CS 557 Computer Graphics Shaders

Project #4

Cube Mapping Reflective and Refractive Surfaces

Name: Rachel Xing (#934229189)

Email Address: xingru@oregonstate.edu

Video Link: https://youtu.be/e8Kg1s8Rvt4

Description:

This project implements a shader to simulate reflection and refraction effects on a bump-mapped 3D model. The selected model is “cow.obj”, and the used cube map is the Nvidia lobby texture provided in the course. This project runs in glman using the “cube.glib” file.

Both reflection/refraction and bump-mapping calculations are performed in the fragment shader. Using the normal and eye vectors obtained from the vertex shader, the GLSL functions reflect(eye, normal) and refract(eye, normal, **uEta**) [where uEta is the index of refraction] are applied to compute the reflection and refraction vectors respectively. These vectors are then used to index the cube map, thus generating the corresponding visual effect on the cow model. The **uWhiteMix** parameter adjusts the amount of white added to the cow’s surface color to enhance the model’s visibility, while **uMix** controls the blending between the reflection and refraction.

The bump mapping adds a 3D noisy depth effect on the cow’s surface by perturbing the normal of each fragment. This is achieved by sampling three noise values from a **3D noise texture** generated in glman, each representing the rotation angle of the normal around the x, y, and z axes respectively. As in Project #3B, parameters **uNoiseAmp** and **uNoiseFreq** control the amplitude and frequency of the noise function, thus creating varying reflective and refractive effects on the cow's surface with crinkles but not causing any actual height displacement.

[Screenshots are on the next pages.]

Screenshots:

Fig 1. Refractive effect on the cow model surface with **uMix = 0**.

Fig 2. Blending of refractive and reflective effect on the cow model surface with **uMix = 0.5**.

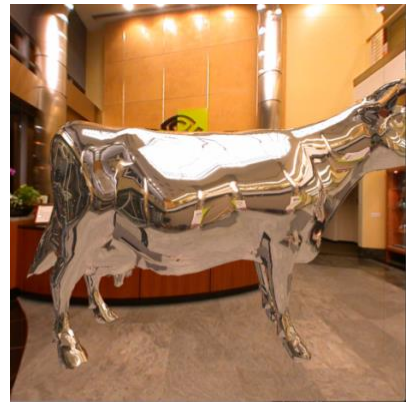

Fig 3. Reflective effect on the cow model surface with **uMix = 1.0**.

Fig 4. **Movement** of the **refractive** cow model along its own coordinates demonstrating correct refraction calculations.

Fig 5. **Rotation** of the **refractive** cow model along its own coordinates demonstrating correct refraction calculations.

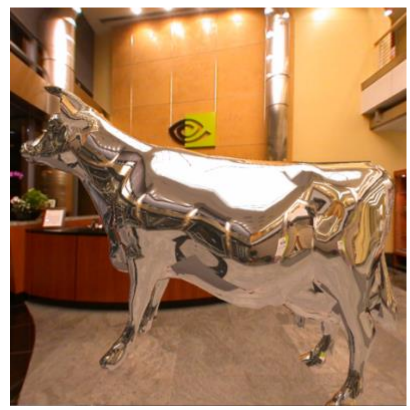

Fig 6. **Rotation** of the **reflective** cow model along its own coordinates demonstrating correct reflection calculations.

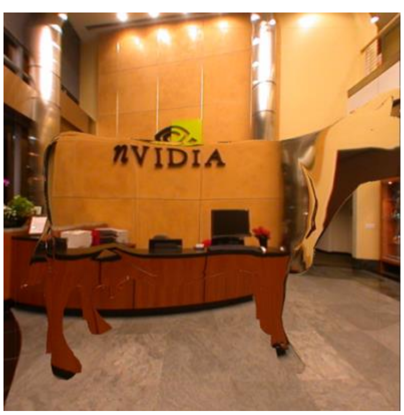

Fig 7. If the model's color is not mixed with white (**uWhiteMix = 0**), the refractive object is difficult to observe.

Fig 8. When **uEta decreases** from the default value in Fig. 1, the **light bends less**.

Fig 9. When **uEta increases** from the default value in Fig. 1, the **light bends more**.

Fig 10. **Refractive effect with noises** generated by bump mapping with **low uNoiseFreq and high uNoiseAmp**.

Fig 11. **Refractive effect with noises** generated by bump mapping with **high uNoiseFreq and high uNoiseAmp**.

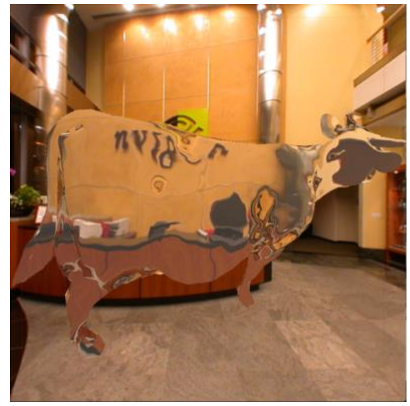

Fig 12. **Rotated** cow model with the noisy refractive effect from Fig. 11.

Fig 13. **Reflective effect with noises** generated by bump mapping with **middle uNoiseFreq and middle uNoiseAmp**.

Fig 14. **Reflective effect with noises** generated by bump mapping with **low uNoiseFreq and high uNoiseAmp**.

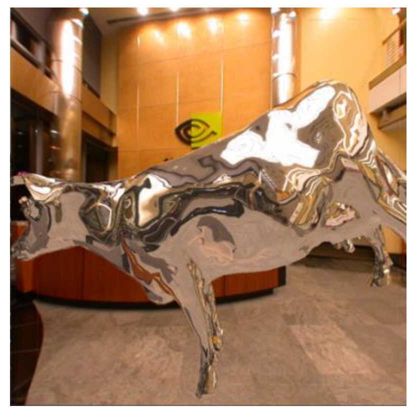

Fig 15. **Rotated** cow model with the noisy reflective effect from Fig. 14.