Dathan Pompa

Computer Graphic and Visualization

10/27/2024

Reflection

When I was selecting my objects for the scene, I chose items that people can right away recognize and can be useful to me to display my knowledge of how to apply OpenGL techniques. I chose a microwave, a pitcher, a bowl, and an ice maker machine. These items let me use different objects in order to recreate how the object looks in my scene. The microwave is a simple box, the ice maker has a round front, and the pitcher has smooth, tapering sides. These different forms gave me the chance to work with cubes, cylinders, and planes, while also letting me play around with textures and transformations. Since I was combining different shapes to create one of the objects, I was able to create texture mapping so that certain textures only show on a specific area of the object I made. For the microwave, I texture only the front of the object as well as I added a button to allow the user to open it.

To allow the user to move around my scene, I made adjustments to how they can navigate around the objects on the table. I added adjustments to the movement speed using the mouse wheel. This helps when you want to zoom in for a closer look or speed up to move through the scene faster. To handle all this, I used view and projection matrices to simulate the camera’s movement and perspective, which helps create a more immersive experience as you move around the 3D space. The virtual camera was implemented using view and projection matrices. By adjusting the view matrix, I was able to simulate camera motion and orientation based on the user’s input. The projection matrix is set up for a perspective view, which ensures that objects closer to the camera appear larger, providing a realistic 3D experience.

I needed to create custom functions that allowed me to implement textures, shaders, and transformations. The transformation functions let me adjust the scaling, rotation, and position for the objects which I used a lot to display the objects in my scene accordingly. It’s really useful because I can just call it with the specific parameters for any object, and it applies the necessary transformations. Whether I’m positioning the microwave or resizing the pitcher, I don’t need to repeat the same lines of code each time. For my setShaderMaterial function, it lets me manage the material properties of each object, such as the color or shininess. I used it to give the bowl a plastic look and the microwave a metallic appearance. The best part is that it’s reusable, so I can switch materials on any object by calling the function and passing in the correct values. When I made the setShaderTexture function, this function binds textures to objects, making it easy to assign different textures to different surfaces. For example, I used it to texture just the front of the microwave while leaving the rest of the body plain. It keeps the code cleaner and more flexible because I can easily apply different textures to different parts of an object without having to rewrite everything.