PROIECT PRELUCRARE GRAFICA

Student: Badea Ioana Cristiana

Grupa: 30237

Contents

[1. Prezentarea temei 3](#_Toc156222732)

[2. Scenariul 3](#_Toc156222733)

[2.1 Descrierea scenei si a obiectelor 3](#_Toc156222734)

[2.2 Functionalitati 3](#_Toc156222735)

[3. Detalii de implentare 3](#_Toc156222736)

[3.1 Funcții și algortmi 3](#_Toc156222737)

[3.1.1 Solutii posibile 3](#_Toc156222738)

[3.1.2Motivarea abordarii alese 5](#_Toc156222739)

[3.2 Modelul graphic 5](#_Toc156222740)

[3.3 Structuri de date 6](#_Toc156222741)

[3.4 Ierarhia de clase 6](#_Toc156222742)

[4. Prezentarea Interfetei grafice utilizator/manual de utilizare 7](#_Toc156222743)

[5. Concluzii si dezvoltari ulterioare 7](#_Toc156222744)

[6.Referinte 7](#_Toc156222745)

# Prezentarea temei

Acest proiect are drept scop crearea unui model tridimensional detaliat al unui peisaj format din dealuri, insotit de alte obiecte pentru imbunatatirea realismului vizual. Obiectivul principal este să se ofere o experiență de vizualizare captivantă și să se dezvolte o scenă complexă, care să evidențieze atât nivelul de detalii, cât și efectul general de fotorealism al întregii compoziții. Prin implementarea unor funcționalități avansate, am dorit să oferim utilizatorului posibilitatea de a explora și interacționa cu această scenă, contribuind astfel la o experiență vizuală captivantă. Imaginea de mai jos ilustrează rezultatul final al proiectului, evidențiind detaliile elaborate ale insulei și integrarea armonioasă a altor obiecte pentru a crea un efect fotorealist remarcabil.

A house on a hill by water

Description automatically generated

# 2. Scenariul

## 2.1 Descrierea scenei si a obiectelor

In scena se gasesc mai multe obiecte, care, grupate impreuna, definesc intregul peisaj. Pe langa acestea, am introdus un obiect separat, integrat in mod distinct, pentru a oferi dinamica si expresivitate scenei.

Scena are un plan de baza care reprezinta suprafata muntelui, folosindu-se functia  “sculpt” din Blender, avand o textura potrivita, adaugandu-se denivelari pentru a se forma dealurile. De asemenea, suprafata apei este alcatuita din alt plan, textura fiind reprezentativa. Scena este formata dintr-o casa, care este luminata, incounjurata de copaci, stalp de iluminat si un proiector. Pe lac se afla o barca, se poate observa un avion, care prezinta animatia scenei.

## 2.2 Functionalitati

In cadrul scenei s-au implementat urmatoarele functionalitati:

-deplasarea in scena pe cele 3 axe, rotirea camerei in toate directiile;

-vizualizarea in cele 3 moduri: punctiform, wireform si solid;

-aprinderea si inchiderea luminilor de tip spot si punctiforme, tranzitia zi/noapte;

- activarea si dezactivarea efectului de ceata;

-activarea unei animatii ;

# 3. Detalii de implentare

## 3.1 Funcții și algortmi

### 3.1.1 Solutii posibile

Atenuarea luminii punctiforme în funcție de distanță poate fi calculată utilizând o formulă patratică, astfel încât intensitatea luminii să scadă proporțional cu pătratul distanței dintre sursa de lumină și fragmentul curent. Această abordare este mai apropiată de comportamentul natural al luminii și contribuie la obținerea unui efect mai realist în reprezentarea scenei 3D. Formula pentru atenuarea patratică este adesea exprimată astfel:

A math equation with black text

Description automatically generated

Unde:

- d reprezintă distanța dintre sursa de lumină și fragmentul curent ;

- constant, linear și quadratic sunt coeficienți care influențează atenuarea. Acești coeficienți pot fi ajustați pentru a obține rezultate vizuale dorite.

Prin aplicarea acestei formule, putem ajusta intensitatea luminii în funcție de distanța dintre sursa de lumină și punctul de iluminare, simulând astfel efectul natural de diminuare a luminii odată cu creșterea distanței.

Implementarea în cod OpenGL ar consta în pasii următori:

1)Calcularea distanței d între sursa de lumină și fragmentul curent în shaderul fragment.

2)Aplicarea formulei de atenuare patratică pentru a obține factorul de atenuare.

3)Multiplicarea factorului de atenuare cu culoarea iluminată pentru a obține culoarea finală a fragmentului.

4)Această abordare oferă un control mai mare asupra modului în care lumina se estompează în funcție de distanță și poate fi ajustată pentru a se potrivi cerințelor estetice specifice ale scenei tale.

