

Ray Marching

Burgalat - Garcia

Table des matières

1	Tout commence par le commencement	2
1.1	Affichage	2
1.2	SdF	2
1.3	Affichage des lumières et ombres	2
2	Opérations	2
2.1	Opérations de base :	2
2.1.1	Union	2
2.1.2	Intersection	2
2.1.3	Soustarction	3
2.1.4	Rotations	3
2.2	Amélioration des effets	3
2.2.1	Fonctions de lissage - Smooth	3
3	Optimisations	3
3.1	Bounding Volumes Hierarchy (BVH-Trees)	3
3.2	Multi-threading	3

1 Tout commence par le commencement

1.1 Affichage

1.2 SdF

Le principe de la Signed Distance Function (SdF) est de calculer la distance entre un point et un objet.

On peut donc donner à chaque forme de base une fonction distance afin de permettre, lors du lancer de chaque rayon, de savoir si le rayon atteint l'objet (la SdF renverra une valeur négative lorsqu'il arrivera à un point dans l'objet), où s'il ne touche pas (à partir d'une certaine distance atteinte par le rayon, on considère qu'aucun objet est atteint et alors on renvoie un fond uniforme).

Exemple de la sphère :

```
1 float SDF_sphere(coord p, coord centre, float rayon){
2     float dist =
3         sqrt((p.x-centre.x)*(p.x-centre.x)+(p.y-centre.y)*(p.y-centre.y)+(p.z-centre.z)*(p.z-centre.z));
4     return dist - rayon;
5 }
```

Les formes de base implémentées sont :

- Plan
- Sphère
- Tor
- Boite
- Pyramide
- Cylindre

1.3 Affichage des lumières et ombres

2 Opérations

2.1 Opérations de base :

2.1.1 Union

Afin de procéder l'union entre 2 objets/formes, il suffit de considérer à chaque fois la plus faible valeur entre la sdf d'un objet et celle de l'autre, ce qui nous permettra, pour chaque rayon de lancé, de capter la présence de la première forme présente sur son chemin.

```
1 float UnionSDF (float d1, float d2){
2     return MIN(d1,d2);
3 }
```

2.1.2 Intersection

Au contraire pour l'Intersection, il suffit de considérer la distance maximale : Si un des 2 objets n'est pas présent, la fonction intersection renverra toujours le maximum des 2 sdf, soit une distance positive et ainsi

aucun objet ne sera atteint. Pour les points sur lesquels les 2 objets sont présents, la fonction renverra bien une distance négative (car les 2 le sont) à l'arrivée du rayon en ce point.

```
1 float IntersectSDF (float d1, float d2){  
2     return MAX(d1,d2);  
3 }
```

2.1.3 Soustraction

La soustraction de F1 par F2 se traduit à $F1 \setminus F2$

Il suffit donc de considérer les points où la sdf de F1 est positive, et celle de F2 négative.

```
1 float SubtractSDF (float d1, float d2){  
2     return MAX(d1, -d2);  
3 }
```

2.1.4 Rotations

2.2 Amélioration des effets

2.2.1 Fonctions de lissage - Smooth

3 Optimisations

3.1 Bounding Volumes Hierarchy (BVH-Trees)

Arbres K-dimensionnels de priorisation des SdF

3.2 Multi-threading