Computer graphics final project

TITLE : 3d car game

GROUP MEMBERS
1 SAMUEL GIRMA - UGR/6303/12
2 FENET SHEWAREGA - UGR/9003/12

Introduction

Our final project is a 3D automobile game utilizing opengl and pygame, with the whole code written in Python. This was one of our challenges because practically all reference code was written in C++. So the first task was to comprehend the entire notion from the prepared code provided on github and transfer it into Python code. Our assignment was completed in three dimensions, as instructed by our lecturer. We just started creating the entire race and the automobile using blender, as suggested by the teacher. Because it is a system agnostic interface, OpenGL enables a wide range of operations to help us accomplish our jobs more efficiently. Indeed, opengl gives a variety of viewing capabilities for our project, which creates a variety of single objects.A minimal collection of transformation functions allows us to acquire viewing and modeling capability. We also noticed something concerning rotation while using the open gl instructions.

Computer graphics

Computer graphics is the art of drawing images on computer screens using programming. It entails computations, data creation, and manipulation. In other words, computer graphics is a rendering tool used for image generation and manipulation. Understanding how light interacts with objects in the physical world and simulating those interactions as closely as possible on a computer are the most accurate and realistic techniques. Reflections, transparencies, and diffuse lighting can all be modeled using a variety of different algorithms, some of which are designed to be physically accurate while others are designed to be computationally efficient. Virtual reality imagery must be created in milliseconds, whereas detailed architectural renderings can take hours to complete..

OpenGl

Allow me to provide some useful information about openGL as an introduction.

OpenGL (open graphics library) is a standard specification that defines a cross-language, cross-platform API for creating 2D and 3D computer graphics applications. Over 250 different function calls are available in the interface, which can be used to create complex 3D scenes from simple primitives. Silicon Graphics Inc. (SGI) created OpenGL in 1992, and it is widely used in CAD, virtual reality, scientific visualization, data visualization, and flight simulation. On Microsoft Windows platforms, it is also used in video games, where it competes with direct 3D.

OpenGL serves two primary functions:

• To hide the complexities of interacting with various 3D accelerators by providing programmers with a single, consistent API

• To conceal the varying capabilities of hardware platforms, all implementations must support the full openGL feature set.

OpenGL has historically had an impact on the development of 3D accelerators, promoting a foundation level of functionality that is now common in consumer-level hardware:
• The basic primitives are rasterized points, lines, and polygons.
• A lighting and transformation pipeline.
Texture Mapping. Alpha Blending. Z Buffering.

Tkinter

Tkinter is the way to create Graphical User Interfaces (GUIs) in Python and is included in all standard Python distributions. In fact, it is the only framework included in the Python standard library. This Python framework interfaces with the Tk toolkit and acts as a thin object-oriented layer on top of Tk. The Tk toolkit is a cross-platform collection of 'graphical control elements,' also known as widgets, for creating application interfaces.

Project Goal

The aim of this project is to develop a 3D graphics car game using transformation operations like translation,rotation, etc on such objects. The package must also have a user-friendly interface that may be menu-oriented, iconic or a combination of both .

Software requirement

- Programming language :python using open gl
- Operating system : windows operating system
- Compiler - vscode

- Graphics library - GL/glut.h
- Python tkinter
- Blender

Hardware equipment

- Ram :2GB and higher
- Keyboard QWERTY keyboard
- Mouse 2 or 3 button mouse

Files

Object (folder): There are.obj files for the blender inside this folder, as stated above, and blender imported files using the app blender.
Blender: is a free application that can be used to design all of the vertices of the objects used in the projects that are imported from blender because every object must first be designed in a blender. Because we are new to this free app, one of our challenges is here. We take our time in order to fully comprehend it.

We put these.obj files containing the vertex used inside the.py file inside the object folder, as listed there, and there is car.obj file which concerns the vertex of the designed car. When they are designed, the vertex environment.obj led.obj road.obj is also used to represent them.

Shader: We can see.txt and.shader files in this folder.
Thus, the files contain shader properties used by the godot engine, a free open-source game engine used to create 2d and 3d games; however, because our project is about 3d car games, we will only see 3d. It uses the godot shading language to store various data types and functions.

carfragmnet.shader, carobject.shader, fragment.txt, and vertex.txt are all included.
Textures: This folder contains.jpg files.
maingame.py contained the entire main code for the 3D car game, which imported all other files.

Implementation

When it came to the implementation part, as stated in the requirements, we used the Python language for implementation as our instructor instructed, and within it we used OpenGL, pygame, and tkinter separately. To run our project, we will pivot in maingame.py as usual. As you can see at the top of this python file, there are imports that connect our maingame.py to other requirements within the file. The purpose of tkinter in this game is to first announce how to play to the player, so we use it as the game's first screen, which contains the game's instructions.

- To move into right
- To move into left
- To move into up
- To move into down

Because the player is new to our game, we need to know how he or she plays. In addition, two buttons on this screen serve as linkers to the entire pygame window.

- Start
- Cancel

As the name implies, it was used to start and stop the game in this level prior to the display of the pygame window.

Then, if the user wants to play the game, he or she can press the start button, and we will see an additional window on the screen that contains both the road and the car inside their environment.

Constraints

We are up against a challenge in this project. Starting with the blender application and the car translation, these two points were a roadblock for us, but with some work, we'll be able to overcome them. However, when it comes to the last one, we are unable to make it as perfect as required. We are attempting to create motion in the car, but after all, we attempted to work on the road. We believe our instructor will take this into consideration.

Reference

https://github.com/totex/Learn-OpenGL-in-python - this github link used for camer.py objectloader and textureloader files
Developing graphics frameworks with python - the whole part