



ÉCOLE
CENTRALELYON

ÉCOLE CENTRALE LYON

RAPPORT

MOS Informatique graphique Rapport - Ray tracing

Élèves :
Ayoub TOUDGHI

Enseignant :
Nicolas BONNEEL

Table des matières

1	Introduction :	2
2	Construction de la scène :	2
2.1	Eclairage lambertien, ombres portées et Correction gamma :	2
2.2	Surfaces miroirs et surfaces diffuses :	2
2.3	Eclairage indirect avec source étendue et ombres douces :	3
2.4	Depth of field :	3
2.5	Maillages triangles avec Bounding Box :	4
2.6	Textures	5
2.7	Repo Github	5

1 Introduction :

Dans le cadre de notre cours d'informatique graphique, le présent rapport détaille les différentes étapes suivies pour parvenir au rendu final du travail sur le ray tracing. Ce rapport présentera en détail les différentes phases de développement, en mettant en avant les résultats obtenus à chaque étape. Des éléments d'implémentation clés seront également discutés pour une meilleure compréhension du processus de création.

Les résultats sont créées avec une résolution de 512x512. L'éclairage est fixé à 2 milliards de lux, cependant, les autres variables sont modifiées à chaque itération pour améliorer la visualisation. Le temps d'exécution est enregistré pour chaque résultat.

