



Surfaces of Revolution in Virtual Reality - CalcVR

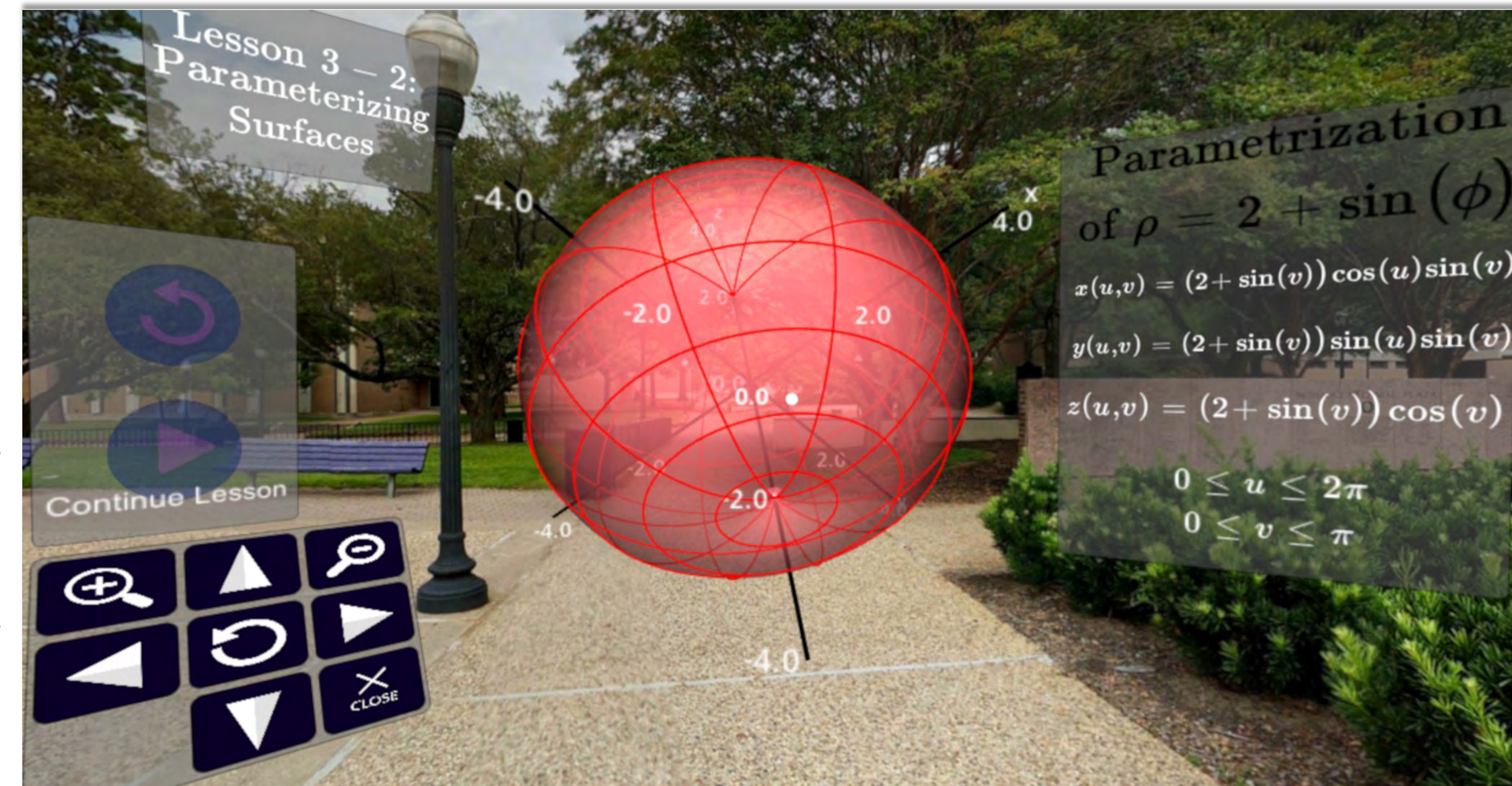
Student: Josh Harris

Faculty: Jeremy Becnel and Nick Long

