

Rapport de TP Textures

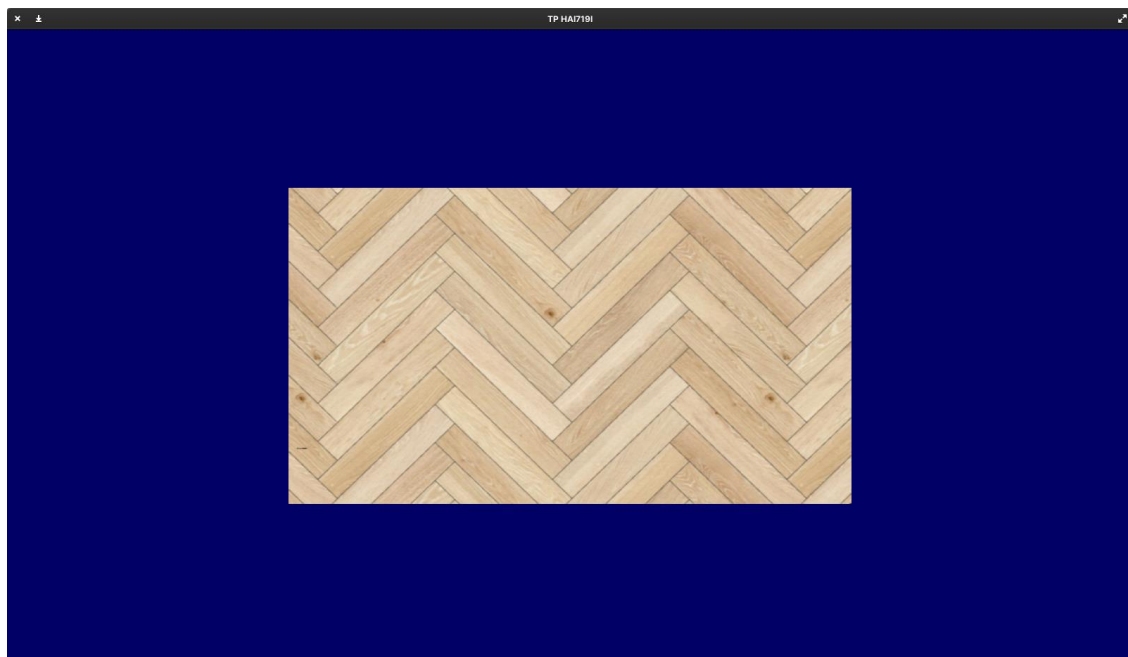
Programmation 3D

- Niveau 0 TP1 :

Pour cette partie j'ai réussi dans un premier temps à appliquer à un simple rectangle (deux triangles) une texture en utilisant comme base de code un TP réalisé précédemment.



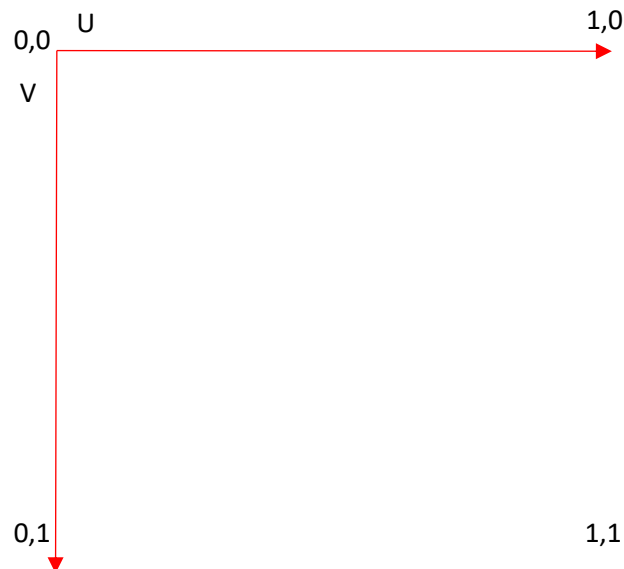
Texture que l'on souhaite appliquer



Résultat obtenu

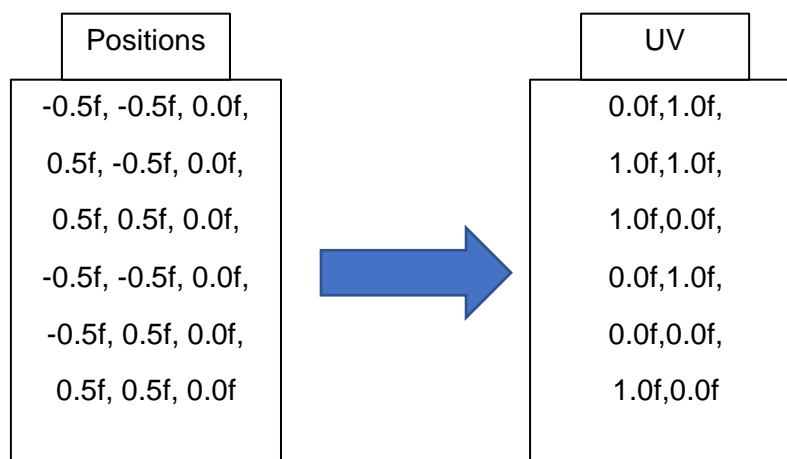
Pour ce faire j'ai créé 3 `ARRAY_BUFFER` que j'ai envoyé au shader (`vec3` position, `vec3` couleur de la forme, `vec2` UV de la texture en fonction de la position) puis j'envoie la `GL_TEXTURE_2D` que je souhaite afficher.

