

The following scientific article was officially published in the journal *International Journal for Numerical Methods in Biomedical Engineering*, published by Wiley. This article's citation is as follows:

Courchesne, Olivier, Francois Guibault, Stefan Parent, and Farida Cheriet. "Patient-specific anisotropic model of human trunk based on MR data." *International Journal for Numerical Methods in Biomedical Engineering*, Vol. 31, no. 9 (2015).

doi: [10.1002/cnm.2724](https://doi.org/10.1002/cnm.2724)

The manuscript, as accepted by the publisher, is reproduced in the following pages.

© 2014 John Wiley & Sons, Ltd. All Rights Reserved.

Patient-specific anisotropic model of human trunk based on MR data

O. Courchesne*, F. Guibault, Stefan Parent and Farida Cheriet

O. Courchesne, Institute of Biomedical Engineering, École Polytechnique de Montréal, o.courchesne@polymtl.ca

SUMMARY

There are many ways to generate geometrical models for numerical simulation and most of them start with a segmentation step to extract the boundaries of the regions of interest. This paper presents an algorithm to generate a patient specific 3D geometric model, based on a tetrahedral mesh, without an initial extraction of contours from the volumetric data. Using the information directly available in the data, such as grey levels, a metric is built to drive a mesh adaptation process. The metric is used to specify the size and orientation of the tetrahedral elements everywhere in the mesh. Our method, which produces anisotropic meshes, gives good results with synthetic and real MRI data. The resulting model quality has been evaluated qualitatively and quantitatively by comparing it with an analytical solution and with a segmentation made by an expert. Results show that our method gives, in 90% of the cases, as good or better meshes as a similar isotropic method, based on the accuracy of the volume reconstruction for a given mesh size. Moreover, a comparison of Hausdorff distances between adapted meshes of both methods and ground-truth volumes shows that our method decreases reconstruction errors faster. Copyright © 2014 John Wiley & Sons, Ltd.

Received . . .

KEY WORDS: MRI; Adaptive mesh refinement; 3D geometric model; patient-specific.

1. INTRODUCTION

Numerical simulations are becoming an important part of research in the biomedical field. Many kinds of simulations can be performed to understand better the impact of a surgery or a medical treatment, in fields such as neuro- [1], hepatic [2] or orthopedic surgery [3]. In particular, some simulators focus on the planning of spine deformity treatments and on understanding the deformity progression over time using bone-based models [5][6]. The goal is to correct spinal deformation and to stop its progression, but there is also an aesthetic aspect of the postoperative result that has an important psychological impact [7]. Therefore the treatment must also focus on improving the external trunk of the patient, such as reducing the back hump. To perform accurate simulations, it is necessary to construct a model that integrates information not only from bones, but also muscles and soft tissue of the internal trunk. Dionne *et al* artificially generated a regular mesh between the external surface of the trunk and the surface of the underlying bone structures. This model was used to predict the impact of a treatment on the external surface of the trunk [8], but this model is not patient-specific and required a lot of manual tuning. A good way to acquire information on the soft tissues of the trunk, without danger for the patient, is to use MR data, and Harmouche *et al* described a way to register them with a 3D model of the spine [9].

There are many techniques to build 3D models from medical datasets using either structured or unstructured meshes. Structured meshes can be used to produce high quality models but it

*Correspondence to: O. Courchesne, olivier.courchesne@polymtl.ca

has been shown that unstructured meshes can be used to perform numerical simulation with an equivalent convergence threshold and accuracy, while using fewer elements [10][11]. Unstructured meshing techniques can be divided into two groups: those that use segmented data and those that use unsegmented data.

In the first category, Sullivan *et al* have developed 3D models of different anatomic structures based on segmented surfaces from the Visible Human Project [12]. Zhang *et al* presented a method to generate adapted tetrahedral and hexahedral meshes for the volume between two isosurfaces extracted from CT-scan and MRI datasets [13]. More recently, they used an octree-based tetrahedral meshing method combined with surface meshing for high-fidelity fluid-structure interaction simulations [14]. Si and Gärtner proposed a Delaunay tetrahedralisation that can be constrained to respect the closed geometry of any object or structure. They also tuned many methods to ensure the high quality level of the resulting mesh [15]. Archip *et al* used a sequence of segmented images to get a cloud of points in 3D. Then, they computed the convex envelope of the cloud using a Delaunay tetrahedralisation which was refined to obtain a smooth result. This method was used successfully to rebuild bones of the skull and pelvis [16]. Salo *et al* performed mesh morphing on tetrahedral meshes to obtain patient-specific finite element models. Those meshes were obtained using surfaces manually segmented from pelvic CT scans [17]. Feng *et al* performed surface extraction from Cryo-EM volumetric data using a marching cube strategy. After smoothing the surface based on the curvature, they performed a Delaunay tetrahedralisation to fill the volume [18]. Marchandise *et al* developed an automatic anisotropic meshing method for tubular segmented geometries. Resulting meshes rely on a metric and a size map [19]. Although anisotropic smoothing approaches were proposed to improve surface quality before generating the 3D model [20][21], automatic segmentation of MRI is still a challenging task. Indeed, MR data typically does not present high contrast or clear edges due to high gray level variability, blurry areas caused by blood flow and partial volume effects [22].

Few techniques belong to the second category. Adams *et al* used anisotropic meshes to reduce a dataset while keeping valuable information [23][24]. Brankov *et al* used a content-adaptive mesh model for non-uniform sampling of medical datasets to reduce the amount of data to process [25]. Even fewer methods can generate 3D models using complete datasets. Hung *et al* developed a new technique to simulate deformations of a complete dataset without any segmentation. They used an intermediary anisotropic mesh, called Virtual Proxy MESH, to deform the whole dataset. That mesh was produced using curvature information within the dataset [26][27]. Goksel and Salcudean developed a very interesting mesh adaptation technique for 2D and 3D datasets called *Variational Image Meshing* (VIM) [28]. This technique is based on a cost function that uses intensity variation and element shape quality measures to place mesh vertices on the boundaries of visible regions while retaining a highly isotropic mesh. It does not either refine the mesh or use geometrical information about structures inside the dataset. It only moves mesh vertices to reduce intensity variability within each element while enforcing regular element shapes.

The method presented in this paper shares similar goals with this last technique such as using image datasets directly without prior extraction of contours of the regions of interest to avoid loss of data. The segmentation step is postponed after the adaptation. However, our method uses anisotropic meshes instead of isotropic. While isotropic mesh focus on regular elements, anisotropic mesh elements shape, size and orientation are defined by a tensor control function. This control function can be computed by extracting information from an image intensity field, such as the second order partial derivatives. Adapted elements will be stretched to follow interesting features within the data. Using an image-based function also allows the implementation of an iterative process. Anisotropic meshes built this way produce interpolation results that are equivalent to those obtained on isotropic meshes while using fewer elements [29].

The modeling process that will be used in this paper iteratively modifies a mesh. It uses an anisotropic mesh adaptation tool called *Object Oriented Remeshing Toolkit* (OORT) [30]. The capacity of this tool to adapt a mesh to satisfy a tensor control function has been demonstrated in [31]. The mesh adaptation process has been recalibrated to take a discrete intensity field extracted from medical images as an input. Even if anisotropic meshes are frequently used in computational

fluid dynamics to capture physical phenomena near edges or boundaries [32], they are rarely used in biomedical applications. However, this approach offers important advantages; in particular, it can significantly reduce the size of mesh models without compromising the results of numerical simulations.

The goal of this paper is to present a way to build a complete human trunk representation including soft tissues to simulate orthopaedic treatments such as a brace or surgery or even to simulate deformities progression. This representation will be a patient specific high quality 3D geometric model, based on tetrahedral elements and using MR data. A preliminary study of the proposed method, performed only on 2D images, has been presented in [33][34]. With the goal of constructing meshes targeted toward the simulation of physical deformations of the trunk, we propose to use 3D MR acquisition sequences for the reconstruction of the discrete 3D trunk model. The rest of this paper is structured as follows: Section II describes the methodology of our proposed approach; Section III presents and analyzes our experimental results. Finally, Section IV summarizes this study and addresses future avenues of research.

2. METHODOLOGY

This section presents the steps to produce valid adapted meshes based on MR data. As these meshes will be used for numerical simulation, the adaptation method is based on finite element principles. The method distributes the interpolation error across all the mesh elements. Céas lemma [35] states that the approximation error is bounded by the interpolation error, which can be defined as:

$$\|F - F_h\| \quad (1)$$

where F is the continuous function and F_h is its discrete representation. With MR data, the function represents the grey level intensity profile. To minimize the global interpolation error, each element must have the same interpolation error and according to Labbé *et al* [36] this leads to an optimal mesh. Using linear interpolation functions, the interpolation error is driven by the first neglected terms in the Taylor expansion, which are the second order partial derivatives (Hessian matrix) multiplied by the square value of the size of each element. This means that minimizing the interpolation error leads to reduced element size where the second order partial derivatives are high and increased element size elsewhere.

