TETRAHEDRAL MESH GENERATION FROM VOLUMETRIC BINARY AND GRAY-SCALE IMAGES

Qianqian Fang and David A. Boas

Martinos Center for Biomedical Imaging, Massachusetts General Hospital, Charlestown, MA 02129

ABSTRACT

We report a general purpose mesh generator for creating finite-element surface or volumetric mesh from 3D binary or gray-scale medical images. This toolbox incorporates a number of existing free mesh processing utilities and enables researchers to perform a range of mesh processing tasks for image-based mesh generation, including the raw image processing, surface mesh extraction and re-sampling, and multi-scale/adaptive tetrahedral mesh generation. Atomic meshing utilities for each processing step can be accessed with simple interfaces which can be streamlined or executed independently. The toolbox is compatible with Matlab or GNU Octave. We demonstrate the applications of this toolbox for meshing a range of challenging geometries including complex vessel network, human brain and breast.

Index Terms — **Mesh generation, Finite element analysis, Medical image**

1. INTRODUCTION

Multi-modal imaging has become increasingly important in medical research and clinical practices, particularly in cancer detection, therapeutic monitoring and brain functional imaging [1]. A critical paradigm of multi-modality imaging is the fusion of structural imaging modalities with functional imaging modalities. By combining the high-resolution image of patient anatomy with physiological-relevant functional images, we can achieve a higher sensitivity and/or specificity than performing the structural or functional imaging separately. PET/CT is an excellent example following this paradigm.

The concept of multi-modality imaging has proliferated into a wide variety of novel imaging methods, for example, the combined digital breast tomosynthesis (DBT) with diffuse optical tomography (DOT), concurrent MR or ultrasound imaging with DOT, MR guided elastography, combined X-ray electrical impedance tomography, combined CT and florescence imaging, and so on. In many of these cases, a modal-based image reconstruction is required to produce the functional image, which is typically done by solving partial differential equation, i.e. the forward model, by numerical methods. Finite-element method (FEM) was found to be a popular choice for solving these forward models.

An important step in finite-element-based modeling is the mesh generation. In the context of multi-modal analyses, this typically requires to produce tetrahedral mesh or triangular surfaces from structural images. Unfortunately, software tools for this purpose is very limited both in choices and functionalities. There exist a few commercial software packages which provide similar functionalities, such as Amira (Mercury Systems, MA, USA), Mimics (Materialise, NJ, USA) and Simpleware (Simpleware Ltd., UK), however, the licenses for these software are usually expensive. In addition, these commercial software are typically shipped in the form of stand-alone packages, and it is difficult to incorporate and automate in a image-based modeling work-flow. A number of publications have reported algorithms or utilities for similar purposes [2,3], however, these software are either not accessible or not scalable for general use.

In this paper, we present a free 3D mesh generation toolbox. It incorporates a number of free mesh processing utilities, and is capable of producing quality 3D tetrahedral mesh or triangular surface directly from binary, segmented or gray-scale medical images. The software is written in Matlab (MathWorks, MA, USA) language and is compatible with both Matlab and GNU Octave (a free clone of Matlab).

In the following sections of the paper, we first present the overall mesh generation work-flow of the software, and then discuss the implementation details of each module. In the Results section, we demonstrate the meshing outputs for a number of medical applications, such as a complex vessel network from Two-photon microscopy, a segmented human brain from MRI scan and a human breast DBT image. We conclude the paper with a brief summary and our plans for future development of the software.

2. METHODS

Due to the great diversity in either the natures of the structural images or the types of modal-based analyses, the focus and needs of image-based mesh generation may vary significantly between applications. In correspondence, we structured this mesh generator in a modular fashion: the mesh generation tasks were break-down into atomic sub-tasks; each atomic subtask has a simple interface which can be called independently or cascaded with other atomic subtasks to achieve more complex goals; high-level wrapper

subroutines were also provided which can perform fully automatic mesh generation by calling the related atomic tasks subsequently.