Une grande difficulté que j'ai rencontrée et la compréhension des UV, je pensais que 0,0 se trouvait en bas à gauche ce qui n'était pas le cas, il se trouve en fait finalement en haut à gauche



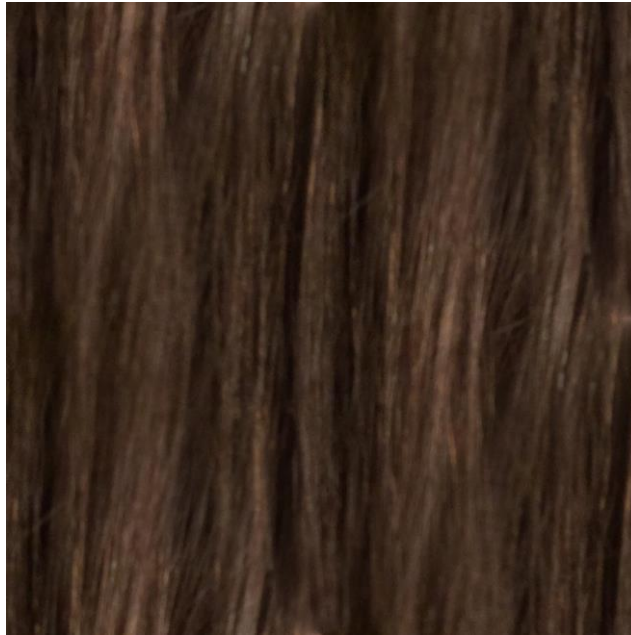
Positionnement des UV (le V est inversé)

Ainsi pour de telle position :

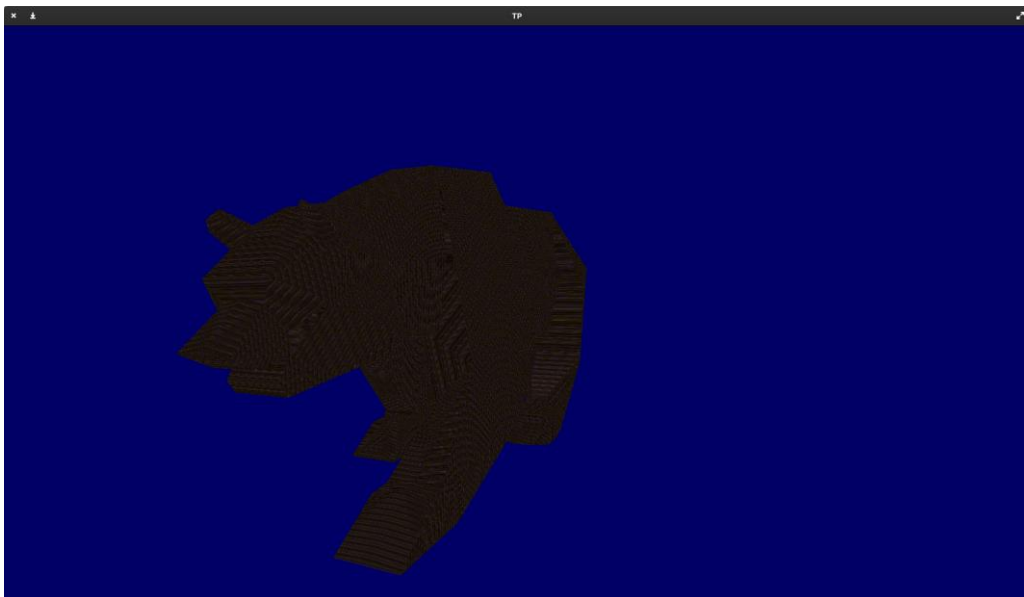


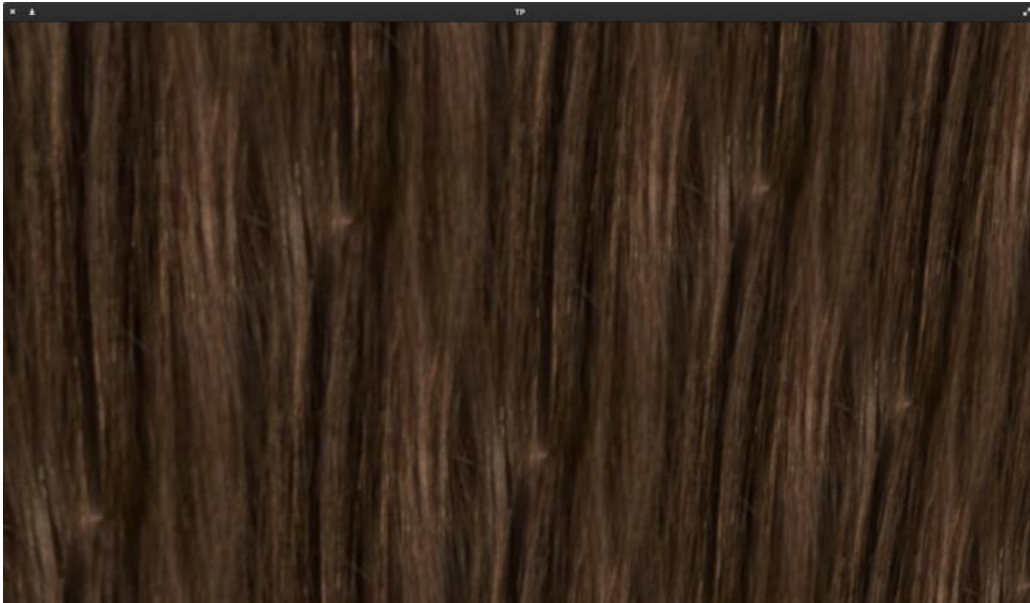
Enfin il suffit d'utiliser les UV et la fonction `texture()` dans le `fragment_shader` pour appliquer la texture souhaitée.

