

Sisteme de prelucrare grafică

Documentație proiect

Student : Bolba Raluca Maria

Grupa : 30235

1. Cuprins

[2. Prezentarea temei 2](#_Toc440142478)

[3. Scenariul 2](#_Toc440142479)

[3.1. Descrierea scenei și a obiectelor 2](#_Toc440142480)

[3.2. Funcționalități 3](#_Toc440142481)

[4. Detalii de implementare 4](#_Toc440142482)

[4.1. Funcții și algoritmi 4](#_Toc440142483)

[4.1.1. Soluții posibile 4](#_Toc440142484)

[4.1.2. Motivarea abordării alese 5](#_Toc440142485)

[4.2. Modelul grafic 5](#_Toc440142486)

[4.3. Structuri de date 6](#_Toc440142487)

[5. Prezentarea interfeței grafice cu utilizatorul / manual de utilizare 7](#_Toc440142488)

[6. Concluzii și dezvoltări ulterioare 9](#_Toc440142489)

[7. Referințe 9](#_Toc440142490)

# Prezentarea temei

Tema proiectului o reprezintă modelul tridimensional al unei scene ce constă într-o cabană dintr-o pădure de brazi. Utilizatorul aplicației are posibilitatea de a se deplasa în cadrul scenei pentru a vizualiza cabana și brazii, dar poate să intre și în cabană, care conține la rândul ei alte obiecte decorative.

Scena de obiecte are două surse de lumini, una reprezentând soarele, iar alta este situată în interiorul cabanei și reprezintă o lanternă. Pentru cea din urmă utilizatorul poate modifica anumite atribute, cum ar fi poziția luminii sau culoarea. Prima lumină este fixă și nu poate fi controlată de utilizator.

Utilizatorul poate de asemenea să controleze vremea din cadrul scenei, mai precis să aleagă dintre opțiunile de ploaie, ninsoare și/sau ceață.

Interfața grafică cu utilizatorul este una simplă, intuitivă, care permite utilizatorului să controleze scena prin intermediul mouse-ului. Totodată, scena poate fi controlată și prin intermediul tastelor.

# Scenariul

## Descrierea scenei și a obiectelor

Scena constă dintr-o pădure de brazi, care sunt realizați prin repetarea unei texturi de brad care dă impresia unui obiect tridimensional, și o cabană situată în mijlocul acesteia. Toate obiectele sunt situate în interiorul unui „skydome” sau altfel spus, o semisferă pe care este aplicată o textură de cer. Semisferă este așezată la rândul ei pe un plan, pe care este aplicat de asemenea o textură.

Interiorul cabanei este și el la rândul lui modelat ca un ”skybox” sau altfel spus, un cub pentru a cărui fețe s-au aplicat texturi de perete, podea și tavan, creând iluzia unei camere.

Obiectele din cadrul scenei sunt :

* O cabană de lemn
* Brazi – realizați prin texturi
* O canapea
* O lampă
* Un ceas
* O pereche de teniși
* O sferă care are rol de lanternă în interiorul cabanei
* O semisferă în interiorul căreia sunt plasate toate obiectele

## Funcționalități

Printre funcționalitățile proiectului se numără :

* Posibilitatea utilizatorului de a se deplasa în scenă prin : translații, rotații și scalări pe toate cele trei axe de coordonate, atât din cadrul interfețe grafice cât și prin intermediul tastelor
* Posibilitatea utilizatorului de a controla sursele de lumini ale scenei; acesta poate sa activeze sau să dezactiveze oricare dintre cele două surse de lumini din cadrul interfeței grafice; pentru sursa de lumină din interiorul camerei poate să modifice din intermediul interfeței grafice poziția acesteia și culoarea ei
* Posibilitatea utilizatorului de a activa sau dezactiva umbrele din cadrul scenei
* Posibilitatea utilizatorului de a controla un obiect din scenă, mai precis pereche de teniși din interiorul cabanei; utilizatorul poate translata, roti sau scala obiectul prin intermediul tastaturii pe toate cele trei axe
* Posibilitatea utilizatorului de a controla din intermediul interfeței grafice vremea din cadrul scenei, respectiv, să activeze ploaia, ninsoarea și/sau ceața
* Posibilitatea utilizatorului de a porni animația de prezentare prin intermediul tastaturii; animația de prezentare are rolul de a ”plimba” poziția camerei în cadrul scenei pentru a vizualiza toate obiectele
* Posibilitatea utilizatorului de a alege unul din următoarele moduri de vizualizare : wireframe, solid, poligonal sau smooth
* Animația unor componente ale obiectelor, respectiv utilizatorul poate observa cum acele ceasului se rotesc, acul care indică ora rotindu-se cu o viteză mai mică decât acul care indică minutele
* Atunci când utilizatorul se deplasează în scenă, în momentul în care ajunge la pereții cabanei, acesta va fi plasat în interiorul acesteia, adică în interiorul skybox-ului; La fel se întâmplă și când observatorul este în interiorul camerei și înaintează spre pereții încăperii, acesta fiind plasat în exteriorul skybox-ului

# Detalii de implementare

## 4.1. Funcții și algoritmi

### 4.1.1. Soluții posibile

* Calcularea umbrelor proiectate

În [1] este prezentat un algoritm pentru calcularea umbrelor proiectate și de volum. În cazul umbrelor de primul tip, trebuie calculate ecuațiile planelor pe care vor fi proiectate umbrele, pe baza a trei puncte ale planului și calcularea matricelor de umbre (necesită produsul scalar dintre ecuația planului și poziția sursei de lumină).

