

Projet INFSI 350 - RayMini

Arnaud Douceur
Daniel Ross

Avril 2011

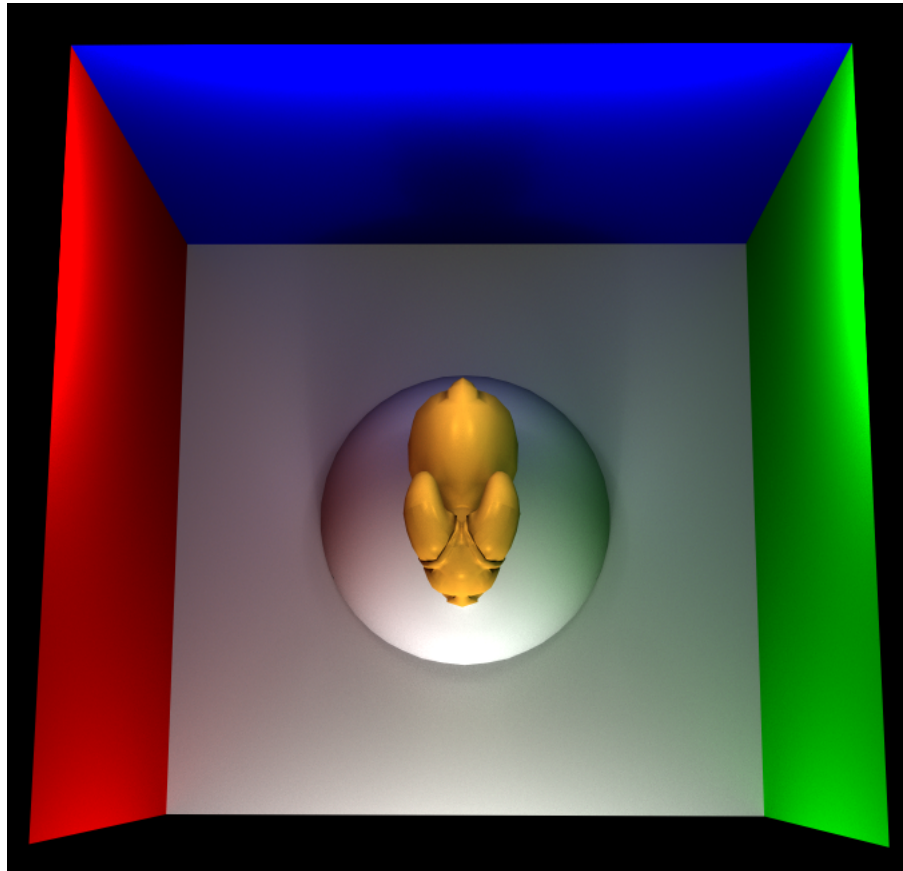


Table des matières

1	Fonctionnalités implémentées	3
2	Introduction	4
3	Éclairage Direct	4
3.1	BRDF de Phong	4
3.2	Ombres Dures	5
3.3	Ombres Douces	5
3.4	Ambient Occlusion	7
3.5	Anti-aliasing	8
3.6	Miroir	9
3.7	Profondeur de champ	9
4	Éclairage Indirect	10
4.1	Pathtracing	10
5	Limites	11

1 Fonctionnalités implémentées

1. Intersection rayon-triangle
2. BRDF de Phong
3. Rendu multi-threads
4. Kd-tree
5. Ombres dures
6. Ombres douces (source de lumière étendu en forme de disque)
7. Ambient occlusion
8. Anti-aliasing
9. Mirroir
10. Profondeur de champ
11. Pathtracing

2 Introduction

Le projet de 3D du module INFSI350 proposait d'implémenter en C++ un moteur de rendu en lancé de rayons (raytracing).

Partant d'une interface graphique en Qt et d'une fenêtre de pré-visualisation avec une scène 3D rendue via OpenGL, le but de ce projet était d'afficher notre propre rendu de la scène.

