

FACULTATEA DE AUTOMATICA SI CALCULATOARE
DEPARTAMENTUL CALCULATOARE

PROIECT

la disciplina

Prelucrare Grafica

Scenă de obiecte 3D

Ureche Simona.
Grupa 30224.
An academic: 2024-2025

Cuprins:

Pagina:

1. Specificația subiectului.....	3
2. Scenariu.....	
2.1. Descrierea scenei și a obiectelor.....	
2.2. Funcționalități.....	
3. Detalii de implementare.....	
3.1. Mișcări.....	
3.2. Lumini.....	
3.3. Umbre.....	
3.4. Efecte.....	
4. Prezentarea interfeței grafice / manual de utilizare	
5. Concluzii și dezvoltări ulterioare	
6. Referințe	

1. Specificația subiectului

Acest proiect se axează pe integrarea și manipularea grafică a obiectelor 3D, utilizând instrumente de modelare precum Blender și tehnologia OpenGL pentru randare. Proiectul explorează cum se pot combina designul artistic și programarea grafică pentru a crea scene realiste, interactive și flexibile. Se pune accent pe utilizarea shaderelor, iluminării, umbrelor și efectelor avansate pentru a oferi utilizatorului o experiență vizuală realistă.

Am realizat un mediu 3D complex, proiectat inițial în Blender, unde am creat și configurat obiectele de bază ale scenei. Modelul principal a fost exportat și integrat în OpenGL, iar elementele suplimentare, cum ar fi ferestre, pahare sau alte componente animate, au fost exportate separat. Această abordare modulară permite o gestionare mai ușoară a scenelor și aplicarea unor efecte vizuale personalizate fiecărui element.

2. Scenariu

Proiectul se concentrează pe realizarea unei scene realiste a unei bucătării, modelată și texturată în Blender, integrată pe urmă în OpenGL. Designul bucătăriei include obiecte detaliate, cum ar fi mobilier, electrocasnice și alte elemente de decor, fiecare exportate separate sau grupate în OpenGL pentru a permite aplicarea de efecte vizuale și dinamice personalizate, cum ar fi iluminarea realistă, umbrele și animații.

2.1. Descrierea scenei și a obiectelor

Scenariul bucătăriei a început cu, construirea și modelarea terenului pe care este așezat cadrul căsuței. Terenul e reprezentat de un plan care a fost sculptat folosind funcționalități Blender, astfel încât să simuleze un cadru muntos care va fi observat de pe geamul bucatariei. După modelarea terenului, am folosit site-uri precum free3d.com, cgtrader.com sau turbosquid.com pentru a găsi obiecte sub format .fbx sau .obj care au contribui la amenajarea spațiului interior al bucătăriei, oferind un aspect modern și totuși funcțional.

- **Insula de bucătărie:** Poziționată central, insula este un element cheie în design. Aceasta include un blat de lucru pe care sunt așezate diverse obiecte, precum un tocător, farfurii și accesorii culinare. Materialele utilizate pentru insulă au fost texturi de lemn și marmură, pentru a oferi un contrast elegant.

- **Corpurile suspendate:** Pe peretele din fundal se află dulapuri suspendate cu design minimalist, realizate din lemn lăcuit.
- **Electrocasnice:** În partea stângă a scenei, se observă un cuptor încorporat, un frigider și o hotă, iar în partea dreapta a scenei am adăugat un aparat de cafea și un fierbător de apă. De asemenea, sub hotă apare și un aragaz. Fiind niste elemente care funcționează pe bază de electricitate, cateva dintre ele o să aiba atașate lumini punctiforme.
- **Masă și accesorii decorative:** Masa de dining este realizată din marmură cu un design minimalist și este însoțită de două scaune cu spătar curbat, având atașată o textură de piele. Pe masă sunt așezate obiecte decorative, precum un vază cu flori, un pahar și un decanter de vin, dar și o bucată de carne așezată pe un tocator de lemn.
- **Podeaua:** Podeaua este modelată cu un model de parchet, oferind o textură detaliată și un ton cald scenei. Textura a fost creată folosind o imagine de înaltă rezoluție aplicată pe un plan curbat.
- **Pereții și tavanul:** Au fost concepute cu un design simplu și modern, puțin rotunjit, acestea fiind elemente care completează scena, oferindu-i un cadru armonios și echilibrat, în care atât funcționalitatea, cât și estetica sunt puse în valoare
- **Șemineul din lemn:** Într-un colț al bucătăriei, se află un șemineu construit din lemn, ce adaugă un sentiment de confort și intimitate.
- **Raftul de metal:** Un element practic și decorativ al bucătăriei este un raft din metal. Pe raft se găsesc diverse obiecte, inclusiv veselă, sticle decorative și alte accesorii de bucătărie.
- **Copacii din exterior:** Pentru a simula un peisaj montan realist și a aduce un element natural în scenă, au fost adăugați copaci vizibili pe geam. Acești copaci au fost realizați utilizând o tehnică simplă și eficientă: două plane perpendiculare, pe care a fost aplicată o textură cu imaginea unui copac decupat, fără fundal
- Toate elementele menționate anterior au fost texturate în Blender, atent și specific folosind imagini potrivite materialelor, găsite pe internet.