Așadar, se desenează întâi planul fără umbre, se multiplică matricea curentă cu matricea de umbră și se desenează obiectul folosind o culoare de umbră, de obicei negru, după care se desenează obiectul normal.

* Realizarea ploii și ninsorii cu ajutorul particulelor

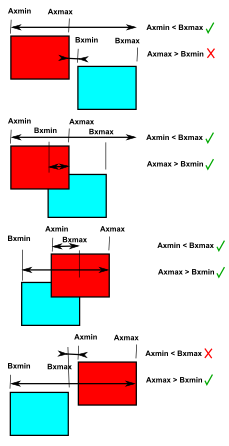
Un sistem de particule este definit în [2] ca fiind o tehnică utilizată în grafica pe calculator de a realiza animații folosind un număr foarte mare de primitive sau modele 3d mici pentru a simula fenomene precum ploaie, ninsoare, foc etc. care sunt foarte greu de realizat prin metode convenționale.

O particulă se poate reprezenta prin intermediul unei structuri de date, întrucât are mai multe atribute precum culoare, poziție, durata de viață, direcție etc. Poziția fiecărei particule este determinată în mod aleatoriu, pentru a crea un efect cât mai natural.

La desenarea particulei, se iau atributele de poziție și culoare, dar și cele ce privesc timpul de viață și durata în care dispare particula. Poziția acesteia este actualizată astfel încât ea să se deplaseze în jos în funcție de viteză dar și de gravitate. Pentru ploaie se desenează linii, iar pentru zăpadă sfere albe.

* Detecția coliziunilor între două obiecte

Detecția coliziunilor este folosită pentru a evita suprapunerea în scenă a două obiecte. Dacă verificarea pentru două pătrate (în spațiul bidimensional) sau a două cuburi (în tridimensional) acest lucru se rezumă la verificarea colțurilor pătratelor precum în poza de mai jos, în cadrul obiectelor, care pot avea diferite forme sau pot avea un număr mult mai mare de vârfuri, trebuie făcute simplificări.

Așadar, pentru a detecta coliziune dintre două obiecte, se poate considera pentru fiecare dintre obiecte o înfășurătoare sub formă de cub [3]. Înfășurătoarea poate fi determinată ușor dacă pentru fiecare obiect cunoaștem poziția și factorul de scalare.

Două obiecte se suprapun așadar dacă proiecțiile lor pe *X, Y, și Z* se suprapun, sau dacă:

shapeA->getX() + shapeA-> getSize() > shapeB->getX()

&& shapeA->getX() < shapeB-> getX() + shapeB->getSize()

&& shapeA->getY() < shapeB->getY() + shapeB->getSize()

&& shapeA->getY() + shapeA->getSize() > shapeB->getY()

&& shapeA->getZ() < shapeB->getZ() + shapeB->getSize()

&& shapeA->getZ() + shapeA->getSize() > shapeB->getZ()

### 4.1.2. Motivarea abordării alese

Algoritmii pentru detecția coliziunilor și pentru generarea umbrelor proiectate descriși mai sus sunt printre cei mai simplii pentru problemele descrise. Algoritmul de generare a ploii și ninsorii este atât simplu, cât și eficient, întrucât modelarea 3d și animarea fiecărui obiect ar necesita foarte multă memorie și timp.

## 4.2. Modelul grafic

Modelul grafic constă din o serie de obiecte amintite anterior (cabana de lemn, brazi, canapea, lampă, pereche de teniși, ceas) dintre care componentele ceasului sunt animate, respectiv acele acestuia. Tot din model fac parte efectele de ploaie și ninsoare, realizate prin intermediul particulele, pentru care s-au declarat structuri de date.

## 4.3. Structuri de date

Singura structură de date folosită în cadrul aplicației este cea pentru particulele de ploaie și ninsoare.

typedef struct {

// Viata particulei

bool alive; // este particula vie?

float life; // durata de viață a particulei

float fade;

// culoare

float red;

float green;

float blue;

// Poziția

float xpos;

float ypos;

float zpos;

// Direcția

float vel;

