# Réalisation d'un lancer de rayons

Beqqi Hamza

# Table des matières

# 1 Introduction

Durant ce projet, on a cré é un lancer de rayon stochastique qui permet de synthé tiser des images à partir d'une description d'un monde 3D, en simulant le trajet des rayons lumineux qui finissent sur la lentille de la camè ra.

# 2 Lancer de rayon stochastique

Afin de produire une image d'une scène 3D, on est amené à résoudre la fameuse équation de rendu qui décrit l'interaction de la lumière avec la scène. Ce qui revient à évaluer plusieurs intégrales. Un lancer de rayons conventionnel estime ces intégrales en échantillonnant la valeur de l'intégrande (la fonction sous l'intégrale) en un seul point du domaine d'intégration. Ceci est clairement une mauvaise approximation. Le lancer de rayon stochastique échantillonne l'intégrande à de nombreux points choisis au hasard, et calcul la moyenne des résultats pour obtenir une meilleure approximation. Ceci lui permet de produire plusieurs effets : Anti-aliasing, réflexion floue, ombres douces, profondeur de champ, …

# 3 Description de la scène

## 3.1 Les objets

Les scènes supportées sont des scènes avec des objects composés de sphère ou des meshs chargée à partir d'un fichier “.obj”.

## 3.2 Les maté riaux

Le lancer de rayons supporte le shading de plusieurs types de surfaces, notament des surfaces lambertiennes, de Phong, transparentes, avec des ré flé xions floues et des mirroires. Il supporte aussi la té xturisation( qui permet de spé cifier la composante diffuse du maté riel).

## 3.3 Les rayons

Un rayon est représenté de façon paramétrique à l'aide de deux vecteurs, une origine et une direction. Un point suivant le rayon est défini par un nombre réel $t$ selon l'équation

$$\overrightarrow{r(t)} = \overrightarrow{origine} + t\overrightarrow{direction}$$

Les rayons sont générés à partir d' une caméra virtuelle qui définit la position et l' orientation de l' observateur. La génération se fait en traçant des rayons à partir de l' œil de l' observateur et des points échantillonnés du plan image.

Soit une caméra définie par sa position $\overrightarrow{eye}$ , et une base orthonormale $((\vec{u}, \vec{v}, \vec{w})$ qui définit son orientation. Pour chaque pixel de notre image, on le subdivise en $N^2$ cases, et on choisit un échantillion par case au hasard, comme on peut voir sur la figure. Soit $(x, y)$ les coordonnées d' un échantillon par rapport au centre du plan image. Alors la direction du rayon à généré est définie :

$$\overrightarrow{direction} = x\vec{u} + y\vec{v} - f\vec{w}$$

Où $f$ est la distance focale de la caméra.

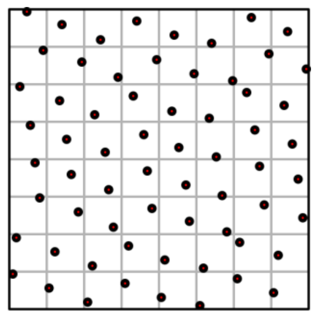

Figure 1: Echantillionnage stratifié

# 4 Shading

Après avoir identifier une intersection, il faut définir la radiance qui sort de ce point-là envers l'observateur. Ceci est fait à l'aide de l'équation de rendu, qui définit la radiance émite par un point $p$ suivant la direction $\overrightarrow{w_o}$ :

$$L(p, \overrightarrow{w_o}) = L_e(p, \overrightarrow{w_o}) + \int_{hémisphère} brdf(p, \overrightarrow{w_o}, \overrightarrow{w_i}) L_i(p, \overrightarrow{w_i}) \cos(\theta_i)\, d\overrightarrow{w_i}$$

$L_e$ : *radiance résultante du charactère emissive de la surface*

$Li$ : *radiance incidente*

$\overrightarrow{w_i}$ : *direction incidente*

$brdf$ : *bidirectional reflectance distribution function*

$\theta_i$ : *angle entre* $\overrightarrow{w_i}$ *et la normale de la surface*

## 4.1 Réflexion floue

La réflexion floue est supportée en prenant compte de $L_i$ sur des rayons qui appartiennent à un cône autour de la direction parfaite de réflexion du vecteur $\overrightarrow{w_o}$ par rapport à la normale.

Soit $\vec{r}$ un vecteur, et $\alpha$ un réel. On veut choisir aléatoirement une direction dans le cône défini par $(\vec{r}, \alpha)$. Pour faire cela, il suffit de choisir aléatoirement un vecteur $\vec{d}$ sur le disque perpendiculaire à $\vec{r}$ et de rayon $\beta\ tel\ que\ \beta \leq \|\vec{r}\| sin\alpha$ . Soit $(\vec{u}, \vec{v})$ deux vecteur non colinéaires qui appartiennent au plan dont $\vec{r}$ est la normale. Soit $\theta$ un nombre choisi au hasard. On a :

$$\vec{d} = \beta(cos\theta\vec{u} + sin\theta\vec{v})$$

Ainsi notre direction qui appartient au cône est :

$$\vec{D} = \vec{d} + \vec{r}$$

Il suffit maintenant d'évaluer $L_i$ sur plusieurs directions et d'en prendre la somme. Il faut noter aussi, qu'il faut adapter la fonction de traçage pour qu'elle soit récursive afin de supporter plusieurs miroirs sur la scène par exemple.

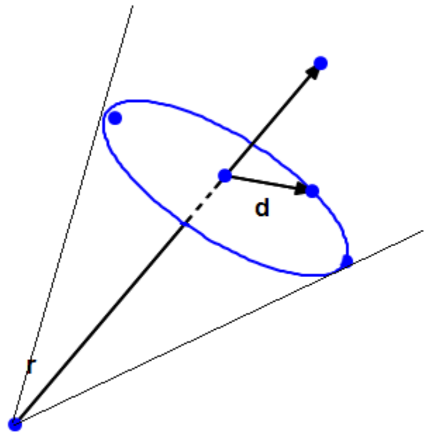


*Figure 2: Vecteur dans un cône*

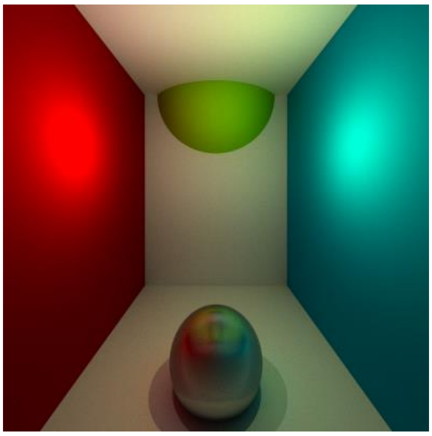

*Figure 3: rendu avec matériel réflexion floue*

## 4.2 Note sur l'intégration Monte-Carlo

Le lancer de rayon stochastique n'est autre qu'un autre nom à l'utilisation de l'intégration monte-carlo pour évaluer les différentes intégrales impliquées dans le rendu d'une scène. Cette technique est très efficace pour évaluer des intégrales dans un domaine de grande dimension, parce qu'elle ne nécessite que la capacité de pouvoir échantillonner ce domaine. Vu ce constat, on peut essayer de résoudre les intégrales de rendu toutes à la fois en ne lançant qu'un rayon aléatoire chaque fois qu'on veut calculer une intégrale (cela revient à échantillonner tout le domaine). Cela veut dire moins de rayons par pixel et un meilleur échantillionnage du domaine.