Department of Mathematics and Statistics

Supported by NSF Grant DUE-1820724

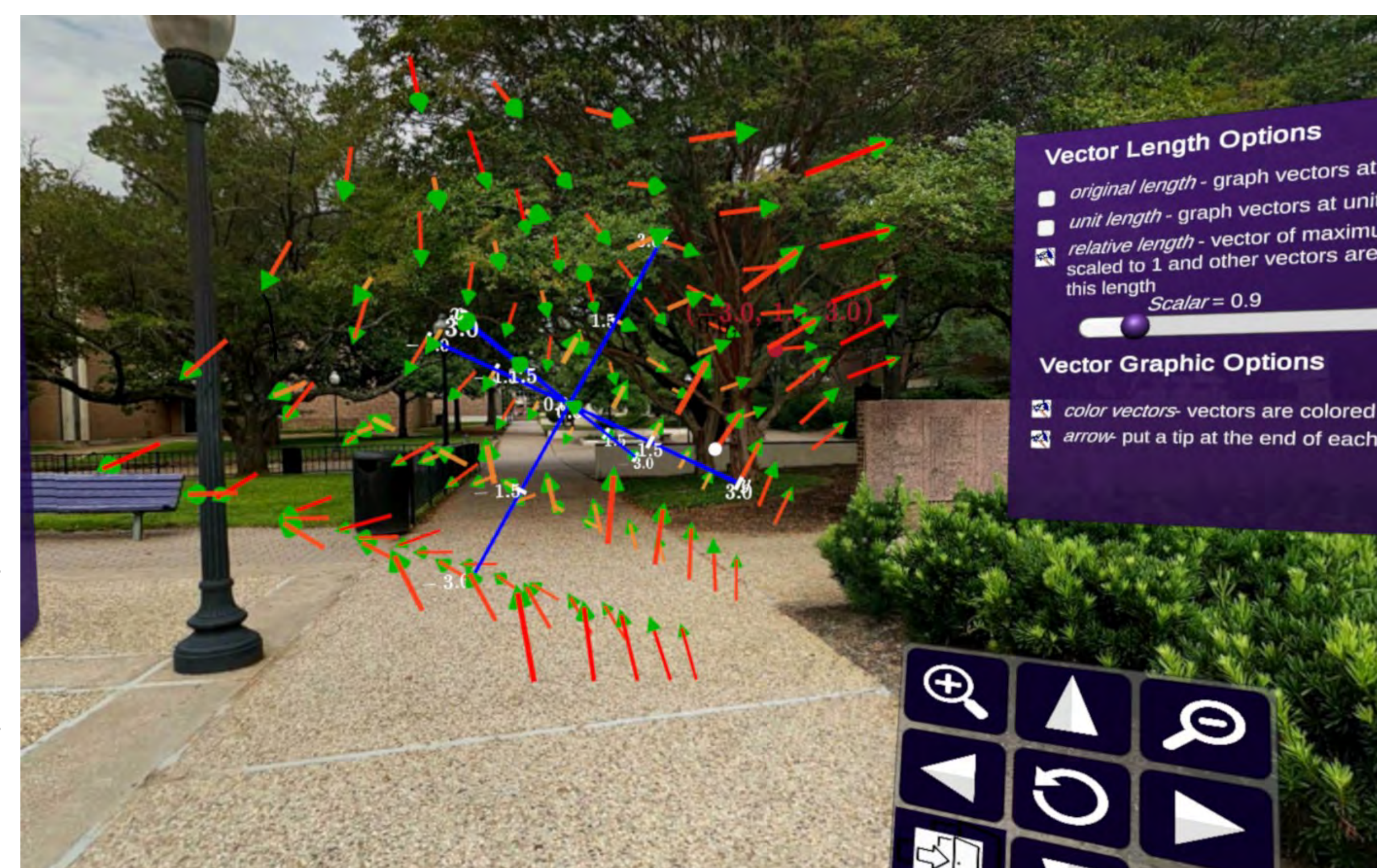
Photo by: Jeremy Becnel



Why is this project needed?

