Tetrahedral Image-To-Mesh Conversion for Anatomical Modeling and Surgical Simulations

Fotis Drakopoulos and Nikos Chrisochoides
Department of Computer Science, Old Dominion University, Norfolk, VA
fdrakopo@cs.odu.edu and nikos@cs.odu.edu

Keywords: Mesh generation, FEM, Biomechanical model, SOFA, 3D Slicer, Surgical simulations

Abstract

We present an Image-To-Mesh Conversion method for building a realistic biomechanical model particularly targeted for surgical simulations. Our implementation generates tetrahedral meshes that conform to the physical boundaries of multi-label segmented images. Our approach, initially creates a Body-Centered Cubic (BCC) lattice that is a coarse approximation of the object boundaries, and then subdivides the lattice using a red-green refinement strategy that guarantees the high quality of the new elements. In a later step, our method deforms the lattice surfaces to their corresponding tissue boundaries using a point-based registration scheme. As a result, the final mesh is smooth and accurately represents the object boundaries allowing a faithful response of the biomechanical properties of the tissues involved in a surgical simulation. Besides, the generated mesh is adaptive with smaller elements in areas where more detail is desired and larger elements in the remainder of the image regions. We evaluate our method qualitatively and quantitatively on isotropic and anisotropic segmented volumetric images. The described implementation will be available within two popular open source software: the 3D Slicer for visualization and image analysis, and the SOFA framework for real-time medical simulations.

1. INTRODUCTION

Mesh generation of medical images is necessary for biomechanical surgical simulation of the brain tissue deformation [10, 15, 14], and other applications like intra-operative non-rigid registration [8, 9, 7]. Although numerous mesh generation methods have been described to date, there are few which can deal with medical data input [5, 2]. Even fewer algorithms have been implemented and evaluated. Software packages that can produce high quality meshes are usually commercial [3, 1].

Mesh generation for medical imaging applications is complicated by the absence of the precise description of the object geometry, which is required by most of the traditional approaches to mesh generation. A robust Image-To-Mesh Conversion method has to satisfy the following requirements: (1) it should work directly with medical data (segmentations or greyscale images); (2) meshes must conform to the region of interest and have good quality (e.g., we can use the minimal dihedral angle of a tetrahedron to evaluate its quality); (3) the algorithm should be capable of producing adaptive meshes; (4) simulation procedures require the algorithm to be very fast.

Some groups [4, 12] presented multi-tissue mesh generation methods based on a Delaunay refinement. However, elements with small dihedral angles (slivers) are likely to occur in Delaunay meshes because elements are removed only when the radius-edge ratio is large. Their dihedral angle quality is completely ignored.

Others [16] presented an octree-based method to generate tetrahedral and hexahedral meshes. This method first identifies the interface between two or more different tissues and non-manifold vertices on the boundary. Then, all tissue regions are meshed with conforming boundaries simultaneously. Finally, edge contraction and geometric flow schemes are used to improve the quality of the tetrahedral mesh.

In this paper we present an Image-To-Mesh Conversion method based on [13, 11], suitable for surgical simulation of highly deformable bodies and anatomical modeling of complex structures like brain Arteriovenous Malformations (AVM).

Our implementation covers a suitable bounding box of the object with a uniform Body-Centered Cubic (BCC) lattice. Next, it subdivides the lattice using a red-green refinement scheme to regularly (red) split any tetrahedra where more resolution is required, and then irregularly (green) split any tetrahedra in such a way to restore the mesh to the state of being a valid simplicial complex. Once the refinement procedure is complete, the topology of the candidate mesh is finalized, and the tetrahedra which are completely outside of the object are discarded. The obtained BCC mesh has a guaranteed quality (minimum dihedral angle equal to 30°), however, it is not an accurate approximation of the image boundaries, and it is not smooth. For those reasons, the surfaces of the BCC mesh are iteratively deformed to their corresponding image boundaries using a customized point-based registration scheme. The mesh after the deformation: (1) accurately conforms to the image boundaries, (2) is non-uniform (adaptive), and (3) is smooth allowing a certain degree of visual reality during a surgical simulation. Next, we will describe the basic

steps of the method and present evaluation results from volumetric isotropic and anisotropic multi-label segmented data.

