Giuliano Costa
11/3/22
CSCI 4229

Progress Report: Interactive Path Tracer

I am still in the discovery phase of this project. But I have been learning a lot. That said, it is proving far more complicated than I could have imagined. I am becoming particularly confused between the number of extensions available and different methods. Luckily I have found a few working repositories online that use GLEW, SDL2, and GLSL. For most of these projects, the performance is surprisingly good on my little Intel integrated graphics CPU. The below render is after only a few seconds on my laptop. Which gives me faith that I'll be able to do the whole development on it instead of my GPU powered desktop.

My goal was really just to write a simple GLSL shader that computed ray intersects using Monte Carlo approximation. Since path tracing/ray tracing will naturally make smooth realistic shadows, that was all I wanted. But I am seeing people use all sorts of other backends with OpenGL, such as Vulkan, CUDA, DirectX, etc… Which I'd rather not learn. At this point I am considering just doing the project in ShaderToy.com, which has a nice web container for GLSL.

Please let me know if doing the project would be acceptable in ShaderToy. From my understanding it is one of the most popular ways for people to show off their GLSL shaders. Otherwise I will just stick to a GLEW/GLFW context in C++ and try to do it the more tedious way.

Render from a GLSL path tracer that uses SDL2. Far too complicated to reproduce unfortunately.
https://github.com/knightcrawler25/GLSL-PathTracer/