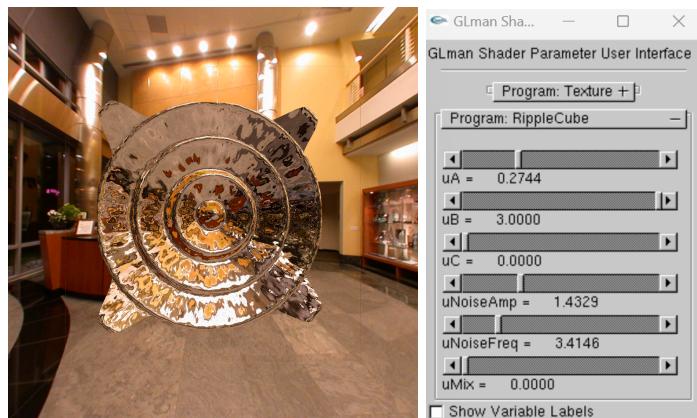
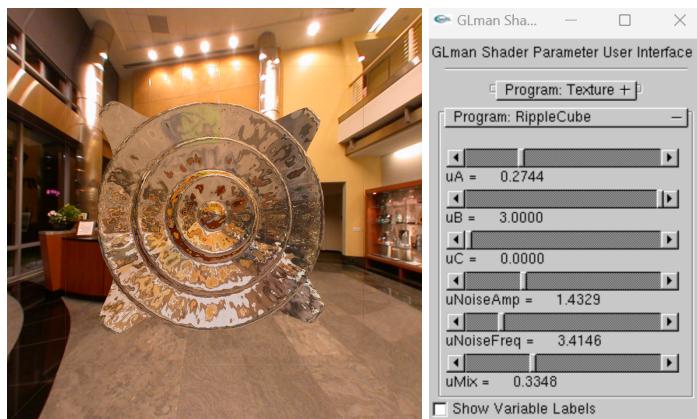


**CS 457 Project #4**  
**Cube Mapping, Reflective and Refractive Bump-mapped Surfaces**  
Annette Tongsak  
[tongsaan@oregonstate.edu](mailto:tongsaan@oregonstate.edu)

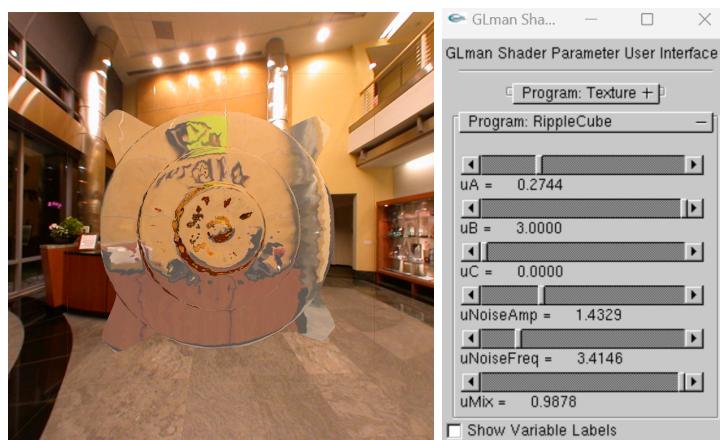
Reflection



Mixed reflection and refraction



Refraction



## [Video link](#)

To create this display, I worked in this order:

### **ripplecube.vert**

1. Implemented the equations used for Project #3's displacement mapping into the template vertex shader file
  - a. Allows for displacement mapping on the QuadXY

### **ripplecube.frag**

2. Implemented the logic from Project #3's bump-mapping into the template fragment shader file
  - a. Allows for bump-mapping on the QuadXY
3. Implemented reflections and refractions to be controlled by the uMix slider in GLman
  - a. The most difficult part was figuring out the last line, where we set gl\_FragColor to `mix(vec4(reflectColor.rgb, 1.), vec4(refractColor, 1.), uMix)`. In this line, we allow uMix to control whether the quad is a reflective or refractive surface.

### **texture.frag and texture.vert**

4. Used the given files to create the six walls
  - a. For wall decorations; these don't actually participate in the cube mapping

### **proj4.glib**

5. Implemented calls to texture.vert, texture.frag, and the NVIDIA lobby cube map textures
6. Set parameters for the following variables so they appear as sliders in GLman:
  - a. uReflectUnit
  - b. uRefractUnit
  - c. uA
  - d. uB
  - e. uC
  - f. uD (set to 0.5 so not shown as a slider)
  - g. uNoiseAmp
  - h. uNoiseFreq
  - i. uEta
  - j. uMix
  - k. uWhiteMix
7. Specified the quad and its sub-quads so there are enough vertices to create a smooth displacement function