2. METHOD

Our approach requires a multi-label segmented image as input, in which label zero denotes the background, and positive integers indicate different tissues. The image can have uniform (isotropic) or non-uniform (anisotropic) spacing. Our implementation consists of two steps: (1) BCC mesh generation and refinement, and (2) Mesh deformation.

2.1. BCC mesh generation and refinement

This step initially creates a homogeneous BCC lattice and then refines the tetrahedra that cross the physical boundaries of the image. The lattice is refined until the desired fidelity (Table 2) is reached. The higher the fidelity, the larger the mesh.

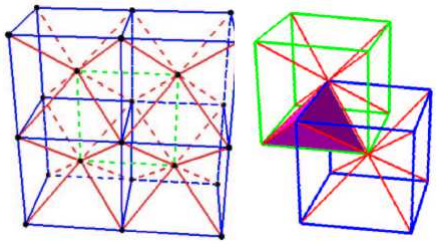
The BCC lattice is an actual crystal structure ubiquitous in nature. It is highly structured and easily refined before or during the simulation [13]. The vertices of the BCC grid are points of two interlaced grids like the blue and the green in Figure 1(a). The edges of BCC lattice consist of edges of the grid and additional edges between a vertex and its eight nearest neighbors in the other grid. The parameter BCCsize (Table 2) determines the distance between the vertices of the BBC grid.

The BCC refinement is performed by a red-green strategy. Initially, all BCC lattice tetrahedra are labeled with a red color. A red tetrahedron can be subdivided into eight children (1:8 regular refinement), and each child is labeled with a red color, as shown in Figure 1(b). There are three choices for the internal edge of the tetrahedron. If the shortest one is selected, the resulting eight child tetrahedra are exactly the BCC tetrahedra except the size is one half of the original BCC. So, the quality of the refined mesh can be guaranteed using this red (regular) subdivision. The red subdivision will lead to T-junctions at the newly-created edge midpoints where neighboring tetrahedra are not refined to the same level. To remove the T-junctions, a green (irregular) subdivision, including the three cases depicted in Figure 1(b), is performed.

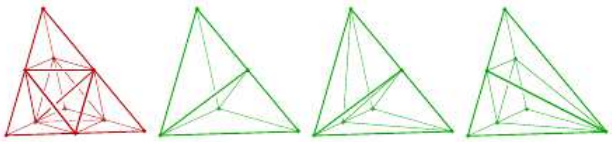
In a multi-label image, a tissue is defined by a set of voxels with the same intensity. Heuristically, the closer the surface of a sub-mesh is to the boundary of a tissue, the more voxels of the tissue are located in the sub-mesh, and the more voxels with the same label this sub-mesh has. To quantitatively evaluate the similarity between the sub-mesh and the tissue region, we define the voxel set S_1: all voxels in the sub-mesh, and $S2$: all voxels in the tissue region. $S_1 \cap S_2$ defines the point set shared by the sub-mesh and the tissue region. We expect the common region to be similar with the sub-mesh and the tissue region. We use the ratio $F_1 = \frac{S_1 \cap S_2}{|S_1|}$ to measure the similarity between the common region and the sub-mesh, and $F_2 = \frac{S_1 \cap S_2}{|S_2|}$ to measure the similarity between the common region and the tissue region. So, the refinement criterion can be defined as:

$$\text{Refine the mesh if: } F_1 < F \text{ or } F_2 < F \tag{1}$$

where $F \in (0,1]$ is the input fidelity listed in Table 2. In the case where the resolution of the input image is very low or the size of the element is very small compared to the voxel size, the number of the voxels inside an element might be zero. For this reason, up-sampling is performed automatically if no voxels are detected in a tetrahedron. To improve the performance, we do not perform up-sampling in the whole image, but restrict it to the bounding box of the tetrahedron.



(a) An example of a BCC lattice. The blue and green edges illustrate the two interlaced grids. The eight red edges lace the two grids together.



(b) Red-Green refinement.

Figure 1. BCC lattice generation and refinement. The figures come from [13].

