

OpenGL - T.P. 1

Franck Hétroy

23 novembre 2005

1 Introduction à *libQGLViewer*

libQGLViewer est une bibliothèque C++ (sous licence GPL) permettant de créer et d'utiliser facilement et rapidement des fenêtres d'affichage OpenGL. Vous n'êtes pas obligés de l'utiliser, mais elle vous simplifie grandement la vie lorsque vous programmez avec OpenGL et que vous voulez visualiser vos résultats. *libQGLViewer* est installée sur les machines de l'ARV, dans le répertoire D:\KITARV\libQGLViexer-1.3.7.



Instructions :

1. Commencez par lire l'introduction à *libQGLViewer* :
<http://artis.imag.fr/Membres/Gilles.Debunne/QGLViewer/intro.html>
2. Lisez également le code de base commenté, pour comprendre comment utiliser *libQGLViewer* avec OpenGL :
<http://artis.imag.fr/Membres/Gilles.Debunne/QGLViewer/commented.html>
Si vous n'avez jamais fait de C++ et que vous avez du mal à comprendre, faites-moi signe.
3. Installez et compilez *libQGLViewer* sur votre compte :
 - (a) copiez le répertoire D:\KITARV\ libQGLViexer-1.3.7 sur votre compte ;
 - (b) démarrez Visual C++ ;
 - (c) cliquez sur l'icône **Open Qt Project** et ouvrez le fichier
Z:\libQGLViexer-1.3.7\QGLViewer\QGLViewer.pro ;
 - (d) Faites **File** → **Open Workspace** et ouvrez le fichier QGLViewer.dsp ;
 - (e) Faites **Build** → **Build QGLViewer.dll**.
4. Lisez quelques-uns des exemples présentés ici :
<http://artis.imag.fr/Membres/Gilles.Debunne/QGLViewer/examples/index.html>
Compilez-les :
 - (a) **Open Qt Project** <nom>.pro ;
 - (b) **Open Workspace** <nom>.dsp ;
 - (c) **Build** <nom>.exe.

Lancez-les ; n'hésitez pas à les modifier pour tester certaines fonctionnalités d'OpenGL !

2 Visualisation d'un diagramme de Voronoï avec OpenGL

Pendant le cours de géométrie algorithmique, on a vu un algorithme de balayage (du à Steven Fortune) permettant la construction d'un diagramme de Voronoï 2D. Cet algorithme repose sur l'observation qu'*un diagramme de Voronoï 2D correspond à la projection sur un plan de l'intersection de cônes ayant pour origines les germes de Voronoï* (cf. figure 1). Le but de ce T.P. n'est pas d'implémenter cet algorithme, mais plus simplement de visualiser un diagramme de Voronoï 2D en suivant cette idée.

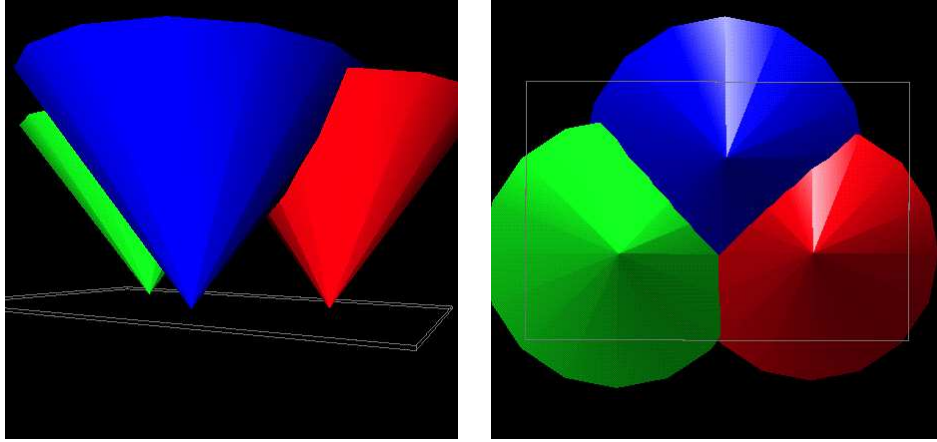


Figure 1: Diagramme de Voronoï 2D vu comme intersection de cônes.

Instructions :

1. Créez, grâce à *libQGLViewer*, une fenêtre permettant l'affichage d'un ensemble de points aléatoirement répartis dans le carré de centre l'origine, de rayon 1 et orthogonal à l'axe z . Choisissez la taille des points pour qu'ils soient visibles à l'écran !
2. Ajoutez la visualisation de cônes d'origines ces points, d'axe z et d'angle 45° .
3. Utilisez le Z-buffer pour afficher le diagramme de Voronoï des points (pensez à placer de manière optimale la caméra).
4. **Pour aller plus loin :**
 - vous pouvez essayer de laisser à l'utilisateur le choix du nombre de points, voire leur position (mais il faut contraindre celles-ci à être dans le carré) ;
 - vous pouvez aussi ajouter la possibilité de sélectionner un point, et d'afficher entièrement le cône dont il est le sommet ;
 - vous pouvez créer une interface multi-vue, avec par exemple une vue de face du carré et une vue en perspective pour visualiser les cônes "au-delà" du diagramme de Voronoï ;
 - les personnes très à l'aise avec OpenGL peuvent essayer de créer une animation balayant le carré par une droite $x = k$ et affichant le diagramme de Voronoï à gauche de cette droite (animation illustrant à peu près l'algorithme de S. Fortune).

