

Alien World open gl animation project

I did this project using c++ and opengl. It includes multiple animations including aliens rolling around on wheels, a missile that can be fired and a spaceship that can liftoff. Everything is constructed using geometry glut library and math with spaceships and aliens being made of many smaller shapes.

Storey

On a remote desert planet a group of aliens have set up their perimeter. They got scared of the ancient pyramids and so decided it was time to leave. They are rolling around their perimeter, getting warm so they can help spin the spaceship to lift off. One of their friends has gone rogue and gets left behind as holding a dangerous weapon.

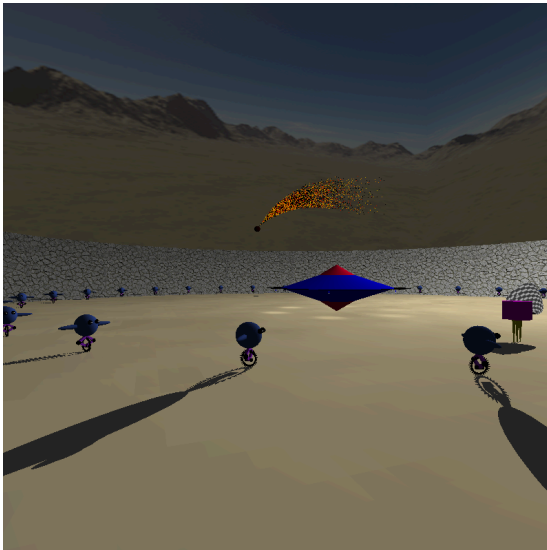


Fig 1

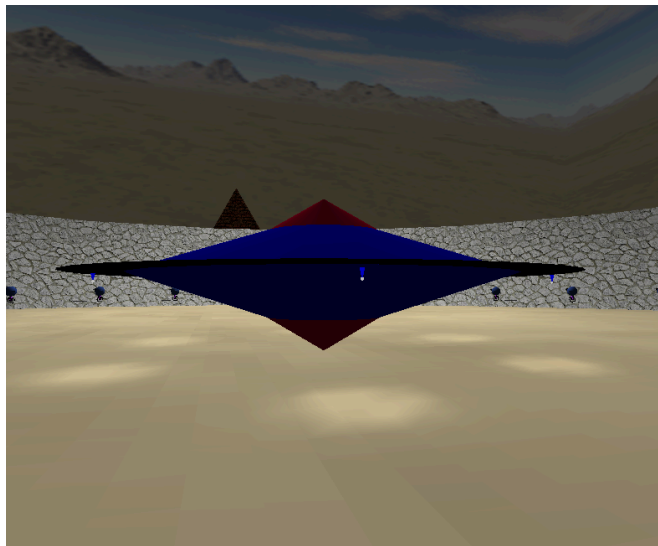


Fig 2

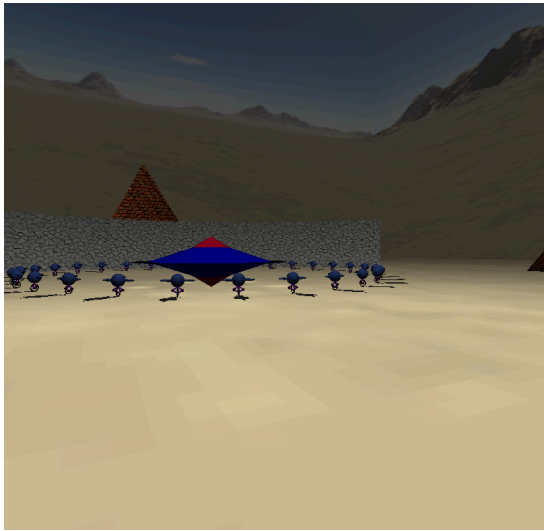


Fig 3

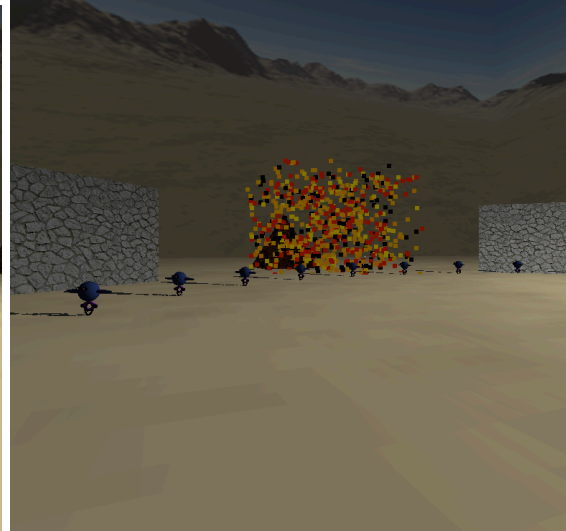


Fig 4

Extra features

Planar shadows:

Alien with weapon has shadow based of sunlight

Projectile fire with right ctr key has planar shadow from sunlight

Aliens that are moving the scene each have planar shadow

Spotlight on moving/rotating object

Shown in figure 2. Spaceship has 6 spotlights on the underside that are shining on the floor.

These rotate as the ship rotates and can see the move faster when ship takes of.

Skybox: Is implement as seen above with clear sky and sandy mountains in distance

A custom-built sweep surface:

Perimeter wall spaceship and aliens is a surface of revolution using quadstrips. Was generated by looping over n_strips and updating θ each time. θ is then used to rotate around the y axis. I did this by setting the x and z coords to be based of \cos and $\sin \theta$. I used 0 and $height_wall$ for y coords.

A surface shape generated using a mathematical formula:

I generated a paraboloid for the main weapon using quad strips and mathematical formula.

Formula created based on Idea of wanting total angle to be 360 degrees - so Will divide that