The proposed adaptation method, which minimizes the global interpolation error, consists of the following steps. First, in an initialization step, a regular mesh is build and, based on the MR data, the Hessian matrix is computed at each mesh vertex. From the Hessian matrix, a tensor control function (called metric tensor) is deduced. Second, in an adaptation step, the initial mesh is modified to respect the specific metric. Third, the metric is interpolated on the resulting mesh; the second and third steps can be repeated. These steps will be described below based on 2D data in order to illustrate them more easily.

2.1. Initialization Step

First, a dual mesh is constructed based on the MR image intensities. This initial mesh is constructed as a regular grid of triangles with one vertex placed at the center of each pixel of the image. A scalar function F_h is defined over this mesh using the constant grey level of each pixel and this value is assigned at the corresponding vertex of the dual mesh. This defines a continuous function that can be linearly interpolated over the mesh. Figure 1 illustrates the construction of a first mesh of quadrangle and its scalar function on an MR section through a human arm with different resolutions. The triangular mesh has the same vertices but each quadrangle is splitted in half. The scalar function is then used to build a Hessian matrix for each vertex of the mesh. In 2D, the Hessian matrix is given by the following:

$$H = \begin{bmatrix} \frac{\partial^2 F_h}{\partial x^2} & \frac{\partial^2 F_h}{\partial x \partial y} \\ \frac{\partial^2 F_h}{\partial y \partial x} & \frac{\partial^2 F_h}{\partial y^2} \end{bmatrix} \quad (2)$$

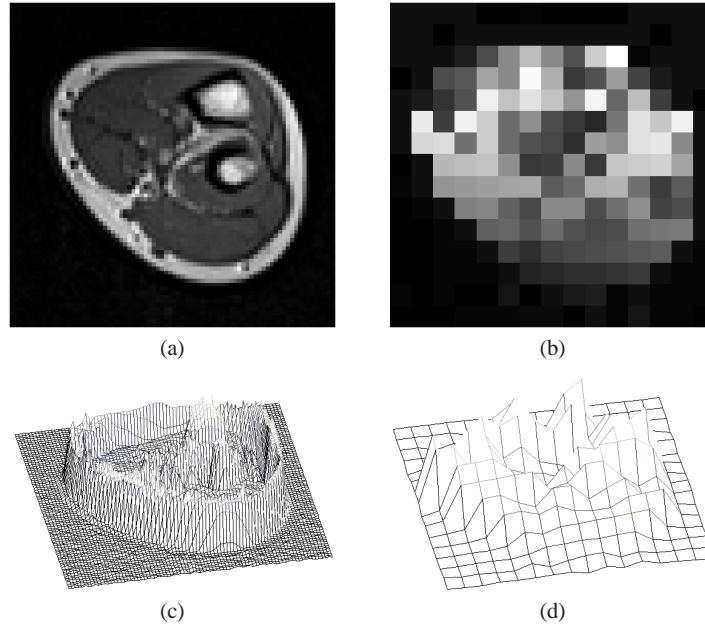


Figure 1. From original image to regular mesh. (a) Original image of size 80x80 pixels. (b) Original image downsampled to 16x16 pixels. (c-d) Regular meshes using grey level values as z coordinates for both images.

In order to fill this matrix, a local reconstruction of second order partial derivatives must be computed. Many reconstruction methods can be used to define the Hessian matrix. Joubarne *et al* showed that the quadratic fitting (QF) method is the most reliable [32][37]. To reconstruct the Hessian of the intensity function at a given vertex, QF selects a number of sampling points in a neighbourhood of one or two levels of elements surrounding the given vertex. A quadratic approximation passing through the sampling vertices is reconstructed. This approximation is then differentiated to yield constant derivatives over the patch. This means that the quadratic function:

$$F_h(x, y) = \sum_{l=1}^6 a_l \varsigma_l(x, y). \quad (3)$$

where

$$\varsigma_l \in \{1, x, y, x^2, xy, y^2\}, \quad (4)$$

is computed to deduce the second order partial derivatives:

$$\begin{aligned} \frac{\partial^2}{\partial x^2} \sum_{l=1}^6 a_l \varsigma_l(x, y) &= 2a_4, \\ \frac{\partial^2}{\partial xy} \sum_{l=1}^6 a_l \varsigma_l(x, y) &= 2a_5, \\ \frac{\partial^2}{\partial y^2} \sum_{l=1}^6 a_l \varsigma_l(x, y) &= 2a_6. \end{aligned} \quad (5)$$

A Riemannian metric, \mathcal{M} , is then constructed based on the Hessian matrix by decomposing it along its eigenvectors, processing its eigenvalues and recomposing it. Since the Hessian matrix H is symmetric, it can be orthogonally diagonalized as:

$$H = RDR^T \quad (6)$$

with R the matrix of eigenvectors and D the diagonal matrix of eigenvalues $\lambda_i, i = 1, \dots, n$. For a given vertex, eigenvalues act as the inverse of the target squared length in the direction given by their respective eigenvectors. Each eigenvalue is then bounded below and above as follows:

$$\lambda'_i = \min(\max(\beta|\lambda_i|, \lambda_{min}), \lambda_{max}). \quad (7)$$

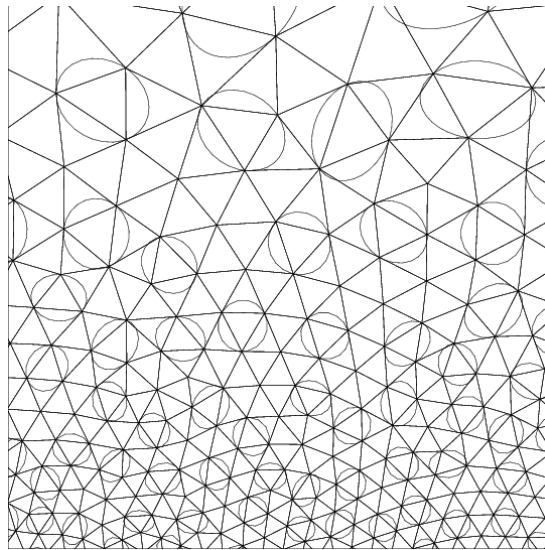


Figure 2. Some metric tensors represented as ellipses over a simple mesh.

In this process, each eigenvalue is made positive and multiplied by a user-defined scaling coefficient β called target edge length in the metric. The user can also specify minimum and maximum edge lengths. Those 3 parameters are defined in the scale of the metric. Edge length is also controlled by a fourth parameter, namely the maximum stretching, that represents the anisotropy factor of the metric. All these parameters are used to define a bounding interval $[\lambda_{min}, \lambda_{max}]$ for each eigenvalue. The processed eigenvalues are then used to compose a diagonal matrix D' . The eigenvalues are non-zero, positive and bounded. That matrix is finally used to recompose the metric tensor by multiplying $\mathcal{M} = RD'R^T$. \mathcal{M} is therefore symmetric positive definite by construction.

To apply user constraints, the distance between two points must be computed in the metric space. The metric distance between the points \vec{p}_A and \vec{p}_B is given by:

$$l_{AB}^{\mathcal{M}} = \int_0^1 \sqrt{(\vec{p}_B - \vec{p}_A)^T \mathcal{M}(\vec{p}_t) (\vec{p}_B - \vec{p}_A)} dt \quad (8)$$

where $\vec{p}_t = \vec{p}_A + t(\vec{p}_B - \vec{p}_A)$.

The resulting constraint metric will limit the size and shape of the mesh elements. It will also align their edges along the directions of variation in the scalar field. Thus, element density will be larger in areas of high field variation and smaller in areas where the field is constant.

For visualization and manipulation purposes, a metric tensor can also be represented as an ellipse for which the main axes are the eigenvectors and the target size in each direction is given by the inverse square root of the corresponding eigenvalue. Figure 2 shows some metric tensors expressed as ellipses over a simple mesh. It can be seen that the metric can precisely describe the size and shape of a mesh element.

2.2. Adaptation Step

A 2D adaptive mesh is obtained by moving vertices (Figure 3 (a)), swapping edges (Figure 3 (b)) and refining or decimating edges (Figure 3 (c) and (d)). Similar operators are available in 3D. The goal of the vertex-moving technique is to regularize or smooth the mesh. Each vertex is moved to the centroid of its neighbors, with distances between vertices evaluated using the metric (Eq. 8). Movements are allowed as long as the topology of the mesh is not modified. For instance, if a vertex

belongs to a topological edge it will be moved towards the centroid but will remain on this edge. This technique is combined with the edge swapping and face swapping (in the 3D case) techniques to achieve geometric and topological smoothing. The goal is always to improve the quality of the elements in the metric space. The edge refinement and coarsening techniques are used to modify the mesh elements' density. The metric defines the element target size everywhere in the mesh. If more elements are required to respect the metric with precision, an edge will be divided to increase the number of elements. On the other hand, if the metric does not vary significantly in a region, the edge coarsening technique will be used to reduce the number of elements.

