



Université
Gustave
Eiffel



INSTITUT
D'ÉLECTRONIQUE
ET D'INFORMATIQUE
GASPARD-MONGE

OpenGL Avancée

Rapport de projet

Réalisés par :

BEN HAMOUDA Amel
DURAND Aurélien

Université Gustave Eiffel
Master 2 Science de l'image
2019/2020

Table des matières

I. Description du projet.....	3
II. Mode d'emploi.....	3
1. Many lights.....	3
2. <i>Normal mapping</i>	4
III. Difficultés rencontrées.....	4
1. <i>Many lights</i>	4
2. <i>Normal mapping</i>	4
IV. Connaissances acquises.....	4

I. Description du projet

Le principe de ce projet est d'améliorer le gltf viewer fait en TP.

Pour cela on a choisi d'ajouter les améliorations suivantes :

→ Many lights : cette amélioration consiste à ajouter de la lumière sur l'objet dessiné que ce soit de la lumière directionnelle ou de la lumière type spot.

→ Normal mapping : cette amélioration consiste à utiliser une texture de normale pour rajouter artificiellement des détails aux objets.

II. Mode d'emploi

Pour l'ensemble des améliorations choisies, on a 2 manières de contrôle sur la caméra :

Manières	Trackball	First Person
Utilisations	<ul style="list-style-type: none">→ Rotation objet : clique molette→ Distance de l'objet : clique molette + Maj→ Position de l'objet : clique molette + Ctrl	<ul style="list-style-type: none">→ Rotation caméra : clique gauche→ Déplacement : z, q, s et d→ Rotation axial : a et e→ Bonus de vitesse : Ctrl + déplacement

1. Many lights

Pour ce sujet on a implémenté trois types de lumières :

→ "Lumière Directionnelle" ("Directional Light") quand une source de lumière est modélisée pour être infiniment éloignée, puisque tous ses rayons lumineux ont la même direction.

→ "Lumière ponctuelle" ("Point lights") est une source de lumière avec une position donnée quelque part dans la scène qui s'éclaire dans toutes les directions, où les rayons lumineux s'estompent à distance. Dans notre cas représentée par 4 cubes de différentes couleurs.

→ "Projecteur" ("Spotlight") est une source de lumière située quelque part dans la scène qui, au lieu de tirer des rayons lumineux dans toutes les directions, ne les tire que dans une direction spécifique. Le résultat est que seuls les objets dans un certain rayon de la direction du projecteur sont éclairés et tout le reste reste sombre.

Pour chaque type de lumières on peut contrôler les paramètres via la GUI.

Paramètres contrôlables via la GUI pour chaque types de lumières :

Types	Directional Light	Point lights	Spotlight
Paramètres	<ul style="list-style-type: none">→ Thêta→ Phi→ Couleur→ Intensité	<ul style="list-style-type: none">→ Choix du cube→ Couleur→ Intensité→ Position (x,y,z)	<ul style="list-style-type: none">→ Couleur→ Intensité→ Dist CuteOff→ Dist OuterCuteOff→ Both CuteOff & Outer→ Dist attenuation spotlight→ Spotlight du curseur/centré

Pour l'ensemble des types de lumière il y a (à la fin de la section "Light") un bouton "Off" qui permet d'éteindre toutes les lumières, puis un bouton "On" qui permet de les rallumer.

Quand on rallume les lumières il faut faire varier l'intensité de chaque types de lumières via la GUI car par défaut au rallumage l'intensité de chaque lumière est nulle (équivalent à 0).

Pour avoir un effet "Smooth" sur les bords on vient créer deux cercles, un cercle intérieur et un extérieur. Cela permet une atténuation entre CuteOff et OuterCuteOff.

2. Normal mapping

Le but de ce sujet est de réaliser du normal mapping à partir d'une texture de normal si elle est présente dans un fichier gltf. Cela permet de faire mieux ressortir certain détails d'un objet 3D et permet aussi si nécessaire de baisser le nombre de sommets utilisés pour représenter un objet 3D tout en améliorant la qualité.

En ce qui concerne l'interface GUI pour le normal mapping on a simplement ajouté un bouton permettant d'afficher ou non le normal mapping quand celui-ci est possible.

III. Difficultés rencontrées

1. Many lights

La difficulté "principale" rencontrée dans ce sujet fut le bon choix des vecteurs de positions des différentes lumières afin de bien projeter la lumière sur l'objet 3D.

Sinon pour le reste le tutoriel était très clair.

2. Normal mapping

Pour ce sujet, le tutoriel explique très bien comment calculer les tangentes et bitangentes pour effectuer un passage dans l'espace des tangentes.

Le plus compliqué fut de récupérer et de comprendre les données des fichiers gltf.

On a mis en place deux versions pour les tangentes. La plus simple étant la suivante, si l'attribut tangentes est présent dans un fichier gltf. Et la plus compliqué étant donc de calculer les tangentes à partir des positions des sommets et de leurs coordonnées de texture. En s'inspirant de la méthode computeSceneBounds de "glTF.cpp" le travail fut bien plus simple.

IV. Connaissances acquises

Pour la partie connaissances acquises lors de l'implémentation des features, on peut citer plusieurs choses tels que :

- l'apprentissage, la manipulation et la mise en place des différents éclairages.
- une meilleure compréhension des shaders, avec l'utilisation des tableaux de ligh et la mise en place de fonction.
- l'implémentation de l'interface GUI qui était nouvelle, puisqu'on a ajouté les paramètres réglables via la GUI.
- Pour le normal mapping cela nous a permis de découvrir ce procédé et l'espace des tangentes. De plus cela nous a permis d'avoir une meilleure compréhension entre vbo et vao ainsi que comment manipuler les données de ces deux objets.