Graphical Processing project documentation

Atacama desert invaded by aliens

Author: Livia-Maria Mitrică
Group 30434/2

Contents

1 Subject specification **2**

2 Scenario **3**
2.1 Scene and objects description 3
2.2 Functionalities 3

3 Implementation details **4**
3.1 Functions and special algorithms 4
3.2 Graphics model 4
3.3 Data structures 4
3.4 Class hierarchy 4

4 Graphical user interface presentation / user manual **5**

5 Conclusion and further developments **8**

6 References **9**

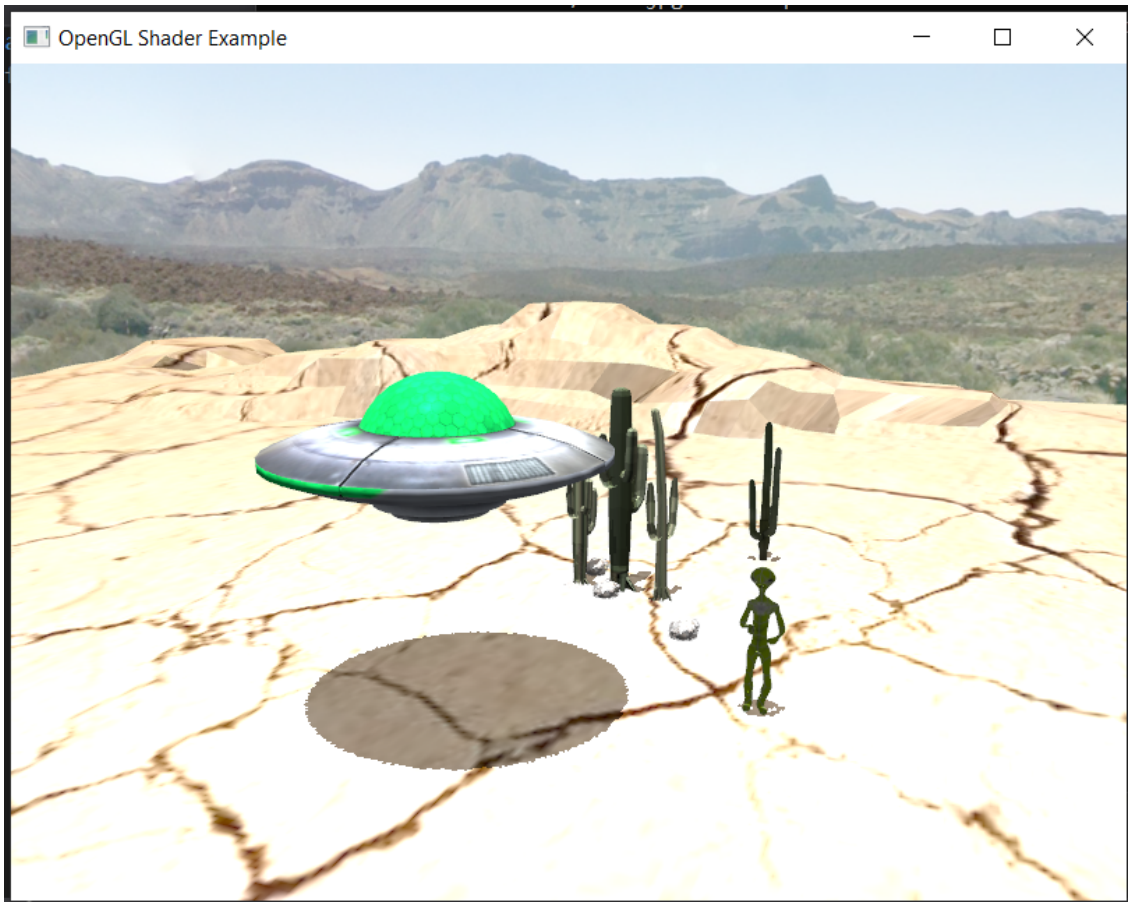
1 Subject specification

The subject of the project consists in the photorealistic presentation of 3D objects using OpenGL library. The user directly manipulates by mouse and keyboard inputs the scene of objects. The scene must contain multiple sources of light, along with shadow mapping techniques.

2 Scenario

2.1 Scene and objects description

The scene is inspired by the desert Atacama, being characterized by a variety of species of cactus along with other plants adapted to intense heat. In order to make the scene more striking, I introduced some SF elements, namely an UFO and an alien, as if they were trying to conquer the desert. The objects were taken from sources [2]-[5] mentioned in the references. The ground was designed using Blender.



