

TD d'infographie n°6

Cours d'infographie

Master 1 : Ingénierie informatique

Modélisation radiométrique

Plaquage de textures, lecture de fichier image au format png.

Le but de ce travail consiste à plaquer une texture sur un objet existant modélisé sous OpenGL.

Téléchargez le fichier **temple.zip** disponible sur le site <http://www-igm.univ-mlv.fr/~piranda>. Ce fichier regroupe un programme proposant une modélisation géométrique d'une scène et des images au format png qui seront plaquées sur le sol et les éléments du temple (cf. image ci-dessous).

Exercice 1 Plaquage de texture sur une scène existante

1. Ajoutez au modèle géométrique fourni, les coordonnées de texture permettant de déformer l'image lors de son plaquage sur les objets : **glSol**, **glBloc** et **glColonne**.

Choisissez les coordonnées de textures de façon à tenir compte des variations de la taille de l'objet modélisé. Les coordonnées de texture doivent être définies de façon à ce que la taille d'une image plaquée soit de 1 x 1 (unité de modélisation) pour les éléments du temple et représente une zone de 4x4 pour le sol.

a) *complément technique : coordonnées de texture*

Dans le cas d'un objet décrit à l'aide de primitives, les coordonnées de textures sont ajoutées grâce à la fonction **glTexCoord**. Elle donne les coordonnées de textures pour les sommets définis ensuite.

```
glBegin(GL_QUADS) ;
glNormal3f(0.,0.,1.); // définition de la normale au sommet
glTexCoord2f(0.2,0.1); // définition des coordonnées de texture au sommet
glVertex3f(-1.,-1.,0.); // un sommet
: : : : : :
glEnd();
```

Dans le cas de déclaration d'un vertex-array, les coordonnées de texture sont placées dans la liste des informations associées à chacun des sommets (les deux premières valeurs dans le mode **T2FN3FV3F** par exemple).

2. Etudiez le fichier **png.cpp** proposant un code permettant de charger une image en mémoire au format png.

Complétez ce code de façon à définir la fonction **GLuint loadPNGTexture (const char *filename)** qui retourne un identifiant de texture OpenGL valide à partir d'un nom de fichier de texture png passé en paramètre.

Charger les images **pierre.png**, **marbre_noir.png**, **sol_herbe.png** et **mur_pierres.png** fournies dans **temple.zip**.

Pour l'édition de liens de ce fichier, il faut penser à ajouter la bibliothèque png : **-lpng**.

3. Plaquer les textures sur chacun des éléments.

Pour plaquer une texture sur un support géométrique sous OpenGL, il faut activer le mode texture 2D puis sélectionner la texture à projeter par les commandes :

```
glEnable(GL_TEXTURE_2D);
glBindTexture(GL_TEXTURE_2D,id_texture);
```

Modifiez la fonction de dessin du fichier **maingl.cpp** afin de sélectionner la bonne texture pour chacun des éléments de la scène.

a) *complément technique : texture OpenGL*

Le code suivant montre comment créer une texture OpenGL (associée à l'identifiant **id**) d'une texture à partir d'un tableau **image** contenant des couleurs des pixels codées sur 3 octets.

```
// demande d'un identificateur de texture libre
glGenTextures(1,&id);
// choix de la texture courante
glBindTexture (GL_TEXTURE_2D,id);
// organisation des plans dans le tableau
glPixelStorei (GL_UNPACK_ALIGNMENT, 1);
// paramètre de répétition de la texture
glTexParameteri (GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_REPEAT);
glTexParameteri (GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, GL_REPEAT);
glTexParameteri (GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_LINEAR);
glTexParameteri (GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_LINEAR);
// copie de l'image dans la texture courante
glTexImage2D(GL_TEXTURE_2D,0,GL_RGB,dx,dy,0,GL_RGB,GL_UNSIGNED_BYTE, image);
// la texture sera mélangée à la couleur du modèle d'illumination
glTexEnvf (GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, GL_MODULATE);
```



Les paramètres transmis à la fonction `glTexParameter[ifv]` permettent de préciser le mode de répétition de la texture sur la surface (paramètres `GL_TEXTURE_WRAP_S` et `GL_TEXTURE_WRAP_T` avec les valeurs `GL_CLAMP` ou `GL_REPEAT`).

Le mode de combinaison de la texture avec la couleur issue du modèle d'illumination est choisi par la fonction `glTexEnv[fv]` (paramètres `GL_TEXTURE_ENV`, `GL_TEXTURE_ENV_MODE` avec les valeurs `GL_MODULATE` pour mélanger, `GL_DECAL` pour imposer la couleur de la texture).

b) *complément technique : texture multi-échelle OpenGL/GLU*

Le mipmapping ou plaquage de texture multi-échelle permet de gérer la qualité de la texture en fonction de l'éloignement de la facette par rapport à la caméra. Les modifications suivantes du code précédent permettent d'utiliser ce mode de texture sous OpenGL.

```
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_LINEAR);
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_LINEAR_MIPMAP_LINEAR);
// copie de l'image dans la texture courante et calcul toutes les images de tailles
// inférieures pour le mipmapping.
gluBuild2DMipmaps(GL_TEXTURE_2D, GL_RGB, lx, ly, GL_RGBA, GL_UNSIGNED_BYTE, image);
```

4. Complétez la modélisation de la scène pour tenir compte des matériaux et de l'éclairage.

Ajouter des paramètres de matériaux la définition des éléments de la scène de façon obtenir un texturage qui dépend de l'éclairage de la scène par une source de lumière blanche.

Faites tourner la source de lumière pour observer les variations de l'éclairage.