Editors: Gitta Domik and Scott Owen

A Summer Blender Camp

Modeling, Rendering, and Animation for High School Students

Mike Bailey and Cathy Law
Oregon State University

In June 2013, we conducted Camp Blender for 21 high school students, under the auspices of Oregon State University's STEM Academy (http://academy.engr.oregonstate.edu). The academy is a nonprofit, extracurricular, precollege education program emphasizing science, technology, engineering, and math (STEM). It enlists local professionals, faculty, and students to share their facilities, equipment, and expertise through hands-on camps, classes, workshops, and in-school outreach.

Blender (www.blender.org) is an open source modeling, rendering, and animation software package. It occupies an interesting intersection of engineering, mathematics, art, and animation and thus fits perfectly with the STEM Academy's goals and clientele. As the Blender website says, Blender represents "3D for all." We became interested in Blender for a STEM Academy class when we realized it would create a unique watering hole for high school students interested in math, engineering, art, or animation. Blender also appeals to a variety of ages. You don't need much math to understand its 3D-modeling concepts. So, in this watering hole, we expected that the students would learn not only from us but also from each other's varied skill sets.

The Students

We were correct about the class appealing to a variety of ages. After the students registered, we had a mix of ages, but unexpectedly skewed to sophomores (see Figure 1).

A precamp survey of the students revealed an interesting variety of experience. Some students had graphics application experience. The most common applications were GameMaker (http://yoyogames.com) and SketchUp (www.sketchup.com), each of which had been used by just under half of the students. The other half had no graphics application experience. About one-third had programming-language experience.

Topics

The camp consisted of three hours of class (9:00 am to noon) over five days. We couldn't cover everything about Blender, so we created this reduced outline:

1. Navigating the Screen Layout.
2. Viewing in 3D.
3. Moving Things Around in 3D.
4. Modeling.
5. Appearance I.
6. Rendering.
7. Appearance II.
8. Particle Systems.
9. Physics Animation.
10. Keyframe Animation.

The first three sections explained the Blender interface and fundamental computer graphics

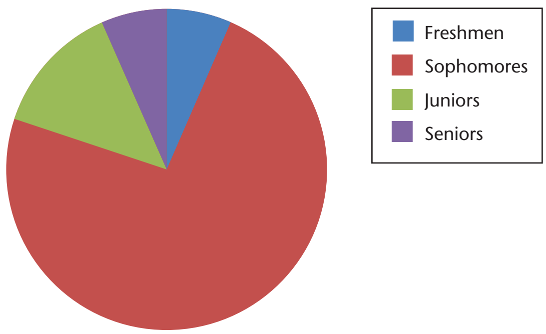

Figure 1. The grade distribution of Blender Campers. We ended up a mix of ages, but unexpectedly skewed to sophomores.

 0272-1716/14/$31.00 © 2014 IEEE

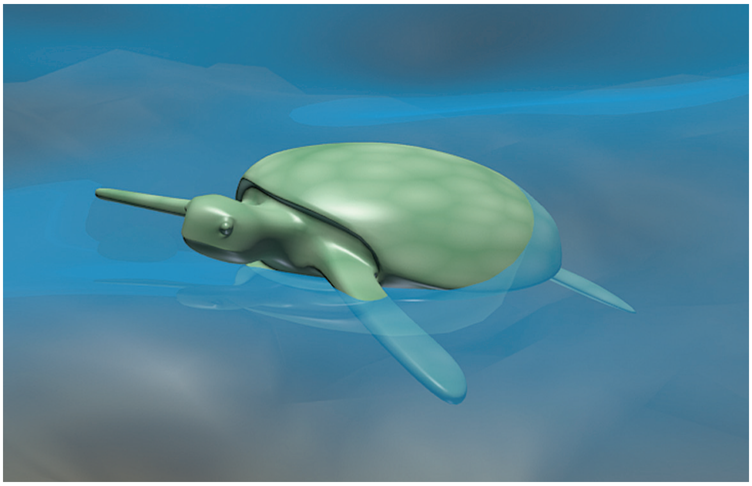

Figure 2. To create this sea turtle, the student started with a cube and repeatedly subdivided and sculpted it.

