

# IMAGE-TO-MESH CONVERSION TOOL FOR IMAGE-DRIVEN SIMULATIONS

Joi Best<sup>1a</sup>, Kevin Garner<sup>a</sup>, Daming Feng<sup>a</sup>, Fotis Drakopoulos<sup>a</sup>, Yixun Liu<sup>a</sup>, and Nikos Chrisochoides<sup>a</sup>

<sup>a</sup>Center for Real-Time Computing Lab, Computer Science Department, Old Dominion University, Norfolk Va

---

## **ABSTRACT**

Grid Generation is a critical component for many (bio-)engineering and science applications. The goal of my research project is to improve upon the current Image-to-Mesh Conversion tool (I2MTool) created by the Center for Real-Time Computing (CRTC). The I2MTool could be proven beneficial in a spectrum of disciplines that spans from Multiscale Materials and Structures Modeling to Finite Element (FE) analysis for NASA's Computational Fluid Dynamics (CFD) 2030 Vision and Vision 2040 for Integrated Computational Structures Modeling. My project involves the development of visualization techniques to make models in simulations look more realistically, impacting research in nuclear physics, aeronautic simulations and health care applications. In this paper, I will present recent work on the CBC3D Slicer extension and the I2Mtool. CBC3D is one of CRTC's 3D image-driven grid generation software that can generate high quality tetrahedral meshes. CBC3D was implemented as an extension within the 3D Slicer package for visualization and image analysis. Some features that were added to the I2MTool include: a feature to display a multi-material (or tissue) labeled mesh, and a loading bar to increase the tool's user-friendliness. In the future, a feature to cut a cross-section of a mesh to view its interior will be implemented.

## **INTRODUCTION**

Finite Element Mesh Generation is an important component for many (bio-) engineering and science applications. Image-to-Mesh conversion tools are widely used for the quantitative analysis of images using the

Finite Element (FE) method [1]. The FE method is essential in modeling tissue deformation, brain tumors, and large-scale simulation of arterial blood flow for image-guided interactive surgery simulations. The goal of this research project is to improve upon the current Image-to-Mesh Conversion tool [2], created by the Center for Real-time Computing (CRTC) [3], for image-driven simulations. The two previously implemented modules added to the tool are Physics-Based Non-Rigid Registration and CBC3D. Physics-Based Non-Rigid Registration uses a linear homogenous (bio-) mechanical model to compute a dense deformation field that defines a transformation for every point in a fixed MRI image to a moving image [4]. CBC3D generates an adaptive multi-tissue tetrahedral mesh from a multi-label segmented image [1].

The CBC3D method targets the anatomical modeling of the Arteriovenous Malformation (AVM), i.e. tangled bundles of abnormal and thin blood vessels connecting arteries and veins in the brain, for surgical simulations. One important category of surgical simulation commonly used in the medical field is Interactive simulation. Interactive simulation plays a vital role in reducing health care costs for medical schools and hospitals because surgical residents are able to train using virtual environments without putting real patients at risk. Interactive simulations require real-time mesh generation to meet the time constraints imposed by end-to-end computations in near-haptic rates [1]. CBC3D is a reliable mesh generator that uses a segmented multi-labeled image as an input and generates graded meshes, which are useful during interactive simulations.

The Image-to-Mesh software should meet the requirements of three essential components for interactive simulations: (1) level-of-detail of tissue morphology, (2) accuracy of the representation of the geometries, and (3) visual reality [1]. Visualization techniques added to the Image-to-Mesh conversion tool will make models in simulations look more realistically, which will impact research in nuclear physics, aeronautic simulations and health care applications. Better visualization of these models will improve the quality of simulations that can be used to prepare individuals for real world tasks such as surgery.

### **RESEARCH OBJECTIVES**

Many visualization tools do not contain full functionality to display both mesh tessellations and images. 3D Slicer [5] is a software for image analysis, but the tool does not display tessellations. CBC3D [1] is one of CRTC's 3D image-driven grid generation software used in Medical Image Computing applications. It is composed of the Body Centric Cubic module that generates a mesh from an input labeled image and the Mesh Compression module that deforms the input tetrahedral mesh towards the boundaries of the input labeled image. In order to utilize the full capability of 3D Slicer, a CBC3D extension was added to measure the visualization techniques, however the tool still lacks the full functionality to display tessellations and images. There is no tool that combines the functionality to display tessellations and images, so the I2MTool was enhanced to incorporate these visualization features.