2.2. Funcționalități

Scena creată oferă utilizatorului multiple funcționalități, ce permit explorarea și interacțiunea dinamică cu mediul. Utilizatorul poate explora bucătăria, dar și cadrul exterior folosind funcțiile camerei. Mișcarea în plan orizontal este realizată cu tastele **W, A, S, D**, în timp ce tastele **U** și **I** permit deplasarea în sus și în jos. Rotirea camerei pentru a schimba direcția de privire se face cu ajutorul săgeților de pe tastatură (**up, down, left, right**). Utilizatorul poate manipula sursa de lumină direcțională prin rotație, utilizând tastele **Q** și **E**. Această funcționalitate adaugă un nivel suplimentar de interacțiune. Scena poate fi vizualizată în moduri diferite pentru o mai bună înțelegere a compoziției sale. **Tasta 2** activează modul **wireframe**, în care se pot observa doar contururile obiectelor din scenă, iar **tasta 3** activează modul **punctiform**, oferind o altă perspectivă a elementelor din scenă. Scena se încarcă inițial cu **lumina direcțională activată**, care poate fi pornită și oprită ulterior de la **tasta L**. De la **tasta O**, utilizatorul poate activa o versiune a scenei în care lumina direcțională este completată de umbre realiste, proiectate pe baza acesteia, iar de la **tasta P**, utilizatorul poate vizualiza scena iluminată de **patru surse de lumină punctiforme**, creând o atmosferă diferită în bucătărie. În plus, apar și niste funcționalități speciale precum o animație interactivă poate fi declanșată prin apăsarea tastei **O**, în care vinul este turnat într-un pahar, iar prin apăsarea tastei **F**, utilizatorul poate activa un efect de fulger, care luminează scena temporar, oferind o atmosferă dramatică.

3. Detalii de implementare

3.1. Mișcări

- Mișcările camerei au fost implementate în clasa Camera și permit navigarea în scena 3D. Funcția `move()` gestionează deplasarea camerei față, spate, stânga, dreapta, precum și sus și jos. Deplasările frontale și laterale sunt calculate folosind `cameraFrontDirection` și `cameraRightDirection`, iar deplasările verticale utilizează `cameraUpDirection`. Prin multiplicarea acestor vectori cu o viteză de mișcare obținem navigarea în scenă.
- Funcția `rotate()` permite rotația în jurul axelor `yaw` (orizontală) și `pitch` (verticală). Se aplică transformări de rotație ce vectorul `cameraFrontDirection`, iar pentru a preveni un gimbal lock valorile sunt limitate între -89° și 89° .
- Funcția `getViewMatrix()` utilizează funcția `glm::lookAt()` pentru a returna matricea de vizualizare a camerei. Această matrice este esențială pentru a transforma coordonatele lumii în coordonatele camerei.
- Este realizat un **tur** animat al camerei prin scenă, care se deplasează între puncte predeterminate într-un mod fluid, folosind o ecuație a liniei parametrice pentru a calcula poziția și ținta camerei în fiecare cadru. Funcția **updateCameraMovement** actualizează progresul camerei pe parcursul turului, mutând camera între două puncte, în funcție de timpul scurs și viteza turului.

După fiecare pas complet între două puncte, se trece la următoarea pereche de puncta.

3.2. Lumini

- Directionala: a fost implementată folosind modelul de iluminare Phong, învățat în cadrul laboratorului. Funcția **computeDirLight()** din shader-ul principal e responsabilă de calcularea celor 3 componente principale ale luminii **ambientala**, **difuza** și **speculara**. Componenta **ambientala** este calculată prin înmulțirea constantei **ambientStrength** cu culoarea luminii asigurând iluminarea globală a scenei. Componenta **difuză** se obține folosind produsul scalar dintre unghiul vectorului normal și direcția luminii, iar componenta **speculara** este responsabilă pentru reflexiile lucioase de pe suprafețe și este calculată folosind vectorul de reflexie și direcția privirii (**viewDirNorm**).