2.2 Functionalities

The user is able to move in the scene using both mouse and keyboard. Viewing in a panoramic manner is also possible. Other functionalities include fog, making the UFO fly, and move the alien in the scene. (Instructions on how to use these features can be found in Graphical User Interface section)

3 Implementation details

3.1 Functions and special algorithms

As far as special algorithms are concerned I intended to detect collision, therefore creating bounding boxes for my objects. However, I realized that after translating an object, the bounding box would not change the coordinates, given that I was only taking min's and max's from the objects' vertices. Unfortunately, I was not able to find a solution for this. One of the useful materials about this I found it here, but it was only giving examples for 2D objects.

3.2 Graphics model

For representing the objects in the scene I used polygonal model.

3.3 Data structures

Apart from the data structures defined in the basic starting project, I have not defined any other data structures.

3.4 Class hierarchy

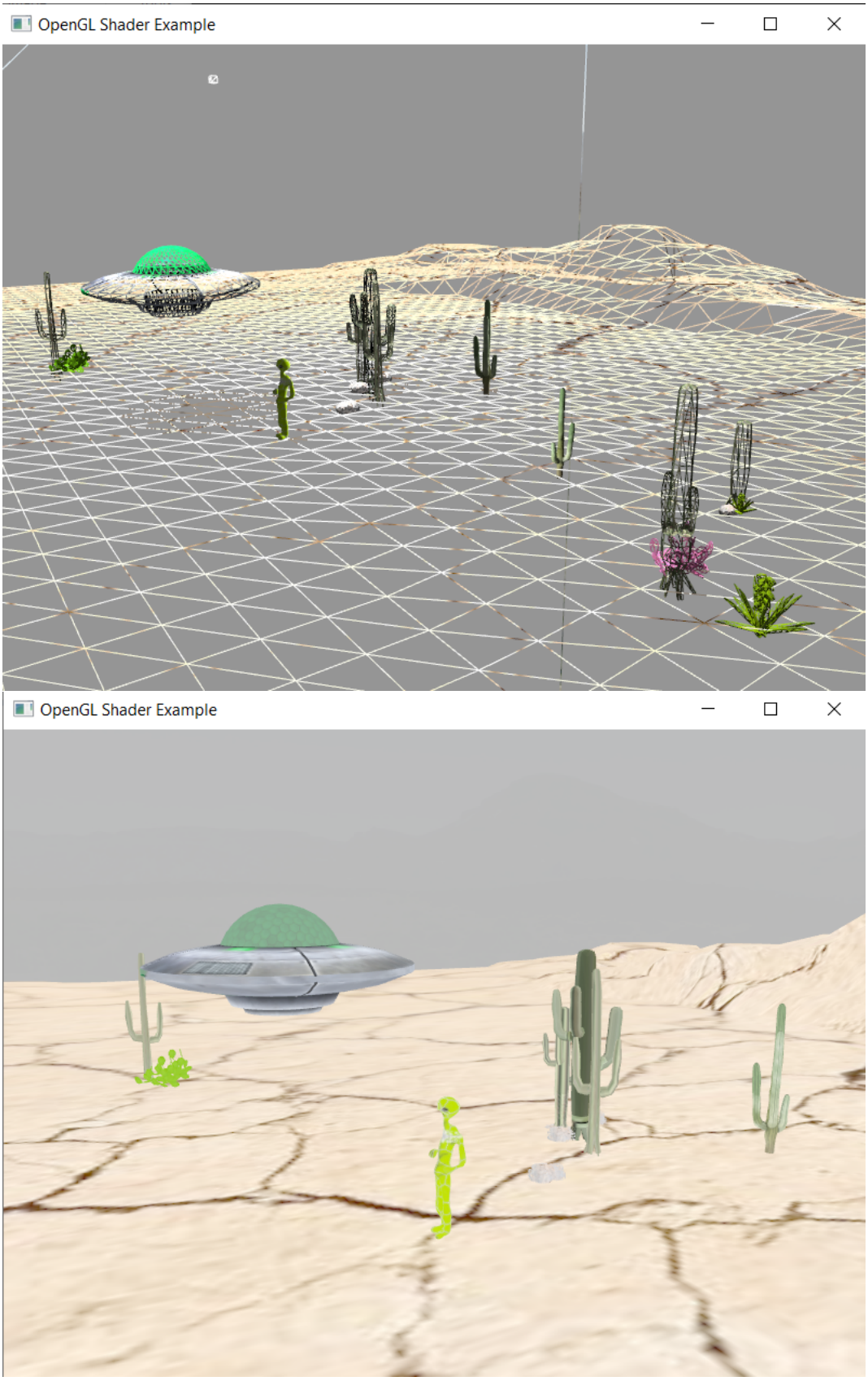
Classes used:

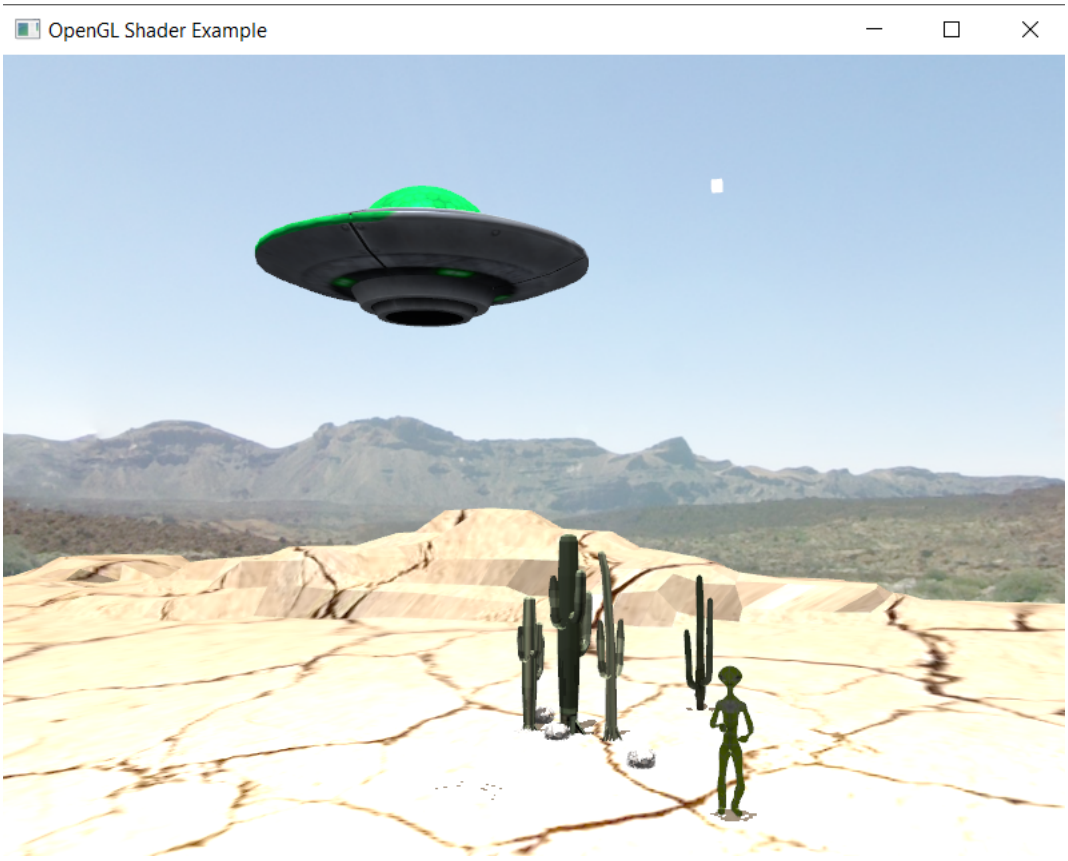
- Camera class [1] - contains attributes and functions that allows the user to freely move around in the 3D scene.
- Mesh class - contains information about object components, like verticces, indices, textures. It is also used for drawing the model.
- Shader class - responsible for reading, loading and compiling the shader files used in the application.
- Skybox - responsible for reading, loading and drawing the images used for the skybox.

4 Graphical user interface presentation / user manual

Functionalities and corresponding keys:

- Change light angle - J(decrease), L(increase)
- Move camera - S(backward), W(forward), A(left), D(right)
- Polygon fill - V
- Polygon line - B
- Polygon point - F
- Polygon smooth - N
- Line Smooth - C
- Animation - flying UFO - UP
- Fog - F
- Rotate UFO object component- Left, Right
- Move alien - 4 (left), 6(right), 2(front), 8(back)
- Rotate alien - Q(left), E(right)
- Panorama view (camera animation) - P


OpenGL Shader Example
OpenGL Shader Example


OpenGL Shader Example

5 Conclusion and further developments

The scene can be altered in many possible ways, from adding more objects in the scene to creating new animations and features and leaving the scene control for the player. One important further development would be succeding the collision detection and resolution, but this would take more time and research.

To sum up, OpenGL is a powerful low-level API that requires a good knowledge of the graphic pipeline and C++ language in order to be useful.

I found it really interesting to work on this project and also to discover how difficult rendering graphics and animations can be.

6 References

- [1] https://learnopengl.com/Getting-started/Camera
- [2] https://free3d.com/3d-models/cactus
- [3] https://www.cgtrader.com/free-3d-models?keywords=cactus
- [4] https://www.cgtrader.com/free-3d-models?keywords=ufo
- [5] https://www.cgtrader.com/3d-models?keywords=aliensuggested=1