The adaptation process combines all these techniques. Algorithm 1 shows the order in which they are applied. This order was determined experimentally over a wide range of cases, so that each iteration improves the global mesh quality. At each iteration of the main loop, a statistical analysis is performed on the mesh edge lengths. If the adaptation process has perfectly converged, each edge length in the scale of the metric will be of size one. In practice, this will not happen, but edge lengths will be within an acceptable range from this target, as specified by the user via a threshold. Assuming a Gaussian distribution, a realistic threshold is based on the standard deviation of the edge lengths in the metric. A small value ensures that the resulting Gaussian distribution is narrow, with a lower dispersion. As this value cannot always be reached, a maximum number of iterations is also specified.

The main loop is divided into three phases. The first one is a topological and geometric smoothing phase. It involves edge swapping and vertex moving techniques. The vertex moving is performed iteratively until either a stopping criterion or a maximum number of iterations is reached. The stopping criterion is a threshold on the mean displacement in the metric space. The user can control the movement speed by specifying a relaxation factor. To determine which elements will be swapped, each element is analyzed to compute its level of deformation. Instead of sorting the elements by their shape deformation, a statistical process on the element shape measure, evaluated using the metric, is performed and returns the average and the standard deviation of the deformation. Still assuming a Gaussian distribution, all the elements above a threshold are treated. This threshold corresponds to a number of times the standard deviation so that the most stretched elements will undergo diagonal swapping.

The second and third phases respectively start with edge refinement and edge coarsening and both end with smoothing. A similar statistical process is also performed in those phases to evaluate the length of the edges according to the metric. The same threshold used for diagonal swapping applies for mesh refinement and decimation, to determine how many elements will be considered. Each of these elements will be considered in the inner loop of refinement and decimation. The mesh refinement loop has its own threshold that determines when an edge is too long. Here, too long means that an edge evaluated in the metric space has a length greater than one. The closer to one this parameter is set, the more time it will take to compute the refinement. For this reason, a maximum number of iterations is again specified for this phase. The edges that are too long are cut in two. The same principles apply to the mesh decimation loop. In that case, edges that are too short are removed.

The global mesh quality of an anisotropic adapted mesh is not easy to assess. Classic quality measures cannot be applied directly because carefully aligned or stretched elements, which can be a good choice in the anisotropic case, will be penalized by such measures. Quality measures can alternatively be computed in the Riemannian space to take into account anisotropic meshes. A comparison of quality measures in the Euclidian and Riemannian spaces can be found in [38].

Another way to verify the quality is to compute how well the mesh respects the target metric. To do so, it is possible to compute the non-conformity coefficient [39]. At any step of the adaptation process, it is possible to define, for an element, a metric \mathcal{M}_K that transforms it from a control space where it is perfectly equilateral, into the Euclidian space. It is also possible to define a target metric \mathcal{M}_S that transforms the element from the Euclidian space into the metric space. The non-conformity coefficient ϵ_K is defined as the matrix norm of the average difference between the current metric of an element \mathcal{M}_K and the targeted metric \mathcal{M}_S . More formally, it is expressed as:

$$\epsilon_K = \|\mathcal{M}_S^{-1}\mathcal{M}_K + \mathcal{M}_K^{-1}\mathcal{M}_S - 2I\|. \quad (9)$$

Algorithm 1 Adaptation cycle

```

while (!convergence) do { //Global adaptation cycle}
  while (!convergence) do { //Topological and geometric smoothing}
    Edge swapping
    Vertex moving
  end while
  while (!convergence) do { //Refinement phase and smoothing}
    Edge refinement
    Edge swapping
    Vertex moving
  end while
  while (!convergence) do { //Coarsening phase and smoothing}
    Edge coarsening
    Edge swapping
    Vertex moving
  end while
end while

```

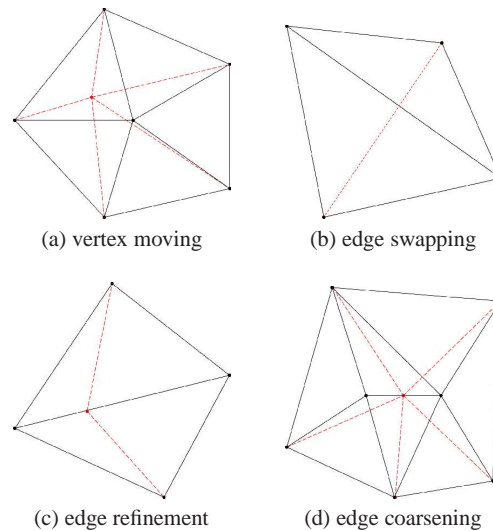


Figure 3. Adaptation techniques (resulting edges and vertices are in red).

Labbé *et al* [36] showed that the proposed method converges and yields adapted meshes with low non-conformity coefficient values. Note that this coefficient is similar to the quality measure used by Marchandise *et al* [19].

2.3. Interpolation Step

When the adaptation step is completed, the intensity field is interpolated over the mesh for visualization purposes and to allow an iterative process. The first iteration can be used to refine the initial intensity field and obtain a fine mesh with a more regular metric. Other adaptation steps can be performed, using the previous mesh as the initial mesh, to obtain a coarser mesh that still reflects the inner field variations. The resulting mesh's validity can be evaluated qualitatively by superimposing it to the actual edges that the mesh must fit, and quantitatively by computing different statistics such as element aspect ratio to ensure that it fits the numerical solvers quality constraints.

Table I. Adaptation Step Parameters Values

Operator	Convergence criteria	Maximum number of iterations
Global loop	0.12	12
Vertex moving	0.1	10
Edge swapping	2	20
Edge refinement	1.3	1
Edge coarsening	0.7	10

3. EXPERIMENTAL RESULTS AND ANALYSIS

To validate the mesh generation process, various datasets were used. This section first shows the parameter values used by the adaptation step of our method, then presents results obtained with two different datasets using those values. First, a synthetic dataset consisting of a sphere centered inside a cube will be presented. Then, the results of using our method on real MRI datasets will be presented. All the datasets were processed with the algorithm presented in this paper and compared with those produced by the method proposed by Goksel *et al* [28] called *Variational Image Meshing* or VIM. Comparisons are based on the accuracy of the volumes of the structures of interest extracted from the adapted meshes, with respect to the ground-truth volumes of the same structures, obtained either analytically (for the synthetic data) or by semi-automatic segmentation by an expert (for the clinical data). Complementary comparisons using the Hausdorff distance between the surfaces extracted from the adapted meshes and their ground-truth counterparts will also be presented to evaluate errors between surfaces [40].

3.1. Calibration

The content of Table I shows the parameter values used to control the mesh adaptation step. The relaxation factor used in this study was kept at its default value of one. For a description of the different parameters, see subsection 2.2 in the previous section.

3.2. Sphere test case

This dataset represents a sphere of radius 1 centered in a cube with edges of length 3. The initial mesh is a grid of resolution 100x100x100 filled with tetrahedral elements. The initial mesh has therefore 1 million vertices and 4.85 million tetrahedral elements. The initial solution for the scalar field is a binary mask of value 0 outside the sphere and 1 inside. The iterative process described in the previous section is then performed to generate the metric and adapt the mesh.

Comparison with VIM method

To compare the accuracy of our method to the VIM method, meshes of various resolutions were adapted and then segmented to keep only elements completely included within the analytical definition of the sphere. By summing the volumes of all the extracted elements, an approximation of the segmented sphere volume is computed. Of course, the approximation cannot be more accurate than the resolution of the initial solution for which the computed volume is 3.97. This is slightly lower than the volume of a perfect sphere, which is $4\pi/3$ or around 4.19. The graph in Figure 4a shows that our method quickly converges to the volume obtained by segmenting the initial volume of 100x100x100 nodes. Moreover, it shows that the VIM method needs a mesh of over 150 000 elements to obtain a segmented mesh volume that converges to 95% of the experimental volume. Our method thus provides more accurate meshes with fewer elements. Indeed, with only 10 000 elements, our methods segmented mesh represents 97% of the experimental volume.