2 Construction de la scène :

2.1 Eclairage lambertien, ombres portées et Correction gamma :

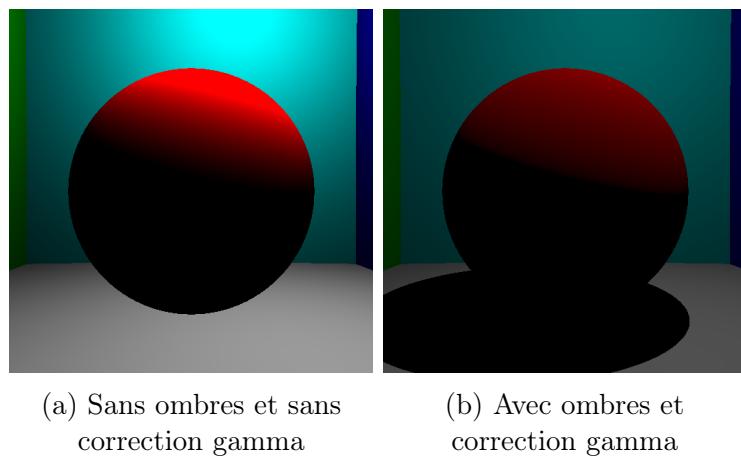


FIGURE 1 – Eclairage lambertien, ombres portées et Correction gamma

2.2 Surfaces miroirs et surfaces diffuses :

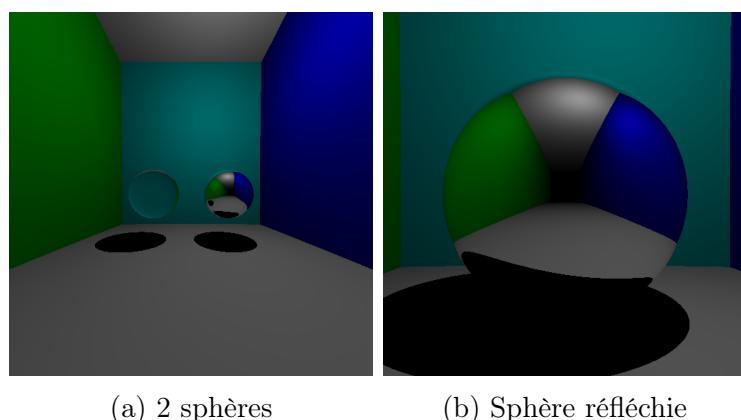


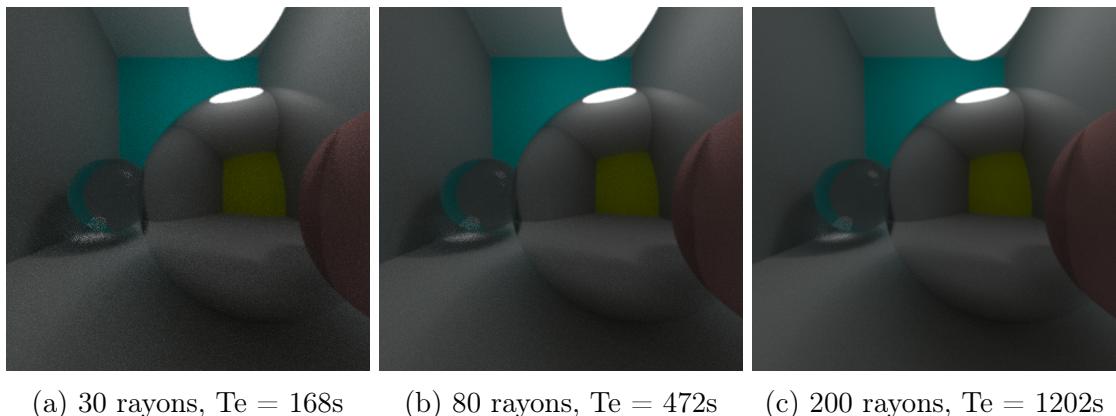
FIGURE 2 – Surfaces miroirs et surfaces diffuses

2.3 Eclairage indirect avec source étendue et ombres douces :

J'ai adopté une approche de rendu global pour calculer l'éclairage indirect de manière récursive avec la BRDF comme j'ai élargi ma méthode pour inclure une source de lumière étendue avec le calcul des ombres douces, améliorant ainsi la qualité visuelle des rendus grâce à des transitions graduelles entre les zones éclairées et ombragées.

De plus, j'ai ajouté une étape d'**anti-aliasing** pour réduire les artefacts visuels tels que l'aliasing et les crénellages dans mes rendus.

Enfin, les résultats sont présentés ci-joint, démontrant l'impact de l'utilisation de différents nombres de rayons envoyés pour éliminer le bruit et améliorer la qualité des rendus avec le temps d'exécution Te et avec $I=2E9$ et une source de lumière étendue de rayon $R = 15$.



(a) 30 rayons, $Te = 168s$ (b) 80 rayons, $Te = 472s$ (c) 200 rayons, $Te = 1202s$

FIGURE 3 – Eclairage indirect avec source étendue et ombres douces

2.4 Depth of field :

Pour simuler des effets réalistes tels que la profondeur de champ et le flou de mouvement dans mes rendus, j'ai opté pour l'utilisation d'un diaphragme pour contrôler la quantité de lumière entrant dans la caméra, offrant ainsi une plus grande flexibilité pour la création d'effets de flou réalistes.

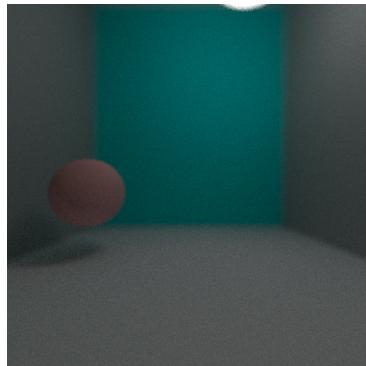


FIGURE 4 – Depth of field $nryas = 30$, $Te = 183s$

2.5 Maillages triangles avec Bounding Box :

La génération des bounding boxes pour une scène donnée a pris 734 secondes pour un nombre de rayon = 10 (fig 6a) et avec un rescale = 0.3 et translation. Afin d'améliorer l'efficacité de ce processus, j'ai opté pour une stratégie hiérarchique et continue. J'ai pu diminuer de temps de calcul à 20s avec nryas = 100 comme illustre ci joint dans fig 6b.