2.1. Mesh generation work-flow

The overall mesh generation work-flow of this package is shown in a block-diagram in Fig. 1. Each connection in the diagram denotes an atomic subtask which process and convert the image or mesh into the target state along the work-flow. Three highlighted paths represent the high-level processing wrapper subroutines provided by this package, which directly convert volumetric binary/gray-scale images into surface or volumetric meshes.
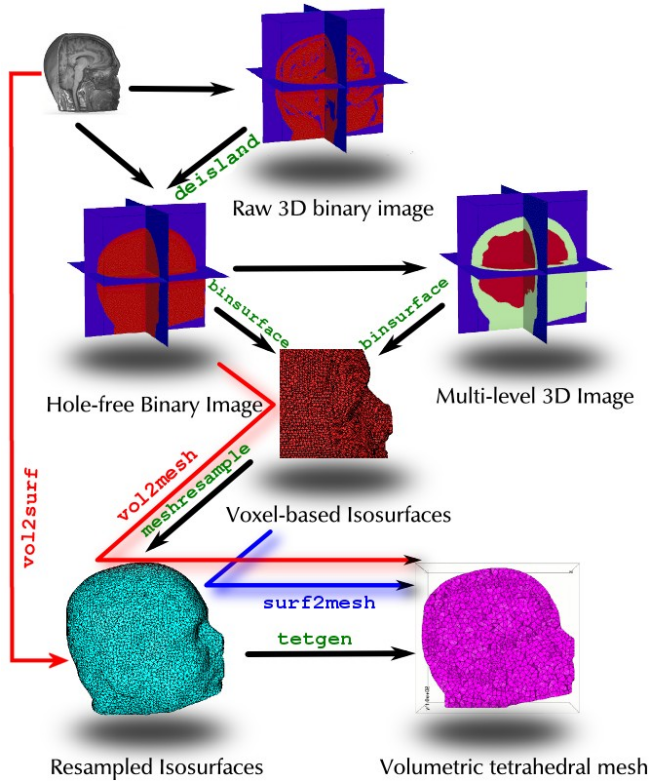

Fig. 1. Mesh generation work-flow diagram

The process of mesh generation can be roughly divided into two subsequent steps. In the first step, triangular iso-surfaces with specified densities are extracted from the input 3D image. If an iso-surface is not closed, we calculate the surface edges located on the bounding box of the volume and add piecewise linear complex (PLC) to form a closed surface. Optionally, a mesh repairing process is applied to the resulting surface to remove isolated vertices, duplicated elements and non-manifold vertices. In the second step, we fill tetrahedral elements for the sub-volumes bounded by the previously extracted iso-surfaces. Holes or sub-regional labels can be supplied so that the resulting FEM mesh may carry hollow structures or sub-domains that correspond to different tissue types.

2.2. Input volumetric images

Any 3D binary image denoted by a 0-1 valued array, segmented volume image containing multiple labeled 3D regions, or 3D gray-scale image can be used as the input of this mesh generator.

2.3. Surface mesh extraction

Two alternative approaches are available for surface mesh extraction: a surface simplification (SS) approach and a constrained Delaunay tetrahedralization (CDT) approach. In the former case, a triangular mesh (may include multiple disconnected surfaces) in voxel-resolution conforming to the inclusion boundaries is created first; a subsequent mesh simplification process is then applied to re-sample the surface mesh based on the user specified surface mesh density. In the later approach, a triangular surface will be directly generated from the input volumetric image. In this case, we used a modified surface extraction program build on a free library, CGAL [4], to perform the extraction.

2.4. Surface mesh repairing

For surface-simplification approach, the initial mesh and the mesh after re-sampling may contain various topological deficiencies, such as isolated vertices – vertices that are not contained by any surface element, duplicated triangles – multiple triangles that shares the same vertices, and non-manifold vertices [5]. These deficiencies will become significant difficulties to proceed further with tetrahedral mesh generation, therefore, it is important to remove these before the 3D element population.

Simple scripts were created to remove the isolated vertices and duplicated triangles. An external tool, build on a free library, JMeshLib [5], is used to find the non-manifold vertices and repair them by replication. This mesh check-and-repair procedure is called before and after the mesh re-sampling step. If the CDT approach is used for surface mesh extraction, this step is not necessary as CDT method ensures a topologically well-posed surface.

2.5. Sub-region labeling and hole specifications

Because the input volume may contain multiple subregions, such as in a segmented anatomical image. To correctly tag the resulting 3D tetrahedral elements with the associated region is of great importance for the subsequent FEM analysis. Fortunately, the external tool we interfaced with (see Section 2.8) supports sub-region labeling. The key is to specify an interior point for each sub-region.

The determination of an interior point for each subregion in a volumetric image is not trivial. Because the extracted iso-surfaces are not necessarily convex. The centroids of these surfaces may locate outside of the enclosed domain; even all the surfaces are convex, the centroid method may still fail

when one surface encloses one or more other surfaces. Furthermore, if one determines a set of interior points from the gray-scale image, we need to ensure that these points have reasonable distances to the boundaries of the subregions, as after surface extraction, these subregions will have slightly altered domain boundaries.

We propose a distance-field method to robustly determine an internal point for each sub-domain. We first find all the edge voxels of each sub-domain (segment), and set them to 1 and 0 elsewhere. Then we apply a Gaussian smoothing kernel for N iterations to the resulting array. This will produce a 3D field where the values are inversely related to the distances from the domain boundaries. We subsequently mask the results by each image segment and select the voxel with the minimum field value as the interior point. Using this method, we can ensure that the resulting points are located strictly inside the specified subregions and are at least N voxels away from the boundaries. We found that N=3 or 4 works robustly for a wide variety of applications with only minimal additional computation.