Another way to compare both methods is to compute the Hausdorff distance between the outer surfaces of each segmented mesh and the perfect sphere as an error estimator. To compute this distance, 20,000 points were projected from the perfect sphere to the segmented meshes. For each evaluation, we considered the maximum and the mean Hausdorff distances. To obtain a ratio, the

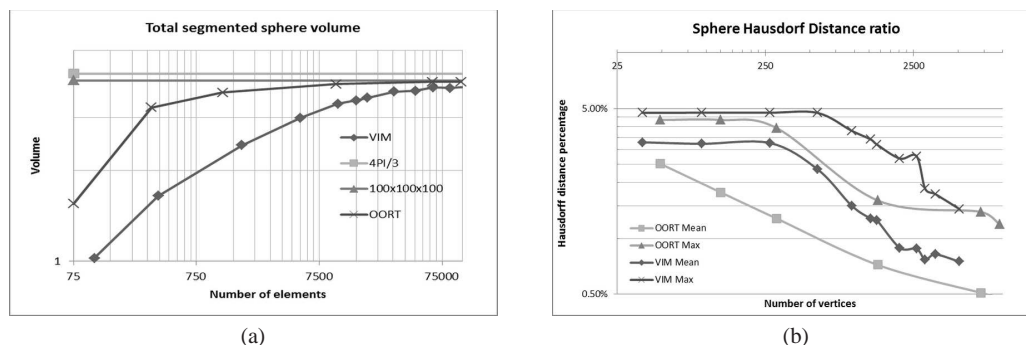


Figure 4. Sphere test case. (a) Total segmented sphere volume. (b) Hausdorff distances of segmented meshes from the sphere

distance is divided by the diagonal of the bounding box of the sphere. The graph in Figure 4b shows that our method (represented by the light grey curves) produces meshes with lower maximum and mean Hausdorff distances for the same number of vertices than the VIM method.

Qualitative evaluation

For the purposes of visualization and qualitative evaluation, it is possible to perform a quick segmentation by using a simple thresholding technique on the mesh produced with our method. By superimposing a perfect sphere over the resulting segmented mesh, it can be seen to what degree the elements follow the spheres surface. Figure 5 shows different views of the adapted mesh segmented with a simple threshold set to 10% of the maximum intensity value to remove the dark background. As can be seen, the resulting mesh doesn't perfectly fit the sphere; indeed, some elements cross the spheres surface. Several cutting planes are also applied to the mesh in these views to assess the mesh elements size distribution. An interesting observation is that the mesh is finer near the surface boundary and much coarser where the scalar field is constant, i.e. in most of the volume inside and outside the sphere.

The main parameter that can influence the global shape of the mesh is the *target edge length in the metric* (TELM). Figure 6 shows the effect of varying this parameter. In all cases, elements are coarser where the scalar field is constant and finer near the edge of the sphere where the scalar field varies. Increasing the TELM reduces the number of elements while trying to preserve significant features. In this case, the only interesting feature is the spheres edge. Even with only 200 vertices in total, this boundary can still be modeled adequately. Therefore, the choice of the TELM should be based on the desired precision around the edges. For this example, a TELM of 5 is enough to have good precision around the edge of the sphere while greatly reducing the mesh size (from 1 million down to 5,832 vertices).

The sphere example also informs us about the impact of the metric initial resolution. Figure 7 shows adapted meshes that were all computed using the same TELM value of 5. It illustrates that when the metric precision increases, more mesh elements are placed closer to the sphere's edge and so the edge precision also increases. Therefore, the precision of the initial dataset is also important.

3.3. Trunk test case

The algorithm presented in this paper was applied to MR data of the human trunk, and the resulting meshes were compared to 3D segmentations of the same data performed by an expert and to meshes obtained using the VIM method proposed by Goksel *et al* [28]. In collaboration with Sainte-Justine University Hospital Centre in Montreal, Canada, MRI datasets of the trunk of 20 patients were obtained. These data were taken in the context of a study on the impact of spine deformity on breast asymmetry [41]. The datasets were acquired from adolescent girls with spine deformity. All acquisitions were performed using a clinical 1.5T MRI system (Achieva XR, Philips Healthcare, Best, The Netherlands) using a dedicated 16 channel breast coil. The patients were in prone position with their arms resting overhead and breasts hanging under free breathing. A

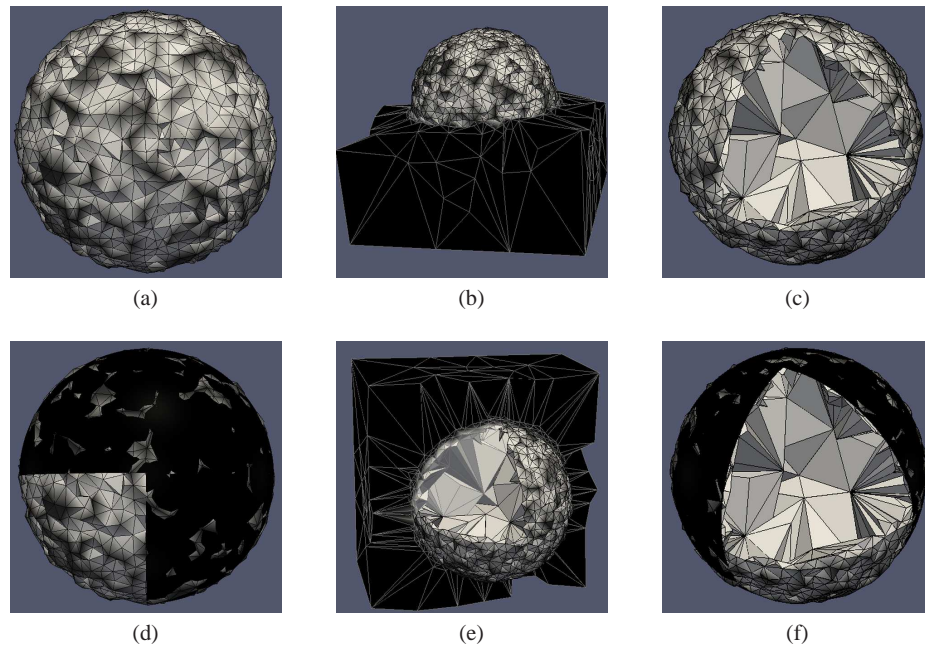


Figure 5. Views of adapted mesh with simple threshold applied and cutting planes. (a) Segmented adapted mesh. (b) Segmented adapted mesh with surrounding elements under a cutting plane. (c) Segmented adapted mesh opened by oblique cutting plane. (d) Perfect sphere around the segmented adapted mesh. (e) Combination of (c) and (b). (f) Combination of (c) and (d).

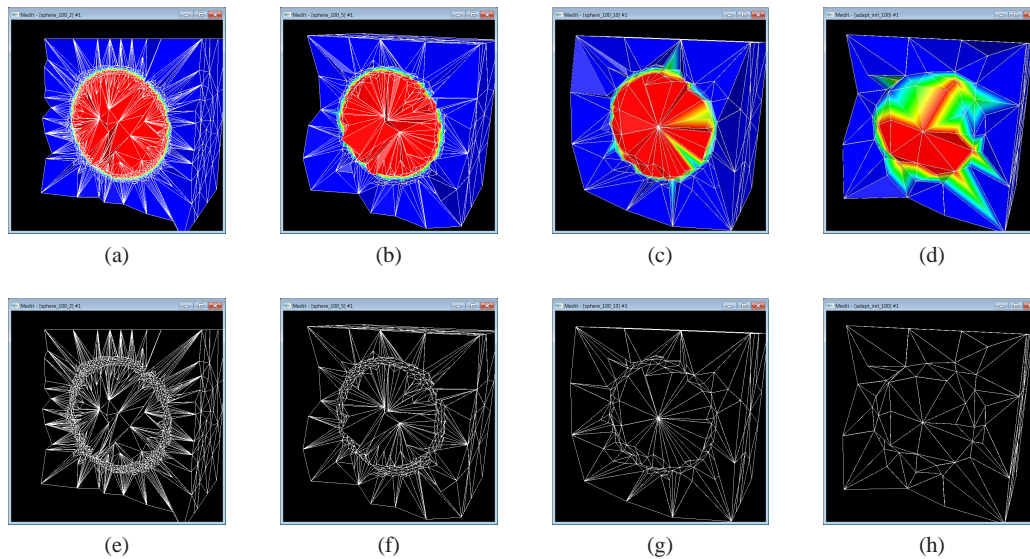


Figure 6. Result of mesh adaptation for different *target edge lengths in the metric* (TELM). Each column shows the same mesh. Colormap is interpolated from red to blue as scalar value decrease.
 (a) & (e) show a mesh with TELM of 2, 35600 vertices and 201780 elements.
 (b) & (f) show a mesh with TELM of 5, 5832 vertices and 32959 elements.
 (c) & (g) show a mesh with TELM of 10, 1109 vertices and 6054 elements.
 (d) & (h) show a mesh with TELM of 20, 200 vertices and 938 elements.