2.2. Mesh deformation

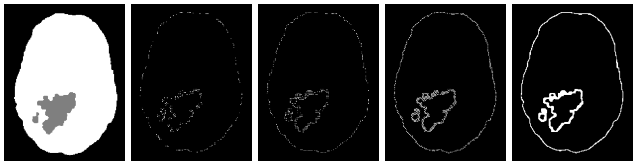
The produced BCC mesh consists of nicely shaped elements. However, it does not provide a certain degree of visual reality during a surgical simulation, because first, it is not an accurate approximation of the image boundaries, and second, it is not smooth. This step overcome those problems by deforming the surfaces of the mesh to their corresponding physical image boundaries.

The mesh deformation is facilitated by a point-based registration algorithm performed between two point sets: a source and a target point set. The source points are the surface verices of the BCC mesh. The target points are the edge points in the multi-label image. We obtain the target points using a Canny edge detection filter [6]. The non-connectivity parameter (Table 2) avoids the selection of target points that are too close to each other and determines the density of the point set. Four patterns are available: vertex, edge, face, and no

non-connectivity (Figure 2). The vertex pattern extracts fewer target points from the image while the no non-connectivity pattern extracts the maximum possible number of points. The larger the number of target points, the more computationally intensive the registration.

The mesh deformation is iterative. The larger the number of iterations, the smoother the mesh. The number of iterations is determined by the user (Table 2). Our experimental evaluation shows that typically, up to ten iterations are adequate to make the mesh look visually appealing. Figure 3 depicts an example of a nidus mesh during deformation.

At iteration i of the mesh deformation step, for each source point, its potential correspondence target point is located in the neighborhood of the source point. Then a biomechanical model is built from mesh i, the computed matching displacements are applied to the model, and a linear system of equations is solved to compute the displacements on the vertices of the mesh i. In the next iteration, the coordinates of the deformed source points at iteration $i+1$ are used to compute the correspondence between the source and the target point set. The biomechanical model is built from mesh $i+1$, and a new linear system of equations is solved. The registration algorithm that drives the mesh deformation is developed within the ITK[1] framework. A non-ITK version of this method was previously described at [11].



(a) Image (b) Vertex (c) Edge (d) Face (e) No

Figure 2. Non-connectivity patterns (vertex, edge, face, and no non-connectivity) for the extraction of the target points from the multi-label image.

3. SOFTWARE INTEGRATION

The described implementation will also be available within the open source SOFA[2] and Slicer[3] software.

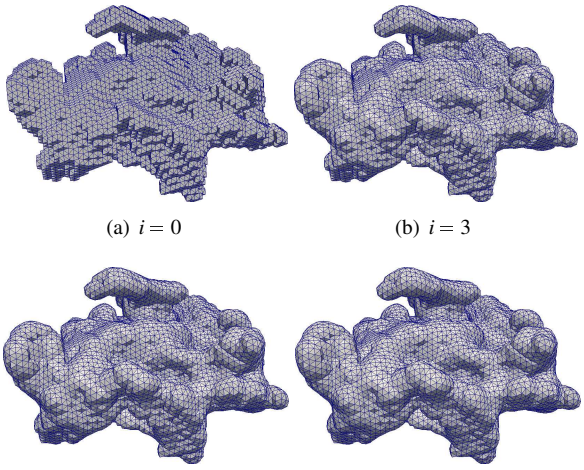
3.1. SOFA Plugin

SOFA is an open source framework primarily targeted at real-time medical simulations. Our Image-To-Mesh Conversion plugin is actually a shared library, loaded at runtime by SOFA. The purpose of the plugin architecture is to allow the developers to add their own components into SOFA, without having to integrate them into the heart of SOFA directories.

[1] http://www.itk.org/

[2] http://www.sofa-framework.org/

[3] http://www.slicer.org/



(a) $i = 0$ (b) $i = 3$

(c) $i = 7$ (d) $i = 10$

Figure 3. A nidus mesh during the deformation step. Iteration $i = 0$ corresponds to the un-deformed BCC mesh.

Figure 4 illustrates the scene graph of a SOFA simulation; our plugin (CBC3D) generates a tetrahedral mesh and a visual model of the mesh is created by mapping the output tetrahedral topology to the surface triangles topology. Figure 5 depicts the output tetrahedral visual model in SOFA's viewer.