- Punctiforma: modelul de iluminare Phong a fost utilizat și pentru implementarea luminilor punctiforme. Aceste lumini oferă o iluminare localizată. În funcția **computePointLight()** se calculează din nou cele 3 componente, ambientala, difuza și speculara, însă de data aceasta se ține cont și de poziția luminii față de fragment și se aplică o atenuare pe baza de distanță. Atenuarea este definită printr-o ecuație dependentă de coeficienții constant, linear și quadratic, astfel încât intensitatea luminii scade pe măsură ce distanța crește.
 - ➔ Luminile punctiforme au fost folosite la hotă, frigider, cuptor și semineu, fiecare având poziții și culori predefinite.



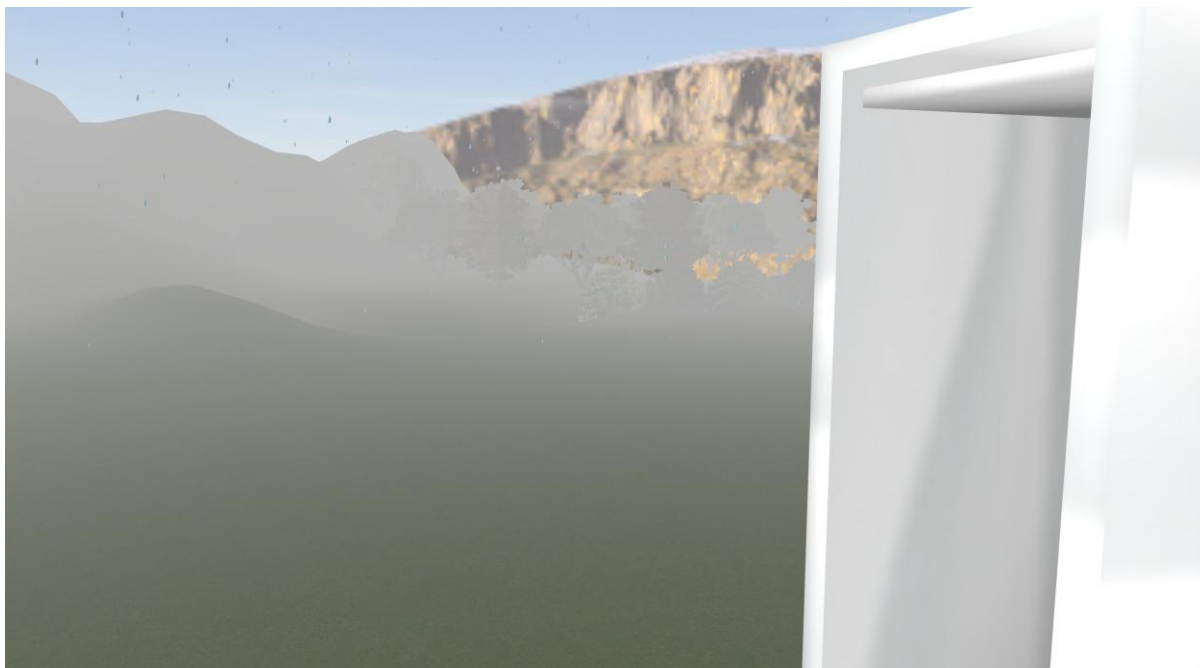
3.3. Umbre

- Sunt aplicate pe sursa de lumina direcțională folosind harta de adâncime. Implementarea acestui efect este bazată pe metoda **shadow mapping**, care utilizează o hartă de adâncime pentru a calcula umbrele proiectate pe obiecte.
- Funcția `computeShadow()` este folosită în shader pentru a determina dacă un fragment din scenă este în umbră sau nu. Transformă coordonatele în spațiul lumi după care compară adâncimea fragmentului cu cea mai apropiată adâncime din harta de umbre și decide dacă fragmentul respectiv este sau nu umbrat.
- Harta de adâncime este generată folosind un FBO(Frame Buffer Object) căruia îi este atașată o textură de adâncime.
- Funcția `computeLightSpaceTrMatrix()` generează matricea de transformare a coordonatelor în spațiul luminii.



