Victor Jarvis

CS 330

Design Decisions and Reflection

The development of the 3D scene for a stage at a venue involved carefully selecting objects that contribute to creating a realistic and dynamic setup, commonly seen in concerts, theaters, or presentations. Essential elements like the flooring and stage platform were included to provide a base for the scene, with the platform adding depth and elevation. These components were chosen, not only for their necessity in the scene, but also to demonstrate the ability to apply textures and basic transformations to objects in a scene. The safety railing around the stage was added to enhance realism and introduce complexity, showcasing the handling of repetitive objects with slight positional variations. Additionally, the inclusion of speakers, which integral to any stage setup, allow for the application of detailed textures, demonstrating proficiency in working with slightly more complex object designs and texture mapping.

User navigation in the scene is facilitated through keyboard and mouse controls, providing an immersive experience. Standard WASD keyboard controls allow the user to move the camera forward, backward, left, right, horizontally, and vertically, while mouse movements adjust the camera’s orientation, offering a first-person view. This ensures smooth and natural interaction with the scene, with functions like Mouse\_Position\_Callback and ProcessKeyboardEvents managing these inputs effectively.

To maintain modularity and organization in the code, several custom functions were developed and refined to accomplish the project goals. The SetTransformations function, for instance, handles the application of scaling, rotation, and translation transformations, making it versatile for positioning an object within the 3D space. Functions like SetShaderColor, SetShaderTexture, and SetShaderMaterial abstract the complexities of shader management, allowing for an easy application of visual effects across multiple different objects. The LoadSceneTextures and DefineObjectMaterials functions ensure that all of the scene’s visual aspects are correctly initialized *before* rendering, contributing to the scene's overall functionality, ensuring that they render correctly.

The development choices for this 3D scene were made to create a realistic and interactive environment, demonstrating the essential OpenGL functionalities such as texture mapping, shader management, and camera control. The functions developed ensure that the code is modular, organized, and reusable, making it adaptable for future projects or different 3D scenes. For instance, additional objects and materials could be added to the scene in the future in order to flesh out the scene even more, such as a bar in the background or additional lighting, on top of the physical lighting objects that would be found in a venue scene.