Par la suite j'ai utilisé la base de code qui nous a été fourni pour l'appliquer cette fois si à un mesh (un mesh d'ours low poly trouvé sur internet).



Texture que l'on souhaite appliquer





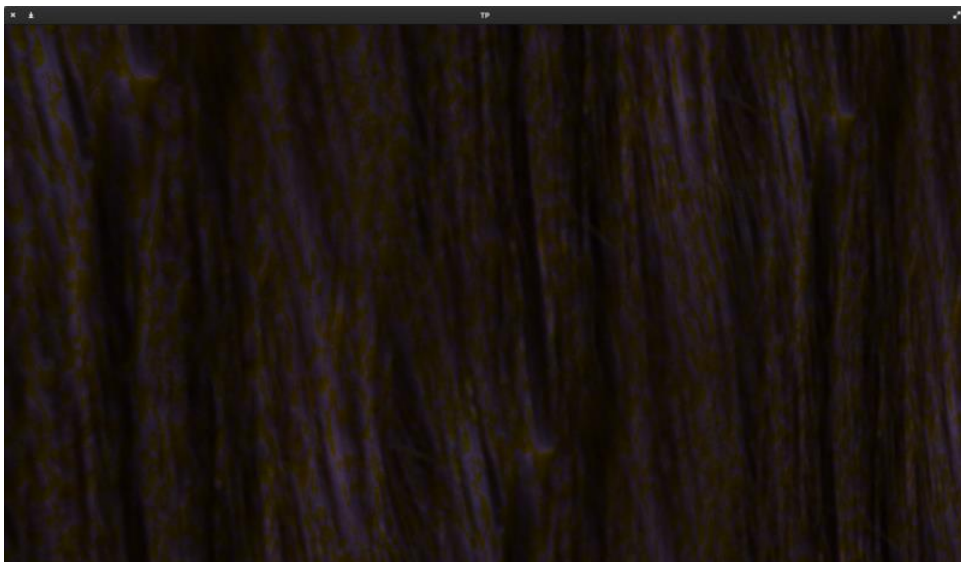
Résultat obtenue appliqué sur le mesh (qui se trouve être finalement vraiment gros) et rendu lorsque l'on zoom

- Niveau 1 TP1 :

A présent le but de cette partie est d'appliquer à notre texture une bump map (ou normal map), pour ce faire j'ai utilisé dans un premier temps cette bump map :



Bump Map que je souhaite appliquer à ma texture



Résultat obtenue

On peut alors voir qu'il y a comme des sortes de taches jaunâtres qui apparaissent et que la texture est beaucoup plus assombrie (ce qui n'est pas bon).

Après de nombreuses tentatives j'ai fini par poser la question à mes collègues de promo qui m'ont conseillé d'utiliser la BoomBox (<https://github.com/KhronosGroup/glTF-Sample-Models/tree/master/2.0/BoomBox>), ce que j'ai fait.

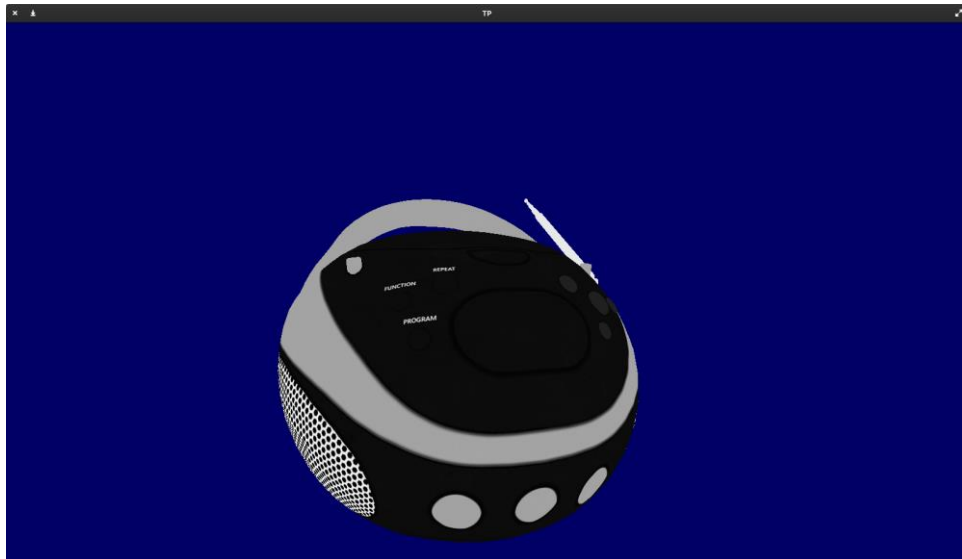
En voyant les différents fichiers disponibles j'ai compris qu'il me faudra utiliser une Normal map plutôt qu'une Bump map et ainsi comprendre comment l'utiliser.

