UTLN/UFRST/Imath/Nguyen

Infographie 3D - Projet

Projet Raytracing

Objectifs

A partir des éléments de raytracing vu en cours et TD, réaliser une application illustrant les notions fondamentales du lancer de rayons récursif. La programmation devra utiliser le langage Python, avec ses modules standards et les modules PIL et Numpy.

Principes du raytracing

Le lancer de rayons modélise l'interaction de la lumière et des objets en usant des principes simples de l'optique : toute source de lumière crée des rayons, ceux-ci percutent des objets, se réflechissent dessus, sont absorbés par eux ou les traversent. Certains de ces rayons finissent par être captés par nos yeux, ce qui permet au cerveau de composer l'image 3D que nous observons. Du point de vue de l'optique, le lancer de rayons suppose que les facettes sont des réflecteurs spéculaires parfaits. Le trajet d'un rayon de lumière se fait en ligne droite à travers des matériaux homogènes. La lumière n'interagit qu'avec la surface des objets. Les propriété de la lumière telles que diffraction, phase, polarisation, longueur d'onde et atténuation avec la distance ne sont pas pris en compte dans le modèle initial.

Le lancer de rayons appartient à la famille des algorithmes qui travaillent dans l'espace discrétisé (*point-sampling algorithm*), qui déterminent les caractéristiques d'une surface à partir d'un nombre fini de points (et interpolant les autres). Cela confère l'approximation de la réalité inhérente au lancer de rayons. Les techniques de rendu d'une scène se ramènent à transformer la description d'une scène 3D en une matrice 2D d'intensités (à afficher sur un écran). Une scène 3D regroupe un ensemble de paramètres qui sont :

- les objets : primitives, formes géométriques, formes plus complexes (formes composées de centaines de triangles),
- les textures (matériaux) : propriétés liées à la couleur (ou à l'association de couleurs ou pattern), transparence, reflexion, rugosité (aspect grossier ou poli), brillance, terne,

- la caméra : le modèle n'est pas celui qui correspond à la réalité mais plutôt celui de la projection perspective. La fenêtre de vue (view plane) est ensuite plaquée sur l'écran.
- les sources lumineuses, dont les propriétés sont les couleurs, l'intensité, la localisation, la taille et la forme. Le phénomène physique étant trop complexe, on a recourt à des simplifications. La source lumineuse est généralement ponctuelle (ramenée à un point) et d'une seule couleur.

Modélisation

La conception orientée objets se prête naturellement à ce type de problème. En dehors des objets mathématiques (comme les vecteurs), tous les objets associés au lancer de rayons trouvent un écho dans le monde réel : couleurs, solides, lumières et caméra. Les descriptions de classes qui suivent ne sont que des suggestions (en particulier, elles supposent la présence des constructeurs, destructeur, accesseurs et mutateurs idoines) :

La classe vecteur :

attributs : origine, extrémité

méthodes : addition, soustraction, multiplication par un scalaire, produit scalaire, produit vectoriel, normalisation, norme

La classe couleur :

attributs : composantes rouge, verte et bleue

méthodes : addition, multiplication (équivalent au filtrage d'une couleur par une autre)

La classe objet :

attributs : position, couleur, facteur de réflexion diffus, facteur de réflexion spéculaire, facteur de réflexion (reflets), ombre (booléen)

méthodes : intersection (retourne le point d'intersection avec une droite le plus proche de la caméra), normale (retourne la normale d'un point à la surface de l'objet)

La classe sphère (sous classe de objet) :

attributs : rayon

méthodes : implante les méthodes virtuelles de la classe objet

La classe plan (sous classe de objet) :

attributs : normale

méthodes : implante les méthodes virtuelles de la classe objet

La classe lumière :

attributs : position, couleur

méthodes : (rien de particulier)

La classe caméra :

attributs : dimensions (largeur, hauteur), position, direction (*look at*), orientation (*up vector*), distance focale

méthodes : rayon (retourne le rayon fonction de la position de son origine sur le plan de projection)

La classe scène :

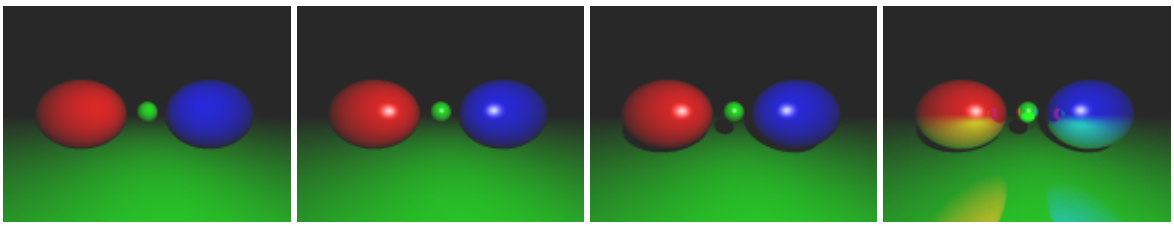
attributs : référence à la caméra, liste des objets, liste des lumières, lumière ambiante, fichier image

méthodes : ajouter la caméra, ajouter un objet, ajouter une source lumineuse, calcul de la plus proche intersection par rapport à un rayon donné, construction de l'image finale (l'algorithme principal)

Implantation

Vu le peu de temps dont on dispose, des priorités doivent être établies qui impliquent une programmation modulaire et incrémentale :

1. conception (ou appropriation) des classes générales indispensables à la réalisation du projet,
2. conception des classes "spécifiques" au projet (couleur, lumière, caméra, scène),
3. implantation du lancer de rayons *non récursif* sans modèle d'éclairement (uniquement l'illumination directe),
4. implantation de modèle d'éclairement de Phong (prise en compte des réflexions diffuses et spéculaires),
5. implantation des ombres portées,
6. implantation des surfaces réfléchissantes par le lancer de rayons récursif,
7. implantation d'objets translucides,
8. implantation du texture mapping.



Evaluation : la qualité de la programmation (modularité, lisibilité, robustesse) sera évaluée sur 3 points, chaque rubrique ci-dessus sera évaluée sur 2 points ; des scènes et des images (dont l'esthétisme sera évaluée sur 1 point) devront être fournies avec le code source.