

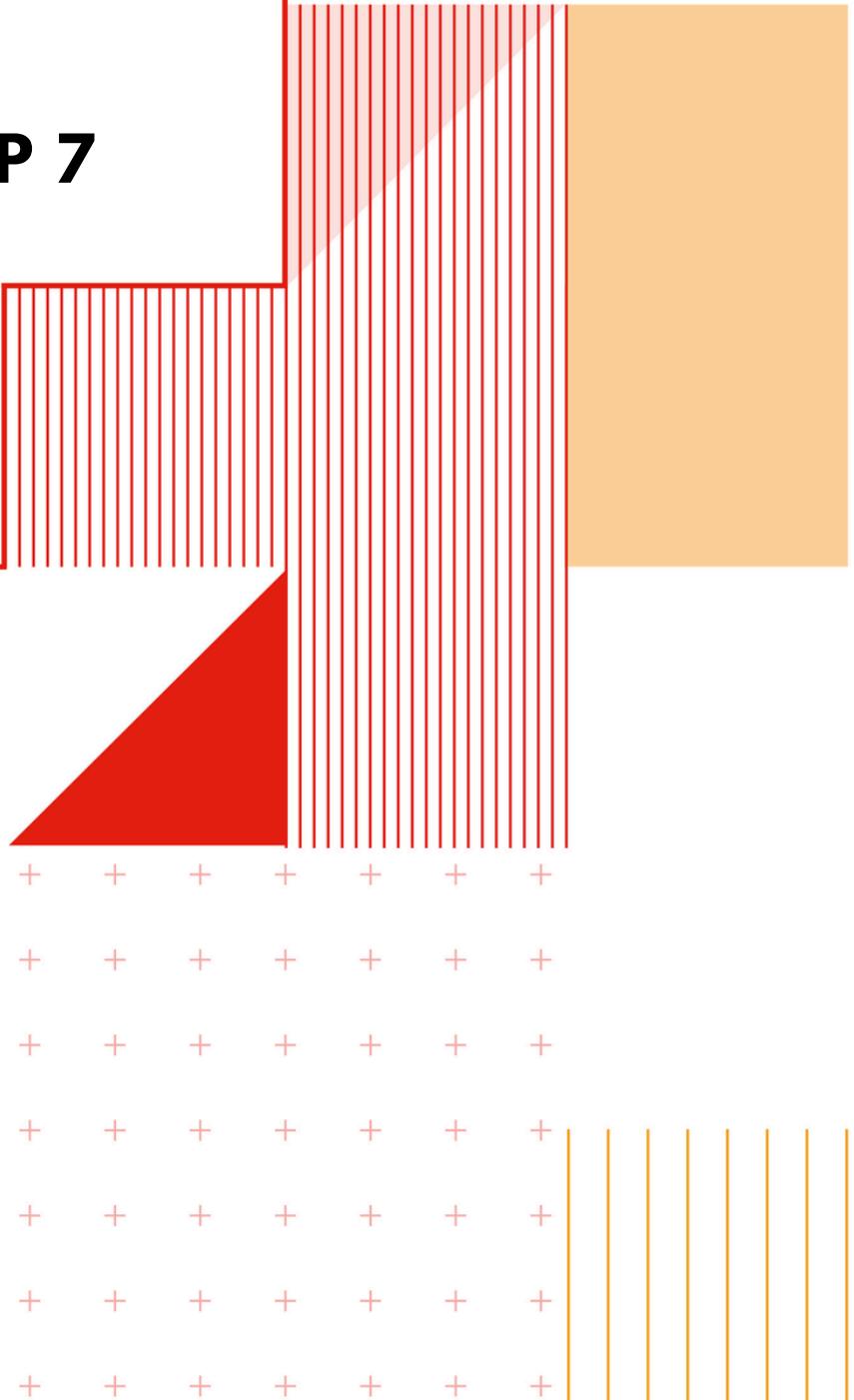
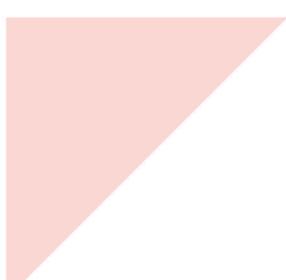
Informatique

Graphique - TP 7

Texturing

GRAINDORGE Amance

VERNET Hector



I. INTRODUCTION

Ce TP a pour but de se familiariser avec les techniques de texturing en informatique graphique. Nous aborderons d'abord les bases du texturing, puis nous étudierons les options de wrapping et de filtering. Ensuite, nous étudierons des techniques plus avancées telles que le mipmapping, le multi-texturing et l'utilisation de cubemaps.

