

# Graphic Processing

## 3D scene of an airport

Student: Cristian Casian-Cristi

Group:30432

## Contents

1. Subject specification
2. Scenario
  - 2.1 Scene and object description
  - 2.2 Functionalities
3. Implementation details
  - 3.1 Function and special algorithms
    - 3.1.1 Possible solutions
    - 3.1.2 The motivation of the chosen approach
  - 3.2 Graphic models
  - 3.3 Data structures
  - 3.4 Class hierarchies
4. Graphical user interface presentation / user manual
5. Conclusions and further developments
6. References

## 1. Subject specification

The project alleged realization of a 3D foto-realist scene using OpenGL. Also, the project implements a bunch of effects for rising the realism of the scene like: shadows, fog, animation.

## 2. Scenario

### 2.1 Scene and object description

The scene represents an airport. Here we have the airport terminal, also an runway for the airplanes and for the helicopter, a park for the cars and a lake. Almost all of the objects were exported in a single obj format, less the objects which has an animation.

The 3D objects that I used are:

1. Grounds
2. Airplanes
3. Helicopter
4. Heliport
5. Helicopter
6. Airport terminal
7. Formula1 car
8. Control tower
9. Ducks
10. Skybox

### 2.2 Functionalities

The project has the next functionalities:

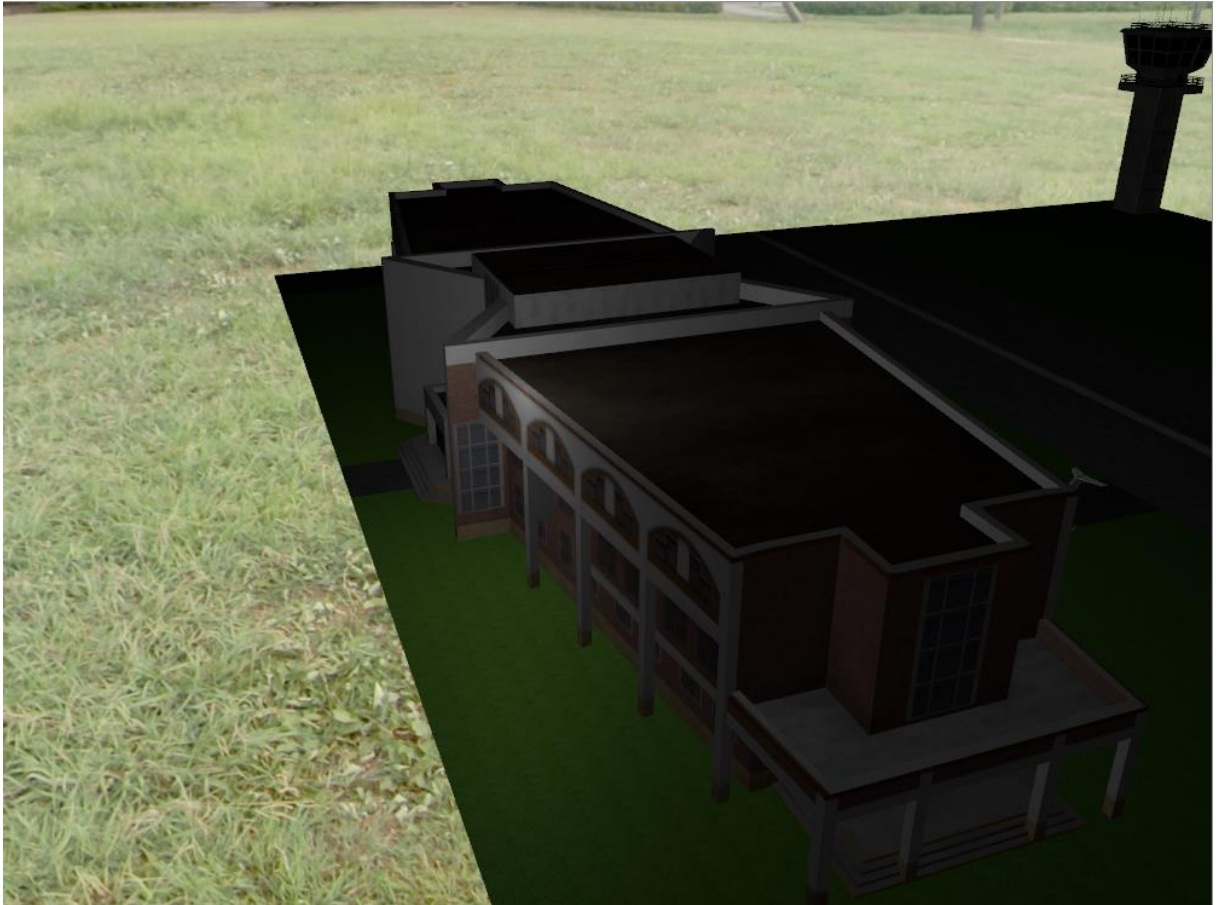
1. Visualization of the scene in all directions using the keys: W, A, S, D
2. Rotation of the scene using the mouse
3. Activation of the directional light using the key P
4. Deactivation of the directional light using the key O
5. Activation of the point light using the key N
6. Deactivation of the point light using the key B
7. Activation of the solid mode using the key 1
8. Activation of the wireframe object using the key 2
9. Activation of the polygonal and smooth surface using the key 3
10. Shadows
11. Animation om the duck
12. Activation and deactivation of the fog using the key I
13. Moving the directional light to the left using the key J
14. Moving the directional light to the right using the key L