Currently, our CBC3D plugin is used for the development of an interactive simulator[4] for neurosurgical procedures in SOFA, involving vasculature Arteriovenous Malformation (AVM). The development consists of anatomical modeling and volumetric meshing of vascular structures, combining a FEM biomechanical modeling with fluid simulation; coupling collision detection and response with haptic feedback; and integrating GPU-based implementations for real-time simulation. Figure 6 depicts some steps of the interactive simulator.

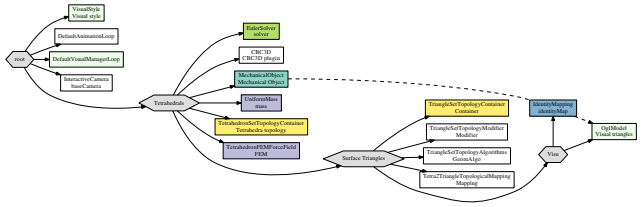


Figure 4. Scene graph of a SOFA simulation that uses the Image-To-Mesh Conversion plugin (CBC3D).

3.2. 3D Slicer Extension

The 3D Slicer is an open source software package for visualization and image analysis. The Slicer software supports

[4] Partnership between Kitware Inc.; the Center for Modeling, Simulation and Imaging in Medicine (CeMSIM) at Rensselaer Polytechnic Institute (RPI); the Departments of Computer Science and Neurosugery at the University of North Carolina (UNC); and Old Dominion University (ODU).

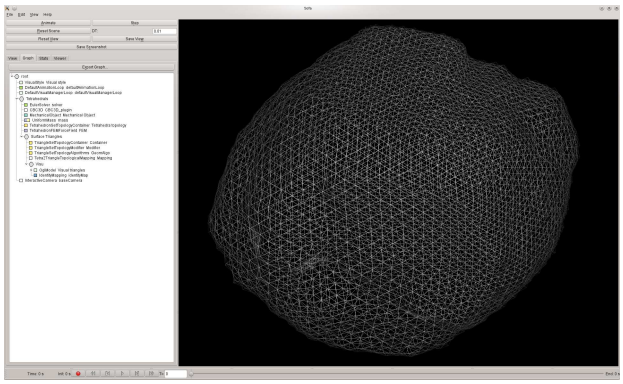
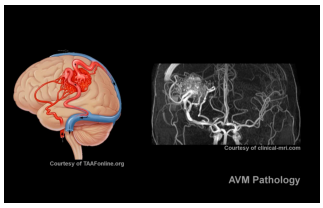
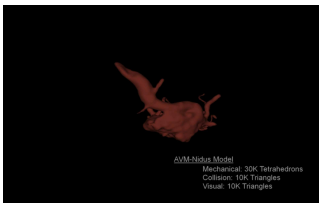


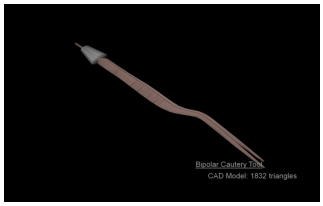
Figure 5. Visual model of the output of the CBC3D plugin, visualized in SOFA.



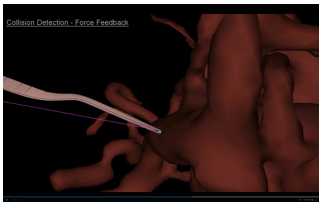
(a) AVM Pathology



(b) AVM model



(c) Cautery tool



(d) Collision detection

Figure 6. Interactive simulator for neurosurgical procedures in SOFA involving a brain Arteriovenous Malformation (AVM). The simulator uses the CBC3D plugin to generate the volumetric meshing of the vascular structures. Courtesy of Kitware Inc.

different types of extensions which can be built outside of the source tree and bundle together one or more modules. Currently, a single-tissue Image-To-Mesh Converion extension[5] is available in Slicer. We are planning to integrate the presented multi-tissue version soon. All of the available extensions can be downloaded from the Slicer server using the Extension Manager[6].

4. RESULTS

We evaluated our method on three volumetric multi-label segmented images:

- Case 1: A brain with a tumor (nidus).
- Case 2: A brain with a set of ventricles.
- Case 3: A brain with an Arteriovenous Malformation (AVM).