Le tutoriel de Learn OpenGL m'as énormément aidé pour cette partie (<https://learnopengl.com/Advanced-Lighting/Normal-Mapping>)

Ainsi après avoir réussi à appliquer la texture fourni (et n'oubliant pas de prendre en compte l'inversion du V) j'obtiens ceci :



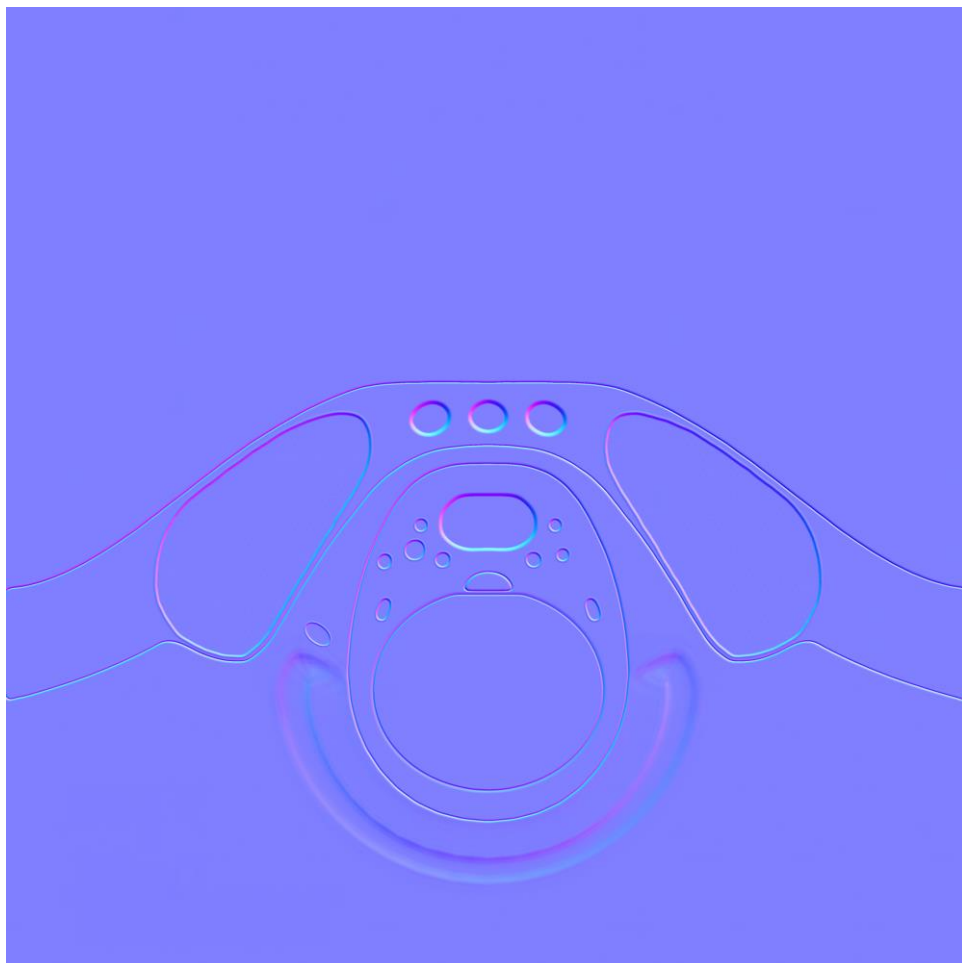
Texture de la BoomBox



Résultat obtenue de l'application de la texture

Le résultat est cette fois si beaucoup plus satisfaisant qu'avec le précédent Mesh.

Maintenant que nous avons une texture appliquée correctement, on peut ajouter un effet de profondeur à l'aide de la normal map disponible sur le git.



Normal Map de la BoomBox

Après avoir suivi le tutoriel de Learn OpenGL et de nombreuses tentatives voici le résultat obtenu :



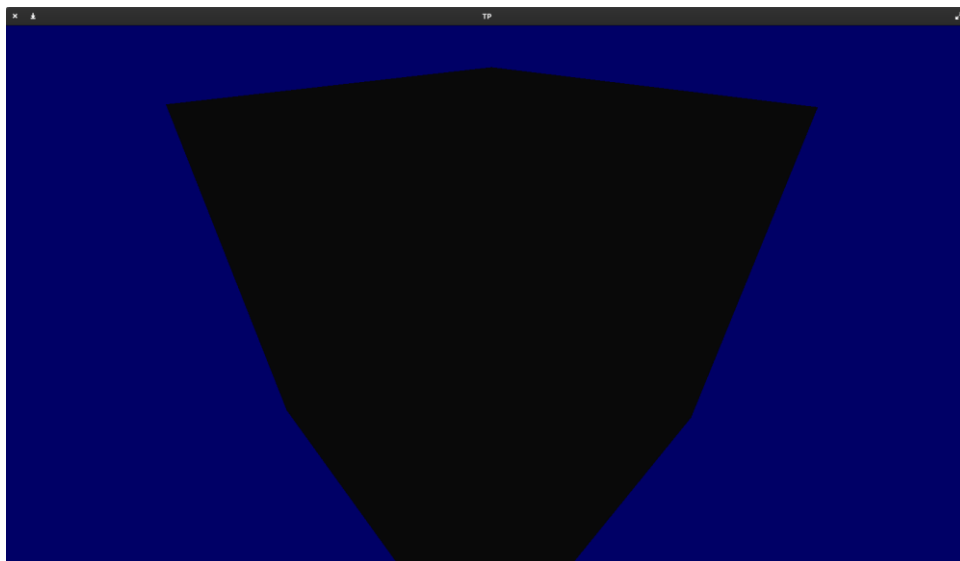
Résultat obtenu en utilisant le modèle de Phong avec une source de lumière situé en haut à gauche devant la BoomBox

- Niveau 2 TP1 :