In there exist hollow regions in the volumetric image, we simply tag these regions with a special label and apply the same algorithm to determine the interior points. These regions can be removed from the final tetrahedral mesh using their regional labels.

2.6. Making water-tight surfaces
It is important to make sure that the surfaces, from which the tetrahedral mesh will be generated, do not have openings or holes. An open surface may happen if users manually extract the surfaces using the utilities from the toolbox. Here we assume that the edges of the open-surfaces are all located inside the volume bounding box. In this case, piecewise linear complexes on the bounding box are needed to enclose the surfaces.

To produce water-tight surfaces, we first calculate all open edges in the surfaces and organize them into a list of disjoint loops. If a loop is located at the intersections of two or three bounding box faces, we further decompose this edge loop into sub-loops that conforms only to a single face. A regional exclusion algorithm is then applied to "punch" the bounding faces by the polygons enclosed by the edge loops.

2.7. Surface mesh smoothing
We implemented 3 surface mesh smoothing algorithms to allow additional optimization of the surface mesh quality. These smoothing algorithms include a Laplacian operator, Laplacian-HC and Low-pass filter methods [6]. The later two methods have shown significantly improved volume conservation than the simple Laplacian operator [6].

2.8. Volumetric mesh generation and adaptive resolution
An external 3D mesh generator, tetgen [7], is called to create tetrahedral meshes from the closed surfaces generated previously. With the subregional labeling, this mesh generator is able to create quality 3D tetrahedral mesh at specified mesh density. Because tetgen generally does not alter the input surface mesh, the outer and subregional surfaces of the resulting mesh will contain all the nodes of the input surface. By controlling the surface and volume element density parameters, we can not only produce meshes with different global mesh densities, but also meshes with adaptive resolution to surface features. Examples of these meshes will be shown in the following section.

3. RESULTS

In this section, we intend to provide a show case for demonstrating the key features of this mesh generator. These example are taken from the up-to-date medical imaging research.

In the first example, we mesh a tissue volume containing a complex vessel network extracted from a two-photon microscopy scan of the somasensory cortex of a rat. A binary representation of the vessel network was created with a manual vessel tracking and segmentation process. An iso-surface of the vessel containing 92035 nodes and 183500 triangles is show in Fig 2(a). Applying the mesh simplification method to this surface mesh, we get a surface with 2032 nodes and 3484 triangles. As explained in Section 2.6, we close this open surface by adding PLCs on the volume bounding box. The cross-cut view of the closed
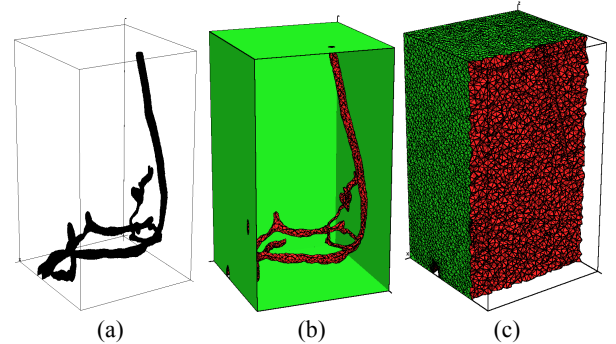

(a) (b) (c)
Fig. 2. Mesh generation for a vessel network: (a) vessel wall raw image, (b) water-tight surface mesh after re-sampling, and (c) volumetric mesh cross-section view.

surface is shown in Fig. 2(b). Finally, we mesh the bounded volumes of this surface to produce an FEM mesh, which has 49796 nodes and 284208 tetrahedra. The cross-cut view of the mesh is shown in Fig. 2(c).

The second example is a human brain MRI image with segmented gray and white matters. The segmentation was

computed using FreeSurfer [8]. A thresholded brain image contains a large number of disconnected regions within the domain. A hole-filling process was first applied to produce a single-connected region. The brain tissue segmentation was subsequently overlapped to produce a multi-regional volume. By using the "vol2mesh" utility with CDT extraction method, we first get a number of closed iso-surfaces for the skin, brain and white matter regions. The final volumetric mesh were subsequently produced where the tetrahedra in each tissue type were tagged correspondingly. The cross-cut views of the final mesh are shown in Fig 3.
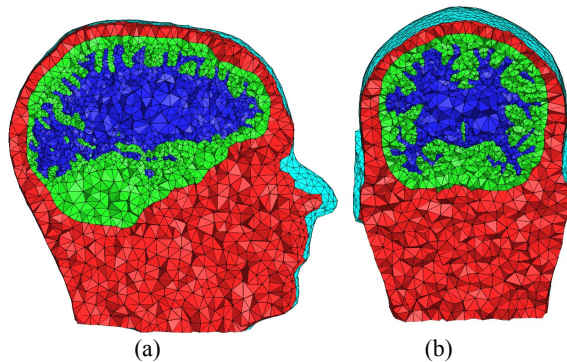