3.4. Efecte

- **Ceata:** Efectul de ceață este realizat prin calcularea unui factor de atenuare a culorii pe baza distanței dintre fragment și cameră, utilizând o funcție exponențială pentru a simula efectul de dispersie a luminii pe măsură ce obiectele se află mai departe de vizualizator. Acest efect reduce vizibilitatea detaliilor în fundal, creând o atmosferă mai naturală și profundă în scenă, cu o tranziție lină între obiectele din prim-plan și cele din fundal.



- **Fragment Discarding:** Efectul de Fragment Discarding este aplicat pentru a elimina fragmentele transparente ale texturii copacilor vizibile prin geamul bucătăriei. Acesta utilizează o condiție în care fragmentele cu un canal alpha mai mic de 0.1 sunt eliminate, împiedicând astfel afișarea părților transparente ale texturii și afișând doar părțile opace.



- **Obiectele transparente:** Obiectele transparente din scena mea sunt gemul, geamul de la cuptorul cu microunde, paharul de vin, decanterul de vin și câteva obiecte de decor de pe insula de la bucătărie. Pentru a reda transparența acestor obiecte am folosit un shader separat în care am adăugat texturii obiectelor o transparență controlată prin canalul alpha. Am adăugat și lumini obiectelor transparente. De asemenea pentru a permite transparența obiectelor se aplică o funcție de amestecare între culoarea obiectului și culoarea de fundal - `glEnable(GL_BLEND)` și `glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA)`.



- **Surse de lumina punctuala:** am mai adăugat încă 3 surse de lumina punctuala, după modelul luminii punctuale principale, schimbând doar poziția și culorile fiecăreia.
- **Ploaia:** Efectul de ploaie este realizat prin simularea unor particule care se mișcă pe o direcție verticală. Fiecare particula are o poziție și o viteză proprie, iar la fiecare apel al funcției `initRainParticles()` se generează un număr aleator de particule între 0 și 9, fiecare dintre ele primind coordonate aleatorii dintr-un codru restrâns în care plouă. Din funcția `updateRainParticles()` oferim mișcare particulelor, iar când acestea ajung la baza terenului, poziția lor este resetată din nou în cadrul în care plouă.
- **Fulger:** Efectul de fulger constă într-o secvență de schimbări rapide de lumină între negru și alb, însoțite de un sunet de fulger, care se activează timp de 2 secunde de întuneric, urmate de o scurtă iluminare albă și apoi întuneric din nou, restaurând scena la starea inițială după 3 secunde.
- **SkyBox:** skybox-ul presupune încărcarea celor șase texturi corespunzătoare fațadelor unui cub (dreapta, stânga, sus, jos, spate, față) pentru a crea o imagine panoramică ce înconjoară scena. Aceste texturi sunt încărcate într-un obiect `mySkyBox`, iar shader-ul asociat se ocupă de proiecția și vizualizarea acestora într-un cub, aplicând coordonatele texturii pentru a reda imaginea skybox-ului în scenă.

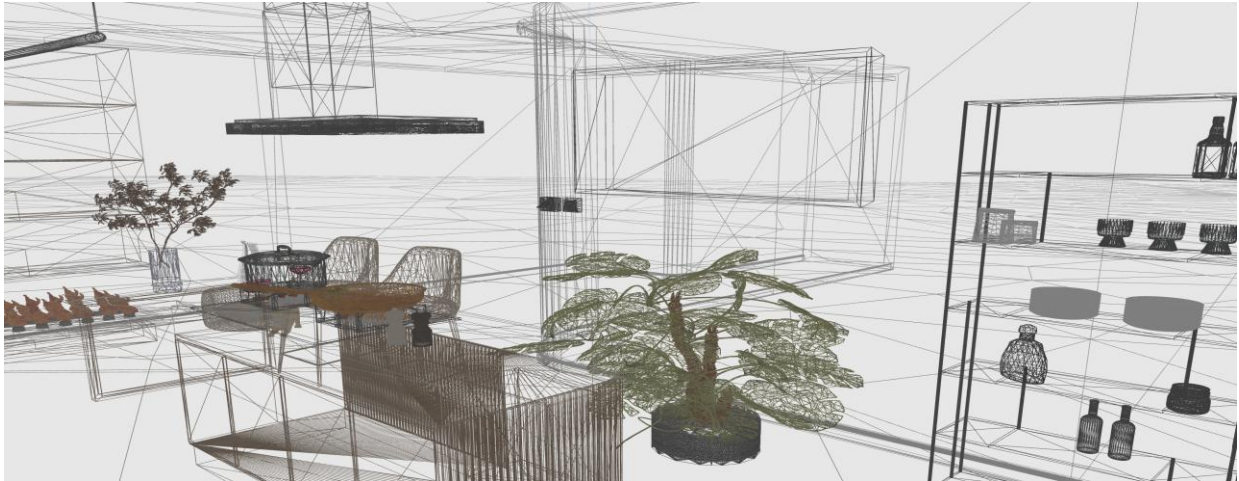
- **Animație vin:** funcția `handleAnimation()` animă mișcarea unui decanter și a unui pahar, mutând decanterul dintr-o poziție inițială într-o poziție finală, apoi adăugând vinul în pahar și restabilind decanterul la poziția inițială, cu o tranziție lină între poziții.



