



UNIVERSITATEA TEHNICĂ DIN CLUJ-NAPOCA

OpenGL Project

Prelucrare Grafica

Autori: Mirisan Octavian

Grupa: 30231

FACULTATEA DE AUTOMATICA
SI CALCULATOARE

Ianuarie 2025

Cuprins

1 Prezentarea temei	2
2 Scenariul	2
2.1 Descrierea scenei si a obiectelor	2
2.2 Functionalitati	3
3 Detalii de implementare	4
3.1 Functii si algoritmi	4
3.1.1 Solutii posibile	4
3.1.2 Motivarea abordarii alese	4
3.2 Modelul grafic	4
3.3 Structuri de date	5
3.4 Ierarhia de clase	6
4 Manual de Utilizare	7
5 Concluzii si dezvoltari ulterioare	7
6 Referinte	8

1 Prezentarea temei

Proiectul are ca obiectiv realizarea unei scene 3D interactive utilizand OpenGL (librariile GLM, GLEW, GLFW) si Blender.

Tema presupune implementarea unei scene fotorealiste, cu texturi si animatii, efecte speciale care ofera posibilitatea utilizatorului de a explora scena utilizand mouse-ul si tastatura.

2 Scenariul

2.1 Descrierea scenei si a obiectelor

Am creat o scena 3D ce reprezinta un complex traditional de temple chinezești. Complexul este înconjurat de un gard de lemn și conține clădiri cu arhitectură chinezescă tradițională, precum acoperisuri curbate și structuri din lemn. Cladirea din centru este templul, care este înconjurată de clădiri mai mici și copaci.

Atât scena cat și obiectele au fost modelate și texturate cu ajutorul aplicatiei Blender, iar apoi au fost exportate într-un proiect OpenGL.



Figura 1: Scena in timpul zile - Daytime.



Figura 2: Scena la apus - Sunset.

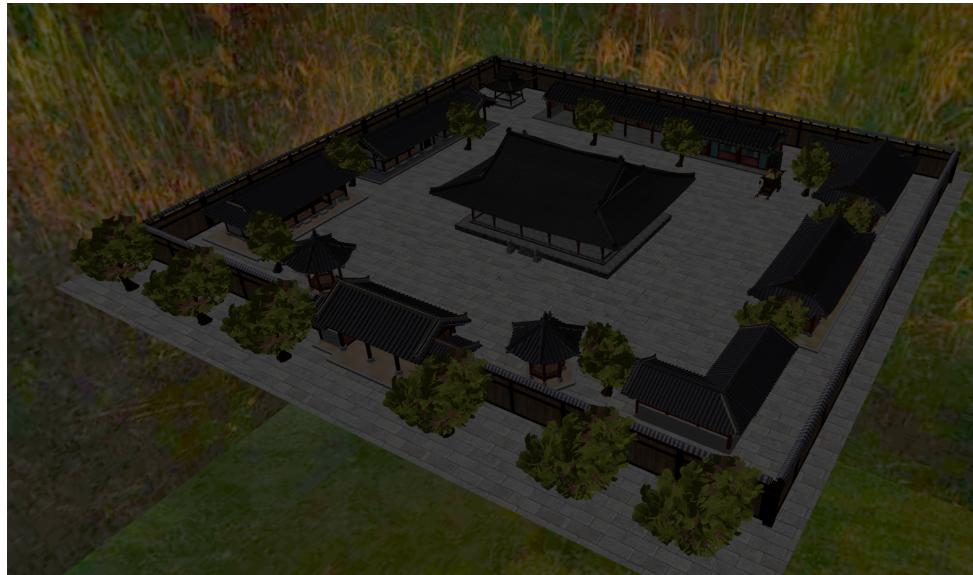


Figura 3: Scena noaptea - Night

Functionalitatile de mai jos au fost implementate cu ajutorul framework-ului OpenGL

2.2 Functionalitati

- Vizualizarea scenei prin rotatie, scalare, translatie
- Posibilitatea de a schimba modurile de randare: solid, wireframe si dots
- Texturarea obiectelor pentru un aspect fotorealist
- Efecte de ceata si animatie pentru caderea particulelor de ploaie pentru a sustine realismul scenei
- Posibilitatea de a comuta intre diferite momente ale zile: day, night, sunset
- Posibilitatea de a misca diferite obiecte din scena

3 Detalii de implementare

3.1 Functii si algoritmi

3.1.1 Solutii posibile

Am folosit tehnici si biblioteci comune pentru grafica 3D, cum ar fi OpenGL si GLFW, si extensiile GLEW pentru OpenGL.

- **GLEW:** Gestionarea extensiilor OpenGL.
- **GLFW:** Crearea ferestrelor si gestionarea intrarilor.
- **GLM:** Operatii matematice pentru grafica 3D.

3.1.2 Motivarea abordarii alese

Fiecare fisier din proiect are un rol specific, impartit in urmatoarele categorii:

- **Gestionarea camerei:**
 - Camera.cpp si Camera.hpp definesc si gestioneaza comportamentul camerei in scena.
- **Manipularea datelor geometrice:**
 - Mesh.cpp si Mesh.hpp definesc structurile pentru manipularea datelor geometrice.
- **Incarcarea si desenarea modelelor 3D:**
 - Model3D.cpp si Model3D.hpp sunt folosite pentru incarcarea si randarea modelelor 3D.
- **Gestionarea shaderelor:**
 - Shader.cpp si Shader.hpp gestioneaza programele de shader.
- **Randarea Skybox-ului:**
 - SkyBox.cpp si SkyBox.hpp se ocupa cu incarcarea si randarea unui Skybox.