II. EXERCICE 1

Ce premier exercice consiste à étudier l'implémentation du texturing dans la classe `TexturedMeshRenderable`. Pour appliquer une texture à un maillage, l'idée est d'associer à chaque sommet du maillage des coordonnées (u, v) sur la texture 2D. Cette association est présente dans le fichier `OBJ` du maillage, et est donc chargée avec la fonction `read_obj()`.

Un exemple de rendu d'un maillage texturé est représenté dans la figure ci-dessous, avec la texture correspondante :

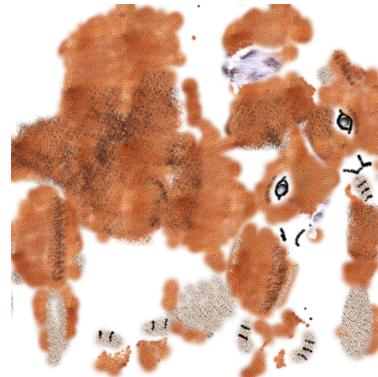
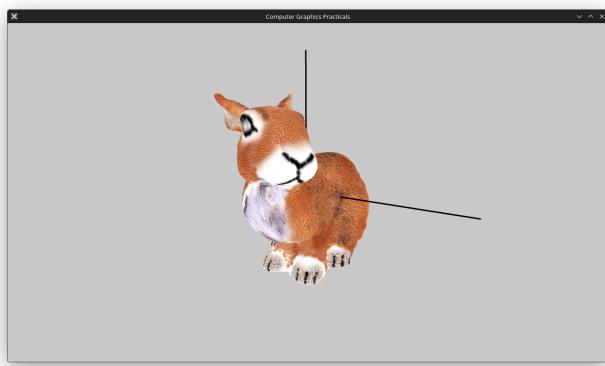


Figure 1 : Rendu d'un maillage texturé et la texture correspondante

III. EXERCICE 2

Cet exercice présente les différentes techniques de wrapping. Le wrapping consiste à déterminer comment la texture doit être appliquée sur une surface qui dépasse sa taille. On peut voir dans l'exemple ci-dessous les différentes options de wrapping appliquées à une même texture sur un plan. La première consiste à simplement agrandir la texture, mais cela rend beaucoup plus visibles les pixels d'image si l'on est trop proche. Les seconde et troisième options utilisent respectivement le wrapping en répétant la texture et en la reflétant. Ces techniques permettent de couvrir une plus grande surface sans agrandir les pixels, mais créent des motifs répétitifs, ce qui peut être indésirable dans certains cas. La dernière option utilise le clamp, qui étire les bords de la texture pour couvrir la surface. Cela crée un effet d'étirement au bord, et ne convient probablement pas pour des textures avec des motifs distincts. Par comparaison, la dernière image montre la texture appliquée sur le plan sans aucune technique de wrapping.

