



UNIVERSITATEA TEHNICĂ DIN CLUJ-NAPOCA

OpenGL Project

Proiectare grafica

Autor: Scuturici Vlad

Grupa: 30235

FACULTATEA DE AUTOMATICA
SI CALCULATOARE

17 ianuarie 2024

Cuprins

1 Prezentarea temei	2
2 Scenariul	2
2.1 Descrierea scenei și a obiectelor	2
2.2 Funcționalități	2
3 Detalii de implementare	3
3.1 Funcții și algoritmi	3
3.2 Modelul grafic	3
3.3 Structuri de date	3
3.4 Ierarhia de clase	3
4 Prezentarea interfeței grafice utilizator / manual de utilizare	4
5 Concluzii și dezvoltări ulterioare	7
6 Referințe	7

1 Prezentarea temei

Proiectul nostru se axează pe însușirea tehniciilor de fotorealism în jocurile video și pe înțelegerea procesării scenelor 3D de către calculator. Folosim biblioteci precum OpenGL, GLFW și GLM, și oferim posibilitatea de a naviga în scenă cu tastatura și mouse-ul, asemenea unui joc. Am selectat o scenă predefinită pentru proiect, pe care am personalizat-o prin aranjarea obiectelor.

2 Scenariul

2.1 Descrierea scenei și a obiectelor

Proiectul se concentrează pe o scenă urbană dinamică și diversificată. Imaginea oricărui oraș este un mozaic complex de elemente arhitecturale și sociale, iar în lucrarea mea am încercat să surprind această esență. Orașul meu imaginar este alcătuit dintr-o varietate de clădiri. Străzile sunt animate de o varietate de semne de circulație, care nu numai că dirijează traficul, ci și adaugă un strat suplimentar de realism scenei. Am inclus și elemente decorative cum ar fi bânci de parc, copaci și flinare, care adaugă un suflu de viață și culoare. În final, prezența unei drone zburătoare adaugă un element modern și tehnologic, sugerând progresul și inovația în cadrul acestui peisaj urban.



Figura 1: Scena

2.2 Funcționalități

In cadrul proiectului am implementat urmatoarele funcționalități :

1. Miscarea camerei cu ajutorul tastaturii (W, A, S, D, SHIFT, SPACE) si al mouse-ului.
2. Schimbul intre lumina directionala si punctiforma. Lumina punctiforma este atasata de drona.
3. Activarea / Dezactivarea de ceata
4. Manipularea pozitiei si dimensiunii dronei, care totodata muta pozitia luminii punctiforme
5. Schimbul intre modurile Solid, Wireframe si Point.
6. Animatii de prezentare utilizand camera si drona.

3 Detalii de implementare

3.1 Functii si algoritmi

Am implementat proiectul folosind OpenGL si am folosit functii precum:

- mouseCallback() - ajuta la utilizarea mouse-ului
- keyboardCallback() - ajuta la utilizarea tastaturii
- processMovement() - identifica anumite miscari in scena
- windowResizeCallback() - permite utilizatorului sa schimbe dimensiunea ferestrei
- initOpenGLWindow() - initializeaza fereastra
- initModels() - initializeaza modelele pentru obiecte
- initUniforms() - initializeaza variabilele de tip uniform
- renderScene() - deseneaza si actualizeaza scena

In clasa Camera:

- move() si rotate() - pentru miscarea camerei
- setPosition() si setTarget() - seteaza pozitia camerei

In shadere:

- computeDirLight() - pentru lumina directionala
- computePointLight() - pentru lumina punctiforma
- computeFog() - pentru ceata

3.2 Modelul grafic

Obiectele folosite au fost aranjate folosind Blender. Texturarea se face cu ajutorul fisierelor de tip .mtl ce atribuie o anumita textura componentelor obiectului .obj. Proiectul a fost compilat si rulat pe Microsoft Visual Studio.

3.3 Structuri de date

Proiectul are la baza structuri de date cum ar fi enumeratii, vectori si matrici din biblioteca GLM, array-uri, vectori din STD sau obiecte de tip GPS.

3.4 Ierarhia de clase

Fiecare clasă din proiect îndeplinește funcții specifice și esențiale pentru a construi și gestiona scena 3D în OpenGL. Clasa Camera gestionează funcționalitățile camerei, permitând navigarea și vizualizarea în spațiul 3D. Window, pe de altă parte, se ocupă de crearea și gestionarea ferestrei de desenare, inclusiv inițializarea și închiderea acesteia. Clasa SkyBox este responsabilă pentru încărcarea, desenarea și aplicarea diferitelor operații asupra skybox-ului, adăugând

astfel fundalul scenic. Mesh joacă un rol crucial în desenarea și manipularea obiectelor 3D. Shaders implementează toate operațiunile necesare pentru funcționarea shaderelor, esențiale pentru renderizarea efectelor vizuale. În sfârșit, bibliotecile externe stb-image și tiny-obj-loader sunt folosite pentru texturarea și încărcarea obiectelor în OpenGL, respectiv, oferind detalii vizuale și structurale obiectelor din scena noastră.

