Justin Osman

Professor Battersby

4/22/2024

Comp Visualization and Graphics

*Design Decisions and Functionalities of a 3D Modeled Scene*

The idea of three-dimensional modeling lies in its ability to simulate environments that resonate with real-world interactions and experiences. The creation of a virtual scene, reminiscent of an interior garage space, entailed a series of calculated design decisions that aimed to infuse the environment with a sense of familiarity and functionality. The choices ranged from the inclusion of commonplace objects to the meticulous programming of scene navigation and object manipulations.

The objects chosen for the scene were carefully curated to depict the functional space of a garage. Shelving units, a staple for organization, and various tools represent the commonality of the setting. These elements were not selected at random but with the intent to construct a space that is universally recognizable. The shelving units, detailed with texture to suggest materials such as wood and metal, were designed to draw the user's attention to the possibilities of interaction within the scene, such as storage or item retrieval. The tools, resting on the shelves, serve as markers of a lived-in space, alluding to ongoing projects or repairs, thereby enhancing the relatability of the environment.

Navigational functionality within the 3D space was programmed for simplicity and accessibility. The virtual camera serves as the eyes of the user, offering a first-person perspective that allows for an immersive experience. The camera's movement is governed by the standard WASD keys, providing a familiar control scheme for those accustomed to video games or simulation software. The mouse facilitates a seamless look around, translating the user's physical movements into the virtual world's camera angles, enriching the explorative aspect of the scene. Moreover, the integration of mouse scrolling for zoom functions enables users to delve into details or withdraw for a broader view, granting control over their interaction intensity.

The code architecture was approached with modularity and organization at the forefront. Custom functions, such as SetTransformations(), were devised to consolidate repetitive tasks, such as setting object scales, rotations, and positions. This function becomes invaluable as it can be applied to any new element introduced into the scene, minimizing redundancy and promoting code reusability. Similarly, the SetShaderMaterial() function abstracts the intricacies of shader handling, presenting a user-friendly interface for applying materials and textures. This not only streamlines the visual enhancement process for objects but also allows for easier updates or alterations in the scene's aesthetic.

These functions exemplify the principles of efficient coding practices, specifically adhering to the DRY (Don't Repeat Yourself) principle. They represent a foundation upon which the scene can evolve, ensuring that future additions or changes remain structured and manageable. The modular nature of these functions also suggests their potential for reuse in other projects, offering a toolkit for rapid development of new scenes with similar requirements.

The development choices made for this 3D modeled scene reflect a balance between aesthetic considerations and functional needs. They were guided by the overarching goal to produce a scene that is not just visually engaging but also intuitive for users to navigate and interact with. The deliberate selection of objects and careful programming of the scene's functionality serve as the backbone for an immersive and realistic virtual experience. By focusing on modularity and organization, the code serves as a robust framework, adaptable and ready for expansion, embodying the principles of sound software development tailored for graphical environments.