Seema Chavesta

Apr 20, 2025

**Final Project Design Decisions – CS330**

The final CS330 project presents a 3D scene built using OpenGL and C++, replicating a simple real-world image featuring household or decorative objects. The centerpiece of the scene is a vase containing a flower, constructed from multiple basic shapes including a tapered cylinder, sphere, and standard cylinder. This project showcases the use of low-polygon modeling, realistic lighting, accurate texturing, and interactive camera navigation.

To fulfill the modeling requirements, the scene includes at least four distinct shapes—plane, cylinder, sphere, and tapered cylinder—with one complex object (the flower in the vase) built by combining multiple primitives. Transformation matrices were used to scale, rotate, and position each shape, ensuring accurate spatial relationships within the 3D world. All models remained under 1,000 triangles to optimize performance while preserving recognizable structure.

Textures were applied to three main components: the vase, the flower, and the tiled floor. All textures were high-resolution (1024x1024 or higher) and royalty-free, selected to enhance surface detail without pixelation or distortion. Texture coordinates were carefully scaled (UV mapping) to maintain visual consistency and realism.

The lighting setup features two sources: a directional light with a soft, pinkish hue to simulate daylight and a point light to cast highlights and define shape contours. Both lights adhere to the Phong lighting model, incorporating ambient, diffuse, and specular components. These sources were positioned to ensure the scene is fully lit from various angles without producing unwanted shadows or dark zones.

Full 3D camera controls were implemented, enabling movement across the X, Y, and Z axes using WASD and QE keys. Mouse input controls the pitch and yaw of the camera, allowing dynamic changes in orientation, while the scroll wheel adjusts movement speed. This configuration provides an intuitive and immersive way to explore the scene.

To support multiple viewing perspectives, a projection toggle was added, allowing users to switch between perspective and orthographic views with a single key press. This functionality enables both realistic and schematic evaluations of the 3D layout.

All code follows best practices for modularity, clarity, and maintainability. Functions such as PrepareScene, SetTransformations, and SetShaderTexture help organize logic, while descriptive comments and clear formatting improve readability and debugging.

In summary, this project demonstrates the effective use of OpenGL to recreate a reference image through 3D modeling, texturing, lighting, and camera control. It reflects a strong understanding of real-time graphics rendering and development principles, culminating in a well-structured and visually accurate scene.