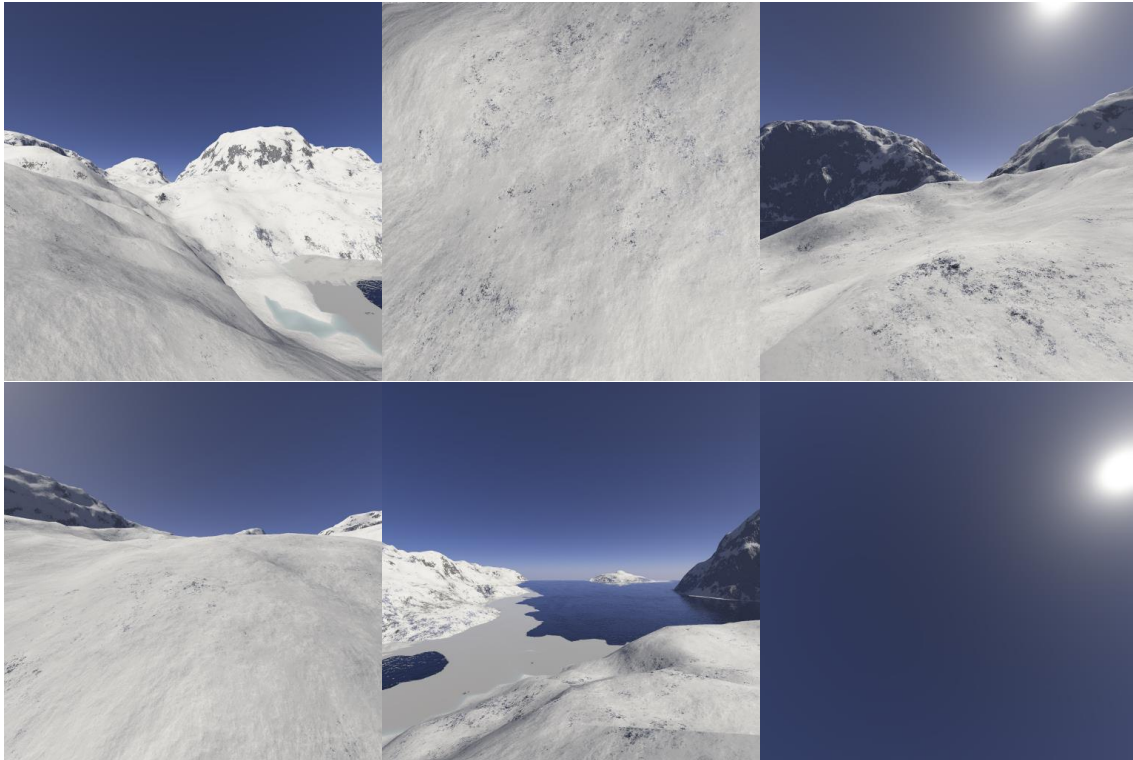
J'ai eu énormément de mal à comprendre cette partie et n'ai à l'heure actuelle toujours pas réussi à obtenir quelque chose de convenable.

Comme pour le normal Mapping, j'ai de nouveau suivi le tutoriel de Learn OpenGL sur le Cubemaps (<https://learnopengl.com/Advanced-OpenGL/Cubemaps>)

J'ai commencé par afficher un cube qui englobe ma BoomBox :



Puis j'ai cherché par la suite une sky-box sur internet :



J'ai dû essayer beaucoup de choses différentes afin d'obtenir quelque chose de satisfaisant et j'ai ainsi fini par obtenir ceci :



Rendu de la skybox avec une lumière positionnée en fonction de la source de lumière



BoomBox au milieu d'une nouvelle Skybox

Ainsi maintenant que notre SkyBox est fonctionnelle il est possible de transformer notre mesh pour qu'il reflète la skybox et ainsi donner l'illusion qu'il est conçu en miroir :



BoomBox avec l'effet "miroir"



Autre mesh avec effet “miroir”

- Niveau 0 TP2 :

► Albedo (Couleur + Texture)

De ce que je pense comprendre, Albedo s’agit d’un simple plaquage de texture.

Ainsi si on reprend notre mesh de poisson et on utilise Phong en lui appliquant sa normal associé :