3D gradient-echo also called FFE sequence without fat suppression was used (TR/TE = 7.6/4.6, flip angle: 12 degrees, FoV: 340 x 340 x 220 mm, NSA: 1, Scan duration: 3m40s). This sequence minimized BO inhomogeneities by using an automated image-based shimming procedure. Also

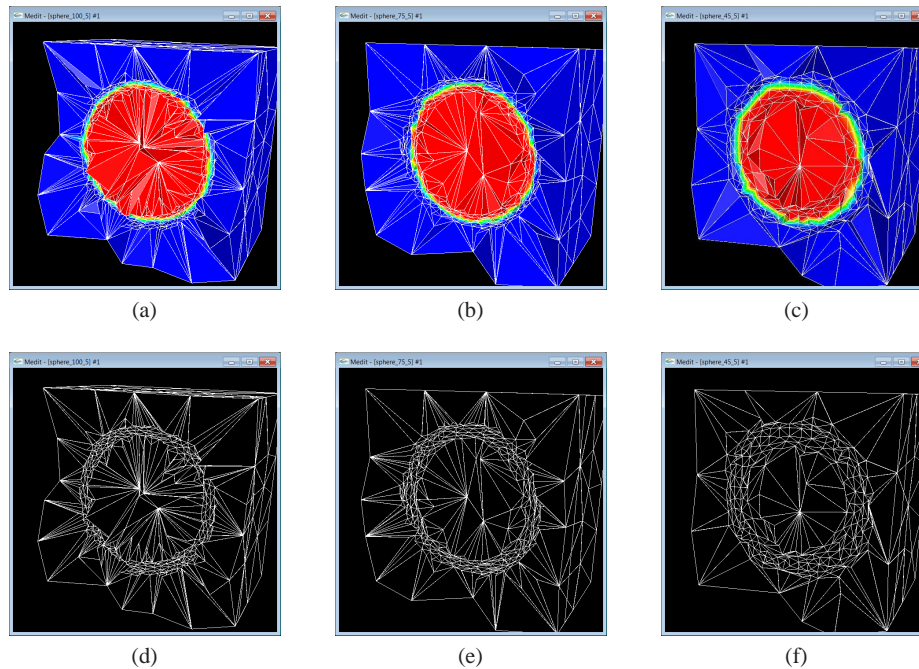


Figure 7. Impact of initial metric precision for mesh with TELM of 5. The initial metric precision decreases from left to right along with the adapted mesh vertex localization accuracy. Colormap is interpolated from red to blue as scalar value decrease.

- (a) & (d) initial precision of 100^3 , adapted mesh of 5832 vertices and 32959 elements.
- (b) & (e) initial precision of 75^3 , adapted mesh of 4426 vertices and 25075 elements.
- (c) & (f) initial precision of 45^3 , adapted mesh of 2453 vertices and 13913 elements.

the patient position combined with a specialized acquisition antenna lead to negligible breathing artifacts [42]. The data reconstructions used by the expert were $352 \times 352 \times 70$ pixels with a resolution of $1 \text{ mm} \times 1 \text{ mm} \times 3 \text{ mm}$.

Comparison with VIM method

To compare the meshes produced using the VIM method with those obtained using our proposed method, we ran both methods on the datasets. We aimed to obtain meshes with a similar number of vertices. Results for the VIM method were produced using Matlab code provided by the authors (VIMesh 1.1). Samples of both meshes are shown in Figure 8; they respectively have 25,725 and 22,543 vertices. As expected, VIM produced a highly isotropic mesh, while our method produced an anisotropic mesh. Another important difference between the two methods lies in the intensity field interpolation. VIM defines a constant value within each mesh element, whereas our method defines a value at each vertex.

The expert used ITK_SNAP 2.4.0 [43] to perform semi-automatic 3D active contour segmentation of the breasts, followed by rigorous pixel by pixel manual validation to obtain an accurate breast volume segmentation. This result was first used to confirm that meshes produced using the proposed method have elements aligned with significant anatomical features such as organ boundaries. The experts segmentations also serve to measure the accuracy of a given reconstructed volume. Each dataset was used without any modification to enhance contours or reduce its resolution. Figure 9 shows the breasts segmentations made by the expert for subjects 15 and 20 in the study and the resulting adapted meshes segmented with a simple threshold set to 10% of the maximum intensity value in order to extract the outer chest surfaces. The segmented surfaces obtained from the adapted meshes (center images) are not smooth but give a good idea of what could be obtained with a more advanced segmentation method. The segmentations were performed using the threshold filter function in the scientific visualization tool ParaView [44].

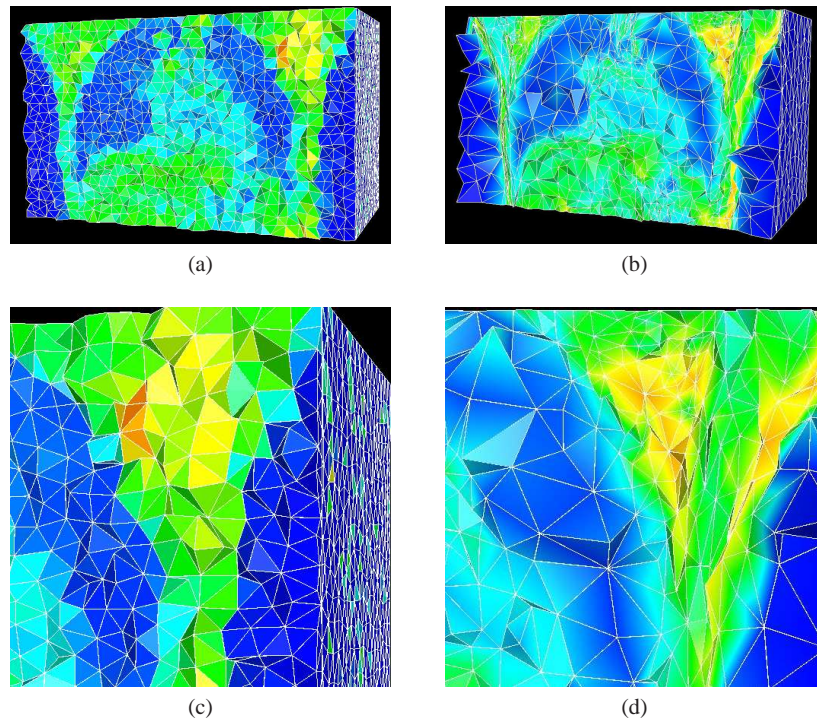


Figure 8. Mesh comparison between VIM method (left column) and proposed method (right column) for subject 15. Colormap is interpolated from red to blue as scalar value decrease.

Figure 10 shows an axial slice through the resulting mesh with the proposed method. Two regions containing boundaries between internal organs and at the outer trunk surface were selected to assess the elements' orientations. Enlarged views of the two regions are also displayed. It can be seen that the elements naturally follow intrinsic boundaries within the dataset. This important feature will be useful when we need to characterize mesh elements with biomechanical constraints. In order to compare tetrahedral positioning precision and allow quantitative comparisons, the expert segmentations were used to extract subsets of interest from the meshes obtained using the two methods. Specifically, only elements completely inside the expert-segmented volume were extracted. Segmentations of the right and left breasts were considered separately for each subject in the study. Multiple meshes were produced with different resolutions. For the VIM method, the number of voxels per tetrahedral element was tested with values of 1, 2, 5, 10, 15, 25, 55, 75 and 100. This was the only parameter for VIM that varied in our tests. For the proposed method, the TELM parameter was tested with values of 3*, 5*, 5, 6, 10 and 20. The first two cases (marked with asterisks in the previous list) used a regular mesh as the initial mesh, while the remaining cases used the mesh with TELM of 3 as the initial mesh.

For both methods, the higher the value of the given parameter, the coarser is the resulting adapted mesh. Volume ratios were computed between each segmented adapted mesh and the corresponding expert segmentation of the (left or right) breast. The volume ratio is computed as the sum of the volumes of the extracted tetrahedral elements over the volume of the expert-segmented breast. The graphs in Figure 11 show the volume ratios computed for subject 15 in the study. It can be seen that our method systematically places elements inside the structure of interest with more precision than VIM. For a mesh of 50 000 elements, our method is 4%-5% more accurate with this dataset.

For comparison purposes, we then considered meshes of size 10k, 30k and 50k elements among all those produced from the patient trunk datasets. We found that our method produces meshes with as good or better placement accuracy than those from the VIM method. As can be seen in Table II, which shows the results for 5 cases, our method generates better volume approximations in 90% of the cases (winner of each comparison marked with asterisk).

