



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

A parallelised ray-tracing program



School of
Applied Computing

Sam Mounter P144579

School of Applied Computing, University of Wales Trinity Saint David, Mount Pleasant, Swansea, SA1 6ED

Email: smounter.144579@student.uwtsd.ac.uk

Introduction

In computer graphics, ray-tracing is when a series of light rays are cast from an image plane into a scene, with the colour of each pixel determined by how its corresponding ray interacts with the objects.[1] Due to the computational costs required, however, this technique tends to be used more often in pre-rendered imagery than with real-time systems.

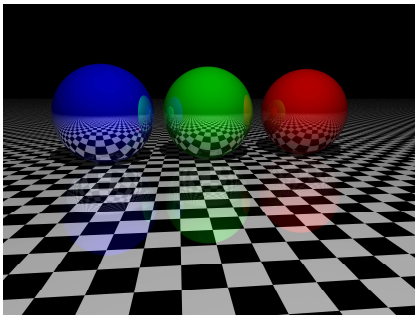


Figure 1: Reflecting spheres[2]

Background Research

Studies conducted in this field are conflicted about how rendering times should be reduced. Some proposals recommend a hybrid model, whereby textures imitate reflections,[3] while others prefer GPU parallelisation.[4] The latter technique will likely be the more feasible one due to complexity and time constraints, so that will be the focus of the impending experiment.

Aims and Objectives

The key goals of this project are to conduct research into ray-tracing techniques and parallelisation, design and build a C++ ray tracing program with this research, and test this program when multithreaded through the CPU and then the GPU. The rendering times of the program when run on the two units will be compared.

Development Methodology

This project will follow a positivist research philosophy, which focusses on data observation, with deductive thinking used to monitor cause-and-effect relationships. Previous research has suggested that images are rendered faster when information is multithreaded through the GPU, and this research is expected to continue the pattern.



Figure 2: Comparison of ray-tracing libraries[4]

Several risks have been identified, which include a potential lack of knowledge on the subject, and an unsustainably large amount of work to complete. These can be avoided with thorough research and requirements analysis early on in the project.

Work cited

- [1] K. Suffern, *Ray Tracing from the Ground Up*. Boca Raton: CRC Press, 2016.
- [2] Shen, "Raytracing_reflection," *Wikimedia Commons*, 2006. [Online]. Available: https://commons.wikimedia.org/wiki/File:Raytracing_reflection.png. [Accessed: 03-Oct-2017].
- [3] M. Fridh, "Hybrid Ray-Traced Reflections in Real-Time: in OpenGL 4.3," Blekinge Institute of Technology, 2015.
- [4] D. Norgren, "Implementing and Evaluating CPU/GPU Real-Time Ray Tracing Solutions," MALARDALEN UNIVERSITY, 2016.