Monoblock Frame:

Rendered in OpenGL

Gage Biros

Department of Computer Science

Southern New Hampshire University

CS-330: Comp Graphics and Visualization

24 April 2022

**ABSTRACT**

We have been instructed to recreate a 3-Dimensional environment using Microsoft Visual Studio along with OpenGL API that allows users to render objects in an environment. This course created a strong fundamental base on some fundamental mathematics and uses that knowledge to create virtually anything in this application. This is an informative piece, along with my personal experiences while using these tools.

**Brainstorming the Scene**

In the beginning stages of this application, we were tasked with choosing a 3-Dimensional environment that we were going to recreate in the OpenGL API within Microsoft Visual Studio. The photo I chose was of a packaging machine from an automation company I worked for. I chose this photo as it has many simple shapes as well as some more complex shapes. There are a lot of rectangular prisms in this single machine that make up the frame, as well as the electrical box located on the front of the machine. There are some cylindrical shapes such as the main shaft in the center, some smaller support shaft on the deck of the table, and some of the items that are scattered on the top and middle decks. I would probably use planes for the decks and the mounts that are throughout the machine to recreate the flat surfaces. There are a couple of bearings that surround the shaft that I might combine a torus shape and a plane since the bearing is mounted on a plate, if it is more applicable, I could always use a cylinder, but I would like to see what the torus would like here. To make this simple I decided to omit everything in the background and focus on the machine itself. In this photo, our machine is well lit so we should have more than one light source. The textures are also metallic so we should see reflective properties on the machine, and the light should look realistic.

**Navigating the Scene**

Recreating the environment is only a small piece of the puzzle. Users should be able to navigate through the environment, like they would in the actual environment. Luckily OpenGL and Visual Studios, allows users to utilize the tools these programs have provided and gives that power to the users. These functions often are a separate class, which allows users to create one program and apply it to different projects. The “camera class” has certain functions that we can modify to allow users to move like they are in the environment; these functions have certain key bindings which may or may not have a default setting, but users are able to modify these functions to make their experience unique.

**Design Decisions**

I probably could have made my program more efficient if I knew every function within OpenGL, there is so much opportunity, if I had more time, I am sure I would be able to harness the power these programs supply and reduce the size of all my files/functions. I unfortunately was on a time crunch and had to revise my initial plan so that I could meet the deadline. I was able to get the frame of the Monoblock completed, along with the decks of the machine, the electrical box, and the box supports; These items were also textured to their respective texture files. I also located the machine on a plane so that the piece seemed like it was in a real environment. I have an aerial light as well as a light from the front of the machine. I really wish I had more time on this project as I had a lot of fun learning a lot just from rendering the environment I created.

Reference

“Silver.jpg”

[https://www.google.com/url?sa=i&url=https%3A%2F%2Fpxhere.com%2Fen%2Fphoto%](https://www.google.com/url?sa=i&url=https%3A%2F%2Fpxhere.com%2Fen%2Fphoto%25)2F1644929&psig=AOvVaw3YFXECo-wWztMAV0ucY\_ek&ust=1650936624576000&source=images&cd=vfe&ved=0CAwQjRxqFwoTCPjijqqIrvcCFQAAAAAdAAAAABAL