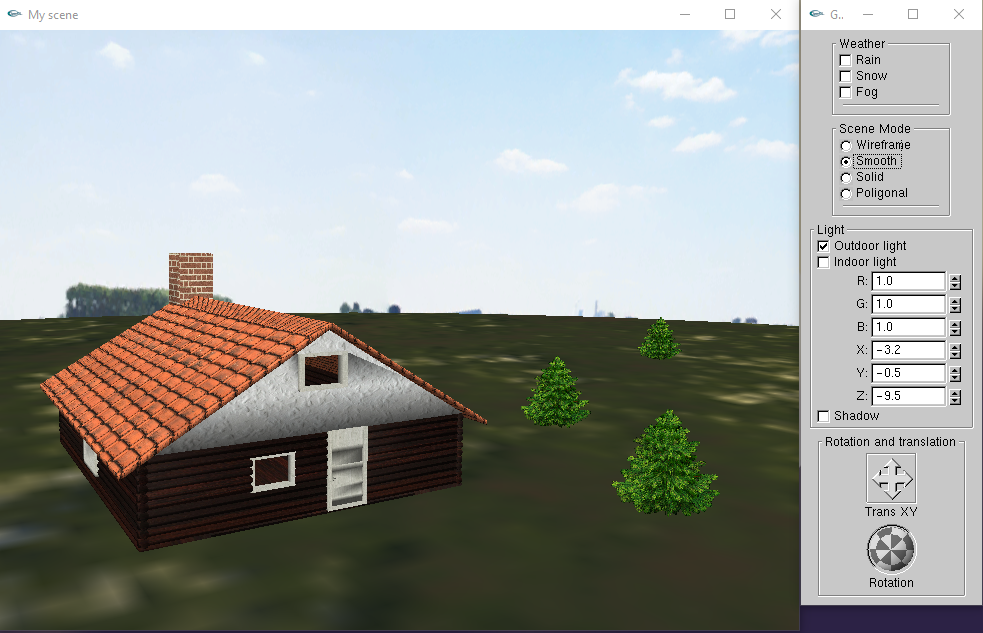
// Gravitatea

float gravity;

}particles;

Pentru aceasta se definesc mai multe atribute, cum ar fi culoarea acesteia, poziția în scenă, gravitatea, direcția, durata de viață și un atribut care indică dacă particula este ”vie”.

# Prezentarea interfeței grafice cu utilizatorul / manual de utilizare



În partea stângă a aplicației se află fereastra ce conține scena tridimensională, iar în partea dreaptă meniul de interfață cu utilizatorul. După cum se observă, utilizatorul poate controla vremea prin activarea sau dezactivarea unuia dintre cele trei checkbox-uri, respectiv *Rain, Snow, Fog.*

Pentru modificarea scenei de vizualizare, utilizatorul va alege una din variantele *Wireframe, Smooth, Solid, Poligonal* reprezentate ca niște radio buttons.

Activarea și dezactivarea luminilor se realizează bifând căsuțele *Outdoor light*, adică lumina globală, sau *Indoor light*, lumina din interiorul cabanei. Pentru lumina din interiorul cabanei se poate modifica culoarea prin modificarea celor trei canale de culori (*R, G, B)* sau poziția acesteia prin modificarea coordonatelor X, Y și Z.

Bifarea căsuței *Shadow* determină afișarea umbrelor din scena de obiecte. Umbrele cabanei și a brazilor sunt calculate în funcție de lumina globală, iar umbrele pentru obiectele din interiorul cabanei sunt calculate în funcție de lumina de tip spot.

Tot din interfața grafică se pot realiza deplasări în scenă, prin rotații și/sau translații.

În ceea ce privește controlul tastelor, acestea și acțiunile lor sunt :

* Translație pe axa X – tastele 1 și 4
* Translație pe axa Y – tastele 2 și 5
* Translație pe axa Z – tastele 3 și 6
* Rotație pe axa X – tastele w și s
* Rotație pe axa Y – tastele a și d
* Rotație pe axa Z – tastele q și e
* Scalare pe axa X – tastele j și i
* Scalare pe axa Y – tastele o și k
* Scalare pe axa Z – tastele p și l
* Pornire animație – tasta ;
* Translație obiect pe axa X – ↑ și ↓
* Translație obiect pe axa Y – → și ←
* Translație obiect pe axa Z – tasta *PageUp* și *PageDown*
* Rotație obiect pe axa X – tasta *F1* și *F2*
* Rotație obiect pe axa Y – tasta *F3* și *F4*
* Rotație obiect pe axa Z – tasta *F5* și *F6*
* Scalare obiect pe axa X – tasta *F7* și *F8*
* Scalare obiect pe axa Y – tasta *F9* și *F10*
* Scalare obiect pe axa Z – tasta *F11* și *F12*

# Concluzii și dezvoltări ulterioare

În ceea ce privește proiectul, se pot aduce îmbunătățiri precum :

* Adăugarea umbrelor de volum
* Modificarea obiectelor sau a texturilor pentru a fi mai realiste
* Modificarea obiectelor astfel încât ușa cabanei
* Adăugarea de umbre pe toate planele din interiorul cabanei
* Creșterea fotorealismului
* Generare efectului de vânt
* Înlocuirea texturilor de brad cu obiecte tridimensionale

# Referințe

[1] Laboratory work 9

[2] Particle system - <https://en.wikipedia.org/wiki/Particle_system#.22Snowflakes.22_versus_.22Hair.22>

[3] 3D Theory – Collision Detection

<http://www.euclideanspace.com/threed/animation/collisiondetect/>