- ❑ Surfaces of Revolution is many student's first experience with three dimensional mathematical objects.
- ❑ The calculations required are not the inhibitor of the student's understanding.
- ❑ Visualization of these solids along with the appropriate set up are the greatest hindrance to a deep understanding of surfaces of revolution.
- ❑ Other tools that create computer generated visualizations of surfaces and curves in space render them as two-dimensional projections of a three dimensional object.
- ❑ Other tools typically do not have the capabilities of guided instruction, exploration of concepts on carefully chosen examples, and the ability to discover new facets based on user input.

Photo by: Jeremy Becnel



App Creation Tools

- ❑ Google Cardboard
 - ⇒ Available on most Android and Apple phones.
 - ⇒ Viewers (pictured on right) are inexpensive: \$5 to \$25.
- ❑ Unity
 - ⇒ Primarily a video game engine.
 - ⇒ Allows developer to quickly and easily port projects to multiple platforms.
 - ⇒ Built-in functionality which minimizes extensive

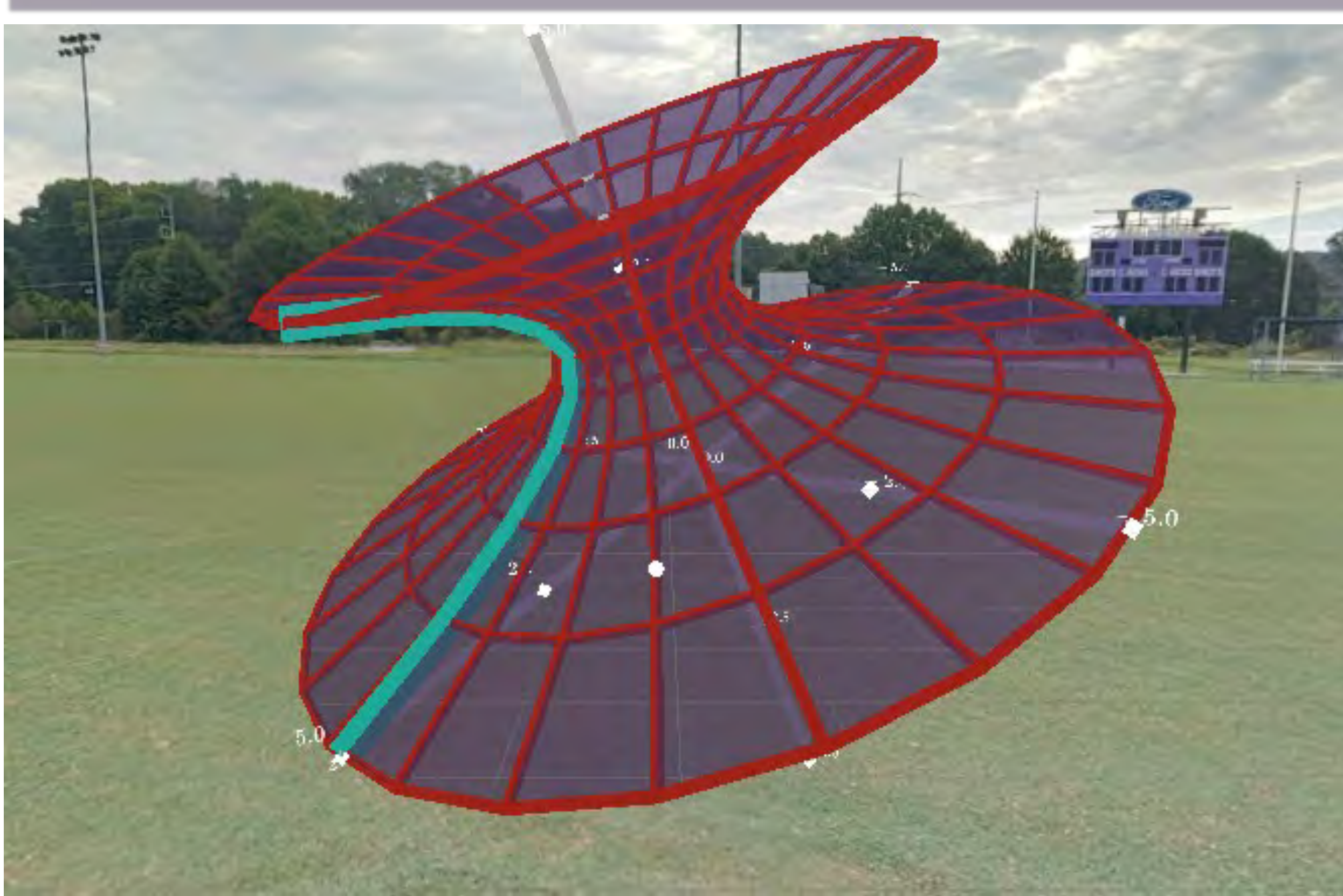
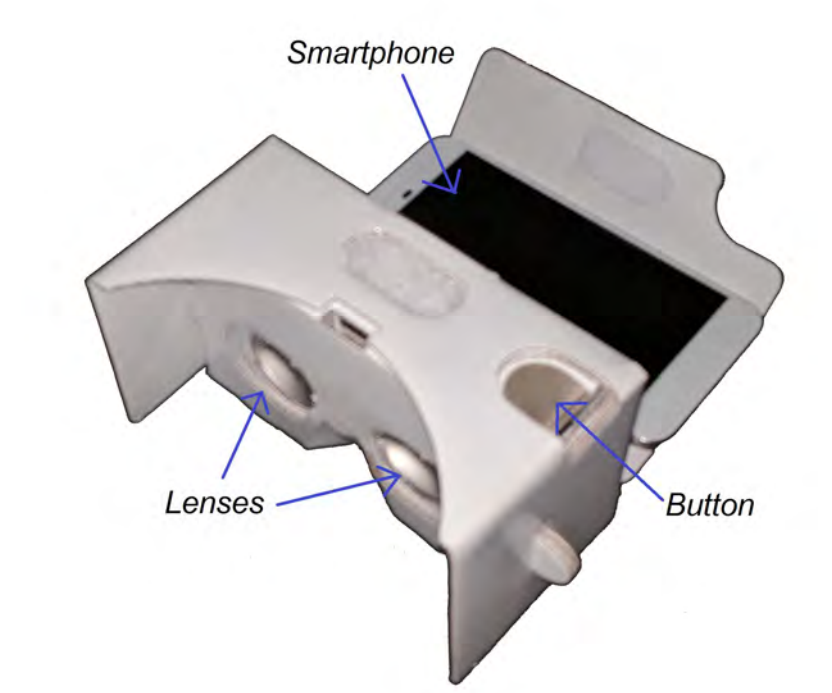
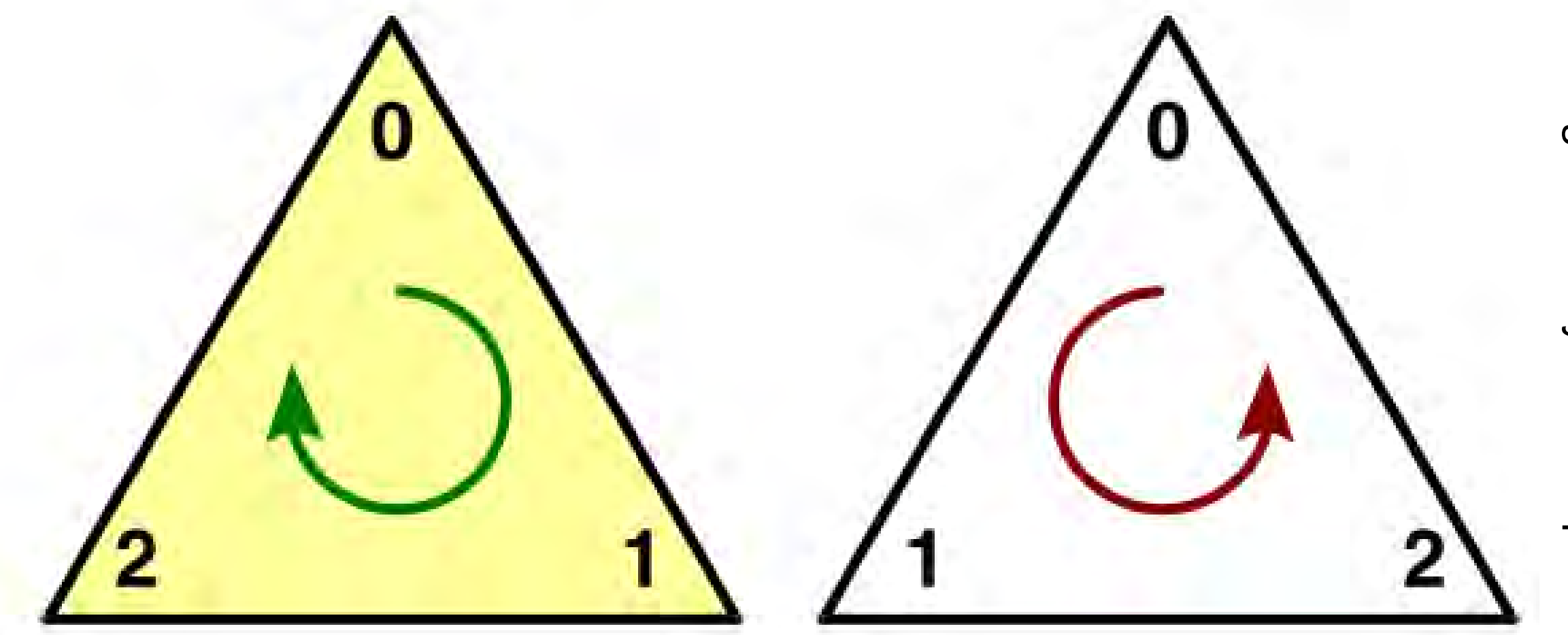