Lumina de tip spotlight în OpenGL reprezintă o sursă de lumină direcționată care emite raze luminoase într-o anumită direcție, în loc să difuzeze lumina în toate direcțiile, așa cum fac luminile omnidirecționale. Acest tip de lumină este adesea utilizat pentru a simula efectul unui reflector sau a unei lanterne, unde lumina este direcționată într-un con specific.

În OpenGL, o lumină de tip spot este definită de următoarele proprietăți principale:

1)Poziție: Coordonatele în spațiul 3D de unde lumina spot este emisă.

2)Direcție: Direcția în care lumina spot este orientată, adică direcția conului de lumină.

3)Unghiul de Tăiere (Cut-Off Angle): Unghiul care definește conul de lumină al reflectorului. Doar fragmentele situate în interiorul acestui con primesc lumină.

4)Atenuare: Similar cu luminile punctiforme, o lumină de tip spot poate beneficia de atenuare pentru a simula diminuarea intensității luminii în funcție de distanță.

Implementarea unei astfel de lumini în OpenGL implică determinarea dacă fiecare fragment se află în interiorul conului de lumină. Aceasta se face, în general, prin calcularea direcției de la poziția fragmentului la poziția sursei de lumină și compararea acestei direcții cu direcția reflectorului. Dacă fragmentul se află în interiorul conului, atunci se aplică iluminarea corespunzătoare.

Această abordare oferă un control precis asupra modului în care lumina spotlight interacționează cu obiectele din scenă, permitând simularea realistă a efectelor de iluminare specifice acestui tip de lumină.

Efectul de ceață reprezintă o realizare vizuală complexă, obținută prin manipularea atentă a atributelor asociate acestui fenomen. Prin personalizarea acestor atribute, nu doar că putem influența atmosfera într-un mod distinctiv, dar avem și posibilitatea de a îmbunătăți percepția adâncimii în axa Z.

Pentru a obține o armonie vizuală coerentă, este esențial să alegem culoarea de fundal în funcție de nuanța specifică a efectului de ceață. Acest detaliu este reflectat în codul de setare a culorii de fundal, unde valori precum (0.5, 0.5, 0.5, 1.0) indică o paletă cromatică specifică.

În versiunile anterioare ale OpenGL cu un pipeline fix, existau trei metode distincte de calculare a ceții: liniară, exponențială și exponențială pătratică. Într-un mod similar, aceeași funcționalitate poate fi realizată și în versiunile moderne cu pipeline programabil prin integrarea calculelor de ceață în shader-ul nostru de fragmente.

Una dintre metodele de calculare a ceții este cea liniară, care utilizează o funcție de interpolare liniară pentru a amesteca culoarea ceții cu cea a fragmentului. Acest proces se desfășoară în spațiul de vizualizare, iar rezultatul este determinat în funcție de distanța fragmentului față de observator.

A black text on a white background

Description automatically generated

Pentru a obține un control mai precis asupra efectului de ceață, este important ca valorile factorului de ceață să se situeze în intervalul cuprins între 0.0 și 1.0. O optimizare semnificativă poate fi realizată prin luarea în considerare a reducerii intensității luminii în funcție de distanță.

Factorul de atenuare utilizat în acest context reprezintă densitatea de ceață, care este considerată constantă în întreaga scenă. Această densitate uniformă generează o scădere accentuată a factorului de ceață în comparație cu abordarea liniară. Astfel, intensitatea luminii este rapid atenuată pe măsură ce obiectele se află la distanțe mai mari de observator.

Această strategie oferă un control mai fin asupra modului în care ceața interacționează cu obiectele din scenă, contribuind la realizarea unui efect vizual autentic și detaliat. Prin ajustarea constantă a densității de ceață și a factorului de atenuare în funcție de distanță, obținem un rezultat mai sofisticat și estetic în ceea ce privește atmosfera și percepția de adâncime în reprezentările grafice.

A black and white text

Description automatically generated

Formula pentru ceața exponențială pătratică este practic aceeași cu cea precedentă, singura diferență provenind din faptul că distanța fragmentului și densitatea de ceață este ridicată la pătrat.

A black text on a white background

Description automatically generated

### 3.1.2Motivarea abordarii alese

Am optat pentru implementarea formulei de ceață exponențială pătratică pentru a asigura obținerea rezultatelor optime în ceea ce privește efectul de ceață. A screen shot of a computer code

Description automatically generated

## 3.2 Modelul graphic

Modelele folosite folosite se impart in scena propriu-zisa si obecte separate care vor fi animate. Modelele grafice folosite sunt urmatoarele :

-scena: reprezinta scena propriu-zis, fara modificari;

-avion: obiectul avion realizeaza o animatie in scena, reprezentata printr-o translatie de-a lungul axei X.

