

Lyon Ray Tracer

Maxime GAUDIN

11 novembre 2011

1 Auteur

Je suis Maxime GAUDIN, élève ingénieur à l'INSA de Lyon et pour le moment en échange avec l'École Polytechnique de Montréal (attachée à l'Université de Montréal).

J'étudie en génie informatique et logiciel et le pire c'est que j'aime ça !

2 Description

2.1 Qu'est ce que c'est ?

Lyon Ray Tracer, en hommage à ma ville natale : Lyon, est un moteur de rendu utilisant la technique du *ray tracing* pour produire des images photoréalistes. Pour ceux qui ne connaissent pas encore le *ray tracing*, je vous invite à consulter notre ami *Wikipedia* (http://fr.wikipedia.org/wiki/Lancer_de_rayon).

Par conséquent, mon moteur prend en entrée des informations comme "il existe une sphère de centre (0,0,0) et de rayon 10 qui possède un coefficient de réflexion de 0.5", et produit ce genre d'image :

2.2 Pourquoi ?

Bien évidemment parce que j'aime ça ! Mon école m'a proposé de réaliser le projet de mon choix et c'est ce que j'ai choisi. De plus, je trouve que le concept est formidable : Créer à partir de quelques informations textuelles des images ultra réalistes et respectant les principes physiques de propagation de la lumière.

2.3 Comment ?

En C++ et en utilisant le plus d'outils libres et standards. Pour vous donner une idée de mon toolkit, je vous conseil d'aller jeter un coup d'œil du côté des dépendances.

2.4 C'est cool, je peux participer ? ?

Oh oui !

Un moteur de *ray tracing* c'est comme des Légo, on peu toujours ajouter quelques choses (surtout en ce moment, puisque le moteur n'est pas encore très évolué). C'est donc pour toi l'occasion de participer à quelques choses au tout début de son évolution.

Tu peux par exemple :

- Me laisser un petit mail pour m'encourager !
- Essayer de compiler le programme et m'avertir des différents problèmes que tu as pu rencontrer (si possible avec la solution ;))
- M'envoyer des *pull requests* (même si c'est pour un coquille ou ajouter un peu de documentation).
- M'aider à créer une vraie page web au lieux de ce PDF :p

2.5 Fonctionnalités

Lumières

- Lumière directionnelle.

Caméra

- Caméra classique avec prise en compte de la perspective.

Géométrie

- Sphère.
- Plan infini.

Matériaux

- Couleur diffuse, spéculaire, ambiante.
- Indice de réfraction.
- Indice de réflexion.

Importeurs

- 3DS.

Exporteurs

- PNG.
- JPG.

3 Dépendences

La compilation du projet nécessite :

1. Un compilateur C++ récent.

2. make
3. libjpeg et/ou libpng
4. lib3ds

Autant dire que le projet est fait pour tourner sur le plus grande nombre de plateformes.

4 Licences

À priori, LGPL.