
Proiect Prelucrare Grafica

Scena 3D interactiva

*Facultatea de Automatică
și Calculatoare*

-14 ianuarie 2025-

Barna Ioana Melisa

Grupa: 30235

Cuprins

1. Cuprins
2. Prezentarea temei
3. Scenariul
 - 3.1. Descrierea scenei și a obiectelor
 - 3.2. Funcționalități
4. Detalii de implementare
 - 4.1. Funcții și algoritmi
 - 4.1.1. Soluții posibile
 - 4.1.2. Motivarea abordării alese
 - 4.2. Modelul grafic
 - 4.3. Structuri de date
 - 4.4. Ierarhia de clase
5. Prezentarea interfeței grafice utilizator / manual de utilizare
6. Concluzii și dezvoltări ulterioare
7. Referințe

2. Prezentarea temei

Tema proiectului meu este inspirată de **viața simplă și liniștită de la țară**, reprezentând modul în care oamenii trăiesc departe de aglomerația urbană. **Scena 3D** creată în **Blender** ilustrează elemente specifice vieții rurale, în care natura și armonia sunt predominante.

Proiectul oferă utilizatorului posibilitatea de a explora această scenă interactivă folosind diverse operații:

-vizualizare și manipulare a scenei: Utilizatorul poate scala, transla și roti obiectele din scenă;

-mișcarea camerei: Camera poate fi deplasată liber, utilizând tastatura și mouse-ul, pentru a oferi o perspectivă diferită asupra vieții rurale;

-moduri de vizualizare: Scena poate fi redată în moduri precum solid, wireframe, poligonal și smooth, evidențiind detaliile geometriei și ale texturilor;

-animații și efecte vizuale: Elementele scenei sunt animate pentru a sugera dinamismul subtil al vieții la țară;

-maparea texturilor: Texturile sunt aplicate pe obiecte pentru a reda materiale realiste, precum lemnul, iarba sau cerul, asigurând un nivel înalt de fotorealism.

În cadrul proiectului, am pus accent pe calitatea texturilor și detaliile acestora, reușind să obțin un efect vizual autentic, care să reflecte atmosfera liniștită și relaxantă a vieții la țară.

3. Scenariul

3.1. Descrierea scenei și a obiectelor

Scena proiectului reprezintă o reprezentare a vieții liniștite de la țară, evidențiind elementele esențiale ale acestui mod de viață: lucrul pământului, îngrijirea animalelor și preocuparea pentru sănătatea trupului și a sufletului. Atmosfera rurală este recreată printr-o combinație armonioasă de obiecte și texturi, care oferă autenticitate și realism scenei.

Obiectele principale din scenă includ fermele și hambarele, importate de pe platforme dedicate descărcării obiectelor 3D. Acestea sunt completate de prezența animalelor specifice vieții la țară, cum ar fi caii, cerbii, oile și caprele. Pentru a optimiza performanța aplicației și a evita eventualele erori, am utilizat **modele 3D cu un număr redus de vârfuri**, asigurând astfel un echilibru între detaliu și eficiență.

Texturile au jucat un rol esențial în definirea scenei, contribuind semnificativ la aspectul vizual și atmosfera acesteia. De la materialele hambarelor până la pielea animalelor și frunzișul copacilor, texturile aplicate adaugă profunzime și realism fiecărui element.



Scena în sine a fost creată pe baza unui **plan importat în Blender**, care a fost ulterior **modificat și modelat** pentru a prinde viață sub forma unui mic sat rural. În modul *shape*, am oferit detalii și structură elementelor, conferindu-le formă și caracter.



Un alt element important al scenei îl reprezintă copacii, care adaugă nu doar un plus de diversitate vizuală, ci și culoare. Am utilizat patru tipuri diferite de copaci, fiecare cu caracteristici distincte, pentru a îmbogăți textura și varietatea scenei, creând astfel o atmosferă autentică, vibrantă și relaxantă.