Although the CBC3D module generates multi-material meshes, the previous version of the I2MTool could not differentiate between various tissues by using different colors when displaying meshes. The tool could only display a single mesh with one solid color. The tool also required some improvements to its graphical user interface

(GUI) and features that would enhance its accessibility. The research objectives for the project were to implement several visualization techniques on the I2Mtool, more specifically:

O1: implement a feature to display a multi-tissue labeled mesh using different colors

O2: enhance the GUI to increase the tool's user-friendliness

O3: implement a feature that will allow the user to cut a cross-section of a mesh to view its interior

### **METHODS**

The objectives above were accomplished by utilizing three software programs: the GUI source code (QT) [6], the Visualization Toolkit (VTK) [7], and the Insight Segmentation and Registration Toolkit (ITK) [8]. I used some existing algorithms implemented within the VTK library and integrated these methodologies within the I2MTool source code to modify the way colors are assigned to meshes based on label data within the mesh file. Figure 1 depicts the GUI of the I2MTool, which uses several modules (such as CBC3D, image resampling, affine registration, and more), visualization tools, saving, loading, and printing images/meshes.



*Figure 1: The GUI of the I2MTool allows for using several visualization tools*

A loading bar will be added to the GUI to show the progress of mesh generation and other methods, enhancing user-friendliness. ITK will be utilized in creating four distinct points while QT will be used to create a method that directly allows point manipulation (on the GUI) to create a cross section, where all of the mesh elements located on a certain side of the cross-section will be displayed (depending on the user's choice).

## RESULTS

A loading bar was added to the Image-to-Mesh Conversion tool to track the progress of mesh generation. As shown in Figure 2, the loading bar pops up in the middle of the screen and shows the progress in percentage. The percentage is based on the time it takes to read the input image, generate the mesh, warp the mesh, compress the mesh, and write the output mesh. Once, the loading bar reaches 100%, the mesh is outputted to the screen.

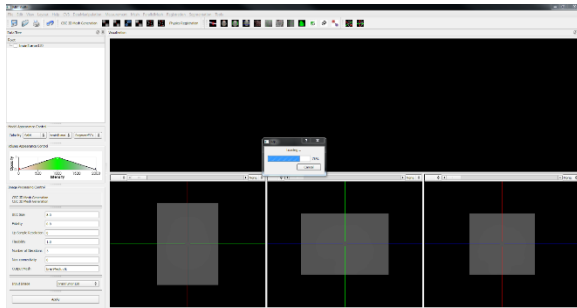


Figure 2: Loading bar within the I2MTool

A feature to display a multi-tissue labeled mesh using different colors was implemented by utilizing the CBC3D method and the Multiple Surface Mesh Generation method. Figure 3 shows an image of the I2MTool displaying a mesh using the CBC3D method. The GUI for this module can be seen on the left with the top panel showing the deformed mesh that this module generates.

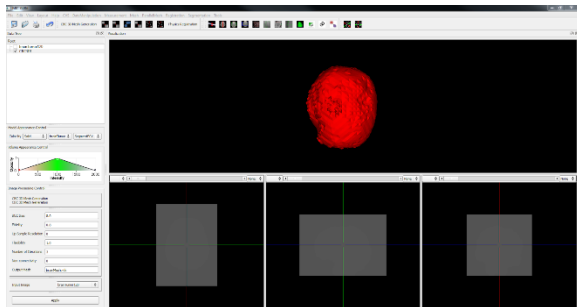


Figure 3: CBC3D method within the I2MTool

To display a multi-tissue labeled mesh using different colors, the Multiple Surface Mesh generation method shown in Figure 4

was used. This method generates the different mesh files based on labels.

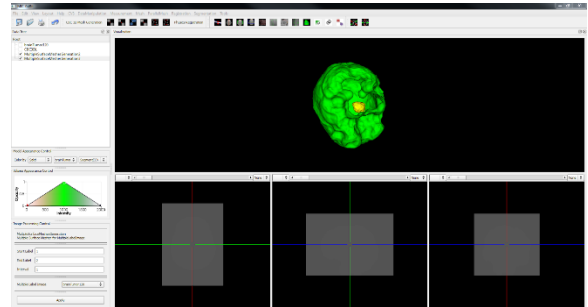


Figure 4: Multiple Surface Mesh Generation method within the I2MTool

The GUI for Multiple Surface Mesh Generation method can be seen on the left with the top panel showing the meshes that this module generates. The brain tumor in this particular test case is yellow and the brain is green. The CBC3D method in Figure 3 was then used to generate the mesh with the same labeled data. On the data tree, to visualize the multi-labeled mesh, the CBC3D mesh gets selected and the label of the tumor. This will display the brain mesh and the tumor as different colors as shown in Figure 5.

