., Furusu, K., Miki, K., and Hasegawa, J. (2002). Development of a finite element model of the total human model for safety (thums) and application to injury reconstruction. In *Proceedings of the International Research Council on the Biomechanics of Injury conference*, volume 30, pages 12–p. International Research Council on Biomechanics of Injury.

Iyer, N., Jayanti, S., Lou, K., Kalyanaraman, Y., and Ramani, K. (2005). Three-dimensional shape searching:

state-of-the-art review and future trends. *Computer-Aided Design*, 37(5):509 – 530. Geometric Modeling and Processing 2004.

Jiang, L., Zhang, X., and Zhang, G. (2013). Partial shape matching of 3d models based on the laplace-beltrami operator eigenfunction. *Journal of Multimedia*, 8(6):655–661.

Koenderink, J. J. and van Doorn, A. J. (1992). Surface shape and curvature scales. *Image and Vision Computing*, 10(8):557 – 564.

Lalonde, N. M., Petit, Y., Aubin, C.-E., Wagnac, E., and Arnoux, P.-J. (2013). Method to geometrically personalize a detailed finite-element model of the spine. *IEEE Transactions on Biomedical Engineering*, 60(7):2014–2021.

Lavou, G. (2011). A multiscale metric for 3d mesh visual quality assessment. *Computer Graphics Forum*, 30(5):1427–1437.

Lavou, G., Cheng, I., and Basu, A. (2013). Perceptual quality metrics for 3d meshes: Towards an optimal multi-attribute computational model. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pages 3271–3276.

Pouget, M. and Cazals, F. (2016). Estimation of local differential properties of point-sampled surfaces. In *CGAL User and Reference Manual*. CGAL Editorial Board, 4.9 edition.

Poulard, D., Bermond, F., Dumas, R., Bruyere, K., and Compigne, S. (2012). Geometrical personalisation of human fe model using palpable markers on volunteers. *Computer methods in biomechanics and biomedical engineering*, 15(sup1):298–300.

Robin, S. (2001). Humos: human model for safety–a joint effort towards the development of refined human-like car occupant models. In *17th international technical conference on the enhanced safety vehicle*, page 297.

Rubner, Y., Tomasi, C., and Guibas, L. J. (1998). A metric for distributions with applications to image databases. In *Computer Vision, 1998. Sixth International Conference on*, pages 59–66.

Shapira, L., Shamir, A., and Cohen-Or, D. (2008). Consistent mesh partitioning and skeletonisation using the shape diameter function. *The Visual Computer*, 24(4):249–259.

Sidi, O., van Kaick, O., Kleiman, Y., Zhang, H., and Cohen-Or, D. (2011). Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. *ACM Trans. on Graphics (Proc. SIGGRAPH Asia)*, 30(6):126:1–126:10.

Siek, J. G., Lee, L.-Q., and Lumsdaine, A. (2001). *Boost Graph Library: User Guide and Reference Manual, The*. Pearson Education.

Tangelder, J. W. H. and Veltkamp, R. C. (2007). A survey of content based 3d shape retrieval methods. *Multimedia Tools and Applications*, 39(3):441.

Theologou, P., Pratikakis, I., and Theoharis, T. (2014). A review on 3d object retrieval methodologies using a part-based representation. *Computer-Aided Design and Applications*, 11(6):670–684.

Vezin, P. and Verriest, J. P. (2005). Development of a set of numerical human models for safety. In *19th International Technical Conference on the Enhanced Safety of Vehicles, Washington DC*, pages 6–9.

Yaz, I. O. and Loriot, S. (2016). Triangulated surface mesh segmentation. In *CGAL User and Reference Manual*. CGAL Editorial Board, 4.9 edition.

Zhou, L. and Pang, A. (2001). Metrics and visualization tools for surface mesh comparison. In *Visual Data Exploration and Analysis VIII, 99 (May 3, 2001)*, volume 4302, pages 99–110.