

# IMN401 - Infographie et jeu vidéos - TP 4

Guillaume GILET - [guillaume.gilet@usherbrooke.ca](mailto:guillaume.gilet@usherbrooke.ca)

## 1 Consignes

- Travail a faire par groupe de 3.
- Le travail sera à rendre avant le **Lundi 17 avril 2023**.
- Livrables :
  - Une archive contenant le code source du TP4. A rendre sur [turnin.dinf.usherbrooke.ca](https://turnin.dinf.usherbrooke.ca).

## 2 Objectif

Nous allons maintenant appliquer des textures discrètes de couleur et de normale sur notre objet. Nous utiliserons pour cela la classe **Texture2D** présente dans le projet. Vous pouvez commencer par lire le code source de cette classe afin de retrouver les différents éléments OpenGL vus en cours.

## 3 Travail a rendre

L'objectif final est d'afficher deux objets 3D (le bunny et le sol) éclairés à l'aide du modèle de Phong, texturés avec un mélange de deux textures de couleur et d'une carte de normale. Les étapes suivantes vous indiquent pas à pas la marche à suivre.

### 3.1 Chargement des textures

Commencez par créer un nouveau matériau avec dans son constructeur un pointeur vers un objet **Texture2D** passé en paramètre. Nous allons dans un premier temps afficher le bunny avec des textures de couleur.

- Commencez par récupérer dans le *vertex shader* les coordonnées de textures (c'est ici un *vec3* arrivant dans l'attribut numéro 3 et transmis automatiquement par le modèle géométrique) et transmettez ses deux premières composantes au *fragment shader*. Afin de vérifier que les coordonnées de textures sont bien définies, affichez l'objet avec comme couleur r et g les deux premières composantes des coordonnées de texture.
- Créez ensuite dans la fonction init de EngineGL un objet **Texture2D** contenant la texture "**Bunny1.png**". Cette texture est rendu résidente en mémoire sur le GPU par le constructeur de la classe.
- Créez dans le *fragment shader* une variable uniforme de type **sampler2D** et transmettez lui le **handle** 64bits de la texture (cf. cours et classe **Texture2D**). Assignez comme couleur diffuse et ambiante du modèle de Phong la couleur issue de la texture en chaque fragment.
- Ajoutons maintenant une deuxième texture de couleur. Chargez et transmettez une deuxième texture "**Bunny2.png**" au matériau. Cette texture contient un canal alpha indiquant la transparence. Nous souhaitons que la couleur de notre objet soit un mélange des deux textures T1 et T2 en utilisant l'interpolation linéaire suivante :  $C.rgb = T2.a * T2.rgb + (1 - T2.a) * T1.rgb$

### 3.2 Normal mapping

Nous voulons maintenant réaliser un normal mapping dans l'espace tangent pour ajouter du relief à notre objet. Commencez par charger dans une variable **sampler2D** du *fragment shader* la texture **Bunny\_N.png**. Nous effectuerons le calcul de Phong pour chaque fragment dans l'espace local à la surface. Nous aurons pour cela besoin de mettre à jour les shaders.

#### 3.2.1 Vertex Shader

- Récupérez dans un premier temps le vecteur tangent (c'est un *vec4* arrivant pour chaque sommet dans l'attribut numéro 4).

- En vous aidant du cours, créez une matrice  $3 \times 3$  TBN permettant le passage du repère local au repère objet. Nous souhaitons effectuer les calculs d'illumination dans le repère local. Il faut donc inverser/transposer cette matrice pour obtenir la matrice de transformation du repère objet vers le repère local.
- Transformez les vecteurs  $L$  et  $V$  dans le repère local et transmettez les au *fragment shader*.

### 3.2.2 Fragment Shader

- Récupérez  $V$  et  $L$  et normalisez les.
- Afin de tester la transformations effectuez le calcul de Phong à partir de  $V$  et  $L$  et d'une normale  $N = (0, 0, 1)$ . Le résultat doit être identique au résultat de la section précédente.
- Récupérez maintenant la normale depuis la carte de normale et effectuez le calcul de Phong avec cette normale (Attention, la normale contenue dans la texture encode des valeurs entre 0 et 1 pour chaque canal. Il faut transformer ces valeurs afin qu'elles soient entre -1 et 1)

Le code finalisé doit afficher un lapin texturé et présentant du relief éclairé par le modèle de Phong. Ajoutez finalement les textures de couleur "BrickL.png" et de normale "BrickL\_n.png" au modèle du sol.