Figure 3. Another student's dining room scene using, among other techniques, cloth modeling, NURBS (nonuniform rational basis spline) surfaces, and a particle system for the candle glow.

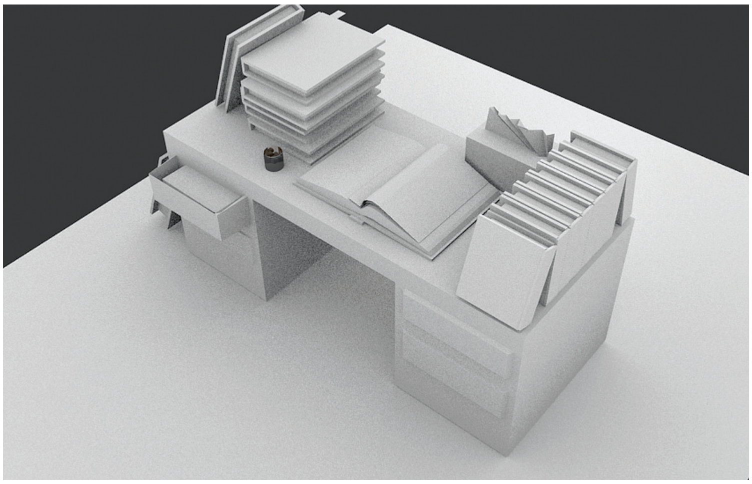

Figure 4. Another student created this desktop scene. It consists of many mesh cubes scaled and sculpted in different ways. The open book is a sculpted subdivided cube.

principles such as a right-handed coordinate system and right-handed rotations.

The Modeling section was probably the most involved because Blender offers so many ways to create and edit geometry. We covered metaballs (because they're fun) but spent most of the time discussing polygonal meshes. Polygonal meshes are the most-used geometry type in Blender and have the most tools. We covered modeling methods such as subdividing, extruding, vertex editing, proportional editing, constraints, and modifiers. Editing, by both interactive sculpting and applying modifiers, was probably the most popular topic because the students could create a basic geometry and then use their creativity to tweak it in interesting ways. For example, to create the sea turtle in Figure 2, the student started with a cube and repeatedly subdivided faces and pulled vertices.

None of the students had much experience with lighting. Even though people deal with lighting every day, few of them stop to observe and quantify its behavior. Students often created and rendered the perfect scene, only to find that everything was black. Appearance I covered lighting and colors; Appearance II covered texturing.

Regarding animation, particle systems were popular for creating waterfalls and glowing visual effects. Figure 3 shows how a student used a particle system to create the glow around a candle flame.

Results

The students seemed to enjoy Camp Blender. They thrived in an atmosphere in which they could learn and interact with each other. We were correct about Blender appealing to both future engineers and future artists. We were also correct about students with different backgrounds and interests learning from each other at a common watering hole.

The students' diverse backgrounds, combined with Blender's user interface, which exposes all features, made keeping the class together difficult. Once we discussed modeling, some students figured out more on their own and ran ahead. This wasn't bad but made the teaching less manageable than we had expected.

Blender provides multiple ways to accomplish roughly the same thing. It was interesting to see how each student approached what they were trying to create, especially when they did it in ways we weren't expecting. The turtle in Figure 2 is a good example.

