Joseph Iadarola

CS 330 Reflection

The scene I chose to recreate for this project was based on the simple objects I found around my desk. This means that I selected four objects that I thought I would be able to recreate with the least amount of frustration while also giving enough challenges to show my skill at building a 3d scene. The two complex objects I chose were my hat and a soda can both were made up of different simple shapes so that by creating both I would meet the requirements for the number of simple shapes used. My hat needed a half sphere at minimum and other shapes to make its brim which I decided to be a short cylinder after seeing how close the combination came to looking like the original. Along with those shapes I also included a full sphere for the tip of the hat in case the half sphere did not count. The brim and tip both had the same fabric so that saved me time when I was searching for texture images to use, and the main part of the hat was a solid grey which turned out well at the end. The can was a more challenging task because the torus I used for the top rim created too many problems from being only half filled with triangles to needing too many triangles to seem whole that it slowed down the load time of the program and made my triangle count go far over the limit set in the directions. Also, the texturing of the can sides became a genuine problem when at the end of my problem-solving stage the textures stopped updating even when I changed the values used for the texture coordinates. This did not take away from object obviously being a soda can with a Coke label but beyond that recognition the labeling is poorly done. The remaining objects were more straightforward because of their simplicity. The metal grinder is only three cylinders stacked on top of each other with varying number of sides to show the original grip textures that allow for the sections to come apart. The card box turned out to be the best of the objects because of the proper label that matches the original and how everything is oriented correctly and lines up with the original picture that was used.

Navigation of the 3D scene can be done using the WASD keys to move forward, left, back, and right respectively. The QE keys to move up and down, and the mouse cursor to change the viewing angle of the camera. The BV keys will change the objects from a 3D to 2D view and back while the left and right arrow keys will change the view mode from filled to wireframe line views. The left and right alt keys will also stop and start the moving light source that was meant to act as a moving sun, while the scroll of the mouse will make moving speed increase and decrease.

This program is currently designed to build a specific 3D scene, but it can be modified easily to change the scene that is created. The main function that allows the program to be modular is the CreateScene function which creates a new GLMesh type object, loads its properties, calls the corresponding build function, and then adds the mesh to the scene vector for rendering. By changing the GLMesh objects and their properties in this function, the programmer will have an easier time modifying the scene that is created. This also includes modifying the scale, rotation, and translation of any mesh objects created through the prop (properties) vector generated inside the function. Along with the CreateScene function, the functions that start with UBuild are semi-modular in the sense that each creates a simple shape and correctly generates vertices and indices vectors that are stored in the mesh object. The part of the functions that are not modular are the texture coordinates which are hard coded into each function. These coordinates will mostly work for images that do not need to be lined up correctly, such as the current fabric textures, but not for complex images like the card box used in the UBuildPrism function.