A computer screen with text

Description automatically generated

## 3.3 Structuri de date

Folosirea bibliotecii GLM pentru structurile de date, cum ar fi matrice și vectori, reprezintă o alegere înțeleaptă pentru implementarea proiectului tău, întrucât GLM (OpenGL Mathematics) oferă un set bogat de funcționalități pentru matematica necesară în programarea grafică. Aceasta include operații cu matrice, vectori, și alte concepte matematice esențiale în contextul graficii 3D.

De asemenea, definirea unor clase personalizate precum Camera, Mesh, Shader, Window, și Model3D aduce o structură clară și modulară proiectului tău. Această abordare orientată pe obiect facilitează gestionarea entităților diferite din scenă și permite o organizare mai eficientă a codului.

## 3.4 Ierarhia de clase

Descrierea ierarhiei de clase reflectă o structură bine organizată și modulară a proiectului tău, ceea ce este esențial pentru gestionarea eficientă a funcționalităților diverse. Iată o sumarizare a ierarhiei descrise:

1)Camera:-Descrie acțiunile aplicabile asupra camerei.

-Gestionarea perspectivelor și acțiunilor specifice camerei, precum mutarea și rotirea.

2)Main:-Reprezintă clasa aplicației principale.

-Probabil conține funcționalități precum inițializarea OpenGL, gestionarea buclei de evenimente, și integrarea altor clase.

3)Window:-Include operațiile necesare pentru gestionarea ferestrei aplicației.

-Deschiderea și închiderea ferestrei, gestionarea evenimentelor de input.

4)Mesh:-Se ocupă de încărcarea texturilor pentru obiectele din scenă.

-Probabil oferă funcții pentru a încărca și atribui texturi la obiecte.

5)Model3D:-Include și execută toate operațiile legate de încărcarea și gestionarea obiectelor în scenă.

-Poate avea funcționalități pentru manipularea, afișarea și gestionarea detaliilor complexe ale modelelor 3D.

Această ierarhie reflectă o abordare organizată a proiectului, unde fiecare clasă are un scop specific și se concentrează pe responsabilități bine definite. Acest lucru face ca codul să fie mai ușor de înțeles, de întreținut și de extins pe viitor.

# 4. Prezentarea Interfetei grafice utilizator/manual de utilizare

Deplasarea camerei se poate realiza folosind urmatoarele taste:

-tasta W: miscarea camerei inainte ;

-tasta S: miscarea camerei inapoi;

-tasta A: miscarea camerei la stanga;

- tasta D:miscarea camerei la dreapta;

- tasta Q: rotirea camarei spre stanga ;

- tasta E: rotirea camarei spre dreapta ;

Efectul de ceata se activeaza prin apasarea tastei 1, iar dezactivarea se realizeaza prin tasta 2.

Stingerea luminii directionale si aprinderea luminilor punctiforma si de tip spot se realizeaza prin tasta 3, iar aprinderea luminii directionale si stingerea celorlalte lumini se relizeaza prin tasta 4, astfel se creeaza efectul de zi-noapte.

Apasandu-se tasta Z, scena se vizualizeaza in mod solid, tasta X prezinta scena in mod wireframe, iar tasta C in mod poligonal.

# 5. Concluzii si dezvoltari ulterioare

Implementarea acestui proiect a oferit o oportunitate valoroasă pentru aprofundarea cunoștințelor generale legate de OpenGL și grafică în contextul lucrului cu obiecte 3D. Pe parcursul dezvoltării, am dobândit înțelegerea practică a conceptelor esențiale în domeniul graficii, contribuind semnificativ la consolidarea fundației de cunoștințe.

Proiectul a facilitat nu doar aplicarea noțiunilor fundamentale discutate în laboratoare, ci și explorarea aspectelor avansate, cum ar fi implementarea și gestionarea luminii de tip spot și dezvoltarea animațiilor dinamice pentru obiecte individuale. Această extindere a cunoștințelor a adăugat un nivel suplimentar de complexitate și expresivitate proiectului.

În perspectivă, dezvoltările ulterioare ar putea viza îmbunătățirea experienței vizuale prin implementarea unui algoritm de detecție a coliziunilor pentru toate obiectele prezente în scenă. Adăugarea umbrelor, introducerea mai multor tipuri de animații și posibilitatea de a genera dinamic obiecte pentru un nivel crescut de fotorealism reprezintă direcții potențiale de extindere și rafinare a proiectului. Aceste adiții ar contribui la crearea unei experiențe vizuale mai complexe și realiste, consolidând astfel competențele în domeniul graficii 3D.

# 6.Referinte

1. <https://learnopengl.com/Lighting/Light-casters#:~:text=A%20good%20example%20of%20a,the%20radius%20of%20the%20spotlight>.

2. Indrumator de laborator