De plus, j'ai ajouté le lissage de Phong en moyennant la normale par les coefficients calculés lors de l'intersection, ce qui a permis d'améliorer la qualité visuelle des rendus (fig 6c-6d).

Les résultats sont inclus ci-joint. Il m'a fallu un certain temps pour les générer en fonction des caractéristiques de lumière utilisées et du nombre de rayons. Pour une meilleure visibilité, vous pouvez consulter les résultats sur le dépôt GitHub, dans le dossier "résultats".

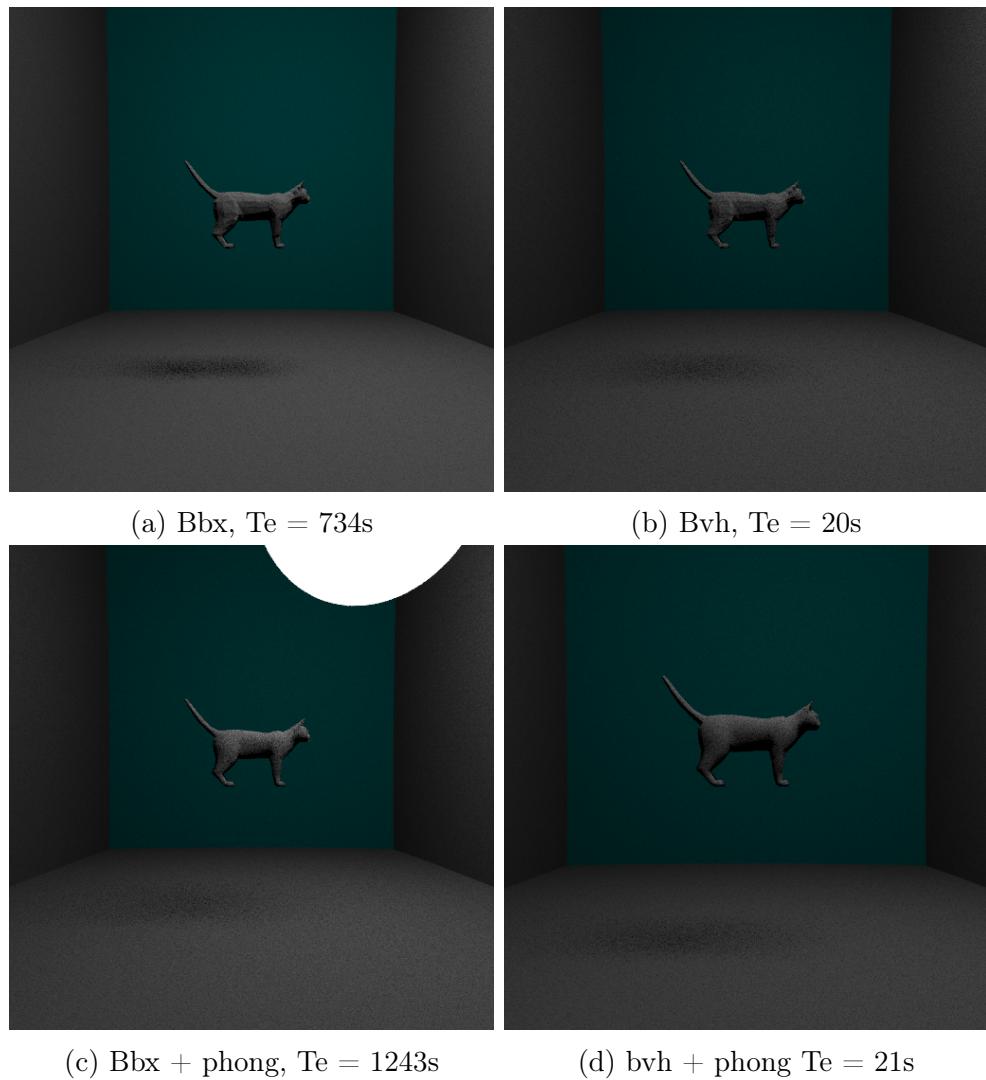
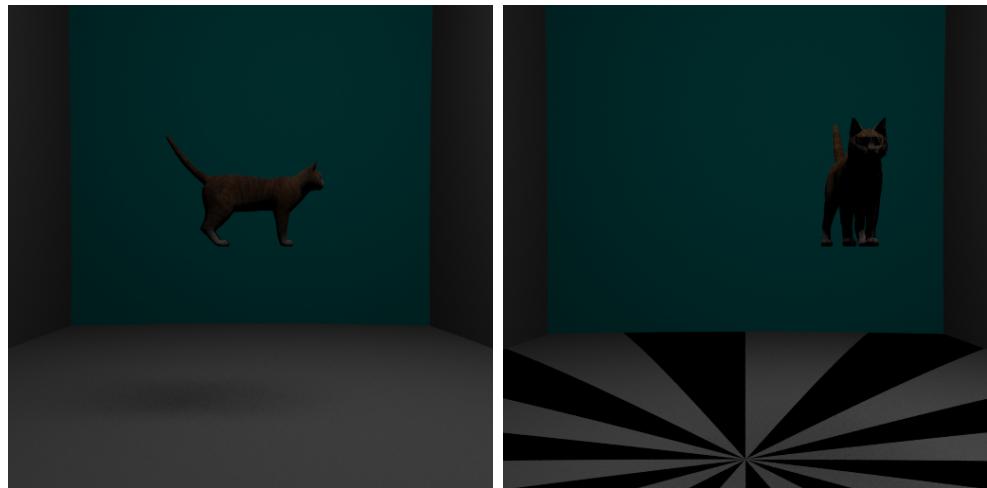


