Final Project Reflection

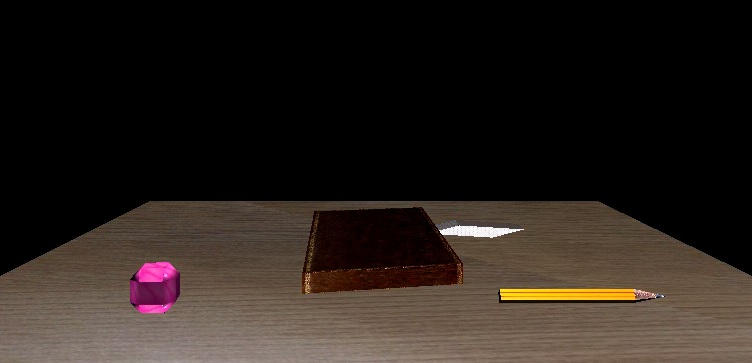
Andrew Hwang

CS-330-T4518 Computational Graphics and Visualization

April 17, 2022

The goal of the project was to recreate a scene of real-world objects with 3D renderings using OpenGL and Visual Studio. Below is a side-by-side of my inspiration photo and the scene that was created.

A picture containing text, businesscard

Description automatically generated

The four primitive shapes used are: a plane (tabletop), cube (book, pencil base, eraser), pyramid (pencil tip), and sphere (the pink ball). Texture has been added to the book, the tabletop, and the pencil parts to make it appear more realistic. There are two light sources included – one stationary above the scene and a second one that can orbit the scene changing the lighting as it moves.

We used a plane to represent the tabletop because it was the most logical to use and flat for us to place the other objects appropriately. A sphere was used to represent the wrist/stability trainer ball as a simple representation. A cube was the best approximation of the book and eraser. A cube and a pyramid together were used for the pencil so that I could use appropriate textures (with straight lines for the pencil base) and it was easy to program them/group them together in terms of manipulating placement, scale, and rotation. In the representation, not as much rotation was applied to all objects because it was skew/warp the shape as it was rotated/matrix multiplication occurred.

Navigation is simple and can be done using either the mouse or keyboard. For the mouse, movement will move the camera and the wheel adjusts the movement speed. For the keyboard, the navigation buttons are as follows:

|  |  |
| --- | --- |
| Camera Directions: | Other controls: |
| W – Forward  S – Backward  A – Left  D – Right  Q – Up  E - Down | P – Changes between orthographic and perspective view  L – Start orbiting light  K – Stop orbiting light  ESC – Close |

Of note: the initial lamp/orbiting light will launch not moving but can be started by pressing L on the keyboard.

User defined functions in our program were created to initialize the program, set window size, redraw graphics when resized, and most importantly to render graphics on the screen of various shapes. We created functions for the shapes to be created and labeled them appropriately. Because separate functions were created for each primitive shape, that part of the program was able to be reused to create multiple objects. For example, we used the same function to create the book and eraser – both are cubes, but we only had to manipulate the scale/rotation/placement and added textures. The rest of the code was written following the same patterns – create vertex shader and fragment shader source codes, in main we create all the objects & load texture, then we use a loop to render the objects.