CS 457 Project #3 – Displacement/Bump Mapping and Lighting

By: Rick Menzel **Email**: menzelr@oregonstate.edu

Screenshot(s):

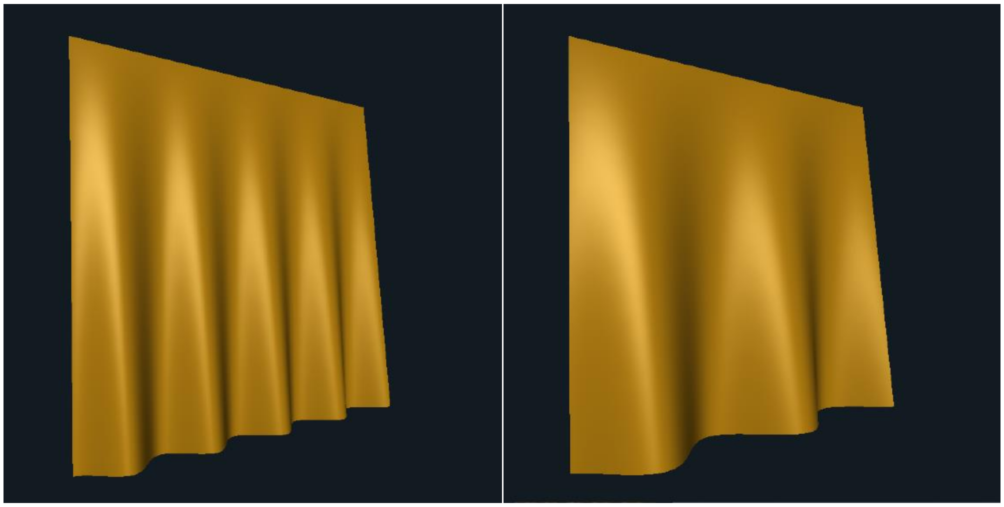

Fig 1. Displacement Variations

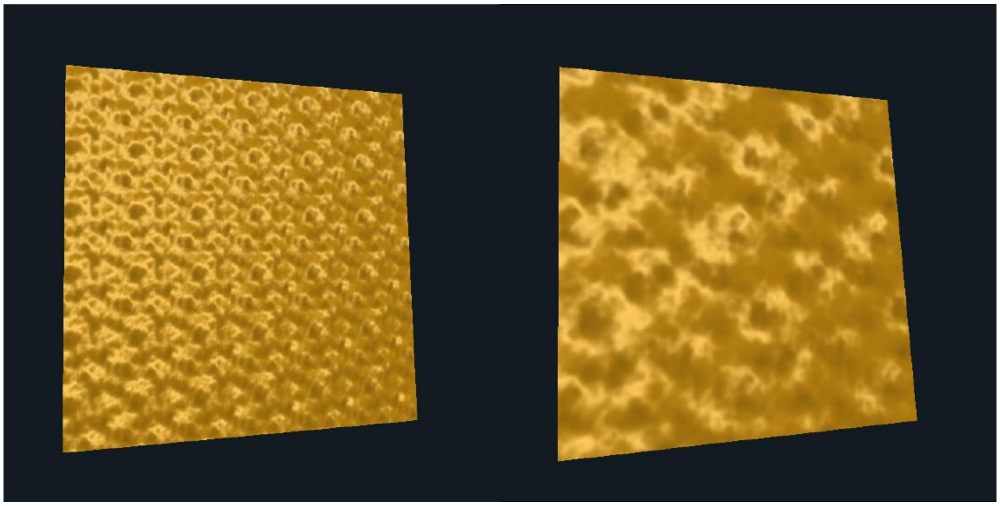

Fig 2. Bump Mapping Variations

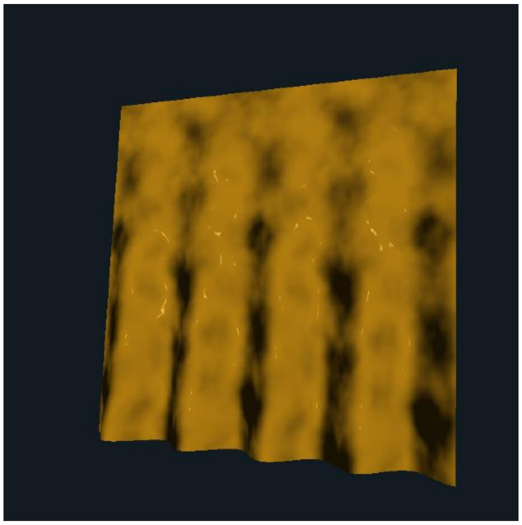

Fig 3. Displacement with Bump Mapping and Adjusted Lighting

Link:

https://media.oregonstate.edu/media/t/0_92qtsmhr

Description:

I began this project with lighting on a composite quadrilateral. This begins in the vertex shader with calculating 3 vectors: a surface normal, a vector from the vertex to the eye position, and lastly a vector from the vertex to the light position. Note that I did make it possible to adjust the light position using sliders. These vectors are then passed to the fragment shader where they are used to compose ambient, diffuse and specular lighting components in conjunction with a slider-based lighting level for each component and the material or specular color as appropriate. The three lighting components are then combined to get the fragment color.

Next, I turned to the displacement mapping. This was relatively simple and involves displacing the z-component of the vertex using a sine function based on the y-position. Essentially the sine displacement is multiplied by a factor found by determining how far the point is from the top of the curtain. After the vertex is updated with this new z-coordinate, an updated normal is calculated using the cross product of two tangent vectors to the vertex.

Finally, the bump mapping is performed. This involves getting a noise value and generating an angle for displacing the normal in each of the x and y directions. These angles are adjusted based on the desired amplitude and frequency of the displacement and then used to adjust the normal produced by the displacement mapping function, thus creating the illusion of a textured surface.