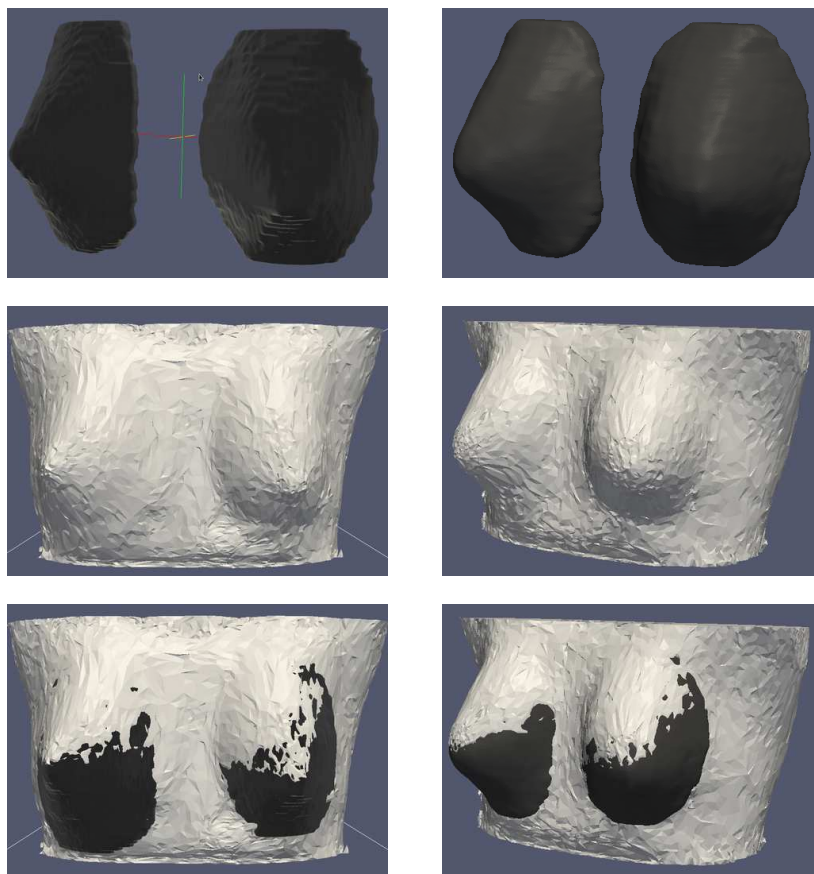


Figure 9. Breasts segmentation for subjects 15 and 20. Top images are the experts segmentations. Center images are the threshold-segmented meshes using our method. Bottom images show both meshes superimposed.

Furthermore, we considered the Hausdorff distance as a complementary comparison measure, and computed it between the adapted meshes from both methods and the ground-truth breast volumes. Distances between extracted meshes and expert segmentations were normalized by the diagonal of the bounding box of the expert segmentation to facilitate comparisons. Considering subject 15, the graphs in Figure 12 shows that the mean Hausdorff distances for both methods decrease at a similar rate, but our method always produces a lower mean value. The maximum Hausdorff distances are similar for both methods, but the VIM method yields in general slightly lower maximum values. Similar results were obtained for all cases. Table III and Table IV show the maximum and mean values for the same five cases as in Table II and for different numbers of vertices. It can be seen that maximum Hausdorff distances are similar (equal in 60% of the case) but as the number of vertices increase the VIM method is usually slightly better. Also, Table IV shows that our method always produces meshes with equal or lower mean Hausdorff distance values.

4. CONCLUSION AND FUTURE WORK

The method proposed in this paper successfully produces 3D geometric patient-specific mesh models of the human trunk without any segmentation or preprocessing steps.

The adapted mesh is anisotropic. The generation process builds a metric based on the second order derivatives of the gray levels. The elements shape, size and orientation are determined by this metric. Using an anisotropic mesh allows us to compress the dataset or reduce the mesh size

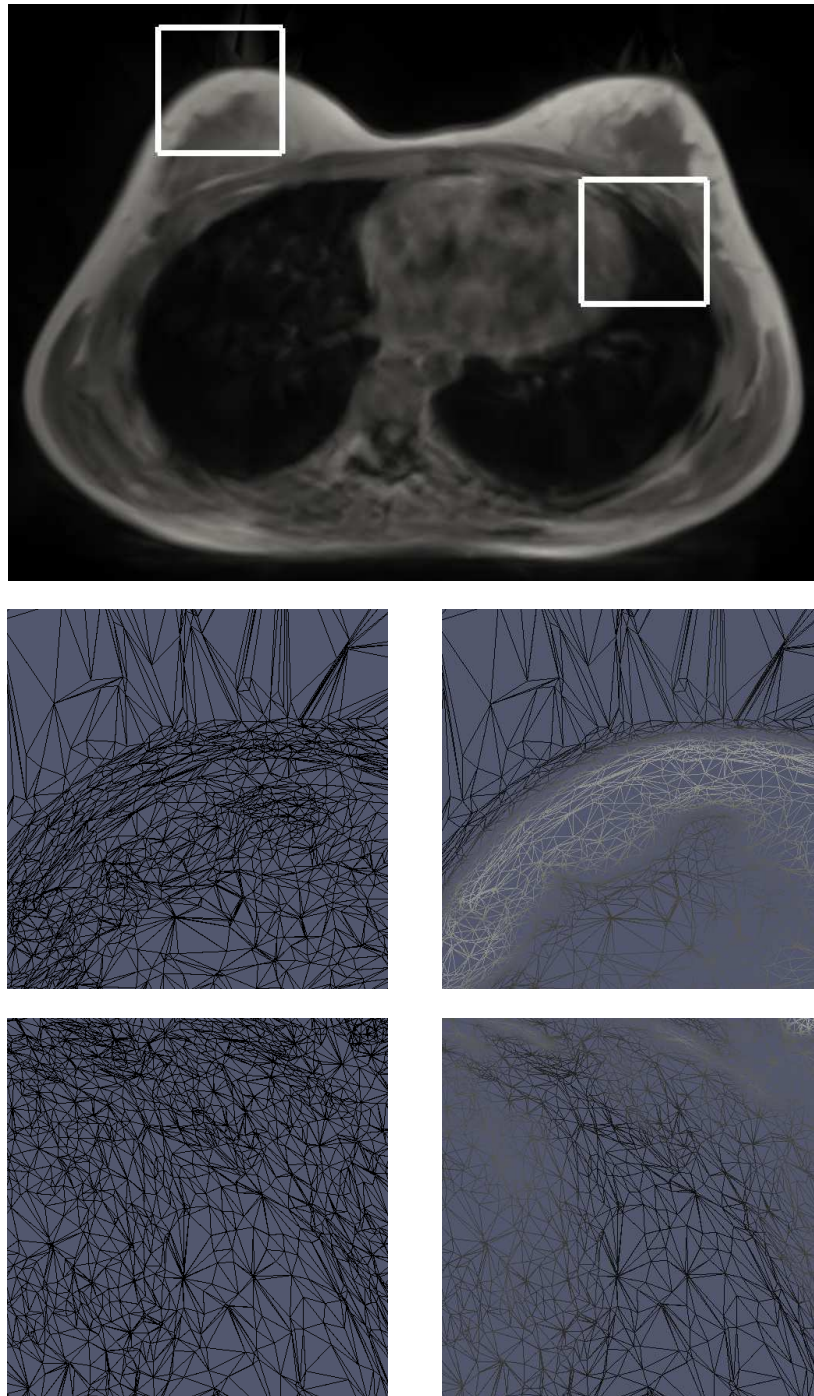


Figure 10. Adapted mesh for subject 15. Top image shows an axial slice through the adapted mesh with grey level interpolation. The two square regions enclose tissue boundaries. Center and bottom images show enlarged views of the left and right square regions respectively. Close ups are in wireframe with and without grey level interpolation.

without compromising its quality. As the type of model to be built for our context must span the whole trunk, reducing the size of the model is not only useful but necessary to perform numerical simulations in a reasonable delay. Another way to reduce the model size is to produce a coarser mesh, which can be achieved quite simply with our method by increasing the TELM parameter.