Table 1 lists some of the image properties. Case 1 is anisotropic and cases 2-3 are isotropic. Table 2 lists the input parameters for the experiments. We chose a BCC lattice spacing that is approximately 8-10 times the image spacing. The mesh fidelity $\in (0,1]$ is set equal to 0.95 for a more accurate representation of the tissue boundaries. The non-connectivity and the number of iterations are set equal to the face pattern and ten, respectively, for smoothing the mesh within a reasonable amount of time. Figures 7(a)-7(b), 8(a)-8(b) and 9(a)-9(b) depict cross sections of the generated tetrahedral meshes for cases 1, 2, and 3, respectively. In the cross sections, we show the conformity between the brain sub-mesh (green) and the other sub-mesh (red) by visualizing them simultaneously. Generally, our method guarantees the conformity between an arbitrary number of sub-meshes.

Table 1. The volumetric labeled image data of this study. x: axial; y: coronal; z: sagittal.

Case	Image spacing $sp_x \times sp_y \times sp_z$ (mm)	Image size $s_x \times s_y \times s_z$ (voxels)
1	$0.86 \times 0.86 \times 0.73$	$256 \times 256 \times 185$
2	$1.00 \times 1.00 \times 1.00$	$240 \times 240 \times 134$
3	$0.88 \times 0.88 \times 0.88$	$256 \times 256 \times 104$

Table 2. The input mesh parameters for all the experiments.

Parameter	Value	Description
BCCsize	8	grid spacing in mm
Fidelity	0.95	fidelity to tissue boundaries
Non-connectivity	face	pattern for the target points extraction
NumIterations	10	iterations for the mesh deformation

Figures 7(c), 8(c), and 9(c), depict the conformity of the mesh surfaces to the physical boundaries of the image. White and gray represent the different image labels. The green line represents the intersection between the brain sub-mesh and the image slice. The red line represents the intersection between the other sub-mesh and the image slice. We depict the Image-To-Mesh conformity on three different image slices: axial, sagittal, and coronal. According to Figures 7(c), 8(c), and 9(c), the multi-tissue meshes are an accurate representation of the multi-label image boundaries.

Table 3 presents some quantitative results of this study. The complex brain-AVM geometry requires a larger number of elements (about 1.2 million) to satisfy the input fidelity (0.95), compared to the other two cases. Currently, our implementation guarantees a minimum and a maximum dihedral angle

[5] http://www.slicer.org/slicerWiki/index.php/Documentation/Nightly/Extensions/CBC_3D_I2MConversion

[6] https://www.slicer.org/slicerWiki/index.php/Documentation/Nightly/SlicerApplication/ExtensionsManager

only for the BCC mesh generation-refinement step. Because of the nature of the BCC lattice and its specific red-green refinement procedure, the generated BCC mesh has a superior quality, with a minimum and a maximum dihedral angle equal to 30 and 116.5 degrees, respectively. On the other hand, our method does not provide any angle guarantees for the mesh deformation, thus it deforms the tetrahedra located nearby the image boundaries in favor of a better Image-To-Mesh conformity and a smoother mesh surface. We are planning to bound the angles of the mesh deformation step in the future.

Table 4 presents some performance results. Our sequential implementation can generate and refine a BCC mesh consisting up to about 1.2 million tetrahedra in less than 3 minutes. The mesh deformation step is the most computationally intensive and a smaller number of iterations or a different non-connectivity pattern can drastically reduce the execution time. In the future we will parallelize both steps of our method in order to provide suitable, realistic tetrahedral meshes appropriate for surgical simulations, in real-time.

Table 3. Quantitative evaluation results.

Case	# Tetrahedra	# Vertices	Min dihedral angle	Max dihedral angle
1	497928	88907	0.22°	179.66°
2	541943	95801	11.19°	166.18°
3	1222551	257570	0.20°	179.68°

Table 4. Performance results (in seconds). The experiments conducted in a workstation with 8 Intel i7-2600@3.40 GHz CPU cores and 16 GB of memory.