Now I will put some photos presenting the scene and the effects that I implemented.

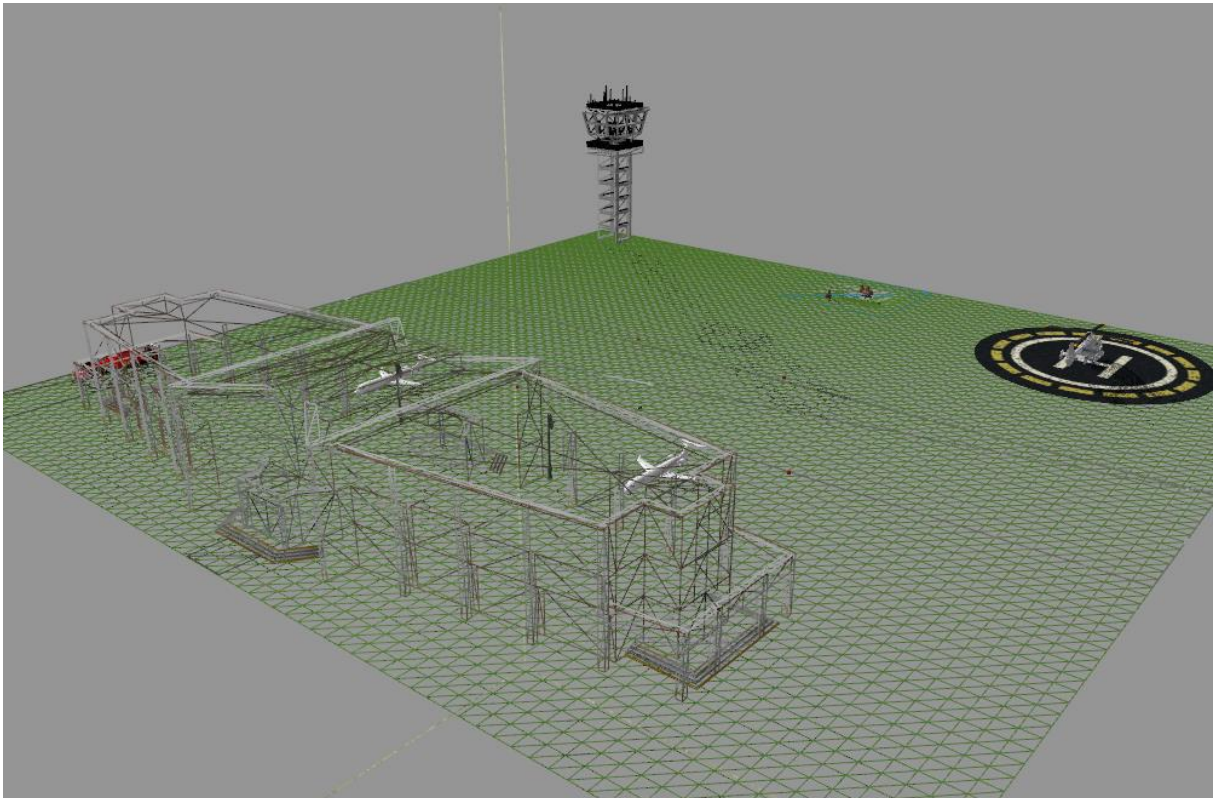
The visualization of the scene with shadows:



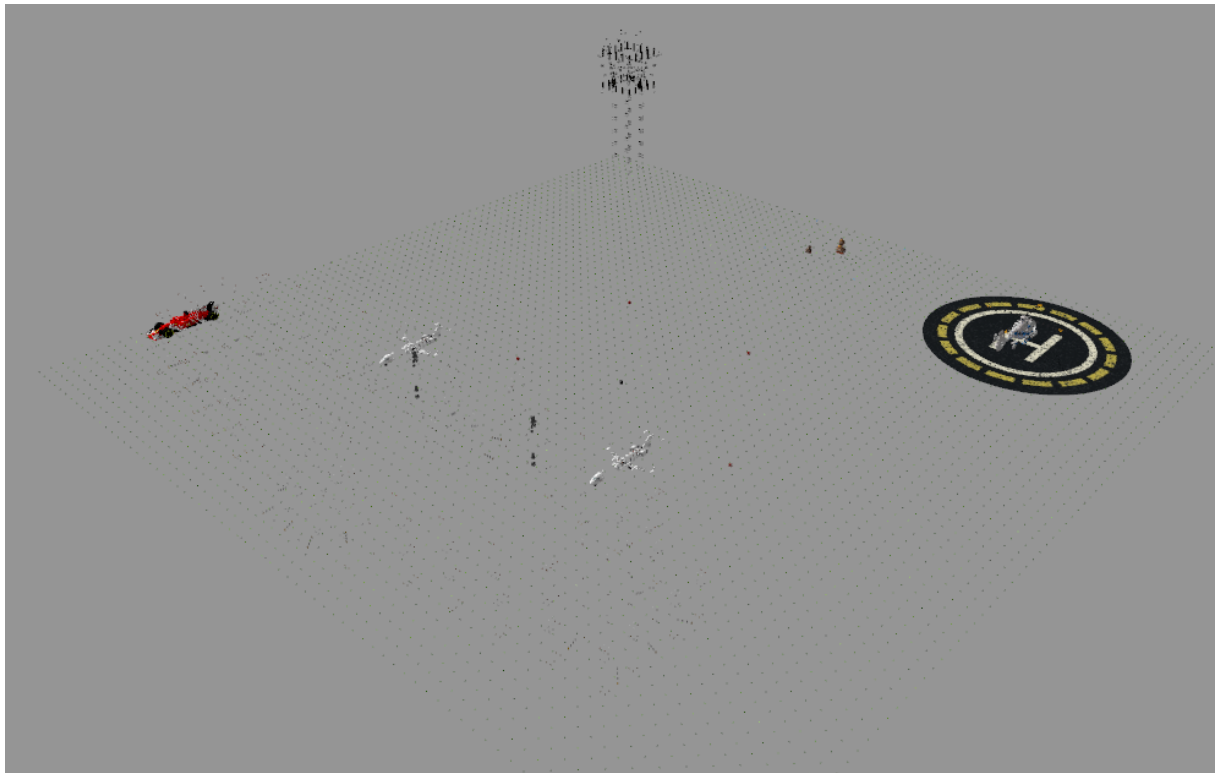
Point light:



Wireframe:

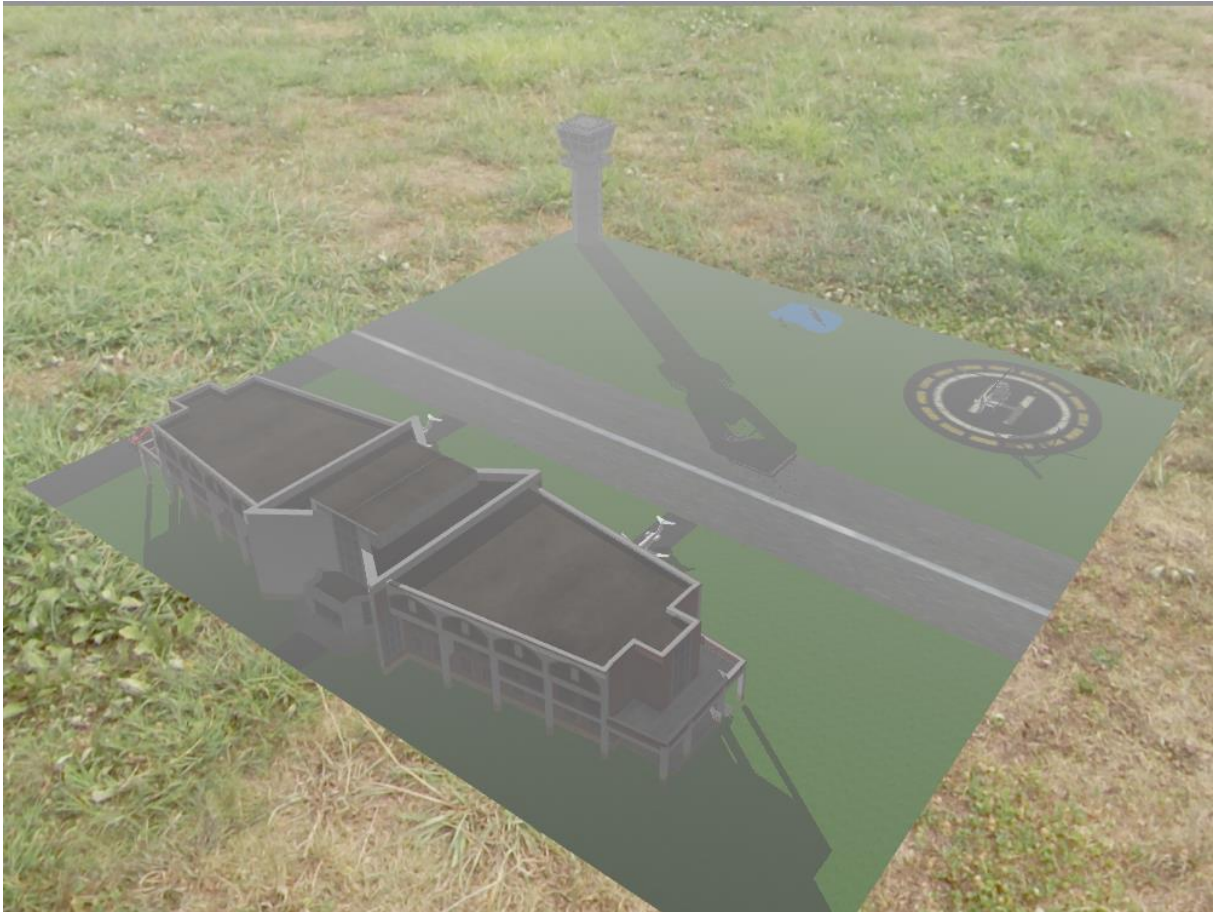


Polygonal and smooth surface:



Fog:





### 3. Implementation details

#### 3.1 Functions and special algorithms

##### 3.1.1 Animation of the duck

For making the animation of the duck, they must have different objs. The duck is moving on the Oy axis until the coordinate is reaching the 1.5f and after that is going behind to the initial point. I used two variables one for store the current position which is incremented with a step, after reaching the 1.5f I changed the sign of the step in order to make the duck move back to the initial position.

#### 3.2 Graphics model

Graphic model used is based on the ShadowMapping, Phong model.

#### 3.3 Data structures

I used data structures specific of GLM library and also the structures from the OpenGL library.

#### 3.4 Class hierarchy

Camera.hpp – defines the camera and its movement

Mesh.hpp – defines the vertex of one object

Model3D.hpp – defines an object

Shader.hpp – defines a shader

SkyBox.hpp – defines a skybox

#### 4. Graphical user interface presentation / user manual

For using the application, you should run the executable file and then read the functionalities from the chapter 2.

#### 5. Conclusions and further developments

In conclusion, this project was a really good opportunity to be familiar on working with OpenGL and how to make and scene in 3D with different functionalities.

For further developments it can be implemented: a new type of light like spot, animations on the airplanes or helicopter, introduce sounds, camera animation.

#### 6. References

<https://free3d.com/>

<https://sketchfab.com/feed>

<https://learnopengl.com/Lighting/Basic-Lighting>

<https://learnopengl.com/Advanced-Lighting/Shadows/Shadow-Mapping>