Students tended to take one of two approaches. Some stuck with one small lifeboat of techniques (see the turtle in Figure 2 or the hexahedron-based desk scene in Figure 4). Others went to the opposite

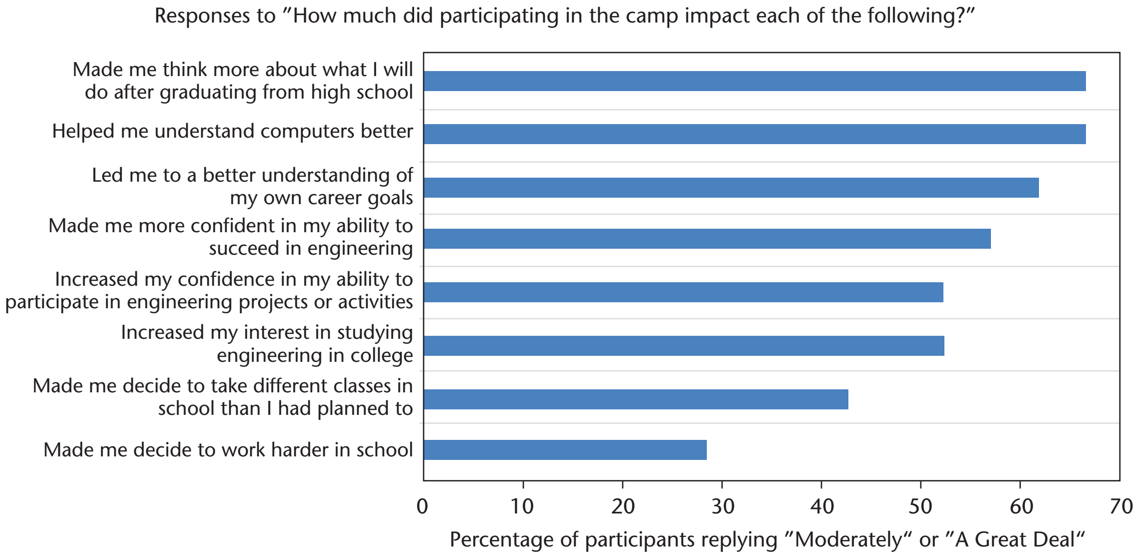

Figure 5. Responses to the postcamp impact survey. At first, we were disturbed about the low response to "Made me decide to work harder in school." However, we realized that about half of the students were on their school's robotics team and were probably working pretty hard already.

end of the spectrum and tried to incorporate as many effects as they could, such as in the dining-table scene in Figure 3.

At the end of the last day, the students gave presentations describing their favorite scene that they created and how they did it. This motivated progress and ended up teaching us all something new that each student had uncovered.

At the start, we were curious about what this class would do for students' enthusiasm for computer graphics and how it might make them think about their career trajectories. Figure 5 shows responses to the postclass survey. Almost 70 percent of the students indicated that this experience affected how they thought about life after high school. At first, we were disturbed about the low response to "Made me decide to work harder in school." However, we realized that about half of the students were on their school's robotics team and were probably working pretty hard already.

We plan to run Camp Blender again. We might break it into two levels: one for raw beginners and one for more advanced students. This way, we can keep each class a little closer together, rather than having some students being confused while others run on way ahead.

The notes for Camp Blender are at http://cs.oregonstate.edu/~mjb/blender. Anyone can use them for educational purposes. For a list of books about Blender, see the sidebar.

Further Reading

For those wanting to learn Blender, here are three excellent references:

- John Blain, *The Complete Guide to Blender Graphics*, Taylor & Francis, 2012.
- Roland Hess, *Blender Foundations: The Essential Guide to Learning Blender 2.6*, Focal Press, 2010.
- Jason van Gumster, *Blender for Dummies*, John Wiley & Sons, 2011.

Acknowledgments

We acknowledge the Blender Foundation and the many developers who have openly contributed to Blender. Together, you've created an amazing educational opportunity for so many.

Mike Bailey *is a professor of computer science at Oregon State University. Contact him at mjb@cs.oregonstate.edu.*

Cathy Law *directs Oregon State University's STEM Academy. Contact her at catherine.law@oregonstate.edu.*

Contact department editors Gitta Domik at domik@uni-paderborn.de and Scott Owen at sowen@gsu.edu.

Selected CS articles and columns are also available for free at http://ComputingNow.computer.org.