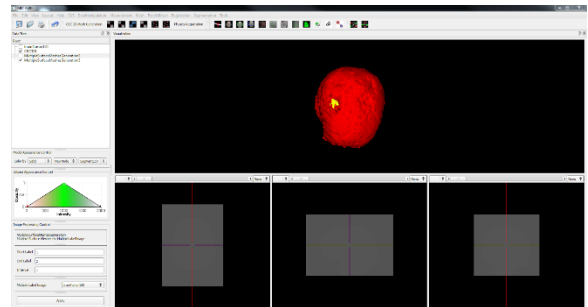


Figure 5: Multiple Surface Mesh Generation method combined with the CBC3D method within the I2MTool

The GUI for this module can be seen on the left with the top panel showing the meshes that this module generates. The brain tumor is yellow, and the brain is red. In the near future, I plan to implement a feature that will allow the user to cut a cross-section of a mesh to view its interior. Research will be done to learn how to cut the mesh based on

different planes. The window containing the mesh is where the user will be able to directly move the points of the cross-section and visualize the cuts.

Along with improving visualization features on the Image-to-Mesh Tool, work was done to implement a CBC3D extension on 3D Slicer. 3D Slicer is a package for visualization and image analysis from Harvard Medical School. The CBC3D 3D Slicer extension can generate high quality tetrahedral meshes from brain labeled images. The extension is separated into two methods: The Body Centric Cubic module and the Mesh Compression module.

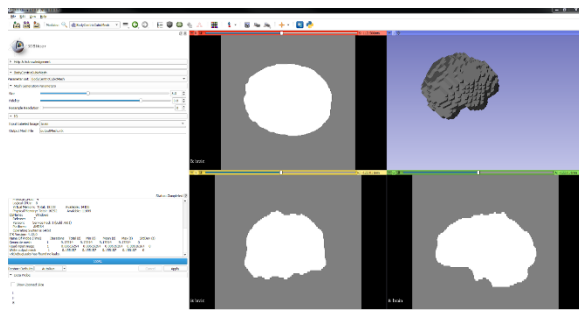


Figure 6: Body Centric Cubic Mesh Module within the CBC3D 3D Slicer extension

The Body Centric Cubic (BCC) module generates a mesh from an input labeled image [9]. As shown in Figure 6, the GUI for this particular module can be seen on the left with the top right panel showing BCC tetrahedral mesh that this module generates. The output of this module is then used as input for the Mesh Compression Module.

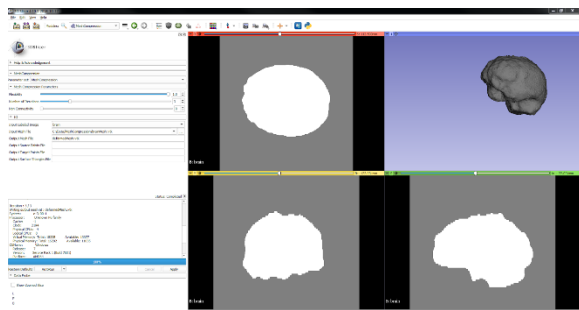


Figure 7: Mesh Compression Module within the CBC3D 3D Slicer extension

The Mesh Compression module deforms the input tetrahedral mesh towards the boundaries of the input labeled image [10]. As shown in Figure 7, the GUI for this particular module can be seen on the left with the top right panel showing the deformed mesh that this module generates. 3D Slicer has a feature to slice across a mesh.