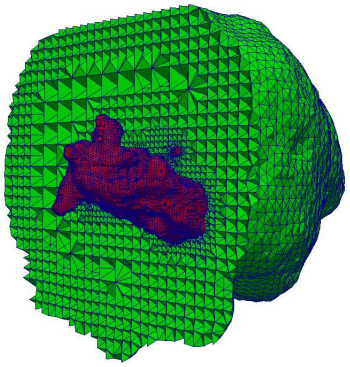
Case	BCC generation-refinement	Mesh deformation	Total
1	42.53	236.77	283.63
2	20.49	173.79	198.82
3	157.01	838.55	1006.47

5. SUMMARY AND CONCLUSION

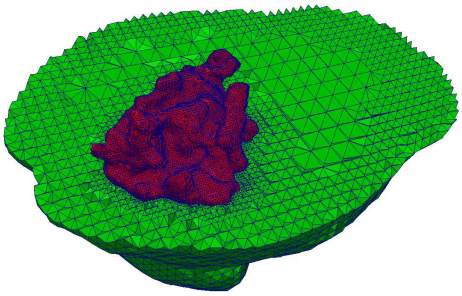
We presented an Image-To-Mesh Conversion method for realistic anatomical modeling of complex brain structures and surgical simulations. Our method is a heuristic, which is using implicit representation of the object as input, and produces a tetrahedral mesh specifically suited for applications that exhibit high deformation. Our approach, initially generates a structured BCC mesh from a multi-label segmented image. Then it refines the mesh using a red-green subdivision strategy that guarantees the high quality of the produced elements. In a later step, it deforms the surfaces of the BCC mesh to their corresponding physical image boundaries.

The produced mesh: (1) is adaptive, with smaller elements in areas where more detail is desired (tissue boundaries/interfaces) and larger elements in the remainder of the image regions, (2) it accurately approximates any complex image boundaries, allowing a faithful, realistic response of the biomechanical properties of the tissues involved in a simulation, and (3) is smooth, providing a certain degree of visual reality during a surgical simulation.

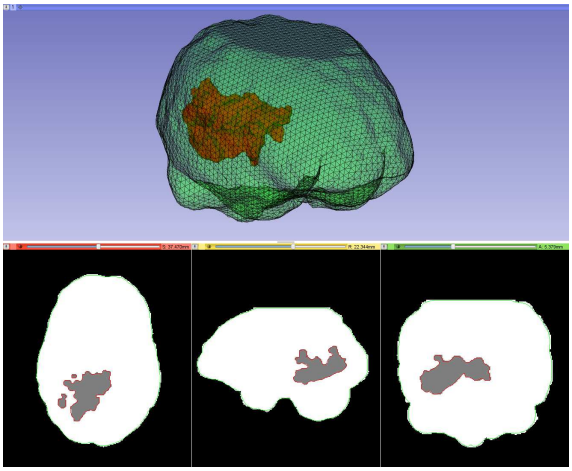
In the future, we will incorporate multiple fidelities in the



(a) Sagittal cross section.

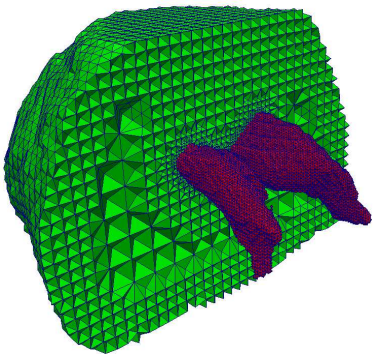


(b) Axial cross section.

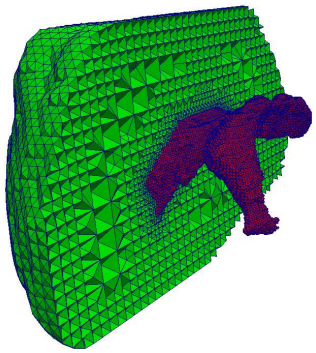


(c) Image-To-Mesh (brain-nidus) conformity, shown in axial (left), sagittal (middle), and coronal (right) slices. The green line represents the intersection between the brain surface mesh and the brain boundary in the image slice. The red line represents the intersection between the nidus (interior) surface mesh and the nidus boundary in the image slice.