Photo by: Josh Harris

Scene User Flow

- ❑ The user selects 1 of the 6 predefined functions within the scene
 - ⇒ A line render of that function is then drawn in front of the user on the axes
- ❑ The user then selects an axis of rotation
 - ⇒ The line render may be redrawn depending on the axis selected
- ❑ The user click the "Rotate" button and their surface of revolution is created
 - ⇒ The shape, a wire frame, and the original function are all displayed in front of the user



Creating A Mesh

- ❑ The most important and most challenging aspect of this project is creating a mesh. The mesh acts as the surface of revolution within this scene.
 - ⇒ Meshes are used to visualize complex shapes/objects within Unity.
 - ⇒ A mesh is made up of many small triangles, a round surface will require more triangles than a flat surface.
 - ⇒ As represented by the photo above, these triangles must be created "clockwise" and "counter clockwise" so that the mesh is viewable from all directions.
- ❑ The challenge: The script used within the scene must dynamically create these meshes.
 - ⇒ Meshes for each surface of revolution will vary in shape and size.
 - ⇒ The number of triangles will vary per mesh.

Upcoming Functionality

- ❑ The user will be able to enter their own radial function to rotate.
 - ⇒ An input area that allows the user to enter a function and function bounds is needed.
- ❑ The ability for the user to draw their own line to rotate around the desired axis.
 - ⇒ Cursor position must be tracked at a very short interval.
 - ⇒ Drawn line must constantly be checked to ensure a valid line is being drawn.