4 Prezentarea interfeței grafice utilizator / manual de utilizare



Figura 2: Solid

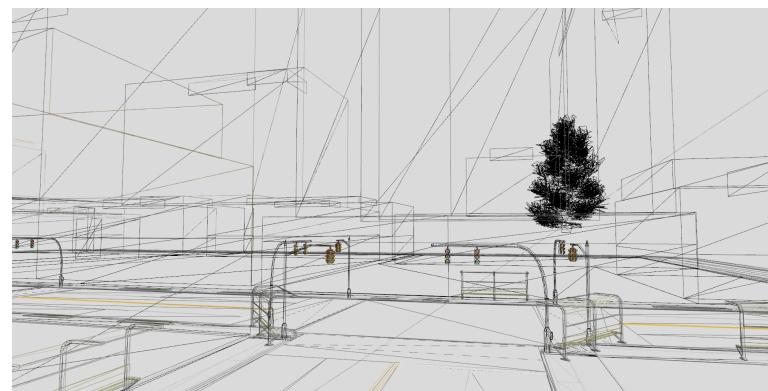


Figura 3: Wireframe



Figura 4: Point

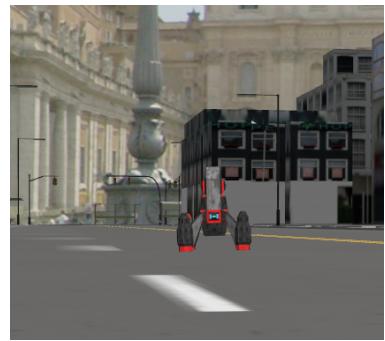


Figura 5: Drona



Figura 6: Lumina Punctiforma din drona

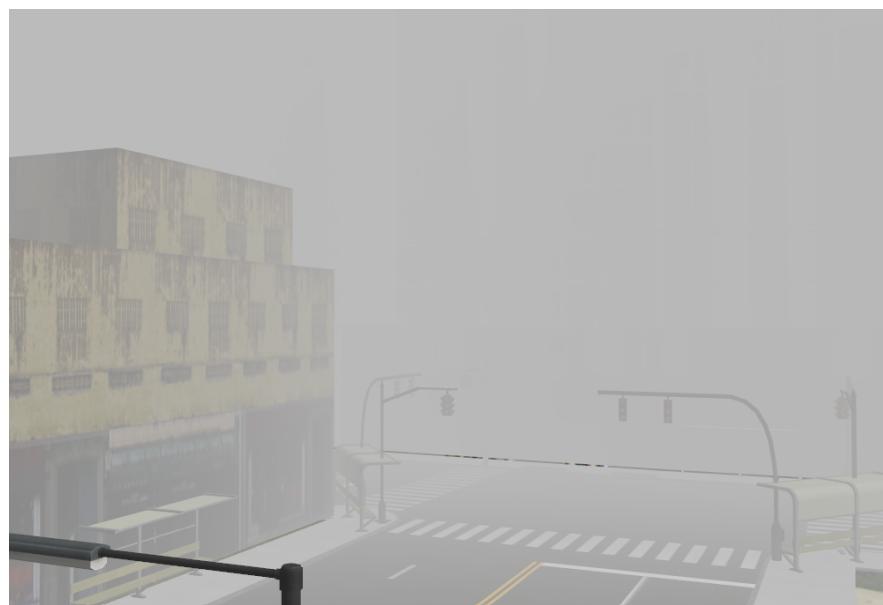


Figura 7: Ceata



Figura 8: Cladiri



Figura 9: Banca

Manual de utilizare

- W, A, S, D - pentru miscarea camerei
- Q, E - rotatia camerei
- SPACE, SHIFT - pentru ridicarea / coborarea camerei
- Arrows - miscarea dronei
- '[,]' - rotatie drona
- 1, 2 - miscare drona in sus si in jos
- O, P - scalare drona
- F - activare / dezactivare ceata
- L - schimbare intre lumina directionala si punctiforma
- TAB - schimbare intre modurile solid / wireframe / point
- M - incepe o animatie care are la baza drona

- N - drona revine in pozitia initiala
- RIGHT SHIFT - animatie de prezentare

5 Concluzii și dezvoltări ulterioare

Acest proiect este destinat să faciliteze introducerea studenților în domeniul modelării 3D utilizând OpenGL. În cadrul acestuia, am dobândit competențe esențiale în texturarea obiectelor, modelarea 3D și aplicarea de transformări asupra obiectelor. Crearea primei mele aplicații de acest tip a reprezentat o experiență educativă și placută.

6 Referințe

Am folosit pentru a căuta diverse obiecte și scene: CGTrader Free3D Turbosquid
De asemenea, am laboratoarele și cursurile de Proiectare Grafică de pe Moodle: Moodle PG
În final, am utilizat documentație OpenGL: Open GL Documentation