GEIL0

Gameplay

3D Geometry

Source - https://learnopengl.com/Model-Loading/Model

The game 3 distinct models:

- Spaceship
- Asteroids
- Map

Spaceship and asteroids are free open source models from https://www.cgtrader.com , the map was built using blender – it's texture, again, being a free sample image from https://opengameart.org/textures/all.

Playable + Advanced Gameplay + Win/Lose Condition

The objective of the game is to complete three laps without crashing into asteroids more than two times. The player's got three lives, each hit (indicated by the word "hit" hovering over the spaceship and being printed in the console) subtracts one life. The player is moving on a cubic hermit spline, automatically following the map curvature, to evade asteroids, however, the player can jump or move sideways. (both within fixed ranges of motion)
This is true to the notion of old racing games like F-zero and the like, basically combining rail shooter controls with quick evasion manoeuvrings.
The lap times as well as the final summarized time is tracked and printed in the console after each lap and after completing the game respectively.

The player wins the game if they manage to complete three laps without dying. The game is lost if the player crashes into the third asteroid. Both win and lose messages are generated in the game and printed to the console.

Min. 60FPS and Framerate Independence

Please deactivate VSync or GSync for testing this if possible. I couldn't deactivate VSync on my PC (Geforce Experience seems to override NVIDIA settings) and could only play my game in a 60fps lock mode.

Intuitive Controls

I – Switch from game cam to ECG/Free motion cam

B – Toggle Bloom

W – Accerlerate

A – Move to the left

D – Move to the right

SPACE – Jump

Intuitive Camera

The game camera is clamped to a fixed point a little above and a varying distance behind the spaceship. As usual in games of this sort, the camera is not rotatable by the player. However, to include a free moving camera, the ECG cam has been left in the game and is able to freely move around the scene. This is achieved with the usual ECG camera controls. (right click to strafe)

Press I to switch between game and debug mode/cam.

Illumination Model

There are 134 point lights and one directional light in the game. Bloom is implemented as well, more about that in the effects section.

Textures

The game contains multiple spaceship textures, one asteroid texture, one map texture and 72 frames (=3 seconds at 24 fps) of a clip implemented as video texture.

Moving Objects

The spaceship moves.

Documentation

See here.

Adjustable Parameters

The game's parameters can be adjusted in the settings file. (bin/assets/settings.ini) This is also where cel shading can be turned on and off.

The following settings are adjustable:

- width = 1280
- height = 768
- refresh_rate = 60
- fullscreen = true
- title = GEIL0
- near = 0.1
- far = 500.0
- celShading = false
- brightness = 1

Collision Detection

Each asteroid has a box collider, so does the spaceship. When two box colliders collide, the spaceship isn't decelerated to a complete halt but rather just decelerated a certain amount. This decision has been made to improve the game flow.

When colliding with a asteroid the player also loses one life.

Effects

CPU Particle System

Source: http://www.opengl-tutorial.org/intermediate-tutorials/billboards-particles/particles-instancing/

The spaceship shoots fire particles out its back.

Hierarchical Animation

The ship's tail fin moves with the steering direction.

Video Texture

At the game start a screen shows a scene from F-Zero GX, the inspiration for this game. The clip is exactly 3 seconds long. (72 frames with 24fps) The clip is usable under creative common rights, it is taken from the game's official trailer from 2002.

Cel Shading

Cel shading can be activated by setting celShading to "true" in the settings file.

Bloom/Glow

Source: https://learnopengl.com/Advanced-Lighting/Bloom

Bloom can be toggled by pressing the B key.

Included libraries

ASSIMP has been included for model loading.

Source: https://www.assimp.org/