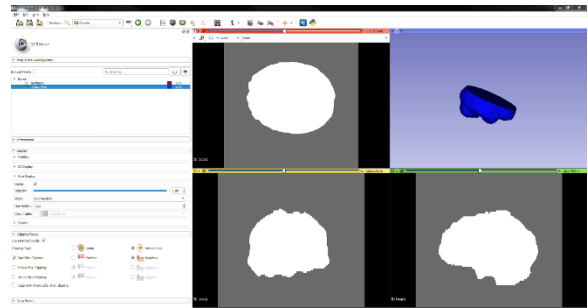


Figure 8: Slice Mesh Compression method mesh within the CBC3D 3D Slicer extension

As shown in Figure 8, the GUI for this particular module can be seen on the left, and the top right panel shows the newly registered image after being sliced through the X axis. The extension can currently work with Slicer 4.9 and the upcoming release of 3D Slicer version 4.10.1. The Slicer source code was built using cmake and QT5. Visual Studios 2015 was utilized to create the library executables for the Body Centric Cubic Mesh and Mesh Compression modules to be used on 3D Slicer. The CBC3D extension was added to measure the visualization techniques of 3D Slicer, however the tool still lacks the full functionality to display tessellations and images. The Image-to-Mesh Conversion tool will continue to get updated and improved to maintain the functionality to display tessellations and images to enhance the quality of simulations.

## CONCLUSION

Mesh generation is important for many (bio-) engineering and science applications. The goal of this research project is to improve upon the current Image-to-Mesh Conversion tool for image computing. A feature that displays a multi-tissue labeled mesh using different colors was implemented to view different tissue meshes. The GUI was enhanced by adding a loading bar. In the future, a feature to cut a cross-section of a mesh to view its interior will be implemented. This study has the ability to improve user interactions with the tool and can help with image-guided simulations. This research directly aligns with NASA's Space Technology Mission Directorate, which strives to "advance broadly applicable, transformational technology to infuse solutions into applications for which there are multiple customers" and "result in new inventions, new capabilities and the creation of a pipeline of innovators aimed at serving future national needs"[11]. This project involves the development of visualization techniques that serve to enhance end-user productivity with applications utilizing mesh generation and refinement. Models in simulations will look more realistically, impacting medical, military, and aeronautic simulations. This research not only utilizes mesh generation techniques that are proven to be efficient but will contribute to solving several problems in the Computational Fluid Dynamics (CFD) Vision for 2030 and Vision 2040 for Integrated Computational Structures Modeling.

## ACKNOWLEDGMENTS

Financial support was provided through a VSGC Undergraduate Research Scholarship Award. In addition, it is partially supported by the Computer Science Department and Richard T. Cheng Endowment. The content is solely the responsibility of the authors and does not necessarily represent the official views of the VSGC.

## REFERENCES

- [1] Drakopoulos, F., Ortiz, R., Enquobahrie, A., Sasaki-Adams, D., & Chrisochoides, N. (2014). Tetrahedral image-to-mesh conversion software for anatomic modeling of arteriovenous malformations. 24th International Meshing Roundtable, TX. Retrieved January 23, 2018.
- [2] Garner, K., Feng, D., Drakopoulos, F., Liu, Y., & Chrisochoides, N. Image-to-Mesh Conversion Tool. Retrieved January 23, 2018.
- [3] (2018). Center for real-time computing. Retrieved from <https://crtc.cs.odu.edu/index.php>
- [4] Liu, Y., Kot, A., Drakopoulos, F., Yao, C., Fedorov, A., Enquobahrie, A., ... Chrisochoides, N. P. (2014). An ITK implementation of a physics-based non-rigid registration method for brain deformation in image-guided neurosurgery. *Frontiers in Neuroinformatics*, 8, 33. doi:10.3389/fninf.2014.00033
- [5] 3D Slicer. Retrieved from <https://www.slicer.org/>
- [6] Qt | Cross-platform software development for embedded & desktop. Retrieved January 23, 2018 from <https://www.qt.io/>
- [7] VTK- The Visualization Toolkit. Retrieved January 23, 2018 from <https://www.vtk.org/>
- [8] ITK – Segmentation & Registration Toolkit. Retrieved January 23, 2018 from <http://www.itk.org>.
- [9] BodyCentricCubicMesh. Retrieved from <https://www.slicer.org/wiki/Documentation/Nightly/Modules/BodyCentricCubicMesh>
- [10] MeshCompression. Retrieved from <https://www.slicer.org/wiki/Documentation/Nightly/Modules/MeshCompression>
- [11] Hall, L. (2015, March 16). About Us. Retrieved January 23, 2018, from [https://www.nasa.gov/directorates/spacetech/about\\_us/index.html](https://www.nasa.gov/directorates/spacetech/about_us/index.html)