Project One Reflection

Christopher Richards  
[Christopher.richards4@snhu.edu](mailto:Christopher.richards4@snhu.edu)  
08/13/2023

The realm of computer graphics and visualization provides a vast playground for creativity. As budding developers and designers, we've been equipped with tools, mathematical knowledge, and the principles of aesthetics to create something meaningful. For our project, we embarked on the journey of designing a 3D scene, a humble representation of a low-poly model with elements of real-world physics, lighting, and interaction.

**1. Development Choices and Justifications**

When tasking myself with creating a 3D scene, I chose to showcase a serene setting featuring candles atop distressed wooden pedestals, juxtaposed with a candle housed within a glass jar. Hovering above this jar, a lamp, casting a gentle glow, warms the candle, melting its wax in a dance of light and shadow. The question arises, why this scene? The answer lies in the juxtaposition of simplicity and complexity. While the models themselves are low-poly, a course requirement, they still represent the challenges inherent in crafting realistic visuals. The scene’s aesthetics echo tranquility, but the under-the-hood operations are a testament to intricate programming and graphics principles.

However, our scene isn't just about the visual appeal. It's also about understanding the technical intricacies and ensuring our program achieves the desired functionality. Each candle's static point light, while not flickering, serves its purpose in illuminating the scene and adding depth to the visual narrative.

**2. Navigating the 3D Scene**

The user's interaction with any 3D scene largely determines its success. Our scene ensures a seamless and intuitive user experience. Specifically, functions like **UProcessInput**, **UMousePositionCallback**, **UMouseScrollCallback**, and **UMouseButtonCallback** dictate how users navigate within our virtual space. The **UProcessInput** function, for instance, defines the scene's movement mechanisms, allowing users to traverse the environment with ease. Concurrently, the mouse-based callback functions cater to look-around functionalities, zooming in or out, and enabling interactions with objects or changing perspectives. The control over the virtual camera offers an immersive experience, ensuring users can witness the scene from multiple angles and truly appreciate the design intricacies.

**3. Modular Code and Function Encapsulation**

Diving deeper into the programming choices, the project became an exploration of best coding practices, object encapsulation, and modularity. As developers, we’re often advised to prioritize code readability and reusability – and this project provided a real-world application of that advice.

Central to the program's structure are **UGLProp** objects and lighting structures like **GLLight**. These encapsulations not only store properties related to objects and lighting but also contribute to the project’s structural clarity. Such a system of encapsulation promotes scalability and ease of debugging. Each class or structure adheres to its specific responsibility, making future integrations or adjustments easier to manage.

Additionally, our shaders, which bring the scene to life, exemplify the strength of function encapsulation. Each shader function, from **calculateScatter()** handling light scattering to **calculateAttenuation()** focusing on light attenuation, serves a dedicated purpose. These functions not only ensure the shader's optimized performance but also provide clarity and isolation, essential for future refinements or debugging. The modular approach ensures that our program remains adaptable, laying a foundation for future projects.

**Conclusion**

In this journey of 3D scene development, beyond the acquired technical proficiency, lies a broader lesson. It's about marrying art with science, aesthetics with functionality, and user experience with intricate programming. This project wasn't just an academic endeavor; it was a lesson in balancing creativity with technical prowess, and it provided an avenue to apply mathematical and programming knowledge to create something both functional and visually appealing. Through this experience, we've learned not only how to create but also how to reflect on our choices, ensuring our projects are always a step forward in our continuous learning journey.