**Milestone Reflection**

Tyeika Willis

CS-330 Computational Graphics and Visualization

Professor Philip Enkema

June 29, 2024

Creating a 3D scene for this milestone project was both challenging and rewarding. It required me to dive deep into various aspects of 3D graphics programming, including shape creation, texture application, lighting, and camera navigation. This reflection will explore the reasoning behind my development choices, the mechanisms I implemented for user navigation, and the custom functions I developed to ensure my code was modular and organized.

I took a picture of my scene from my living room. I wanted something that was challenging for me but still achievable. The objects included a table, vase, pocketbook, picture frame, and a sphere. Each object was chosen for its unique properties and the specific challenges it presented. The table was a foundational object, providing a broad surface ideal for experimenting with wood textures. I chose this object to understand how textures can add depth and realism to a scene. However, aligning the texture correctly on the table's surface proved tricky and required multiple adjustments. The vase, made of a sphere and a tapered cylinder, allowed me to explore complex shapes and glass textures. This object helped me understand how different materials interact with light, but achieving the right level of transparency and reflection was challenging. I had to fine-tune the material properties several times to get the desired effect. The pocketbook added a larger element to the scene and helped me practice applying detailed, abstract textures. Ensuring the texture mapped correctly to the pocketbook's surface without distortion was a significant challenge. The picture frame added realism and decoration to the scene. It allowed me to experiment with cement textures, which added a rugged and industrial feel. The sphere was essential for demonstrating metallic textures and lighting effects on curved surfaces. While working with the sphere, I faced the challenge of balancing the specular highlights to ensure they looked realistic without being overly bright or washed out.

To achieve the required functionality, I divided the project into distinct components, each handled by specific classes and methods. The SceneManager class was the core of the project, managing texture loading, material definition, lighting setup, and scene rendering. The ShaderManager class was responsible for managing shaders, ensuring efficient rendering by the GPU. One major challenge I faced was integrating the shaders correctly to handle the complex lighting and material interactions. Debugging shader issues was particularly tough, as minor errors could lead to significant rendering problems. I spent considerable time troubleshooting these issues, but it was satisfying when everything finally worked as intended.

I aimed to make user navigation within the 3D scene intuitive, utilizing both keyboard and mouse inputs. The keyboard controls allowed users to move forward, backward, left, and right using the W, A, S, and D keys. This familiar setup, common in many 3D applications and games, ensured users could navigate the scene comfortably and efficiently. Implementing smooth camera controls was another challenge. Initially, I faced issues with the camera movement being either too slow or too fast, making navigation difficult. Setting up the virtual camera involved configuring the view and projection matrices. The view matrix was updated based on user inputs, allowing the camera to move fluidly through the scene. The projection matrix was configured for perspective projection, creating a realistic sense of depth and space within the 3D environment,

To keep the code organized and easy to read, I used custom functions in the SceneManager class. These functions handled specific tasks, making the code modular and reusable. The DefineObjectMaterials function set up material properties for each object. This kept all material definitions in one place, making them easy to manage. The SetupSceneLights function was crucial for configuring the lighting in the scene. This function allowed me to define multiple light sources, including ambient, directional, and point lights. Debugging the lighting setup was challenging, especially when lights interacted with different materials in unexpected ways. Fine-tuning the light positions and intensities took time, but it was rewarding to see the scene come to life with realistic lighting. The LoadSceneTextures function handled the loading and binding textures. By keeping texture management separate from other parts of the code, I ensured that the texture loading process was streamlined and did not interfere with the rendering logic. One significant challenge was managing the texture coordinates for different objects to avoid stretching or distortion. This required careful attention to detail and frequent testing. By carefully selecting objects, setting up intuitive navigation controls, and using modular functions, I created a functional and visually appealing 3D scene. Throughout the project, I faced numerous challenges, from shader debugging to texture mapping and camera control. Overcoming these challenges not only improved my technical skills but also reinforced the importance of planning, organization, and modularity in software development. This project has given me a deeper appreciation for the complexity and artistry involved in creating immersive 3D environments.