By amount of slices, for how many quads there wall be around the 360 degrees

$2 * m_pi$ is to convert to radians.

$\theta_1 = (1.0 * i / slices) * 2 * M_PI;$

$\theta_2 = (1.0 * (i + 1) / slices) * 2 * M_PI;$

$X1 = radius * \sqrt{height} * \cos(\theta_1)$

$Y1 = \text{radius} * \text{sqrt}(\text{height}) * \sin(\text{theta}1)$

And $z = x^2 + y^2$ Is easily adjustable with slices variable, and height, and radius

Physics models:

projectile_z = z_velocity * time; // D = VT equation

projectile_y = 0.5*(y_velocity) * time; // Height equation $y = 0.5v * t$ with

// v being lowered by gravity

// z/ travel direction is lowered by air_resistance.

y_velocity -= g; // gravity

z_velocity -= air_resistance;

Spaceship liftoff: Slowly speeds up including help from aliens, before lifting off

Particle systems:

Projectile has fire behind it, and explosion.

Control Functions

void special(int key, int x, int y)

Special keys: For special glut keys including arrows keys and right ctr key

Up and Down keys move the camera forward and backward.

Left and right rotate the camera by 5 degrees until left and right key are released

void specialup(int key, int x, int y):

Checks when the arrow keys are released, this so can have smoother camera movement

void keyboard(unsigned char key, int x, int y):

Space bar down to prepare and launch ship. Takes time to gain speed.

Build commands.

Was developed using Visual Studio code on the lab computers.

Build command: `/usr/bin/cmake --build /csse/users/ber30/2024/Graphics/Assignment1/
alienProject/build --config Debug --target AlienWorld.out -j 22 -`

References.

For paraboloid equations

https://math.libretexts.org/Courses/Misericordia_University/MTH_226%3A_Calculus_III/Chapter_12%3A_Vectors_and_the_Geometry_of_Space/12.6%3A_Quadric_Surfaces

For particles – I didn't realize was gonna be lecture slides on it at the time

<https://learnopengl.com/In-Practice/2D-Game/Particles>

Name: Bradley Erskine

Date: 25-6-24