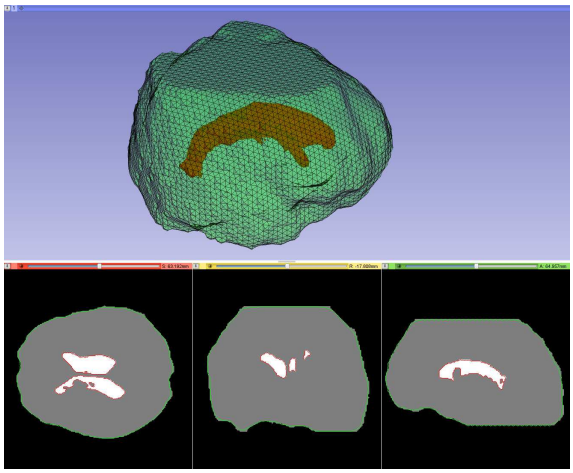
Figure 7. Brain-nidus mesh (case 1). Number of tetrahedra: 497928.



(a) Sagittal cross section.

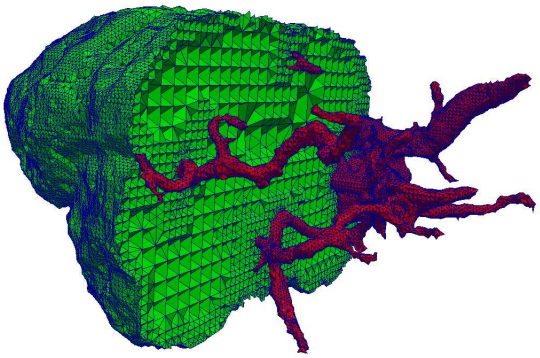


(b) Coronal cross section.

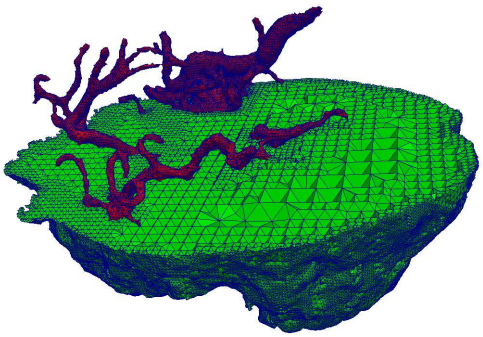


(c) Image-To-Mesh (brain-ventricles) conformity, shown in axial (left), sagittal (middle), and coronal (right) slices. The green line represents the intersection between the brain surface mesh and the brain boundary in the image slice. The red line represents the intersection between the ventricles (interior) surface mesh and the ventricles boundary in the image slice.

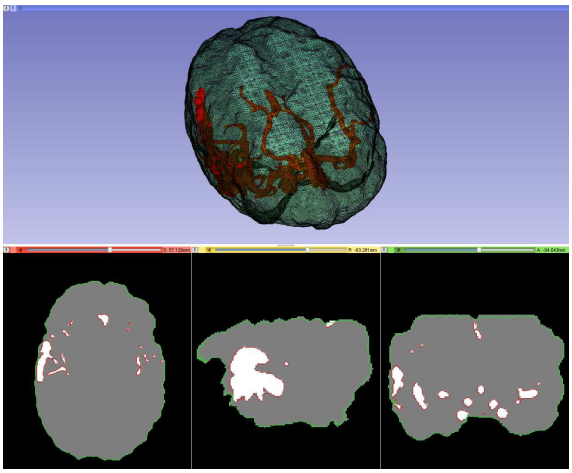
Figure 8. Brain-ventricles mesh (case 2). Number of tetrahedra: 541943.



(a) Sagittal cross section.



(b) Axial cross section.



(c) Image-To-Mesh (brain-AVM) conformity, shown in axial (left), sagittal (middle), and coronal (right) slices. The green line represents the intersection between the brain surface mesh and the brain boundary in the image slice. The red line represents the intersection between the AVM (interior) surface mesh of the AVM boundary in the image slice.

Figure 9. Brain-AVM mesh (case 3). Number of tetrahedra: 1222551.

mesh refinement step (e.g. different fidelity value for each tissue), and we will provide bounds for the minimum and maximum dihedral angles after the mesh deformation. Aditionally, we will parallelize our method to generate suitable tetrahedral meshes appropriate for surgical simulations in real-time.