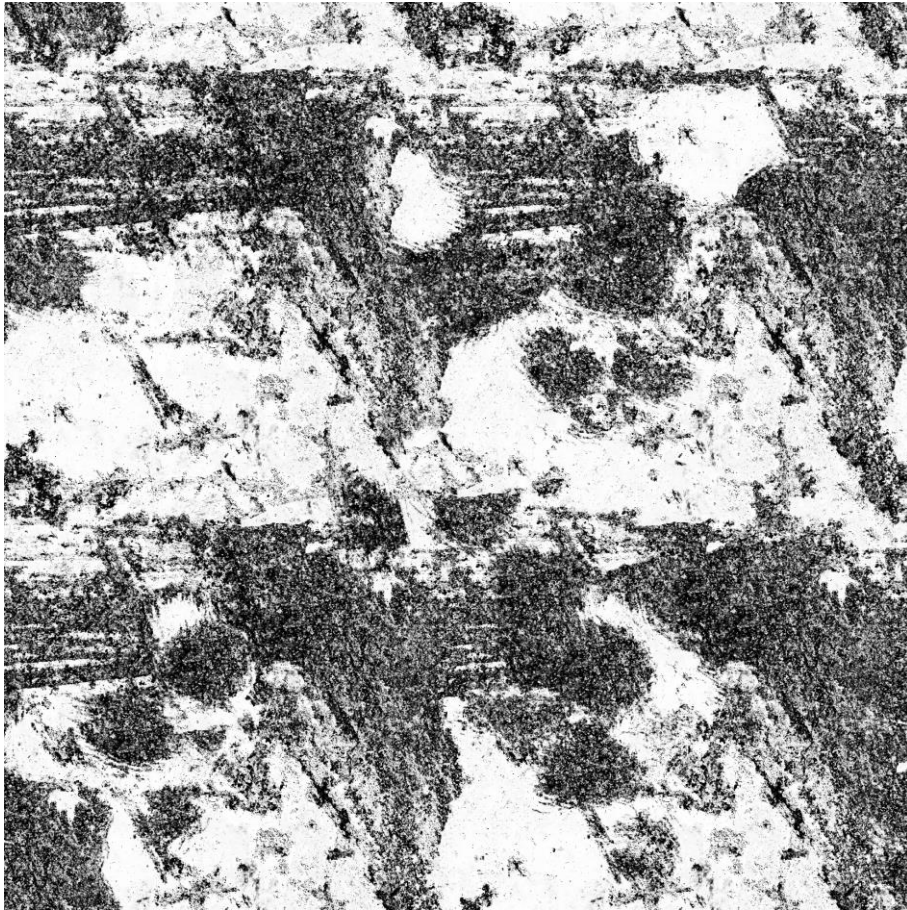
Résultat obtenu pour Albedo

► Metalness (Valeur + Texture)

Pour Metalness j’ai utilisé cette map et cette texture

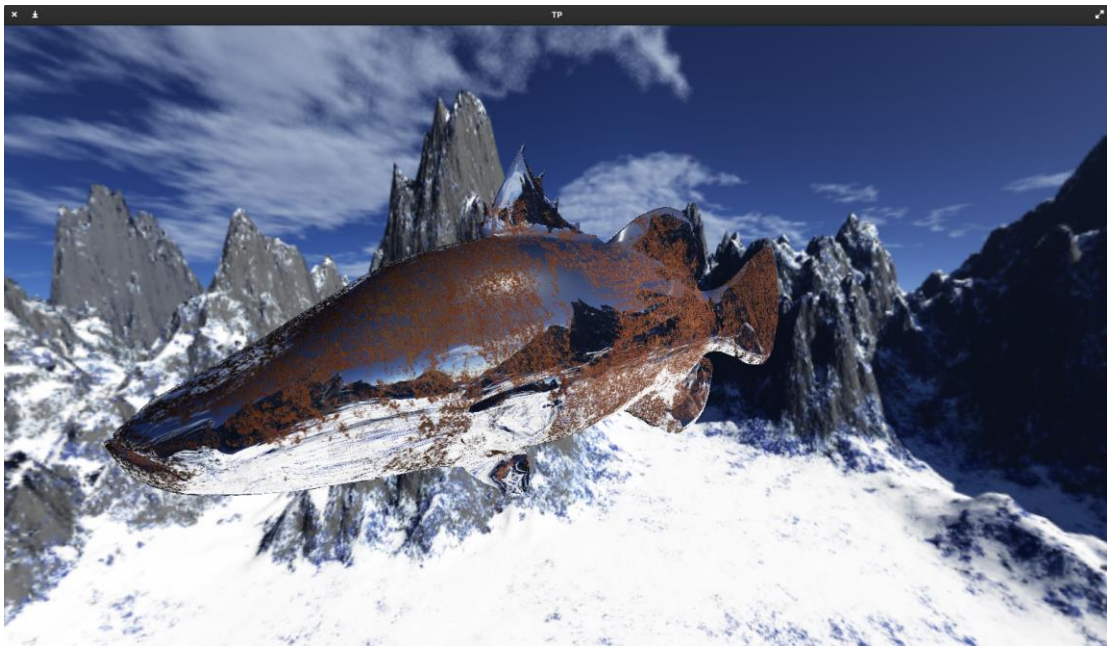


Texture effet "rouillé"



Metalness Map

Selon s'il y a du noir ou pas avec la metalness map on affiche la texture ou le reflet.
Et obtiens ainsi ce résultat :

