Google Summer of Code 2015 Proposal: *Easy Mesh Cutting* for BRL-CAD

Brad Eric Hollister

March 3, 2015

Abstract

Maintaining and improving BRL-CAD’s usability is a goal of the community. Although BRL-CAD’s modeling tools are considered one of its many strong suits, it would benefit both new and experienced users alike to have a more intuitive interface for mesh manipulation. One such method, *Easy Mesh Cutting* [1], is a relatively straight-forward application using the Sketch-based modeling approach [2]. I propose to integrate *Easy Mesh Cutting* into BRL-CAD.

1 Contact Info

1. **Name:** Brad Eric Hollister
2. **email:** behollis@soe.ucsc.edu, brad.hollister@gmail.com
3. **phone:** 864-395-5560
4. **physical address:** 102 Moore Street; Santa Cruz, CA 95060

2 Benefits to BRL-CAD

BRL-CAD will gain a very intuitive tool for mesh cutting. *Easy Mesh Cutting* will facilitate modeling for new and experienced users alike, thus attracting less experienced users while retaining current ones. By adding *Easy Mesh Cutting*, it is hoped that the stage will be set for incorporating similar sketch-based modeling tools in the future.

3 Deliverables

The implementation will consist of C++ code integrated with relevant modules. I have obtained the original source code from [1]. Modifying the original research code, instead of implementing it from scratch, will help to ensure the project’s success.
4 Project Details

Modelers will start by marking two parts of a mesh they intend to cut apart. Each part, a foreground sub-mesh and a background sub-mesh, is designated by a marking line on each part of the model. After doing this, each vertex from the entire mesh will be labeled as belonging to one or the other sub-mesh. The separation metric used in determining which sub-mesh a vertex belongs to is based on the minima rule and salience theory from cognitive vision theory [1].

The automatic separation seeks to find two disparate parts of the model as suggested by the modeler’s markings. Once the algorithm assigns all vertices to either sub-mesh, a cutting boundary for the two parts of the mesh is created. The user can then interactively re-position the boundary for the separation of the mesh.

Because mesh segmentation is based on an extended isophotic metric from [1], BRL-CAD will have functionality for calculating mesh surface features based on curvature. This functionality might additionally be used for mesh feature visualization.

For more information, see the Easy Mesh Cutting project page: http://www.math.zju.edu.cn/ligangliu/CAGD/Projects/SketchBasedMeshCutting/EasyMeshCutting/

5 Project Schedule

I can begin work on May 25th, when the coding period begins, and continue full-time until August 31st (total 14 weeks).

Milestones:

1. Implement mesh marking of sub-meshes, including ramping up knowledge on relevant modules. Total time: 2 weeks.
2. Implement isophotic metric calculations. Total time: 2.5 weeks.
3. Implement vertex assignments to sub-meshes. Total time: 2.5 weeks.
4. Implement cutting boundary refinement by snake. Total time: 3 weeks.
5. Implement user manipulation of cutting boundary. Total time: 1 week.
6. Implement mesh cut along cutting boundary. Total time: 1 week.
7. Implement refinements in GUI interface to tool, code, etc. Total time: 2 weeks.

I plan on working 40+ hours per week. I currently do not have any other conflicts with the proposed schedule. However, I may apply for a summer teaching assistantship which will require less than 20 hours per week.

6 Bio

I am currently a PhD candidate at the University of California at Santa Cruz. I have extensive professional programming experience in computer graphics. Additionally, I’ve worked on Draco, a solid geometry CAD system at Lawrence Livermore National Labs.

In the past, I have worked on Draco at the Lawrence Livermore National Labs: [http://people.ucsc.edu/~behollis/webcutImpl_hollister.final102ack.pdf](http://people.ucsc.edu/~behollis/webcutImpl_hollister.final102ack.pdf)

I successfully completed GSoC 2014. See project info here: [https://www.google-melange.com/gsoc/project/details/google/gsoc2014/brad_h/5639274879778816](https://www.google-melange.com/gsoc/project/details/google/gsoc2014/brad_h/5639274879778816)

I have also submitted a few patches to the Blender open source project username: `brad_h@developer.blender.org`. I plan to help out further with bug fixes when time permits.

My research field is the visualization of ensemble vector fields. You can view my prior research and thesis proposal here: [http://people.ucsc.edu/~behollis/hollister_prop.pdf](http://people.ucsc.edu/~behollis/hollister_prop.pdf)

**References**
