Project Proposal: Geocold Ray Tracer (Differentiable?) in C++

Authors: Prakash Chaulagain, Nishar Arjyal, and Pramish Paudel

Roll Numbers: 076BCT045, 076BCT042, 076BCT047

Submitted to the Department of Electronics and Computer Engineering in Partial Fulfillment of the Requirements of the 3rd Year **Computer Graphics** Course at

Pulchowk Campus

IOE, Tribhuwan University

June 27, 2022

Accepted by: .....

Mr. Basanta Joshi
Assistant Professor at
The Department of Electronics and Computer Engineering

Date of Submission: June 27, 2022
Expected Date of Completion: July, 2022

Geocold

by Prakash Chaulagain, Nishar Arjyal, and Pramish Paudel

Submitted to the Department of Electronics and Computer Engineering
on June 27, 2022
in Partial Fulfillment of the Requirements for the 3rd Year Computer Graphics Course in Computer Engineering

Abstract

Ray tracing has a rich history in the history of computing and computer graphics. With this manuscript, we propose to build an offline ray tracing software using the Vulkan graphics/compute library in C++. Our renderer is supposed to work generically, as in take as input any file containing geometric data, perform a mesh render pass in order to render a mesh of the described scene and then perform ray tracing with a separate pass. In this paper, we cover the mathematical principles that we follow as we build our ray tracing software.

Table Of Contents

1 Acknowledgement

Our project idea is the product of excellent supervision of all of our instructors, most notably our lecturer Mr. Basanta Joshi. We could not have been able to develop interest in computer graphics as a field of study without the constant inspiration provided to us by all of our lecturers and lab instructors and assistants. Some of the credit should also go to the college administration for their efforts in the smooth functioning of all of our classes and labs safely and securely despite the unprecedented times of the pandemic.

2 Objectives

- To understand the graphics pipeline.
- To become familiar with modern GPU architectures.
- To become familiar with GPU programming models and GPU computing (massively parallel computing).
- To understand and uncover existing ray tracing techniques.
- To gain a degree of familiarity with common graphics and GPU compute APIs, and understand their abstraction mechanisms.

3 Introduction

Rendering is the task of taking a scene composed of many geometric objects arranged in 3D space and computing a 2D image that shows the object as viewed from a particular viewpoint. The goal of our project Geocoldis to implement an obj file loader which then creates a mesh of our scene, then finally we implement a ray tracer which will correctly color every object in the scene. Over the next few sections, we will try and set the mathematical basis/principles used in our project and the API that we have attempted to design based on those principles.

4 Mathematical Basis

4.1 On Basic Geometric Primitives

5 Existing Systems

The package is suppposed to have the following directory structure:

```
Geocold
├── papers
├── src
├── testbench
└── external
```

5.1 System Block Diagram

The following is a block diagram of our system:

6 Project Scope

7 Project Schedule