Pour cela, nous avons dans un premier temps implémenté l'éclairage direct avec des effets tels que la BRDF de Phong, les ombres dures et douces, l'ambient occlusion et l'anti-aliasing. Dans un second temps, nous avons choisi d'implémenter le pathtracing stochastique ainsi que des effets supplémentaires comme la profondeur de champ et les matériaux miroirs.

3 Éclairage Direct

3.1 BRDF de Phong

Afin d'implémenter une BRDF de Phong, nous avons ajouté un attribut *shininess* à la classe *Materiel* pour satisfaire à l'équation vu en cours :

$f(\omega_i, \omega_o) = k_d(n \wedge \omega_i) + k_s(r \wedge \omega_o)^S$ avec $r = 2n(\omega_i \wedge n) - \omega_i$ et k_d , k_s et S les coefficients diffus, spéculaire et brillance.

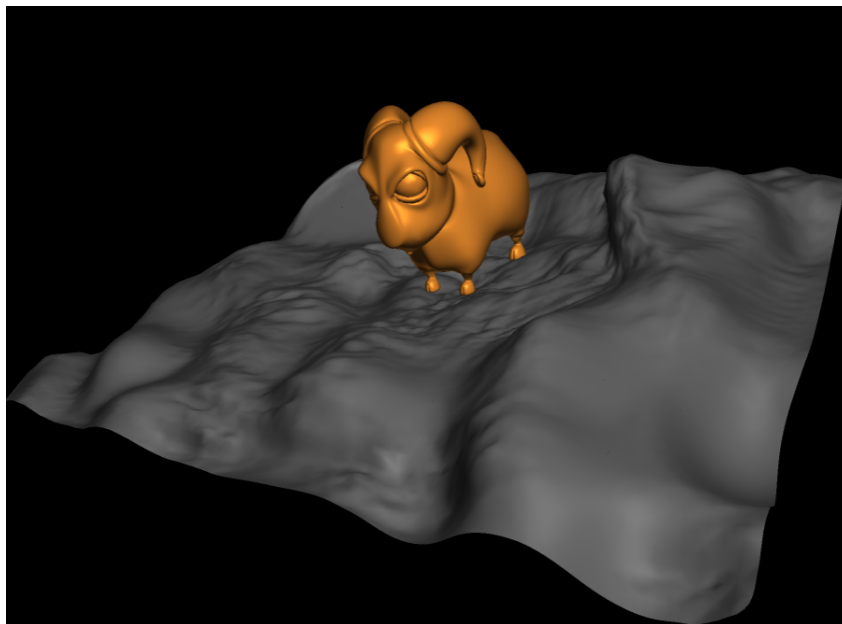


FIG. 1 – BRDF de Phong

3.2 Ombres Dures

À partir d'une source de lumière ponctuelle, nous avons défini une méthode binaire qui colorie les pixels qui sont visibles en effectuant une BRDF de Phong.

