Introduction to Computer Graphics: Intermediate Report

João Capucho 113713

April 10, 2025

Introduction

My project is based around recreating the concept behind the *Portal* series of games. *Portal* is a series of puzzle games where the player is given access to a device capable of putting a pair of portals in surfaces that instantly transport any object that comes into contact with one of them to the other.

Figure 1: Image from the game *Portal* 2

The project doesn't yet have a name. The project is hosted at `https://github.com/JCapucho/ICG` and a working version can be accessed through `https://jcapucho.github.io/ICG/`.

Expected results

The final version is expected to implement a 3D environment where the user controls a character that has collisions and follows some basic laws of physics, like gravity and inertia. The user can then press a button to cause a projectile to be spawned with an initial velocity and isn't affected by any acceleration. Once it hits a surface this projectile will spawn one portal, and when two are spawned they will link allowing the player to walk into one and come out from the other.

The environment will be modeled with dynamic lights and using textures (taken from the internet) using Physically Based Rendering (PBR) provided by three.js. The character will use a model that contains animations (taken from the internet) that will be played according to the user's input (idling, running, falling, ...).

A UI for a main menu and pause state will also be developed.

What is done

Currently, the recursive portal rendering is mostly working. This required setting up a camera per portal, that renders onto a texture instead of the default framebuffer. This presented a cyclic dependency between the texture that needs to be rendered as a portal and will be rendered to. To fix this an intermediate texture is used that the camera renders to, and later is copied into the texture that the portal draws.

The portal cameras also needed to react to the player's camera to simulate the player "peeking" into the portal from different angles and positions. This was implemented through some vector and quaternion math.

A physics engine *Rapier* was also integrated for collision detection and force generation. The player character is fully integrated into the physics world, allowing it to move around and collide with floor and other objects.

What is missing

The portal rendering still has some problems with frustum culling that causes it to render objects behind the portal. This will need a new projection matrix to be calculated per camera, to fix the clipping plane.

The portals need to be made interactive by allowing objects (such as the player) to enter through one end and exit the other while conserving momentum.

The portals are statically created, while it's expected that the user can create new portals by shooting projectiles.

Finally, a complete level showcasing the game mechanics needs to be modeled and implemented.