



Moteur 3D

Rapport de Projet

Yoann SOCHAJ

Janvier 2022

1 Introduction

Le but de ce projet était de comprendre le principe d'open Gl. Pour comprendre le principe il fallut aller le plus loin dans les TPs proposés. Je tiens à préciser que mon code est commenté donc il n'y aura pas d'explications PRÉCISES sur le code.

2 TP1

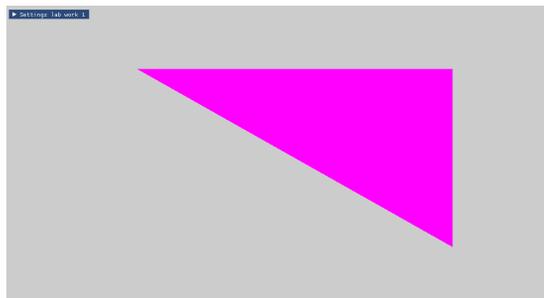


Figure 1: Résultat TP1

Le but de ce TP était de se familiariser avec l'Open gl. En effet il suffisait de pouvoir comprendre commencer marcher un vbo et un Vao, ainsi que la liaison avec le vertex et fragment shader. Une fois que l'on avait compris, en suivant

les explications, il fut simple d'afficher un triangle. Étant donné que j'étais en avance j'ai aussi mis en place un système de Wrapper afin d'initialiser des shader, program, vao et vbo plus facilement. J'ai aussi essayé de faire l'exercice bonus du TP2.

3 TP2

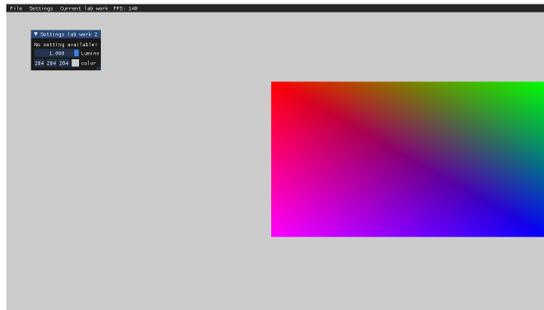


Figure 2: Résultat TP2

Le but de ce TP était un peu d'améliorer le programme du TP1. Il fallait au lieu d'avoir un triangle, un rectangle qui en plus se déplacer. Il fallut donc mettre en place plus de points dans le vbo, et aussi un ebo. L'ebo pouvait permettre de dire quel triangle on allait tracer avec des points donnés. Par exemple 1,2,3,4,2,0 on trace un triangle avec les points 1 2 3 et le deuxième avec les points 4 2 0. Il fallut aussi comprendre le système d'uniforme, c'est-à-dire envoyer des informations aux vertex, ici c'était pour effectuer une translation.

4 TP3

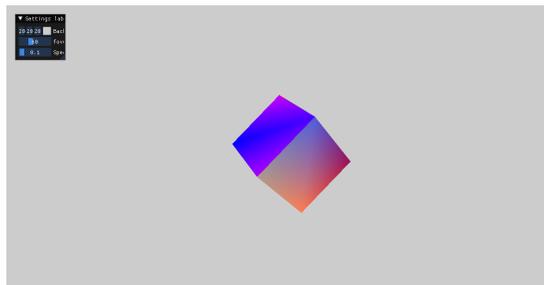


Figure 3: Résultat TP3

Le TP3 encore une fois est une suite directe au dernier TP2. Cette fois-ci on passait enfin à la 3D! Il fallut afficher un cube qui effectuer une rotation en

fonction du temps. Vu que nous étions en 3D, j'ai dû finir d'implémenter une classe caméra, c'est-à-dire compléter afin de pouvoir récupérer les différentes matrices. Nous avons donc besoin des matrices comme View Matrix, et Projection Matrix. La prochaine étape était de représenter notre cube à l'aide de notre ebo, vbo et vao. Pour comprendre mieux le problème, et afin de mieux identifier les triangles du cube à afficher, j'ai utilisé un schéma sur papier. Une fois notre cube en place sur le code, il fallait pouvoir envoyer nos 3 matrices (Model/Projection/View) à nos shaders. On faisait ça à l'aide d'uniforme et grâce à ces matrices nous pouvions calculer la position des éléments en fonction de la rotation et des espaces.

5 TP4



Figure 4: Résultat TP4

C'est à partir de ce TP qu'on commençait les choses sérieuses, c'est aussi à partir de celui-là que j'ai commencé à avoir pas mal de problèmes mais nous en parlerons dans une partie associée. Ce Tp était en deux parties, la première partie il fallut implémenté un système de calcul de lumière sur un modèle simple de lapin, pour ensuite l'utiliser sur une scène contenant plusieurs modèle.

Premier temps il fallait pouvoir load le modèle, c'est-à-dire compléter une classe avec les vbo, Vao, ebo correspondant. Sauf que cette fois nous avions qu'un seul vbo pour 5 attributs, il fallut donc utiliser les offsets (ou j'ai passé beaucoup de temps alors que c'était assez simple :(). Une fois les offsets bien configurés nous arrivons à afficher un lapin. Il fallut ensuite un par un ajouté des lumières pour calculer notre couleur. Pour cela on utilise une multitude de vecteur. Personnellement en plus des cours je me suis aidé d'une ressource sur internet qui est [ceci](#). Une fois les 3 lumières calculées on pouvait les additionner afin d'afficher un résultat final.

6 TP5



Figure 5: Résultat TP5

Ce TP5 parlait des textures, en premier temps il fallut appliquer une texture sur un lapin. Il fallut utiliser des Maps à envoyer dans le fragment shader qui allait à l'aide d'un calcul de texture modifier notre couleur. Si nous avons une map nous faisons donc le calcul de la texture, sinon le calcul normal comme au TP4. Il fallut implémenter les Maps pour l'éclairage spéculaire et diffus.

La suite de l'exercice parlait des filtres en effet il fallait en modifiant le code fourni donner un rendu plus réaliste. En effet on pouvait pour sûr le lion par exemple, un effet de suréchantillonnage. Il fallait donc appliquer un filtre de magnification en suivant les indications et le cours.

Il y avait aussi un problème de sous-échantillonnage sur les bannières. Pour ce faire il fallait faire l'effet inverse c'est-à-dire appliquer un filtre de minification..

7 Problèmes rencontrés

Durant ce projet j'ai rencontré pas mal de problèmes, surtout certain qui m'a pris pas mal de temps. Le TP4 pour moi était le plus compliqué à comprendre, je ne saurai pas expliquer mais j'avais vraiment beaucoup de mal avec. Au début il m'a bloqué bêtement sur les offset, je n'arrivais pas à bien comprendre la subtilité. Ensuite quand j'avais l'impression d'avancer sur mon éclairage j'avais toujours un problème (cf.: problème sur la table :() mais une fois ce problème passer un m'a vite rattrapé c'était le problème du dernier exercice du TP5. J'ai vraiment eu du mal à comprendre comment je pouvais avancer, j'ai essayé pas mal de méthodes différents mais sans succès. Je ne compte pas aussi toutes les erreurs dues à une faute d'écriture que je n'ai pas vue... J'ai aussi eu un problème de petite lumière rouge sur mon sponza, mais j'avais aucune idée d'où cela pouvait venir.



Figure 6: Le problème en question