3.2. Funcționalități

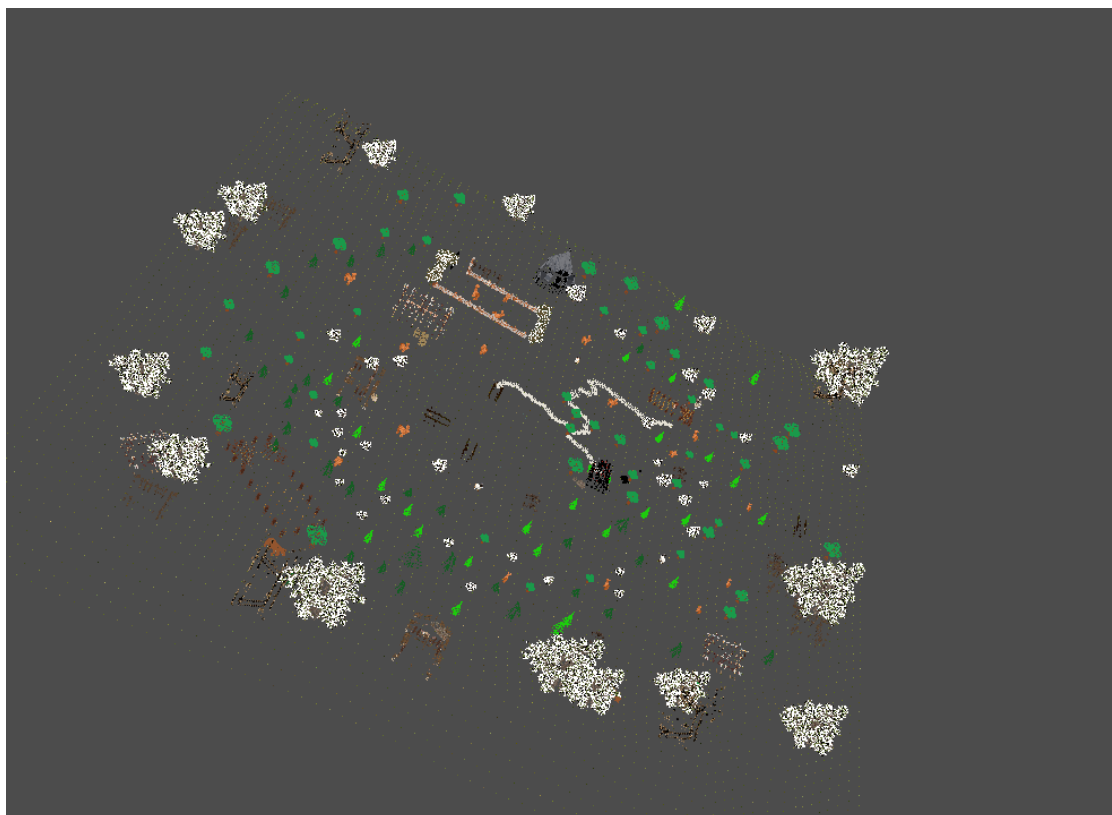
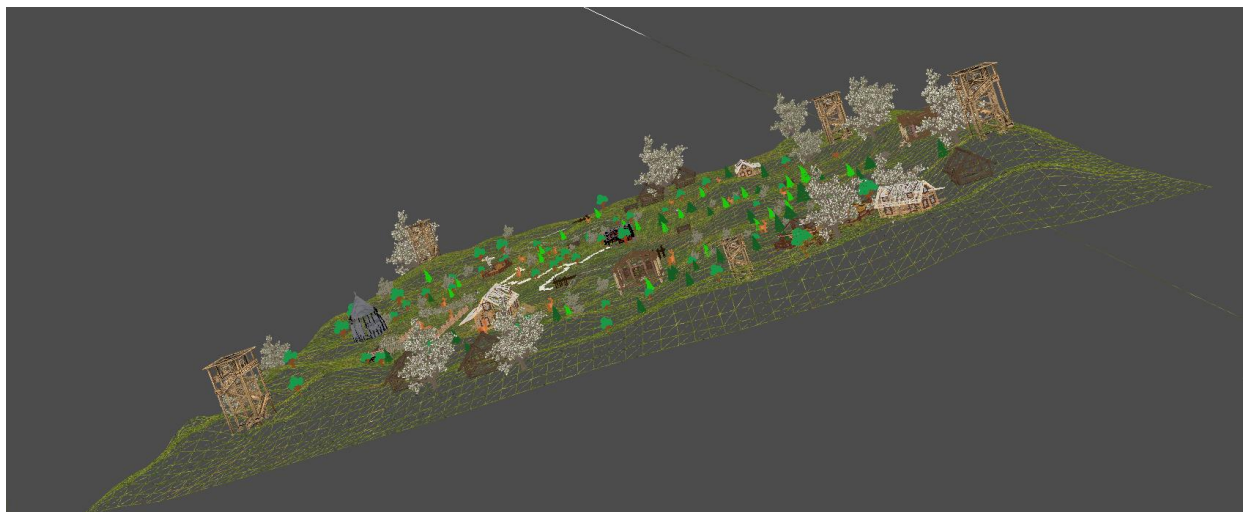
Proiectul include o serie de funcționalități esențiale pentru explorarea și manipularea interactivă a scenei 3D, oferind utilizatorului o experiență captivantă și realistă. Printre acestea se numără posibilitatea de a **vizualiza scena prin scalare, translație și rotație**. Aceste operații permit utilizatorului să mărească, să deplaseze sau să rotească obiectele, astfel încât să obțină perspective diferite și să examineze în detaliu fiecare element al scenei. De asemenea, **camera poate fi deplasată liber în spațiul 3D, fie utilizând tastatura** (tastele “A”, “S”, “W”, “D”), fie **mouse-ul**, ceea ce facilitează navigarea și oferă o flexibilitate sporită în explorarea mediului rural recreat.