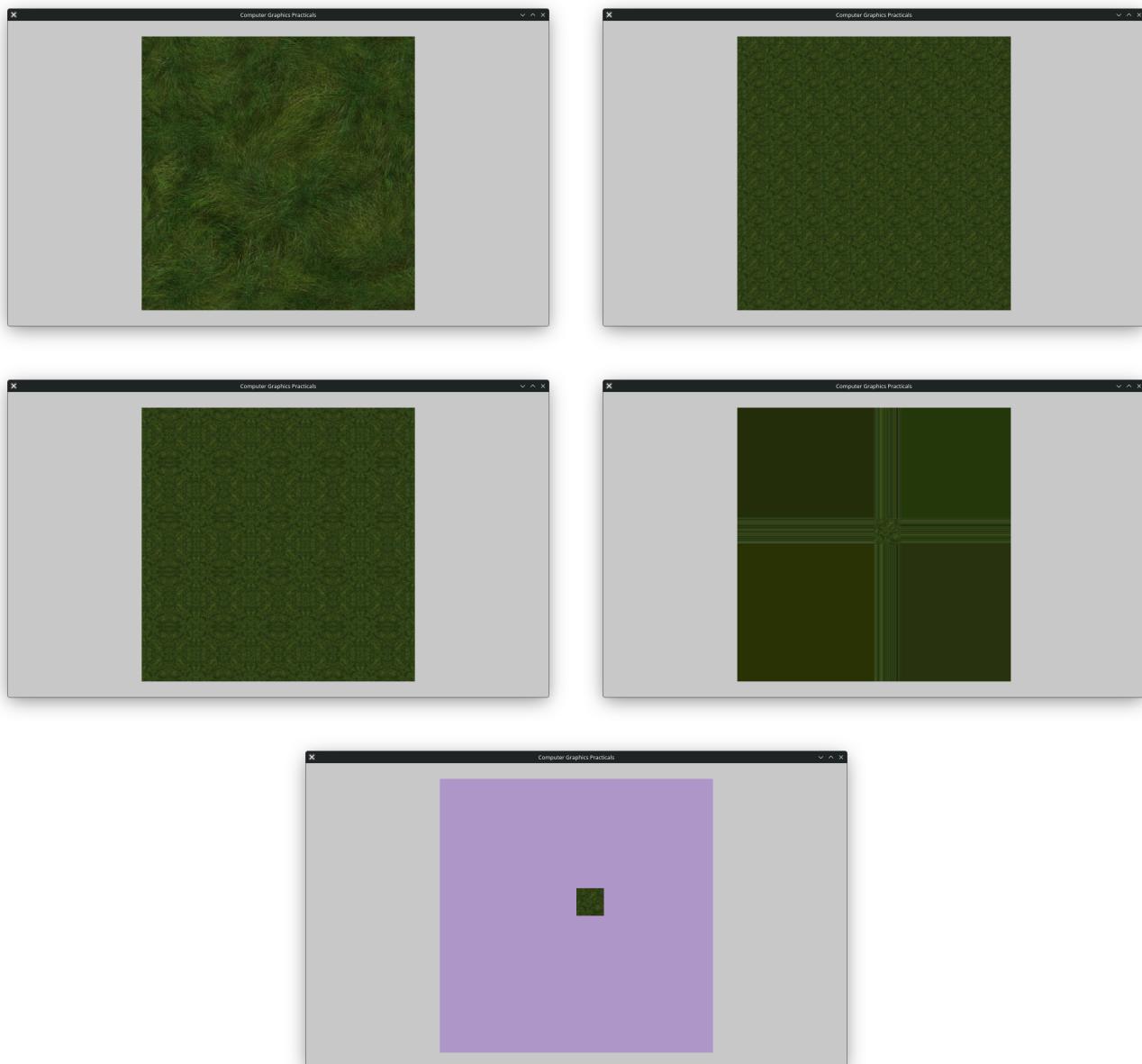


Figure 2 : Différentes techniques de wrapping appliquées à une même texture sur un plan

IV. EXERCICE 3

Ce troisième exercice présente les différentes techniques de filtering. Le filtering détermine comment les pixels de la texture sont échantillonnés lorsqu'ils sont appliqués à une surface. La première image utilise le filtering nearest-neighbor, qui sélectionne le pixel le plus proche de la coordonnée de texture. Cela peut créer un effet pixelisé lorsque la texture est agrandie. La seconde image utilise le filtering linear, qui interpole entre les pixels voisins pour obtenir une transition plus douce. Cela réduit l'effet pixelisé, mais peut introduire un flou lorsque la texture est agrandie.

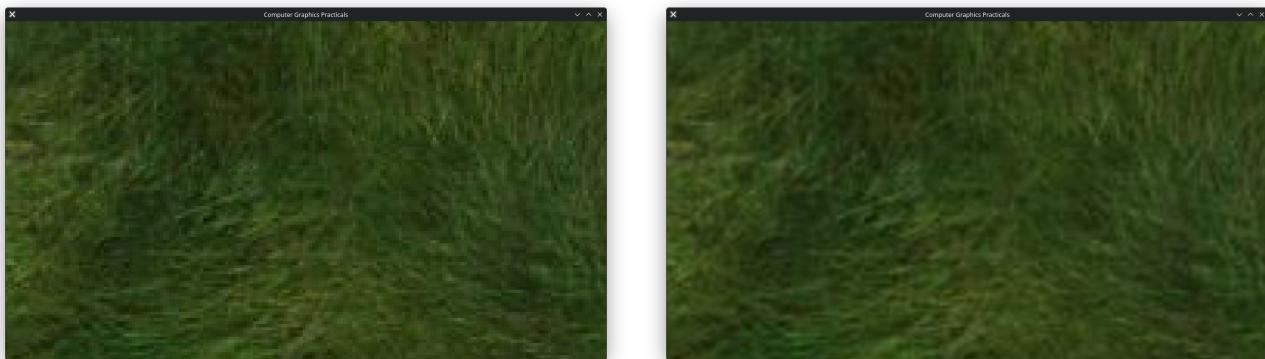


Figure 3 : Différentes techniques de filtering appliquées à une même texture sur un plan

V. EXERCICE 4

Cet exercice explore le mipmapping, une technique qui utilise plusieurs versions pré-calculées d'une texture à différentes résolutions. Lorsque la texture est vue de loin, une version de résolution inférieure est utilisée, ce qui améliore les performances et réduit le scintillement. Les exemples Figure 4 montrent deux cubes : celui de droite utilise le mipmapping, tandis que celui de gauche n'en utilise pas.