3.2 Modelul grafic

Modelele au fost descarcate de pe internet, si exportate in format compatibil cu OpenGL (.obj). Fiecare model are asociate texturi pentru diffuse si specular.

3.3 Structuri de date

- **Vertex:**

- Contine informatii despre un varf din geometria unui obiect 3D.
- Structura include:
 - * **Position:** pozitia varfului in spatiul 3D (tip `glm::vec3`).
 - * **Normal:** vectorul normal asociat varfului, folosit pentru calculul iluminarii (tip `glm::vec3`).
 - * **TexCoords:** coordonatele de textura pentru aplicarea texturii pe suprafata (tip `glm::vec2`).

- **Texture:**

- Reprezinta o textura aplicata pe un obiect 3D.
- Structura include:
 - * **id:** identificatorul OpenGL al texturii (tip `GLuint`).
 - * **type:** tipul texturii (de exemplu, "ambient", "diffuse", "specular").
 - * **path:** calea catre fisierul de textura.

- **Material:**

- Descrie caracteristicile unui material utilizat in scena.
- Structura include:
 - * **ambient:** culoarea ambientala a materialului (tip `glm::vec3`).
 - * **diffuse:** culoarea difusa a materialului (tip `glm::vec3`).
 - * **specular:** culoarea speculara a materialului (tip `glm::vec3`).

- **Buffers:**

- Contine datele necesare pentru buferizarea geometriei obiectului.
- Structura include:
 - * **VAO** (Vertex Array Object): contine configuratia bufferelor pentru un obiect 3D.
 - * **VBO** (Vertex Buffer Object): stocheaza datele varfurilor obiectului 3D.
 - * **EBO** (Element Buffer Object): stocheaza indicii pentru constructia geometriei.

3.4 Ierarhia de clase

```
SceneProject
  assets
    scene.obj
    scene.mtl
  carriage
    carriage.obj
    carriage.mtl

  shaders
    shaderStart.frag
    shaderStart.vert
    skyboxShader.frag
    skyboxShader.vert
    rain.frag
    rain.vert
  skybox

x64
Camera.cpp
Camera.hpp
main.cpp
Mesh.cpp
Mesh.hpp
Model3D.cpp
Model3D.hpp
Shader.cpp
Shader.hpp
SkyBox.cpp
SkyBox.hpp
stb_image.cpp
stb_image.h
tiny_obj_loader.cpp
tiny_obj_loader.h
```

4 Manual de Utilizare

Interactiunea cu aplicatia se face utilizand tastatura si mouse-ul. Functionalitatile disponibile sunt urmatoarele:

- Tastele W,A,S,D: permit miscarea in scena a utilizatorului
- Mouse-ul: prin miscarea mouse-ului, se modifica directia in care priveste camera
- Tastele Q si E: permit rotirea scenei spre stanga, respectiv spre dreapta
- Arrow Keys: permit miscarea unui obiect (carriage) in scena pe directiile fata-spate-stanga-dreapta
- Tasta K: enable/disable pentru o animatie de prezentare automata a scenei
- Tasta Z: enable/disable efectul de ceata in scena OpenGL
- Tastele X si C: permit modificarea intensitatii cetii in scena. Cu X se creste nivelul de intensitate al acesteia, iar cu C se scade.
- Tastele 1,2,3: permit vizualizarea scenei in trei moduri diferite: wireframe, point si solid
- Tasta O: enable/disable la efectul de noapte(night). La apasare, se modifica skybox-ul precum si culoarea cetii.
- Tasta P: enable/disable la efectul de apus(sunset). La apasare, se modifica skybox-ul precum si culoarea cetii.
- Tasta R: enable/disable pentru efectul de ploaie(rain) in scena.

5 Concluzii si dezvoltari ulterioare

Acest proiect a reprezentat o oportunitate valoroasa de a invata si aplica concepte importante din domeniul graficii pe calculator. Am inteles mai bine modul de interactiune dintre diferitele componente ale unei aplicatii OpenGL, cum ar fi gestionarea shaderelor, implementarea unui skybox, manipularea camerei si crearea unor efecte vizuale precum ceata. In plus, am dobandit experienta in exportarea unui model din Blender si integrarea acestuia intr-o scena grafica.

In ceea ce priveste dezvoltarea ulterioara a proiectului, exista numeroase directii interesante care pot fi explorate, cum ar fi:

- Adaugarea mai multor obiecte in scena pentru a crea medii complexe si interactive.
- Utilizarea unor texturi mult mai realiste pentru a imbunatati aspectul general al scenei.
- Adaugarea unui efect de "splash" in momentul in care particulele de ploaie se lovesc de obiectele din scena
- Implementarea unei modalitati de a manipula intensitatea cetii in functie de zona din scena

6 Referinte

Referințe

- [1] Lucrarile de laborator pentru cursul de Prelucrare Grafica, care au oferit baza teoretica si practica pentru intelegerarea si implementarea proiectului.
- [2] Tutoriale de Blender ale domnului profesor Constantin Nandra care includ crearea de scene, aduagarea si texturarea de obiecte, exportul de obiecte in proiectul OpenGL, crearea de skybox