FIGURE 5 – Résultats du Maillage nrays = 10

2.6 Textures

J'ai intégré la lecture et l'affichage de la texture. J'ai également étendu les fonctionnalités en intégrant des méthodes pour mettre à l'échelle, translater et faire pivoter des objets dans l'espace. De plus, j'ai incorporé une fonctionnalité permettant d'appliquer un type spécifique de texture aux sphères de la scène.



(a) bvh et texture , Te = 295s

(b) bvh et texture , Te = 331s

FIGURE 6 – Resultats du Maillage avec lissage et texture nrays = 100

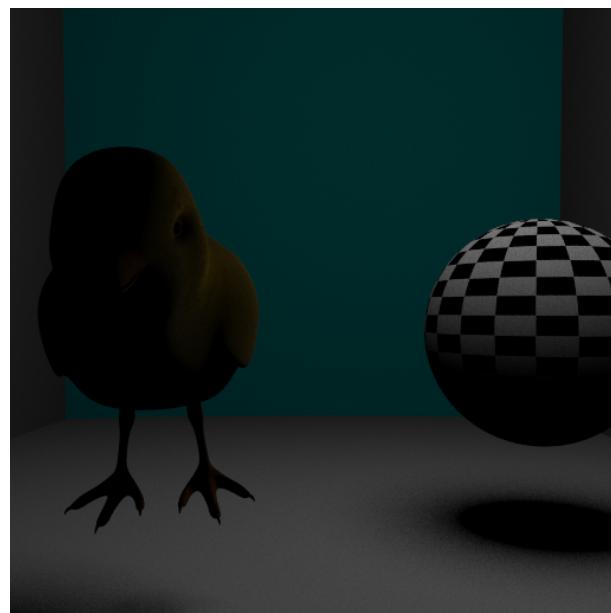


FIGURE 7 – scene nrays= 100, Te=832s

2.7 Repo Github

Dans le fichier main.cpp, vous trouverez tout le code utilisé, tandis que le dossier "Résultats" contient toutes les sorties pour chaque étape nécessaire à la création de la scène.