Pentru a adăuga dinamism și o notă de realism, scena include animații care prezintă treptat elementele principale. **Camera**, de exemplu, **poate urma un traseu prestabilit**, oferind utilizatorului o privire de ansamblu asupra peisajului. Această animație va începe când utilizatorul apasă pe tasta “O”.

Iluminarea joacă un rol important în evidențierea detaliilor și în definirea atmosferei. Proiectul utilizează două tipuri de surse de lumină: **lumină direcțională**, care simulează lumina naturală a soarelui și **iluminează uniform întreaga scenă**, și **lumină punctiformă**, care adaugă accente și creează umbre în zone specifice, oferind profunzime și realism obiectelor. Odată apăsată tasta “Y”, modul inițial al sursei de lumină, adică direcțională, se va schimba în modul punctiform.



Un alt aspect important este posibilitatea de a schimba **modurile de vizualizare** ale scenei. Utilizatorul poate alege între moduri precum **solid**, **wireframe**, **poligonal** sau **smooth**. Aceste moduri permit fie o observare detaliată a structurii interne a obiectelor (wireframe), fie o redare netedă și realistă a suprafețelor acestora (smooth), oferind astfel flexibilitate în analiza și percepția scenei. Pentru aceste moduri avem asignate următoarele taste: “H” pentru modul SOLID, “J” pentru wireframe, “K” pentru poligonal și “L” pentru smooth.



Maparea texturilor și definirea materialelor sunt elemente cheie ale proiectului. Texturile utilizate sunt de înaltă calitate și sunt aplicate cu precizie pentru a reda caracteristicile fiecărui obiect. De exemplu, suprafața hambarelor sau frunzișul copacilor sunt redată în mod realist datorită acestor texturi. În plus, calitatea și nivelul de detaliu al texturilor au fost ajustate pentru a menține un echilibru între performanță și estetică, evitând erorile sau încetinirea aplicației.

