

Projet de LMG

Truong Tan Dat - ISI

Vendredi 12 Janvier 2018

1 Introduction

Le projet demandé était assez ouvert, j'ai décidé de créer des classes réutilisables pour mon projet, regroupant toutes les opérations à effectuer pour des groupes d'objets donnés. J'ai décidé de partir d'une scène basique et d'afficher un objet.

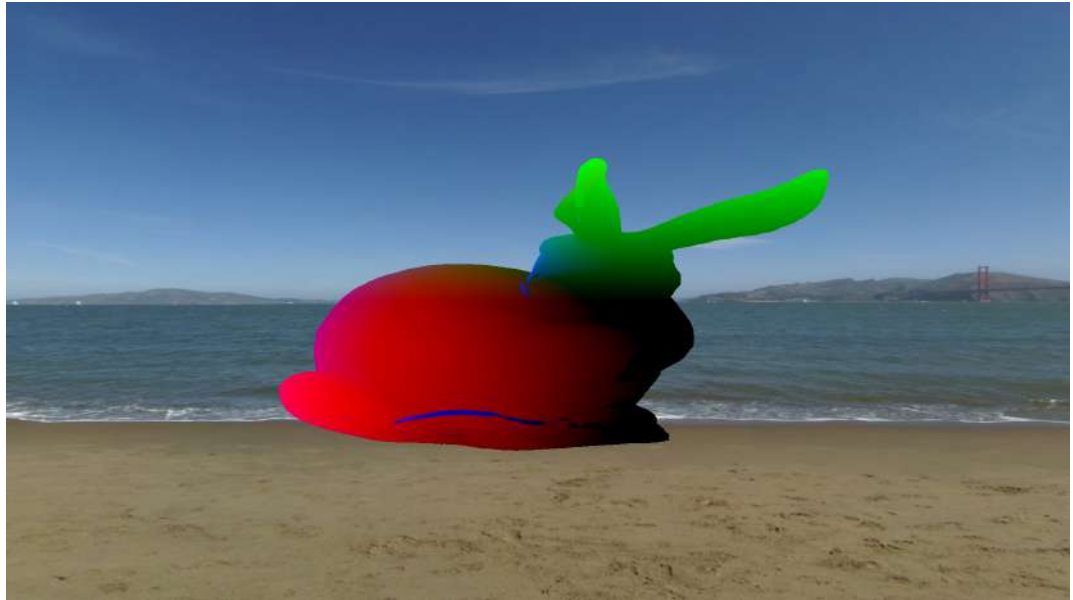
2 Fonctionnalités du projet

- Chargement d'un objet ou d'une texture:

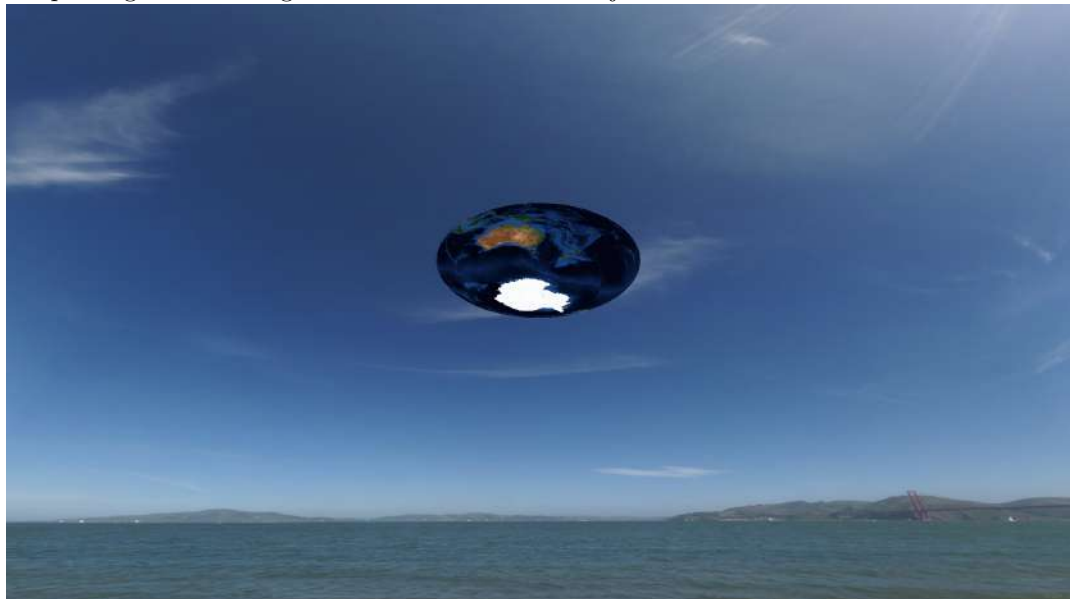


On peut choisir la l'objet, la texture, ainsi que les shader à utiliser.





On peut également bouger la caméra autour de l'objet:



3 Environnement utilisé et dépendances

Le projet a été réalisé sous ubuntu avec des librairies qui peuvent être trouvées dans le fichier install.h, le processus d'installation des dépendances a été simplifié pour l'utilisateur en utilisant simplement une commande qui installera toutes

les dépendances: "make lib-deps"

4 Compilation et lancement du programme

Pour compiler:

"make"

On peut utiliser plusieurs options indépendantes lors de la commande make:

LD=1 pour utiliser le loader fait à la main, la texture est buguée.

NORM=0 pour désactiver la normalisation des sommets de l'objet, il ne sera donc pas centré et à une taille convenable pour la caméra.

DEBUG=0 pour désactiver les logs dans le dossier debug

Exemple de compilation:

"make LD=1 NORM=0"

Pour lancer le programme:

"./main" suivi des fichiers correspondants, on supposera que les fichiers sont au format correspondant: .obj pour les objets, .vert pour les vertex shaders, .frag pour les fragment shaders, .jpg pour les textures. Si aucun fichier n'est spécifié pour n'importe laquelle de ces entrées, une entrée par défaut à été définie, on peut donc lancer "./main" sans aucun argument ou avec un nombre variable d'arguments. Si plusieurs fichiers au format correspondant sont transmis, on utilisera le dernier donné. On peut bouger la caméra avec ZQSD et zqsd. On peut désactiver/activer la rotation de l'objet avec la touche p.