Kyle Marinaro

6/26/2025

CS 330

Project Reflection

For my final project in computational graphics, I developed a realistic desktop environment—a closed laptop, a coffee mug, some books, and a desk lamp, all meticulously arranged on a table. I intentionally selected this scene because it offered a comprehensive platform to apply a diverse range of skills: object modeling, transformation, texturing, and lighting. Careful attention was given to scaling, positioning, and rotating each shape to accurately capture the nuances of real-world objects. The construction began with basic primitive meshes—boxes, cylinders, etc.—and I then utilized transformation matrices to shape these into more complex forms. For instance, the coffee mug’s handle was created by iteratively rotating and connecting segments to achieve a natural curve, while the books were arranged with subtle tilts and scale variations to enhance visual realism. This approach not only satisfied the technical criteria of the assignment but also allowed me to demonstrate creativity and a keen eye for detail in scene assembly.

A particularly significant aspect of this project was the implementation of interactive camera controls using both keyboard and mouse input. I programmed controls enabling users to navigate the scene: ‘W’ and ‘S’ keys for zoom, ‘A’ and ‘D’ for horizontal panning, ‘Q’ and ‘E’ for vertical movement. Mouse actions were mapped to camera rotation, while the scroll wheel adjusted movement speed. These interactive features were achieved through matrix transformations and real-time event handling, resulting in a smooth and intuitive navigation experience. I also incorporated the ability to toggle between perspective and orthographic projections, fulfilling project milestones and enriching the user’s ability to observe the environment from multiple viewpoints.

To maintain code clarity and efficiency, I adopted a modular approach, developing custom reusable functions. For example, my SetTransformations() function accepts parameters for scale, rotation, and position, and applies them to any scene object, minimizing redundancy and streamlining the creation process. Functions like SetShaderTexture() and SetShaderMaterial() similarly expedited the application of textures and materials, ensuring maintainability as the project evolved. This emphasis on modular and reusable code facilitated easier testing and debugging, which was essential as the complexity of the scene increased.

Ultimately, this project significantly deepened my understanding of computational graphics and OpenGL-based 3D rendering. It strengthened my confidence in integrating geometry, lighting, and interactivity to create cohesive virtual environments. While I am comfortable with much of the rendering pipeline, I recognize that advanced lighting models and shadow techniques remain areas I need to explore further. Moving forward, I intend to expand on these skills through courses in game development, simulation, or interactive media. Possessing the ability to construct detailed, interactive 3D scenes will undoubtedly prove valuable in future professional pursuits within software development, virtual design, and computer graphics.