Pentru a completa atmosfera rurală, scena include un efect de **ceată** subtilă, care contribuie la crearea unei **adâncimi vizuale** și accentuează tranziția naturală dintre

prim-plan și fundal. Acest efect adaugă un strat suplimentar de realism, sugerând răcoarea dimineților la țară și oferind un sentiment de calm și liniște. Toate aceste funcționalități se îmbină armonios pentru a oferi utilizatorului o experiență interactivă captivantă și autentică.

4. Detalii de implementare

4.1. Funcții și algoritmi

Un prim algoritm folosit în cod este cel pentru **ceață**, care are rolul de a crea un efect realist de estompere a obiectelor pe măsură ce acestea se află la distanțe mai mari față de cameră.

```
float fragmentDistance = length(fPosEye.xyz);
float fogDensity = 0.1f;
float fogFactor = exp(-pow(fragmentDistance * fogDensity, 2.0));
fogFactor = clamp(fogFactor, 0.0f, 1.0f);

vec4 fogColor = vec4(0.5f, 0.5f, 0.5f, 1.0f);
vec4 finalColor = mix(fogColor, vec4(color, 1.0f), fogFactor);
```

Prima linie de cod are rolul de a **calcula distanța dintre fragment și cameră** folosind lungimea vectorului poziției fragmentului în spațiul camerei. Distanța este folosită pentru a determina cât de afectat este fragmentul de ceață.

Cea de a doua definește **densitatea ceții**, adică cât de rapid începe să se aplice ceața pe măsură ce fragmentele sunt mai departe. O valoare mai mare face ceața mai densă și acoperă obiectele mai aproape de cameră.

Pentru a treia linie de cod se **calculează factorul de ceață** folosind o funcție exponențială. Fragmentele mai îndepărtate au un factor mai mic, ceea ce face ca acestea să fie mai estompate, simulând un efect natural de ceață.

Următoarea linie asigură că **valoarea factorului de ceață rămâne între 0** (ceață completă) **și 1** (fără ceață), evitând valori nevalide sau efecte nedorite în scenă.

Ultima linie **combină culoarea originală a fragmentului cu culoarea ceții**, în funcție de factorul de ceață. Rezultatul este o tranziție naturală, unde obiectele mai îndepărtate apar mai estompate, iar cele apropiate își păstrează claritatea.

Un alt algoritm am putea spune este mișcarea camerei. Funcția `Camera::rotate` este folosită pentru a schimba orientarea camerei prin modificarea direcției în care aceasta

privește, bazându-se pe două valori: **pitchOffset** și **yawOffset**, care reprezintă rotația pe două axe (**sus-jos** și **stânga-dreapta**).

```
glm::vec3 front;
front.x = cos(glm::radians(yawOffset)) * cos(glm::radians(pitchOffset));
front.y = sin(glm::radians(pitchOffset));
front.z = sin(glm::radians(yawOffset)) * cos(glm::radians(pitchOffset));

cameraFrontDirection = glm::normalize(front);
cameraRightDirection = glm::normalize(glm::cross(cameraFrontDirection,
cameraUpDirection));
cameraUpDirection = glm::normalize(glm::cross(cameraRightDirection,
cameraFrontDirection));

cameraTarget = cameraPosition + cameraFrontDirection;
```

Funcția ajustează direcțiile "dreapta" și "sus" pentru a se potrivi cu noua orientare a camerei și actualizează punctul către care camera se uită, în funcție de input-ul primit.

4.1.1. Soluții posibile

Pentru implementarea funcționalităților, ar fi fost posibile mai multe abordări. De exemplu, pentru algoritmul de ceață, **s-ar fi putut folosi un shader mai complex**, care aplică ceața în mod diferit în funcție de detaliile obiectelor din scenă. O altă variantă ar fi utilizarea volumetriei pentru un efect mai natural, dar nu am ales această soluție din cauza complexității suplimentare.

În ceea ce privește mișcarea camerei, **se putea implementa interpolarea liniară** pentru o mișcare mai fluidă, dar am ales o metodă mai simplă pentru a menține un control mai direct asupra camerei.