On peut observer qu'avec la distance, le cube sans mipmapping présente de l'aliasing, c'est-à-dire que les détails de la texture deviennent brouillés et scintillent, car la texture haute résolution est échantillonnée de manière inadéquate. En revanche, le cube avec mipmapping conserve une apparence plus nette et stable, car il change dynamiquement de texture en fonction de la distance, utilisant des versions de l'image avec des résolutions de plus en plus basses (numérotées de 1 à 4).

Plusieurs options sont disponibles pour le mipmapping, notamment le choix du filtre (nearest ou linear) et la méthode de sélection du niveau de mipmap (nearest ou linear). Le choix du filtre affecte la qualité de l'échantillonnage entre les niveaux de mipmap, tandis que la méthode de sélection détermine comment le niveau de mipmap approprié est choisi en fonction de la distance. L'option offrant la meilleure qualité visuelle est le filtre linear combiné à la sélection linear, car elle permet une transition plus douce entre les niveaux de mipmap et réduit les artefacts visuels, mais elle est aussi la plus coûteuse en termes de performance, car elle nécessite plus de calculs pour l'interpolation.

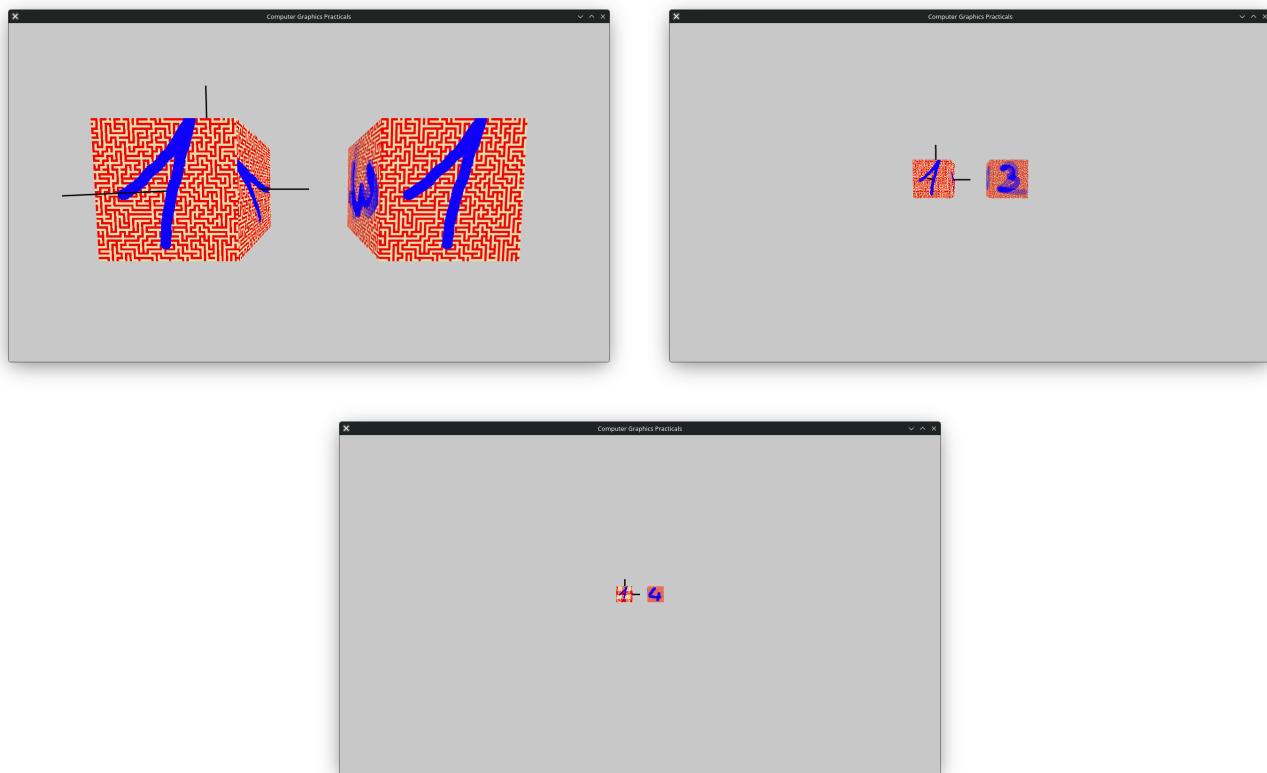


Figure 4 : Comparaison entre un cube avec et sans mipmapping

VI. EXERCICE 5

Ce cinquième exercice explore le multi-texturing, une technique qui permet d'appliquer plusieurs textures à un même objet. L'exemple Figure 5 montre deux cubes :