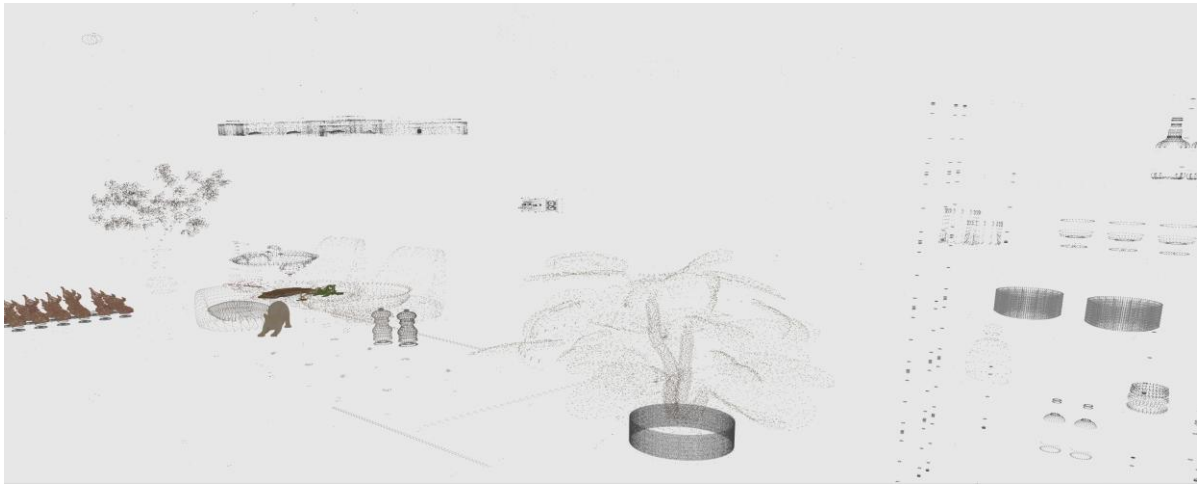
7. Prezentarea interfeței grafice / manual de utilizare

W, A, S, D	Mișcarea în plan orizontal
U, I	Deplasarea în sus și în jos
up, down, left, right	Rotirea camerei
Q, E	Rotația sursei de lumina
1	Modul solid
2	Modul wireframe
3	Modul punctiform
L	Activare lumina direcțională
O	Activare lumina direcțională completată de umbre
P	Activare lumini punctiforme
0	Animație vin
F	Efect fulger

- **Modul wireframe**



- **Modul punctiform**



- **Harta de umbre**



8. Colcluzii și dezvoltări ulterioare

- În acest proiect am dezvoltat o scenă interactivă în OpenGL, cu funcționalități diverse și detalii de implementare care acoperă iluminarea, navigarea camerei și efecte speciale. Scenei i-a fost aplicată o iluminare detaliată, cu efecte de ploaie și fulger, iar interacțiunea utilizatorului este facilitată prin navigarea automatizată a camerei. De asemenea, implementarea include detalii precum obiecte transparente și animații fluide.
- Pe viitor, proiectul poate fi extins prin
 - Implementarea unui algoritm de detectare a coliziunilor pentru particulele de ploaie care ating solul sau alte obiecte.
 - Integrarea unei hărți de umbre pentru obiecte precum frigiderul, pentru a adăuga realism la iluminare.
 - Implementarea unui efect de foc dinamic în semineu, care reacționează la schimbările de iluminare și vânt
 - Extinderea animației pisicii și adăugarea de interacțiuni simple între utilizator și animal.
 - Integrarea de obiecte care reflectă lumina și mediul înconjurător pentru a îmbogăți aspectul vizual al scenei.

9. Referințe

În proiect, am utilizat laboratoarele de prelucrare grafică pentru dezvoltarea mediului 3D și am accesat site-uri precum free3d.com, cgtrader.com și turbosquid.com pentru obținerea modelelor 3D necesare.