Philip Gergis

Shuang Zhao

CS 112

11 December 2021

CS 112 Final Project Report

For my project I implemented three different variations of the Cook-Torrance shading model. Initially, I had planned to only use a singular shading model because I thought there was only one way to implement each unique model. However, after reading on the Cook-Torrance shading model I learned about how flexible a shading model can really be. In the end, I implemented the shading model by editing the normal distribution function each time, while keeping the geometry and Fresnel functions the same. The geometry function I used was the Schlick-GGX model which utilizes the roughness value, viewing direction, and light direction. As for the Fresnel function, I used the Fresnel-Schlick approximation which mainly uses the reflectivity of the surface of the mesh. Regarding the normal distribution functions, I ultimately decided to use the Trowbridge-Reitz GGX function (GGXTR), a variance of the Blinn-Phong model's normal distribution function, and the Beckman normal distribution.

The technical aspect of the work in incorporating the three variations of the Cook-Torrance model revolved around understanding the different functions that made up the Cook-Torrance model and how to code them respectively. This includes writing out the equations in code, learning how to access the needed values for said equations and calculating the values, using WebGL to incorporate different values from the UI and have them accessible to the

Cook-Torrance equation, and ultimately using WebGL to acquire the calculated result and project it as a color in the model.

In the beginning, it was difficult to identify which values were which in the equation and ultimately how to access them. Consequently, I spent a large amount of time trying to determine which variables in the different functions were which since they used several different new values. However, after comparing multiple functions online, as well as the ones present in PA2, I was able to discern the values of each variable in the functions and how to access them in the project. From there, it was a matter of creating new variables to hold different values for each variation of the model which proved to be troublesome due to my lack of knowledge on how WebGL works. By utilizing the examples in PA2 and other online examples, I was able to change the UI to correctly transfer values from my code to the model and vice versa, allowing me to incorporate new values and add new models to the project. Also, by doing so most of the values, including roughness, refraction, light power, and more, are changeable instead of relying on brute force values, as shown in the UI of the project. Overall, most of my difficulties regarding my final project revolved around my lack of knowledge on how WebGL operates and my lack of understanding as to how shading models are rendered.

The most significant things I learned from this assignment include how every component of shading models play into creating said models and how to better code in WebGL. Prior to this project, the only experience I had with shading models was working on PA2. However, the project was very simple and the actual implementation of the UI and most of the values were accessible to use. Because of this, I was able to simply plug values in and compare the result despite not understanding what I was doing at all. On the contrary, when looking up the different normal distribution functions for Cook-Torrance, the confusion that I was faced with when trying to understand these formulas inevitably led to a deeper understanding of how shading models work. For instance, I never understood the purpose of the normal distribution functions prior to this project, but now understand that for models similar to the Cook-Torrance they are used to calculate the relative surface area of microfacets exactly aligned to the halfway vector. In addition, I was unaware of what the variables in the equation stand for, not knowing that k_d and α represent the refracted light energy and roughness in the model. Additionally, having to code using WebGL on our own conveyed how inexperienced I was in applying the information from class in a practical setting. Before this assignment, most projects felt as if I were simply calculating mathematical equations or problems rather than 3D modeling. However, the final project allowed for me to get first hand experience in using WebGL to format the project in my own way and felt as if I accomplished, to some extent, what real 3D modeling should be. Lastly, the main thing I learned from the project was how different functions affect the shading model and why. In conclusion, the project gave me a deeper understanding of how shading models work that was unclear to me prior, and allowed me to gain experience applying my knowledge from class for the first time.