- Le cube de gauche utilise deux textures : une texture de base et une texture de détails superposée. Les deux textures sont combinées en multipliant simplement leurs couleurs.
 - Le cube de droite utilise la technique du normal mapping, qui utilise une texture spéciale (la normal map) pour simuler des détails de surface en modifiant les normales des pixels. Cela crée l'illusion de relief sans ajouter de géométrie supplémentaire. Les trois images montrent le changement de la combinaison des textures et de la normal map en fonction du temps pour créer l'illusion d'un changement de lumière sur la surface du cube.

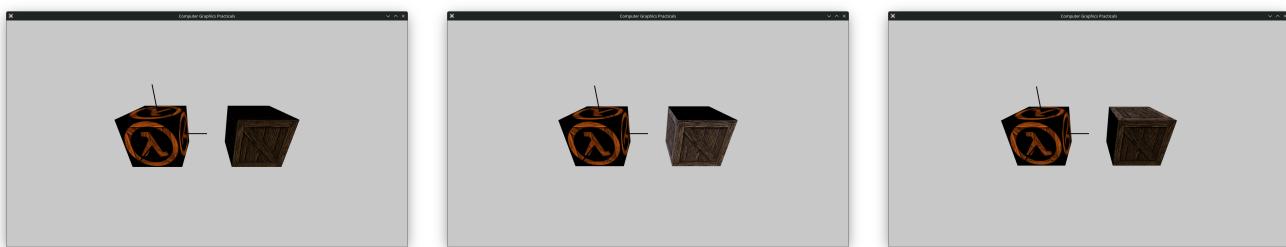


Figure 5 : Exemples de multi-texturing appliqu       des cubes

VII. EXERCICE 6

Ce dernier exercice explore l'utilisation des cubemaps pour le texturing. Les cubemaps sont des textures spéciales qui représentent l'environnement autour d'un point en utilisant six images.

formant un cube. L'utilisation des cubemaps à la place d'une image 360° permet de limiter d'effectuer des calculs moins coûteux en temps réel. La première image de l'exemple Figure 6 montre un paysage texturé à l'aide d'une cubemap, plutôt qu'être rendu comme un objet 3D classique. La seconde image illustre l'utilisation des cubemaps pour l'éclairage environnemental, où la cubemap est utilisée pour simuler la lumière ambiante provenant de l'environnement autour d'un objet.

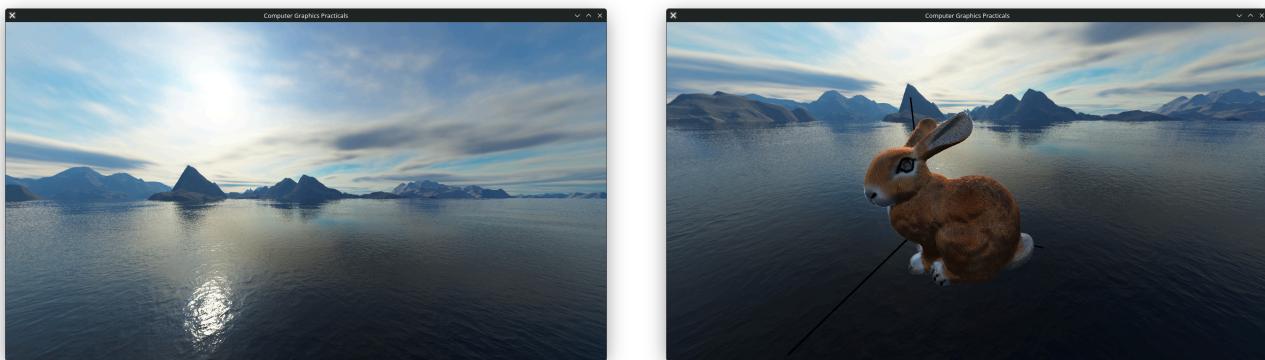


Figure 6 : Utilisation des cubemaps pour le texturing et pour l'éclairage environnemental

VIII. CONCLUSION

Ce TP nous a permis de nous familiariser avec les différentes techniques de texturing en informatique graphique. Nous avons exploré les bases du texturing, les options de wrapping et de filtering, ainsi que des techniques plus avancées telles que le mipmapping, le multi-texturing et l'utilisation des cubemaps. Ces techniques seront utiles pour améliorer la qualité visuelle de notre projet. Nous utiliserons notamment une cubemap pour le paysage environnant.