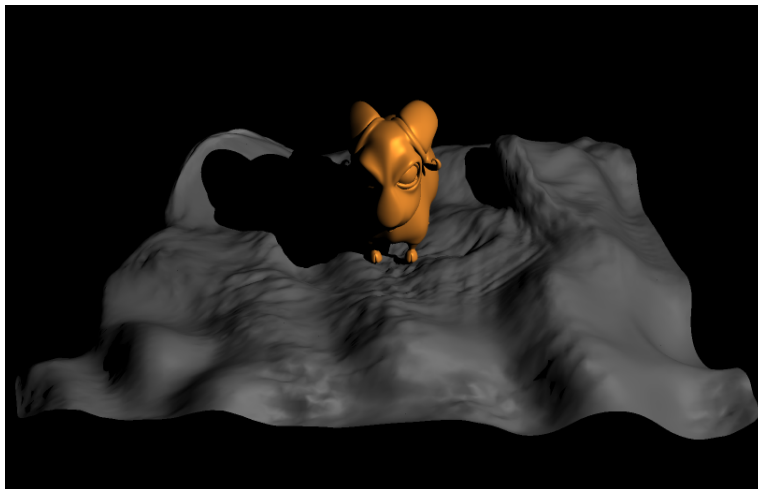


FIG. 2 – Ombres dures

3.3 Ombres Douces

Afin d'obtenir des ombres douces nous avons implémenté une classe Area-Light qui étend la classe Light. La lampe a une forme de disque. Nous simulons ses effets, en comptabilisant un nombre N de rayons tirés aléatoirement dans la direction de la lumière. Le ratio des rayons visibles pondère le résultat de la BRDF de Phong pour obtenir l'effet escompté.

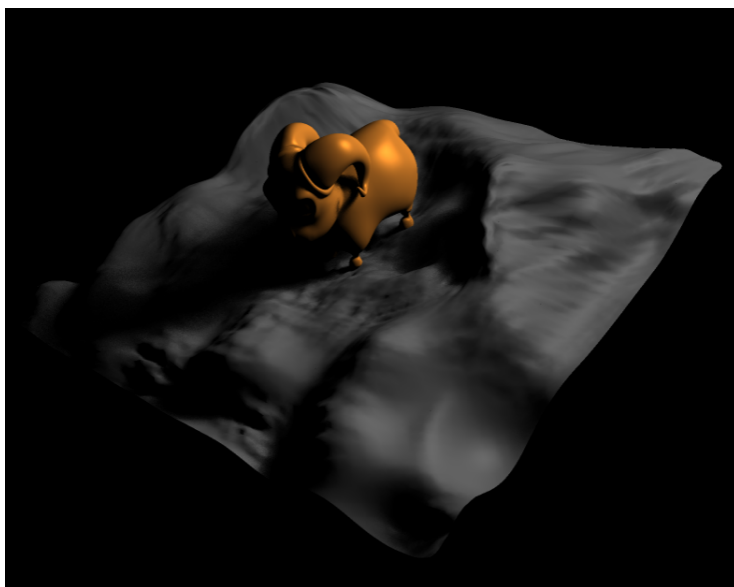


FIG. 3 – Ombres douces

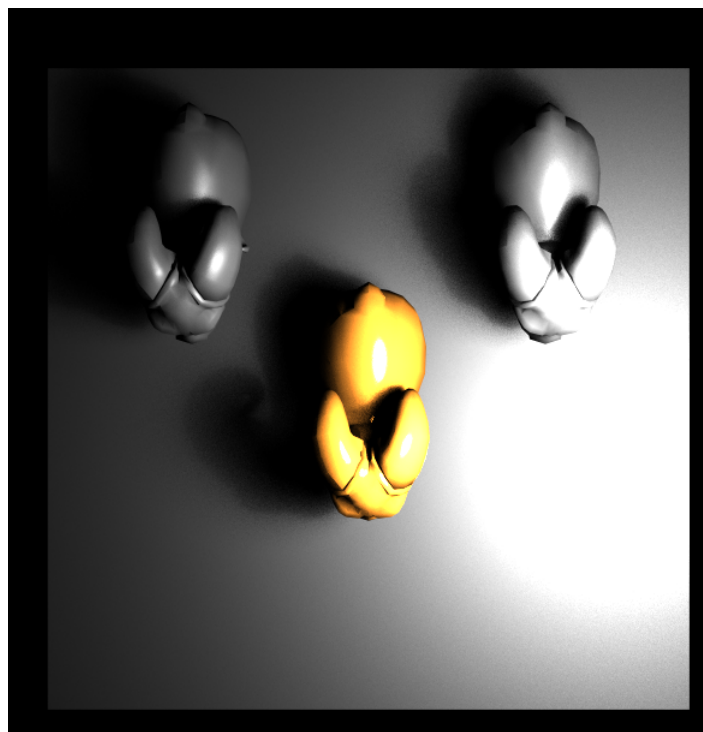


FIG. 4 – Source de lumière étendue

3.4 Ambient Occlusion

L'ambient occlusion (ici sur un modèle rendu sans BRDF de Phong afin de mettre en évidence l'apport de l'ambient occlusion) à été réalisé en considérant la demi sphère, d'un rayon de 5% de la taille de la scène 3D, le long de la normale avec un nombre N de rayons.



