

Prezentarea unei idei de licență

Moisel Rares-Ioan

October 13, 2024

1 Noțiuni preliminare

1.1 Signed distance functions (SDFs)

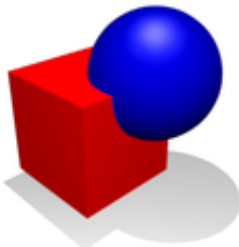
O SDF este o funcție care returnează distanța dintre oricare două puncte din spațiu. În cazul de față va returna distanța dintre oricare punct din spațiu și o suprafață implicită. Suprafața implicită reprezintă mulțimea punctelor în care valoarea funcției este zero. Ca exemplu, dacă dorim să reprezentăm o sferă folosind o funcție SDF, valorile funcției pentru punctele din exteriorul sferei vor fi pozitive, ale celor care se află pe sferă vor fi zero, iar pentru cele care se află în bilă vor fi negative. Această funcție poate fi exprimată astfel pentru o sferă cu centrul în origine și de rază R :

$$f(x, y, z) = \sqrt{x^2 + y^2 + z^2} - R$$

1.2 Geometrie solidă constructivă (CSG)

CSG este o tehnică de creare și modelare de suprafețe geometrice prin folosirea operațiilor booleene. În cazul SDF-urilor aceste operații se realizează astfel:

- Uniunea: $A \cup B = \min(A, B)$
- Intersecția: $A \cap B = \max(A, B)$
- Diferența: $A - B = \max(A, -B)$



Uniune



Intesecție



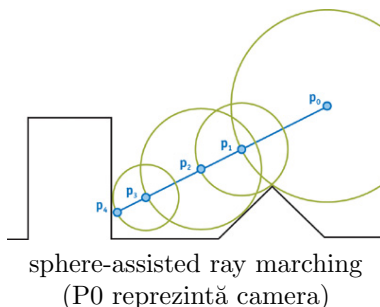
Diferență

2 Algoritmul Ray marching

Ray marching este o tehnică de randare similară cu ray tracing, unde trimitem raze dinspre cameră spre scenă. Aceste raze sunt traversate în mod iterativ, verificându-se la fiecare iterație anumite condiții. Pentru eficiență, folosim o variație a acestui algoritm numită sphere-assisted ray marching. Folosind acest algoritm și SDF-uri, la fiecare iterație a unei raze trebuie să se ia următoarele două decizii:

- Cât de mare să fie acest pas
- Dacă raza a intersectat un obiect

Pentru a decide cât de mult va înainta raza la pasul curent, calculăm distanța până la cea mai apropiată suprafață implicită. Această distanță definește raza unei sfere care intersectează suprafața. După ce am decis lungimea pasului, trebuie să aflăm dacă avem vreo intersecție. Acest lucru se face ușor prin verificarea fiecărui obiect. Dacă vreun SDF este zero, înseamnă că am intersectat o suprafață.



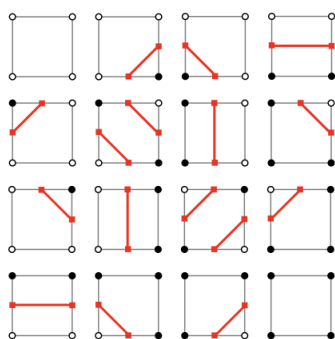
3 Opținerea mesh-ului

Pentru că încă nu am decis ce algoritm voi folosi în acest pas, voi prezenta doi algoritmi care sunt, cel mai probabil, opțiunile mele.

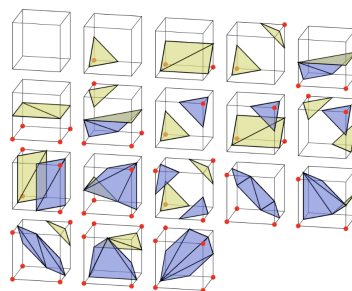
3.1 Marching cubes

Algoritmul Marching cubes consistă din următorii pași:

1. Împărțim spațiul într-un grid de cuburi egale. Pentru fiecare astfel de celulă, verificăm ce colțuri sunt conținute în obiect.
2. Procesăm fiecare celulă care conține colțuri în interiorul obiectului și alegem o suprafață potrivită. Această alegere se face dintr-un lookup table care conține toate cazurile posibile. În cazul 2D, acest tabel conține 2^4 combinații, iar în 3D $2^8 (= 256)$. În cazul 3D, chiar dacă numărul de posibilități este 256, doar 15 sunt necesare, deoarece restul reprezintă doar ogindiri sau rotații ale altora.



cazurile 2D



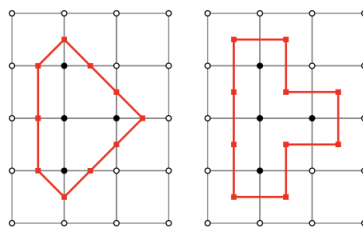
cazurile 3D

Deși în imaginea pentru cazurile 3D sunt 18 posibilități, ultimele 3 nu sunt necesare.

3. Conectăm suprafețele alese din fiecare celulă.

3.2 Dual Contouring

Similar cu Marching Cubes, începem prin a împărți spațiul într-un grid de cuburi care să conțină obiectul nostru. Diferența majoră între cei doi algoritmi este că în Dual Contouring plasăm un nod în interiorul celulelor care intersectează obiectul și unim aceste noduri pentru a construi suprafața.



Marching cubes vs Dual
Contouring

Fața de Marching Cubes la conectarea suprafeței nu mai putem evalua celulele gridului în mod independent, ci trebuie să le considerăm și pe cele adiacente. Acest lucru se realizează prin găsirea fiecărei margini de celulă care conține un colț în interiorul obiectului și un colț în exterior și conectând nodurile din interiorul acestor celule. În imaginea anterioară, aceste margini sunt cele care au un punct alb și un punct negru.

4 Observații

Este de menționat că, pentru toți algoritmi prezentați, am omis unele detalii legate de implementare și optimizare. Nu am subliniat punctele forte și slabe ale variantelor discutate. De asemenea, am lăsat deoparte descrierea structurilor de date care ar putea fi necesare pentru stocarea anumitor informații în cadrul operațiilor CSG.

O sa las aici câteva surse care explică mai în detaliu algoritmii:

- [SDFs si Ray Marching](#)
- [Marching Cubes](#)
- [Dual Contouring](#)