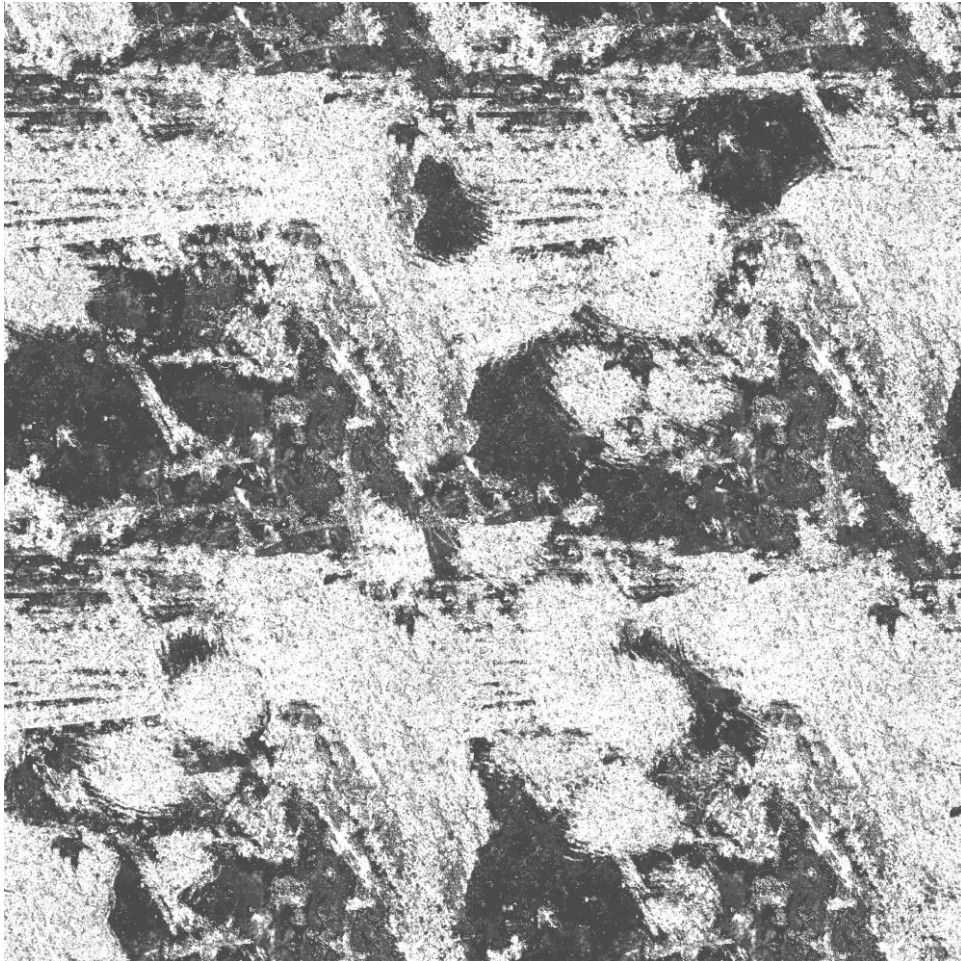


Résultat obtenu pour Metalness

► Roughness (Valeur + Texture)

Pour le Roughness j'ai eu beaucoup de mal à comprendre concrètement ce qu'il fallait faire, ainsi j'ai regardé des textures "rugueuses" sur internet et ai essayé de faire quelque chose qui s'en rapproche le plus par intuition.

J'ai donc utilisé cette Roughness map

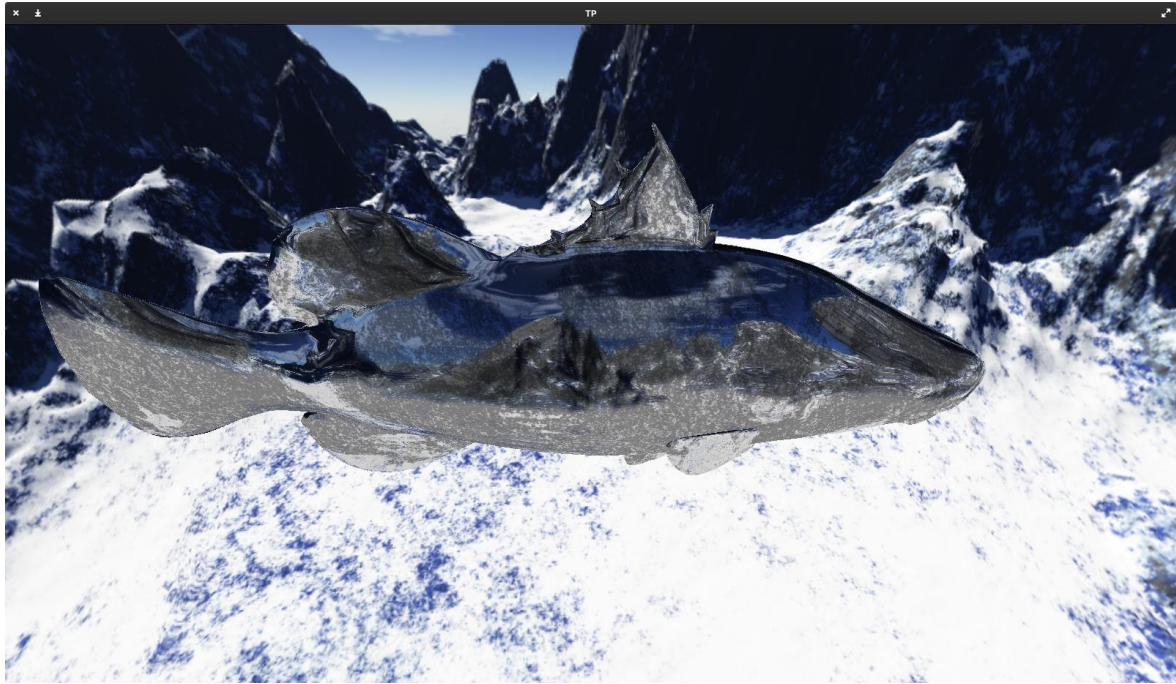


Roughness Map

Ainsi afin d'avoir un effet "non poli" j'ai voulu faire en sorte que mon miroir soit flou, pour ce faire j'ai lancé 10 rayons avec des directions légèrement différentes du rayon principal puis fait la moyenne de ses rayons.

Et ainsi selon la map (plus ou moins de NOIR) afficher plus ou moins la vraie réflexion où la réflexion floue.

Ce qui nous donne comme résultat (avec une couleur légèrement atténuée pour bien voir les parties rugueuses) :