4.1.2. Motivarea abordării alese

Am ales aceste soluții pentru că oferă un bun **echilibru între performanță și realism**. Deși soluțiile alternative ar fi putut îmbunătăți anumite aspecte, am dorit să păstrez simplitatea și să mă concentrez pe crearea unei experiențe vizuale plăcute și autentice, fără a complica implementarea.

4.2. Modelul grafic

Scena 3D a fost realizată în **Blender**, unde am folosit diverse moduri pentru a construi și manipula obiectele. În „**Edit Mode**”, am modelat fiecare obiect din scenă, începând cu structurile principale (ferme, hambare, case) și continuând cu animalele și vegetația. Aceste obiecte au fost create utilizând un **număr redus de vârfuri** pentru a optimiza performanța aplicației.

Pentru fiecare obiect din scenă, am selectat „New Material” și am aplicat o „Image Texture” corespunzătoare pentru a reda aspectul natural al fiecărui element. După ce am adăugat materialul, am căutat textura potrivită din surse locale, pentru a obține detalii autentice. De exemplu, pentru hambare am folosit texturi de lemn, iar pentru copaci am aplicat texturi de frunziș și scoarță. În „**Object Mode**”, am ajustat pozițiile și scalele obiectelor, iar prin diferitele moduri de vizualizare (solid, wireframe, smooth) am optimizat structura scenei.

Astfel, întreaga scenă a fost creată într-un mod detaliat și interactiv, fiecare obiect fiind echipat cu materialele și texturile necesare pentru a reda cât mai fidel atmosfera unei vieți rurale.

4.3. Structuri de date

Pentru gestionarea camerei, am creat clasa **Camera**, care conține vectori pentru poziția camerei și direcțiile acesteia în spațiul 3D: **cameraPosition**, **cameraTarget**, **cameraUpDirection**, **cameraFrontDirection**, și **cameraRightDirection**. Aceste variabile sunt folosite pentru a defini și actualiza starea camerei în scenă. Clasa **Camera** include metode pentru a calcula matricea de vizualizare și pentru a muta și roti camera în funcție de intrările utilizatorului.

Am folosit și o enumerare **MOVE_DIRECTION**, care definește direcțiile posibile pentru mișcarea camerei (înainte, înapoi, dreapta, stânga). Aceste structuri de date ajută la gestionarea eficientă a mișcărilor și a orientării camerei în cadrul scenei 3D.

4.4. Ierarhia de clase

În cadrul aplicației OpenGL, **ierarhia de clase** joacă un rol esențial în organizarea și gestionarea funcționalităților. În mod particular, clasa Main și clasa Camera sunt esențiale pentru interacțiunea cu scena 3D și gestionarea mișcărilor vizuale.

Clasa **Main** reprezintă punctul de intrare al aplicației, responsabilă cu inițializarea și gestionarea întregului context OpenGL. Aceasta se ocupă **cu crearea ferestrei**, gestionarea loop-ului de redare (**rendering loop**), **manipularea inputurilor** și realizarea actualizării și randării scenei. În această clasă se găsesc interacțiunile directe cu utilizatorul, cum ar fi mișcarea camerei și schimbarea surselor de lumină.

Clasa **Camera** este o componentă crucială care **modelează viziunea utilizatorului asupra scenei 3D**. Aceasta gestionează poziția camerei, direcția acesteia și mișcările în scenă, permițând utilizatorului să navigheze prin mediul 3D.