(a)   (b)

Fig. 3. FEM mesh from brain MRI image (two views)

The third example is taken from combined DBT and DOT breast imaging. A breast DBT scan is able to reveal 3D structure of adipose, fibroglandular tissue as well as breast tumors [1]. A finite-element based DOT reconstruction can produce hemoglobin concentration maps that may have diagnostic values. To create breast FEM mesh from DBT images, we first segment the breast to create a binary mask of the fibroglandular tissue. Using the segmented breast, we generated multiple breast meshes with progressively increasing mesh densities. These meshes are shown in Fig. 4, where the node numbers are 31263, 48275 and 100563 respectively. The total time for the volume-to-mesh conversion increased from 23 seconds to 43 seconds on a Core 2 Duo PC.
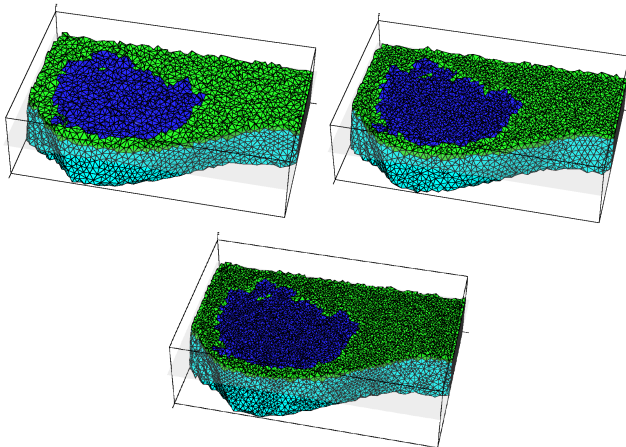

Fig. 4. Breast meshes with progressively smaller element sizes.

4. CONCLUSIONS

We have developed an open-source 3D mesh generator build with modularized processing scripts and free software components. The source code and binaries are available at our website [9]. By innovatively connecting and stream-lining a number of existing free meshing utilities, we built a powerful, efficient yet simple-to-use mesh generation toolkit. We tested this toolbox with a number of medical applications and it has demonstrated great robustness and simplicity in handling challenging structures such as meshing volumes of multiple regions, opened surfaces and structures expanding only a few voxels. This tool may greatly facilitate multi-modality data analysis, particularly for structural and functional image fusion. We will further extend the functionalities of this package and fine tune it to serve for more applications where 3D meshes are needed.

5. ACKNOWLEDGEMENT

6. REFERENCES

[1] F. S. Azar, X. Intes (Eds.), *Translational Multimodality Optical Imaging*, Artech House, Norwood, 2008

[2] O. Courchesne, F. Guibault, J. Dompierre and F. Cheriet, "Adaptive Mesh Generation of MRI Images for 3D Reconstruction of Human Trunk", Springer Berlin / Heidelberg, 2007

[3] A. J. Cuadros-Vargas, L. G. Nonato, R. Minghim and T. Etiene, "Imesh: An Image Based Quality Mesh Generation Technique," Computer Graphics and Image Processing, 2005. SIBGRAPI 2005. 09-12 Oct. 2005 Page(s): 341 – 348

[4] L. Rineau and M. Yvinec. "3D Surface Mesh Generation," In CGAL Editorial Board, editor, *CGAL User and Reference Manual*. 3.4 edition, 2008.

[5] M. Attene and B. Falcidieno, "ReMESH: An Interactive Environment To Edit And Repair Triangle Meshes", In Proc. of Shape Modelling Int. (SMI'06), pp. 271-276. 2006.

[6] R. Bade, J. Haase, B. Preim, "Comparison of Fundamental Mesh Smoothing Algorithms for Medical Surface Models", Simulation and Visualization, pp. 289-304, 2006.

[7] H. Si and K. Gaertner, "Meshing Piecewise Linear Complexes by Constrained Delaunay Tetrahedralizations". In Proc. of the 14th Int. Meshing Roundtable, pp. 147-163, Sep. 2005

[8] A.M., Dale, B. Fischl, M. I. Sereno, "Cortical surface-based analysis. I. Segmentation and surface reconstruction," Neuroimage 9, pp. 179-194, 1999

[9] iso2mesh: URL: http://iso2mesh.sourceforge.net