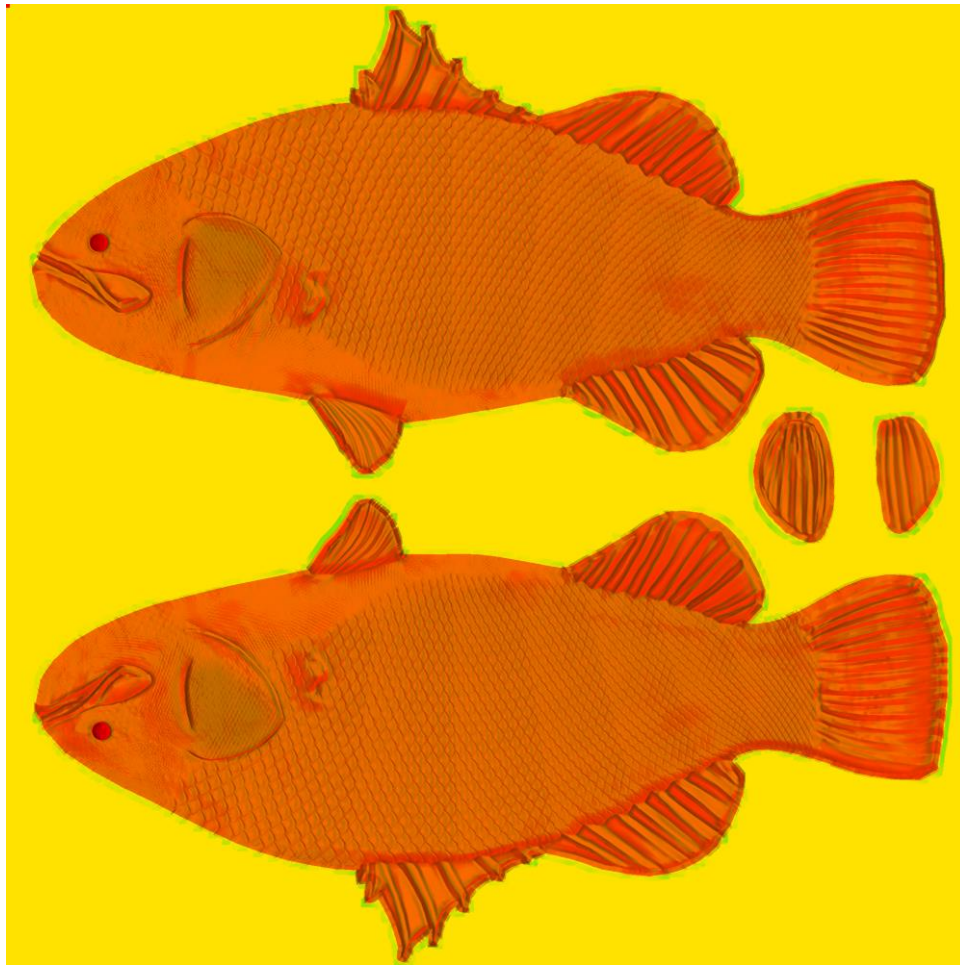


Résultat obtenu pour Roughness

On peut voir sur la gauche qu'il y a une zone non rugueuse ainsi le reflet et parfaitement bien reflété cependant ce n'est pas le cas pour le reste.

► AO (Valeur + Texture)

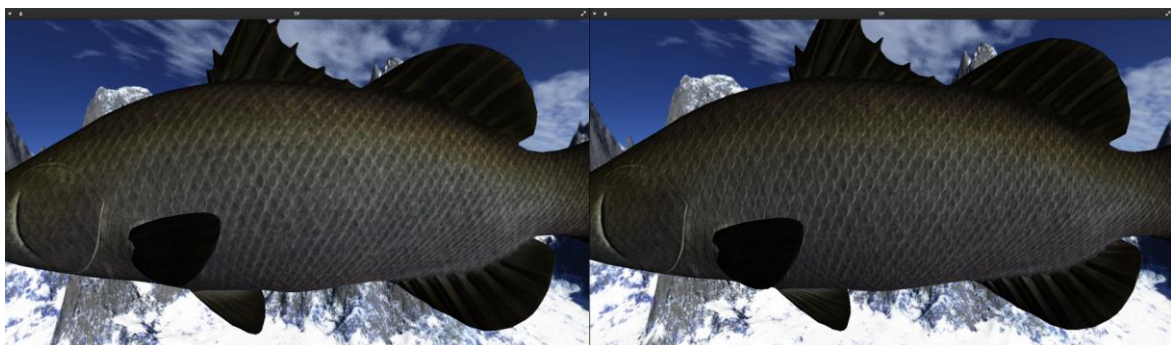
Et enfin pour AO j'ai utilisé la map fourni avec le model du poisson



OcclusionRoughnessMetallic map

Cette map est un peu plus différentes de celles vu précédemment, elle contient en elle 3 map, le canal rouge représente l'AO, le canal vert la Roughness et le canal Bleu la Metallness ainsi on peut voir que sur cette map il n'y a pas de métal car il n'y a pas de bleu présent.

On obtient ainsi (à gauche sans AO et droite avec AO)



Résultat obtenu avant/après application de l'occlusion ambiante

Je n'ai pas compris ce qui était demandé par rapport aux classes de Matières à créer et ne possède plus assez de temps pour y réfléchir dessus.

- Niveau 1 TP2 :

Je n'ai pas eu le temps de réfléchir dessus.

- Niveau 2 TP2 :

Je n'ai pas eu le temps de réfléchir dessus.

- Ce que ça m'a apporté :

J'ai beaucoup apprécié travailler sur ces TP, je commence à avoir une bonne vision de ce qu'il se passe au niveau de la transmission d'informations au niveau du GPU.

Cependant je suis dessus de ne pas avoir pu continuer par manque de temps.

Je pense tout de même le continuer/peaufiner plus tard sur mon temps libre.

- Organisation :

J'ai travaillé environ 8h dispersé sur le long de la semaine la semaine dernière et j'ai depuis lundi 19 travaillé dessus de 14h à 21h tous les jours.

Même si le TP nous a été de base attribuée en début du mois de décembre je n'ai pu commencer à y travailler que lors des vacances et ai quand même dû travailler sur d'autres projets à côté qui devaient être rendu avant.