ACKNOWLEDGEMENTS

Research reported in this publication was supported in part by the Modeling and Simulation Fellowship at Old Dominion University, the Office of The Director, National Institutes Of Health under Award Number R44OD018334 the NSF grants: CCF-1139864 and CCF-1439079 and by the Richard T.Cheng Endowment. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH and NSF. Special thanks for helping with edits and proofreading to Scott Pardue.

REFERENCES

[1] Amira. `http://www.vsg3d.com/amira/mesh`.

[2] Computational geometry algorithms library. `http://www.cgal.org/`.

[3] Simulog. `http://www.simulog.fr/`.

[4] Dobrina Boltcheva, Mariette Yvinec, and Jean-Daniel Boissonnat. Mesh generation from 3d multi-material images. In Guang-Zhong Yang, David Hawkes, Daniel Rueckert, Alison Noble, and Chris Taylor, editors, *Medical Image Computing and Computer-Assisted Intervention MICCAI 2009*, volume 5762 of *Lecture Notes in Computer Science*, pages 283–290. Springer Berlin Heidelberg, 2009.

[5] JonathanR Bronson, JoshuaA Levine, and RossT Whitaker. Lattice cleaving: Conforming tetrahedral meshes of multimaterial domains with bounded quality. *Proc Int Meshing Roundtable*, 2013:191–209, 01 2013.

[6] J Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, January 1986.

[7] O. Clatz, H. Delingette, I.-F. Talos, A. Golby, R. Kikinis, F. Jolesz, N. Ayache, and S.K. Warfield. Robust non-rigid registration to capture brain shift from intraoperative mri. *IEEE Trans. Med. Imag.*, 2005.

[8] M. Ferrant, A. Nabavi, B. Macq, F.A. Jolesz, R. Kikinis, and S.K. Warfield. Registration of 3-d intraoperative mr images of the brain using a finite-element biomechanical model. *Medical Imaging, IEEE Transactions on*, 20(12):1384 –1397, dec. 2001.

[9] A. Hagemann, K. Rohr, H.S. Stiehl, U. Spetzger, and J.M. Gilsbach. Biomechanical modeling of the human head for physically based, nonrigid image registration. *Medical Imaging, IEEE Transactions on*, 18(10):875 – 884, oct. 1999.

[10] Grand Roman Joldes, Adam Wittek, and Karol Miller. Suite of finite element algorithms for accurate computation of soft tissue deformation for surgical simulation. *Medical Image Analysis*, 13(6):912 – 919, 2009.

[11] Yixun Liu, Panagiotis Foteinos, Andrey Chernikov, and Nikos Chrisochoides. Mesh deformation-based multi-tissue mesh generation for brain images. *Engineering with Computers*, 28(4):305–318, 2012.

[12] Miriah Meyer, Ross Whitaker, Robert M. Kirby, Christian Ledergerber, and Hanspeter Pfister. Particle-based sampling and meshing of surfaces in multimaterial volumes. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1539–1546, November 2008.

[13] N. Molino, R. Bridson, J. Teran, and R. Fedkiw. A crystalline, red green strategy for meshing highly deformable objects with tetrahedra. In *12th Int. Meshing Roundtable*, 2003.

[14] S.K. Warfield, M. Ferrant, X. Gallez, A. Nabavi, F.A. Jolesz, and R. Kikinis. Real-time biomechanical simulation of volumetric brain deformation for image guided neurosurgery. In *Supercomputing, ACM/IEEE 2000 Conference*, page 23, nov. 2000.

[15] Adam Wittek, Grand Joldes, Mathieu Couton, Simon K. Warfield, and Karol Miller. Patient-specific non-linear finite element modelling for predicting soft organ deformation in real-time; application to non-rigid neuroimage registration. *Progress in Biophysics and Molecular Biology*, 103(2 - 3):292 – 303, 2010.

[16] Yongjie Zhang, Thomas J.R. Hughes, and Chandrajit L. Bajaj. An automatic 3d mesh generation method for domains with multiple materials. *Computer Methods in Applied Mechanics and Engineering*, 199(58):405 – 415, 2010. Computational Geometry and Analysis.