CS 148 Final Project Report

George Song & Siolé Mayeski

I. Introduction

Our project was inspired by the popular 2D game, Geometry Dash. For this project we wanted to reimagine what its world might look like if its recognizable characters, obstacles, and game mechanics existed in a fully 3D environment. We designed the scene around familiar elements from the game, including the cube character, spikes, rings, jump boosters, and other obstacles, while giving them depth, materials, lighting, and spatial relationships that would not exist in the original 2D game.

We wanted to maintain the bright, colorful, arcade-like aesthetic of Geometry Dash while using Blender and Cycles to make the environment feel more dimensional. Rather than attempting to make the scene totally photorealistic, we focused on creating a stylized 3D interpretation, staying true to the game graphics that captivated us as children (and even now), while still making use of realistic lighting effects we've learned in this course such as reflections, shadows, emissive surfaces, and depth of field.

In addition, instead of sticking to the standard side-view used in the actual game, we explored multiple perspective options. In the end, we were between the POV of the cube and the off-angle we ultimately went with. We made this choice because we felt that using the cube's POV would hinder what is actually visible in the frame, since the environment is relatively compact. By using this off-angle perspective, we are still able to exhibit the structures that make Geometry Dash so captivating.

II. How We Met the Requirements

The primary Geometry Dash-inspired objects in the scene were modeled from scratch in Blender. We recreated recognizable game elements as 3D geometry rather than importing finished models of them. These included the main cube character, multiple spike obstacles, circular rings, jump-related objects, and other environmental elements. We used basic Blender meshes such as cubes, cylinders, spheres, cones, and toruses as starting points and modified, combined, scaled, extruded, and positioned them to create the final objects.

In Geometry Dash, there is a glowing orange sphere and glowing blue pad that you can jump on as a booster jump that we featured in the scene. For the glowing orange sphere, we created a procedural texture using a Noise Texture connected to a Color Ramp, then adjusted the colors to create blended orange and yellow variations across the surface. We also used the noise to add subtle surface variation, which helped the sphere look more like a glowing, textured object instead of a flat solid color. For the cyan glowing pad, we first UV unwrapped the object and

created a custom image texture. In Texture Paint, we painted a cyan base and then added darker blue shapes and lines by hand. We used the Smear and Blur tools to blend the blue and cyan together, which created a softer, glowing spiral effect. We then connected the painted texture to an Emission shader so the painted colors appeared illuminated in the final scene.

To create the glowing light trails behind the characters, we modeled a series of small square shapes and placed them sequentially along the characters' paths of motion. We then applied bright colored materials with Emission shading so that the squares appeared to glow rather than reflect light like a normal surface. By varying their placement and spacing behind each character, the repeated glowing squares create the appearance of a fading motion trail and help reinforce the fast-moving arcade aesthetic of Geometry Dash.

For an advanced Blender feature, we used camera depth of field. The camera was given a specific focus object so that the main Geometry Dash cube could remain in focus while objects located at different depths in the scene became progressively blurred. The camera aperture and F-stop were adjusted to control the strength of the effect.

For the background lighting, we added several spotlights near the back wall and arranged them so that the beams spread outward from one point behind the main objects. To make the beams visible, we added a thin cube in front of the backdrop and gave it a Principled Volume material. This made the light scatter within that area so that the beams could be seen without making the entire scene look foggy (which was our initial problem without the bounding cube). We adjusted the spotlight power, angle, color, and placement until the beams were visible but did not overpower the rest of the scene. We used a light blue color for the spotlights to match the blue lighting in the reference image. This use of volumetric lighting also gave us another advanced Blender/Cycles feature and helped us take advantage of the ray-traced lighting required for the project.

We also added glowing outlines to the cubes and spike obstacles. For these objects, we created a second material and made it emissive. We then added a Wireframe modifier and changed the material offset so that the wireframe used the glowing material while the rest of the object kept its original material. This created the bright outlines around the edges of the obstacles. For the spikes, we replaced the original smooth cones with square pyramids because the wireframe effect worked much better on geometry with clear edges. Because the outlines were created using actual emissive materials and geometry in the scene, the effect was produced directly in Cycles without compositing or post-processing.

III. Who Did What

George created the initial layout of objects as a basic starting point and worked mainly on the lighting and visual effects. He added and positioned the background spotlights and created the volumetric cube that made the light beams visible. He also experimented with the Cycles render

settings, sampling, light paths, and material properties to improve the lighting and reduce noise in the final image. For the obstacles, he created the glowing edge effect using emissive materials and Wireframe modifiers, and replaced the cone-shaped spikes with square pyramids so the effect worked better. He also added the main area lighting and adjusted the floor material and reflections to make the scene look cleaner and reduce graininess.

Siolé created the geometry for the circular portals, the glowing spheres, the rocket, and refined the Geometry Dash-inspired characters. She worked on the materials, textures, and shading for these objects, including the procedural orange sphere texture and the hand-painted cyan jump pad texture. She also added emissive materials to the glowing game elements and created the light trails behind the characters and rocket using repeated small shapes. In addition, she adjusted the placement, scale, and appearance of these objects so they fit the Geometry Dash-inspired scene and worked well from the final camera view.

Both George and Siolé worked together on the depth of field by adjusting the camera focus, focus object, and aperture settings until the main characters remained clear while objects at different depths were gradually blurred.

IV. Downloaded vs. Self-Made Assets

All assets in the scene were self-made using the built in mesh geometries in Blender. We did not download any external 3D assets.

V. Tutorials Referenced

For the spotlight and volumetric lighting, we followed this YouTube tutorial as a starting point for creating the background spotlights and making their beams visible. We used the same general setup, but changed several parts to better fit our scene. For instance, instead of applying the volume effect to the entire world, we created a thin cube with a Principled Volume material behind the main objects.

For the glowing obstacle edges, we closely followed this YouTube tutorial and applied the same method to the different obstacle shapes in our scene.