FIG. 5 – Ambient Occlusion

3.5 Anti-aliasing

Afin d'éviter le problème classique de crênelage, nous calculons chaque pixel en le découpant selon une grille de sous pixels, puis en moyennant ces derniers.



FIG. 6 – Anti-aliasing

3.6 **Miroir**

Bien que le projet ne le demandait pas explicitement, nous avons choisi d'ajouter des matériaux miroirs.

//TODO add Taurus in the box scene

FIG. 7 – Tore Miroir

3.7 **Profondeur de champ**

Nous avons également implémenté un effet flou de profondeur.

//TODO add RAM army

FIG. 8 – Armée de béliers

4 Éclairage Indirect

4.1 Pathtracing

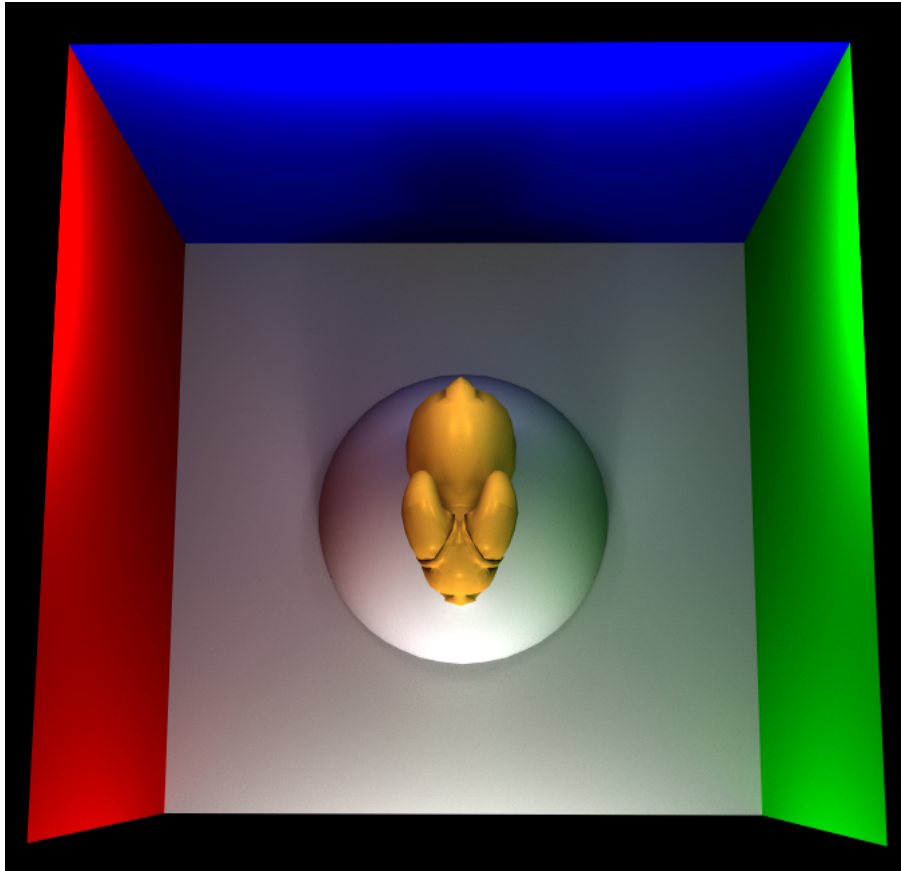


FIG. 9 – Éclairage Global

5 Limites

Ce paragraphe liste certains points qui pourraient être amélioré.