Clasa Main utilizează **instanțe** ale clasei Camera pentru a actualiza și pentru a reda viziunii asupra scenei. Aceasta **interacționează** cu obiectele Camera pentru a modifica

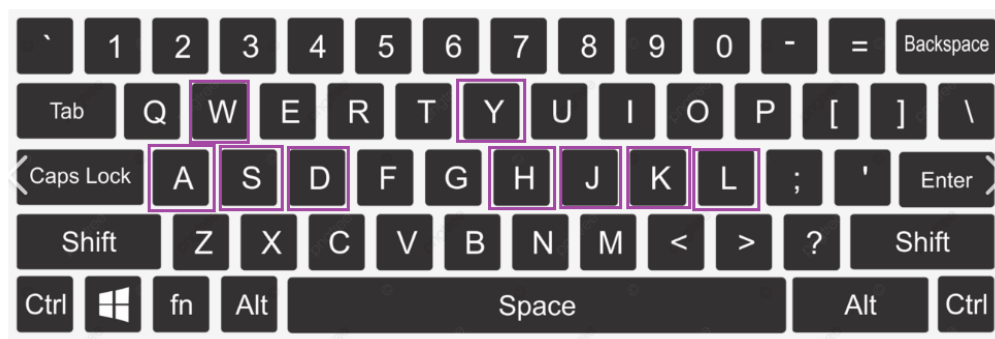
poziția, unghiul și direcția camerei pe baza inputului utilizatorului și pentru a actualiza transformările vizuale ale scenei.

5. Prezentarea interfeței grafice utilizator / manual de utilizare

Interfața grafică a utilizatorului este simplă și ușor de utilizat, având la bază un control intuitiv al scenei 3D. Utilizatorul poate naviga prin scenă folosind tastatura și mouse-ul, având posibilitatea de a manipula obiectele din scenă, de a modifica poziția camerei și de a schimba modurile de vizualizare.

Funcționalitățile includ mișcarea camerei folosind tastele "W", "A", "S", "D", **rotirea camerei prin mișcarea mouse-ului** și modificarea modurilor de vizualizare pentru a oferi o perspectivă diferită asupra scenei. Tastele "H", "J", "K" și "L" sunt asignate pentru schimbarea modurilor de vizualizare: "H" activează modul SOLID, "J" activează modul wireframe, "K" activează modul poligonal, iar "L" activează modul smooth.

De asemenea, schimbarea surselor de lumină sau activarea animațiilor poate fi realizată prin inputuri rapide, ce adaugă dinamism și realism atmosferei rurale din scenă. **Tasta "Y" este folosită pentru a schimba sursa de lumină din modul direcțional în modul punctiform**, oferind o iluminare mai detaliată și specifică în anumite zone ale scenei. Scena poate fi vizualizată atât dintr-o perspectivă generală, cât și detaliată, iar utilizatorul poate explora fiecare element al acesteia pentru a experimenta viața liniștită de la țară într-un mod captivant și interactiv.



6. Concluzii și dezvoltări ulterioare

Proiectul demonstrează cum o scenă 3D simplă poate reda cu succes atmosfera unei vieți rurale, prin utilizarea unor tehnici de modelare, animație și texturare realiste. Funcționalitățile implementate permit o explorare interactivă a scenei, iar algoritmii de mișcare a camerei și ceață contribuie la un efect vizual plăcut.

Totuși, pentru **dezvoltări ulterioare**, se poate lua în considerare integrarea unor efecte de post-procesare mai avansate sau chiar extinderea scenei pentru a include interacțiuni mai complexe, cum ar fi schimbări în mediul înconjurător sau animarea vieții rurale într-un mod mai detaliat. Adăugarea unor elemente de tip volumetric sau îmbunătățirea detaliilor geometrice ale obiectelor ar contribui la creșterea realismului scenei. În plus, implementarea unor tehnologii de realitate virtuală ar putea oferi utilizatorilor o experiență de imersie totală.

7. Referințe

<https://docs.blender.org/manual/en/latest/>

[3D Models for Free - Free3D.com](https://www.free3d.com/)

[Sketchfab](https://sketchfab.com/)

[LearnOpenGL - light casters](https://learnopengl.com/)

[Pedigree RO Alien LeftTown](https://www.pedigree.ro/)

[Spot Lights // OpenGL Tutorial #23](https://www.opengl-tutorial.org/2013/05/23/spot-lights/)

[Tons of free skyboxes - Announcements and Showcase Forum](https://www.blender.org/forum/showthread.php?p=1000000)