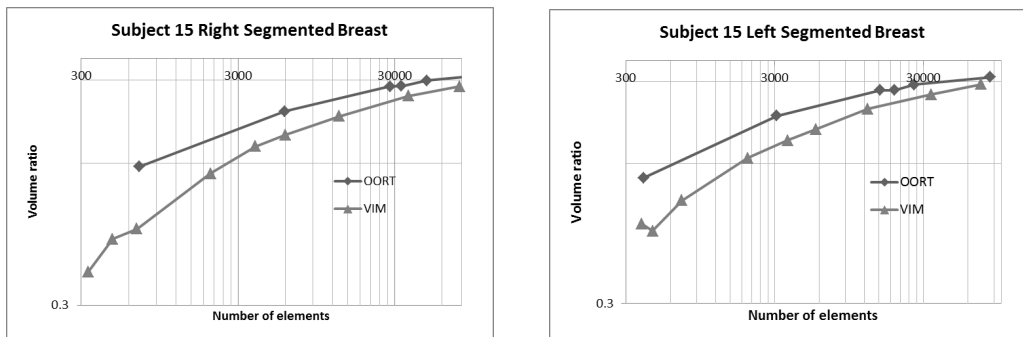


Figure 11. Segmented Breast Volume Ratio for VIM and proposed methods (subject 15)

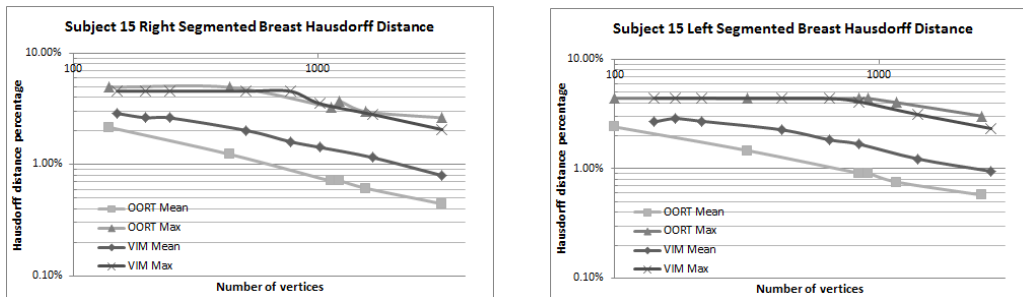


Figure 12. Hausdorff distance between segmented breast volumes for VIM and proposed methods (subject 15)

Table II. Extracted Volume Ratios for VIM and Proposed Methods

	10,000 elements		30,000 elements		50,000 elements	
	OORT	VIM	OORT	VIM	OORT	VIM
15 Right	0.79*	0.72	0.87*	0.81	0.90*	0.85
15 Left	0.82*	0.76	0.89*	0.83	0.90*	0.86
16 Right	0.77*	0.75	0.83	0.83	0.86	0.86
16 Left	0.77*	0.76	0.83	0.83	0.86	0.86
17 Right	0.76*	0.75	0.84*	0.83	0.88*	0.86
17 Left	0.77	0.77	0.82	0.84*	0.86	0.87*
18 Right	0.76*	0.74	0.84*	0.82	0.88*	0.85
18 Left	0.72	0.77*	0.87*	0.84	0.89*	0.87
20 Right	0.80*	0.75	0.84*	0.82	0.87*	0.86
20 Left	0.79*	0.77	0.85*	0.83	0.87*	0.86

Table III. Maximum Hausdorff Distance Ratios for VIM and Proposed Methods

	250 vertices		1000 vertices		2500 vertices	
	OORT	VIM	OORT	VIM	OORT	VIM
15 Right	0.05	0.05	0.04	0.04	0.03	0.02*
15 Left	0.04	0.04	0.04	0.04	0.03	0.02*
16 Right	0.04	0.04	0.03*	0.04	0.02	0.02
16 Left	0.05	0.05	0.04	0.03*	0.02	0.02
17 Right	0.04	0.04	0.04	0.04	0.04	0.02*
17 Left	0.04*	0.05	0.04	0.03*	0.04	0.02*
18 Right	0.05	0.05	0.04	0.03*	0.02	0.02
18 Left	0.04	0.04	0.04	0.03*	0.03	0.02*
20 Right	0.05	0.05	0.04	0.04	0.03	0.02
20 Left	0.04*	0.05	0.04	0.04	0.03	0.02

Table IV. Mean Hausdorff Distance Ratios for VIM and Proposed Methods

	250 vertices		1000 vertices		2500 vertices	
	OORT	VIM	OORT	VIM	OORT	VIM
15 Right	0.02	0.03	0.01	0.01	0.01	0.01
15 Left	0.02	0.03	0.01	0.02	0.01	0.01
16 Right	0.02	0.03	0.01	0.01	0.01	0.01
16 Left	0.02	0.03	0.01	0.01	0.01	0.01
17 Right	0.02	0.03	0.01	0.02	0.01	0.01
17 Left	0.02	0.03	0.01	0.01	0.01	0.01
18 Right	0.02	0.03	0.01	0.01	0.01	0.01
18 Left	0.02	0.02	0.01	0.01	0.01	0.01
20 Right	0.02	0.03	0.01	0.01	0.01	0.01
20 Left	0.02	0.03	0.01	0.01	0.01	0.01

Increasing this parameter also causes the mesh to become more isotropic. So if isotropy is desired, our method can also produce meshes with low anisotropy. For a given mesh resolution, anisotropy can also be controlled by the parameter maximum stretching in the metric, which indirectly controls the aspect ratio. This parameter is very useful because the acceptable level of anisotropy varies from one numerical solver to another. Solvers are also highly sensitive to the elements' orientations, especially when they are anisotropic. On the other hand, if the number of elements is not an issue, our method can exploit anisotropy and even produce meshes with sub-pixel resolution.

Our method yields good results with both artificial and experimental datasets. In each case, the resulting mesh respects the inner geometry of the dataset. Furthermore, results show that elements within the mesh are aligned with organ boundaries even though these have not been explicitly defined a priori. Results show that our method gives as good or better meshes as a similar isotropic method (VIM), based on the accuracy of the volume reconstruction for a given mesh size and the mean Hausdorff distance comparison. However, VIM method vertex positioning seems slightly better according to max Hausdorff distance comparison.

The proposed method is robust in many respects. First, to build the metric of a specific vertex, the method uses a neighbourhood around that vertex, which makes the method less sensitive to noise in the dataset by performing an implicit smoothing over the grey level values. Because noise cannot be avoided when dealing with MRI data, this feature is important for our overall project. Second, all the meshes generated from clinical data presented in this paper were computed using the same set of parameters. This means that, even if the method gives a lot of control to the user, it is possible to define a single set of parameters that can be used with all the datasets from a given acquisition protocol. **A possible drawback is, however, that many parameters might need to be changed when dealing with new datasets.** Third, our method is completely automatic and easy to use. Finally, our results show that increasing the number of elements improves the precision of the volume reconstruction. This means that the method converges to an optimal solution according to the precision of the dataset and that the end user has the ability to balance the competing requirements of computation time and mesh precision.

The method we propose offers control over the resulting meshes in order to respect given criteria based on element shape and size. However, some aspects still need to be improved, especially vertex positioning and computation time. First, although mesh elements follow boundaries within the dataset, mesh vertices could be placed more accurately at the boundaries themselves. To improve this aspect, gradient information could be used when building the metric that drives the adaptation process. **Information from different acquisition protocol or imaging modalities could also be mapped with our data to obtain a more robust metric.** Moreover, when building the metric, an approximation is made to consider the intensity field as a continuous field. A vertex of the dual mesh is placed at the center of each pixel or voxel of the dataset. This causes the field to be shifted by half a pixel on average. If the pixel resolution is low, this fact can partly explain why mesh vertices are not perfectly placed on the boundaries present in the initial dataset. Lastly, even though our adaptation

process is automatic and works with large datasets, computation time is still an issue **that can limit the applicability of the method to very large datasets**. This aspect should be assessed in future work by exploiting parallelization in the initialization and adaptation steps of the method.

In relation with numerical simulation, future work will focus on segmenting the geometric model to characterize the different layers of soft tissue in the human trunk. As the adapted model respects the inner boundaries present in the image data, it should help to segment the bone, muscle and fat layers, which are the main anatomical structures of interest in our application.

ACKNOWLEDGEMENT

The authors would like to thank Philippe Debanné for his valuable help and useful comments and Joyce Ramsay for letting us use MR datasets from her study. This work was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada and the MEDITIS training program (École Polytechnique de Montréal and NSERC).

