

TP 6 : Texture avec OpenGL & SDL

Une texture est une image Bitmap qu'on applique sur un objet opengl. Cette image peut être générée par un programme ou bien stockée dans un fichier (au format jpeg,png,gif, ect..)

Fonctions OpenGL importantes lié à la texture

- void glTexImage2D(GLenum target, GLint level, GLint internalFormat, GLsizei width, GLsizei height, GLint border, GLenum format, GLenum type, const GLvoid * data);

Définit une texture 2D

- target : cible de la texture. Souvent définis à GL_TEXTURE_2D
- level : Niveau de détail . Mettre à 0
- internalFormat : Nombre de composante de couleur dans la texture. 3 pour des couleurs sur 24 bits.
- width et height : longueur et largeur de la texture. En octet. Doit être une puissance de 2. Ex : 256,256
- border : 1 si y a une bordure
- format : type de données des pixels. Ex GL_RGB
- data : pointeur vers les données de l'image contenu en mémoire.

- void glTexCoord2f(GLfloat s, GLfloat t);

A appeler avec chaque définition de sommet. La position (s,t) de la texture sera associée avec la position du sommet défini par la suite. s et t est compris entre 0 et 1.

Ex : `glTexCoord2f(0.5,0.5);glVertex3f(0.2,0.3,1.0);`

le milieu de la texture (position 0.5,0.5) sera au niveau du sommet à la position (0.2,0.3,1.0)

Exercice

1. L'exemple `tex_une_face.c` permet d'appliquer une texture sur une face du cube. La texture est définie dans un tableau d'octets.
 - a) Tester l'exemple.
 - b) Modifier les coordonnées de la texture (défini avec `glTexCoord2f`), en remplaçant 1 par 5. Observer. Faire de même en remplaçant par 0.5.
 - c) Revenir avec le fichier `tex_une_face.c` initial. Modifier le programme pour appliquer la texture sur les 6 faces du cube.
2. La fonction SDL `IMG_Load(char* file);` permet de charger une texture à partir d'un fichier. Elle renvoie un pointeur vers une structure `SDL_Surface`. Appliquer la texture contenue dans le fichier `tux.jpg` sur chaque face du cube. Pour cela appeler la fonction `IMG_Load` et modifier les paramètres de `glTexImage2D` en utilisant les paramètres contenus dans la structure `SDL_Surface`.

Remarque : `IMG_Load` fait parti dans la librairie `SDL_image`. La librairie n'est pas installée

sur le système la télécharger à partir du site : [sdlimage.tgz](http://www.libsdl.org/release/SDL_image.tgz) . Décompresser dans le dossier où se trouve le programme. On doit donc ajouter `libSDL_image.a` à la commande pour compiler , et ajouter au source `#include "SDL_image.h"`

Rappel : La structure `SDL_Surface` contient :

- `format->BytesPerPixel` : nombre d'octet par pixel de l'image
- `w` et `h` : largeur et longueur de l'image
- `pixels` : pointeur vers les données de l'image

3. Jeu de dé. On souhaite modéliser un dé à jouer.

- a) Plaquer la texture du dé (fichier *de.jpg*) sur le cube. Chaque face du dé sur l'image à une dimension de 64x64 pixels.
- b) Animer le dé pour qu'il tourne dans différentes directions quand l'utilisateur clique sur la fenêtre.
- c) Après que l'utilisateur a cliqué et que l'animation à duré quelques secondes, le dé s'arrêtera sur la face correspondante a un nombre tiré aléatoirement (avec les fonctions `srand` et `rand`).