Adam Resemius

CS-330

Project 1

Open GL Design Decisions

Introduction

Creating a lifelike 3D world using OpenGL was both challenging and rewarding during my CS-330 course. This project helped me explore new boundaries and learn about 3D graphics and OpenGL. To better grasp the concepts from our coursework, I extended my learning resources to include online tutorials. Victor Gordan's YouTube channel was particularly helpful, offering valuable insights into mastering OpenGL techniques, especially the creation of detailed and immersive environments – a central aspect of our project's goal.

Design Approach

My approach was to recreate a part of my entertainment center in 3D. I began with a photo taken on my iPhone, using it as a reference. I followed a math-based strategy to arrange objects proportionally within the scene. This technique allowed me to position objects systematically, considering their relationships to one another. I immersed myself in vector manipulation, mapping, and structuring objects, although I had some difficulties seeing everything as a collection of triangles.

Inspired by real-world settings, I encountered challenges and found solutions by referring to the original scene. Imagining real-world objects as triangles within OpenGL helped me convert real-life concepts into visual graphics. For example, turning complex objects like the money tree – with its 49 vertices – into triangles was a significant hurdle. This required careful planning to ensure the vertices fit seamlessly, creating an accurate and appealing representation.

Development Choices

I chose to recreate my entertainment center to make the virtual world feel familiar and personal. This approach helped me connect with the project, making it more engaging and meaningful. This familiarity also enabled me to carefully align objects, ensuring their sizes and positions were true to reality.

I designed controls for moving within the environment. Using the WASD and QE keys mimicked another game movement. I was aiming for something familiar, such as Minecraft. This approach added to the immersive experience. I also incorporated mouse inputs to control the camera's direction, allowing users to change the movement of the camera through left clicking on the mouse and dragging it to change the view angle.

Usage of Modular Programming

To keep my code organized and manageable, I implemented custom functions. These functions encapsulated specific tasks, making the code more coherent. For instance, the function responsible for handling key inputs centralized movement logic, making the code more readable and easier to maintain.

Using Math and Object-Oriented Programming

Mathematics played a vital role in my project. Scaling objects proportionally and managing their positions involved intricate calculations. This math-driven approach maintained accurate relationships and realistic proportions within the scene. To streamline calculations and encourage reusability, I used header files and object-oriented programming. This allowed me to create objects as instances of classes, bundling their properties and behaviors for simpler management and consistent use.

Reusability

My codebase is highly reusable. By adhering to a systematic approach and encapsulating functions within custom classes, adding new objects becomes more straightforward. Adjusting parameters and integrating new elements accelerates development while maintaining code coherence.

References:

GeeksforGeeks. (2022, June 29). *Getting started with OpenGL*. GeeksforGeeks. https://www.geeksforgeeks.org/getting-started-with-opengl/

YouTube. (2022). *OpenGL Tutorials*. *YouTube*. Retrieved August 15, 2023, from https://www.youtube.com/playlist?list=PLPaoO-vpZnumdcb4tZc4x5Q-v7CkrQ6M-.