REFERENCES

1. NRC-CRNC. (2009). *Highlights - Simulation-based brain surgery in Halifax yields breakthrough in surgical training and rehearsal*. Available: <http://www.nrc-cnrc.gc.ca/eng/news/nrc/2009/08/26/virtual-surgery.html>
2. G. Picinbono, H. Delingette, and N. Ayache, "Non-linear anisotropic elasticity for real-time surgery simulation," *Graphical Models*, vol. 65, pp. 305-321, 2003.
3. X. Drevelle, Y. Lafon, E. Ebermeyer, I. Courtois, J. Dubousset, and W. Skalli, "Analysis of idiopathic scoliosis progression by using numerical simulation," *Spine*, vol. 35, pp. E407-12, 2010.
4. B. V. Reamy and J. B. Slakey, "Adolescent idiopathic scoliosis: review and current concepts," *Am Fam Physician*, vol. 64, pp. 111-6, Jul 1 2001.
5. I. Villemure, C. E. Aubin, J. Dansereau, and H. Labelle, "Biomechanical simulations of the spine deformation process in adolescent idiopathic scoliosis from different pathogenesis hypotheses," *European Spine Journal*, vol. 13, pp. 83-90, 2004.
6. L. Shi, D. Wang, M. Driscoll, I. Villemure, W. Chu, J. Cheng, and C.-E. Aubin, "Biomechanical analysis and modeling of different vertebral growth patterns in adolescent idiopathic scoliosis and healthy subjects," *Scoliosis*, vol. 6, p. 11, 2011.
7. W. K. I. Payne, J. W. Ogilvie, M. D. Resnick, R. L. Kane, E. E. Transfeldt, and R. W. Blum, "Does Scoliosis Have a Psychological Impact and Does Gender Make a Difference?," *Spine*, vol. 22, pp. 1380-1384, 1997.
8. O. Dionne, "Developpement d'un modele numerique simplifie du tronc pour simuler l'effet d'une chirurgie de la scoliose sur l'apparence externe d'un patient," Ecole Polytechnique, Montreal (Canada) M.Sc.A., Ecole Polytechnique, Montreal (Canada), Canada, 2010.
9. R. Harmouche, F. Cheriet, H. Labelle, and J. Dansereau, "Multimodal image registration of the scoliotic torso for surgical planning," *BMC Medical Imaging*, vol. 13, p. 1, 2013.
10. S. J. Owen, "A Survey of Unstructured Mesh Generation Technology", in *7th International Meshing Roundtable*, Sandia National Lab, 1998, pp. 239-267.
11. R. Lohner, "Automatic unstructured grid generators", *Finite Elements in Analysis and Design*, vol. 25, pp. 111-134, 1997.
12. J. M. Sullivan, Z. Wu, and A. Kulkarni, "3D Volume Mesh Generation of Human Organs Using Surface Geometries Created from the Visible Human Data Set", in *The Third Visible Human Project Conference Proceedings*, National Institutes of Health, William H. Natcher Conference Center, Bethesda, Maryland USA, 2000.
13. Y. Zhang, C. Bajaj, and B.-S. Sohn, "Adaptive and quality 3D meshing from imaging data", in *Proceedings of the eighth ACM symposium on Solid modeling and applications*, ed Seattle, Washington, USA: ACM Press, 2003, pp. 286-291.
14. Y. Zhang, W. Wang, X. Liang, Y. Bazilevs, Hsu, T. Kvamsdal, R. Brekken, and J. Isaksen, "High-fidelity tetrahedral mesh generation from medical imaging data for fluid-structure interaction analysis of cerebral aneurysms", *Computer Modeling in Engineering & Sciences*, vol. 42, pp. 131-49, 2009.
15. H. Si and K. Grtner, "Meshing piecewise linear complexes by constrained Delaunay tetrahedralizations", in *Proceedings of the 14th International Meshing Roundtable*, Berlin/Heidelberg, 2005, pp. pp. 147-163.
16. N. Archip, R. Rohling, V. Dessenne, P. J. Erard, and L. P. Nolte, "Anatomical structure modeling from medical images", *Computer Methods and Programs in Biomedicine*, vol. 82, pp. 203-15, 2006.
17. Z. Salo, M. Beek, and C. M. Whyne, "Evaluation of mesh morphing and mapping techniques in patient specific modelling of the human pelvis", *International Journal for Numerical Methods in Biomedical Engineering*, vol. 28, pp. 904-913, 2012.
18. X. Feng, K. Xia, Y. Tong, and G.-W. Wei, "Geometric modeling of subcellular structures, organelles, and multiprotein complexes", *International Journal for Numerical Methods in Biomedical Engineering*, vol. 28, pp. 1198-1223, 2012.
19. E. Marchandise, C. Geuzaine, and J. F. Remacle, "Cardiovascular and lung mesh generation based on centerlines", *International Journal for Numerical Methods in Biomedical Engineering*, vol. 29, pp. 665-682, 2013.
20. Z. Zhong, X. Guo, W. Wang, B. Lvy, F. Sun, Y. Liu, and W. Mao, "Particle-based anisotropic surface meshing," *ACM Trans. Graph.*, vol. 32, pp. 1-14, 2013.

21. X. Jiao, A. Colombi, X. Ni, and J. Hart, "Anisotropic mesh adaptation for evolving triangulated surfaces," *Engineering with Computers*, vol. 26, pp. 363-376, 2010/08/01 2010.
22. C. Petitjean and J.-N. Dacher, "A review of segmentation methods in short axis cardiac MR images," *Medical Image Analysis*, vol. 15, pp. 169-184, 2011.
23. M. D. Adams, "An improved content-adaptive mesh-generation method for image representation," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*, 2010, pp. 873-876.
24. M. Adams, "A Flexible Content-Adaptive Mesh-Generation Strategy for Image Representation," *Image Processing, IEEE Transactions on*, vol. PP, pp. 1-1, 2011.
25. J. G. Brankov, Y. Yongyi, and M. N. Wernick, "Tomographic image reconstruction based on a content-adaptive mesh model," *Medical Imaging, IEEE Transactions on*, vol. 23, pp. 202-212, 2004.
26. K. W. C. Hung, M. Nakao, K. Yoshimura, and K. Minato, "Background-incorporated volumetric model for patient-specific surgical simulation: a segmentation-free, modeling-free framework," *International Journal of Computer Assisted Radiology and Surgery*, vol. 6, pp. 35-45, 2011.
27. A. K. Wai, C. Hung, B. M. Nakao, and C. K. Minato, "Automated volume sampling optimization for direct volume deformation in patient-specific surgical simulation," in *Biomedical Imaging: From Nano to Macro, 2009. ISBI '09. IEEE International Symposium on*, 2009, pp. 1051-1054.
28. O. Goksel and S. E. Salcudean, "Image-Based Variational Meshing," *Medical Imaging, IEEE Transactions on*, vol. 30, pp. 11-21, 2011.
29. F. Labelle and J. R. Shewchuk, "Anisotropic Voronoi Diagrams and Guaranteed-Quality Anisotropic Mesh Generation," in *Nineteenth Annual Symposium on Computational Geometry*, San Diego, California, 2003, pp. 191-200.
30. J. Dompierre and P. Labbé, "OORT : Objet-Oriented Remeshing Toolkit," ed. Montral: MAGNU : Laboratoire de Maillage de Gomtrie Numrique, 2006.
31. P. Labbé, J. Dompierre, M.-G. Vallet, and F. Guibault, "Verification of three-dimensional anisotropic adaptive processes," *International Journal for Numerical Methods in Engineering*, vol. 88, pp. 350-369, 2011.
32. É. Joubarne and F. Guibault, "3D Metric-based anisotropic mesh adaptation for vortex capture," *Mathematics and Computers in Simulation*, vol. In Press, Corrected Proof, 2011.
33. O. Courchesne, F. Guibault, J. Dompierre, and F. Chieriet, "Adaptive Mesh Generation of MRI Images for 3D Reconstruction of Human Trunk," in *Image Analysis and Recognition*. vol. 4633, M. Kamel and A. Campilho, Eds., ed: Springer Berlin / Heidelberg, 2007, pp. 1040-1051.
34. O. Courchesne, F. Chieriet, D. Pri, and F. Guibault, "Anisotropic Mesh Generation for Myocardial Biomechanical Modeling," in *Proceedings of MASCOT11 - Meetings on Applied Scientific Computing and Tools*, Rome, 2011.
35. P. J. Frey and P. L. George, *Mesh Generation : Application to Finite Elements*: Wiley, 2008.
36. P. Labbé, J. Dompierre, M.-G. Vallet, and F. Guibault, "Verification of three-dimensional anisotropic adaptive processes," *International Journal for Numerical Methods in Engineering*, pp. n/a-n/a, 2011.
37. M. G. Vallet, C. M. Manole, J. Dompierre, S. Dufour, and F. Guibault, "Numerical comparison of some Hessian recovery techniques," *International Journal for Numerical Methods in Engineering*, vol. 72, pp. 987-1007, 2007.
38. J. Dompierre, M.-G. Vallet, P. Labbé, and F. Guibault, "An analysis of simplex shape measures for anisotropic meshes," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, pp. 4895-4914, 2005.
39. P. Labbé, J. Dompierre, M.-G. Vallet, F. Guibault, and J.-Y. Trpanier, "A universal measure of the conformity of a mesh with respect to an anisotropic metric field," *International Journal for Numerical Methods in Engineering*, vol. 61, pp. 2675-2695, 2004.
40. N. Aspert, D. Santa-Cruz, and T. Ebrahimi, "MESH: measuring errors between surfaces using the Hausdorff distance," in *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on*, 2002, pp. 705-708 vol.1.
41. J. Ramsay and S. Parent, "Is Breast Asymmetry Present in Girls with Adolescent Idiopathic Scoliosis?," *Spine Deformity*, submitted 2013.
42. (2014). *Achieva 3.0 T Breast coils*. Available: http://www.healthcare.philips.com/main/products/mri/options_upgrades/coils/achieva3T/coils_breast.wpd
43. P. A. Yushkevich, J. Piven, H. C. Hazlett, R. G. Smith, S. Ho, J. C. Gee, and G. Gerig, "User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability," *NeuroImage*, vol. 31, pp. 1116-1128, 2006.
44. (2013). *ParaView*. Available: <http://www.paraview.org/>