

INF2705 Infographie

Spécification des requis du système

Travail pratique 1

Utilisation du pipeline graphique et des VBO

Table des matières

1	Introduction	2
1.1	But	2
1.2	Portée	2
1.3	Remise	2
2	Description globale	3
2.1	But	3
2.2	Travail demandé	3
3	Exigences	6
3.1	Exigences fonctionnelles	6
3.2	Exigences non fonctionnelles	6
3.3	Rapport	6
A	Liste des commandes	7
B	Figures supplémentaires	7
C	Apprentissage supplémentaire	10

1 Introduction

Ce document décrit les exigences fonctionnelles et non fonctionnelles du TP1 « *Utilisation du pipeline graphique et des VBO* » du cours INF2705 Infographie.

1.1 But

Le but des travaux pratiques est de permettre à l'étudiant d'appliquer directement les notions vues en classe.

1.2 Portée

Chaque travail pratique permet à l'étudiant d'aborder un sujet spécifique.

1.3 Remise

Faites la commande « `make remise` » afin de créer l'archive « **INF2705_remise_TPn.zip** » que vous déposerez ensuite dans Moodle. (Moodle ajoute automatiquement vos matricules ou le numéro de votre groupe au nom du fichier remis.)

Ce fichier zip contient le fichier Rapport.txt et tout le code source du TP (makefile, *.h, *.cpp, *.glsl, *.txt).

2 Description globale

2.1 But

Le but de TP est de permettre à l'étudiant de mettre en pratique les fonctions de contrôle du pipeline graphique d'OpenGL pour la modification des matrices et la manipulation de la caméra synthétique : `Rotate()`, `Translate()`, `Scale()`, `PushMatrix()` et `PopMatrix()`.

Ce travail pratique lui permettra aussi d'utiliser les fonctions liées aux *Vertex Buffer Objects (VBOs)* : `glGenBuffers()`, `glBindBuffers()`, `glBufferData()` et `glDrawElements()`.

2.2 Travail demandé

Partie 1 : la bestiole

On demande de réaliser un programme permettant d'afficher une bestiole un peu bizarre (probablement d'origine extraterrestre !) avec une tête sphérique de rayon fixe, un corps cylindrique de taille variable (`bestiole.taille`), deux ailes carrées (même taille que le corps) et quatre pattes en parallélipède allongé de taille fixe (`bestiole.longPatte` x `bestiole.largPatte`) formées par des cubes étirés. Cette bestiole pourra aussi transformer son corps en LA théière bien connue en infographie ! La Figure 1 montre cette bestiole sous ses deux formes : cylindre et théière.

Le cylindre, le sphère, le cube et les quadrilatères sont tracés par des appels aux fonctions fournies (sans modifier ces fonctions). La tête de la bestiole est une sphère positionnée sur l'arête supérieure du corps (en X+ et Z+). Chaque aile est composée d'un quadrilatère et articulé selon un angle (`bestiole.angleAile`), tandis que chaque patte est composée d'un cube étiré et articulé selon un angle (`bestiole.anglePatte`) (voir Figure 2). Enfin, la bestiole peut aussi tourner sur elle-même (`bestiole.angleCorps`) et se déplacer (`bestiole.position`) dans l'espace de la boîte (voir Figure 3). Les valeurs de toutes les variables sont contrôlés interactivement.

Partie 2 : la caméra synthétique et utilisation de *Vertex Buffer Objects (VBOs)*

Afin de mettre en pratique l'utilisation des transformations élémentaires, on pourra remplacer l'utilisation de la fonction `LookAt()`, en gardant le même point de vue, par une série de transformations utilisant `Translate()` et `Rotate()`.

Par ailleurs, on pourra représenter le corps de la bestiole par un cylindre ou par la célèbre théière. La théière sera affichée en utilisant deux VBOs (sommets et indices) créés avec les deux tableaux définis dans le fichier déjà inclus « `inf2705-theiere.h` ». Ces VBOs doivent être définis une seule fois à l'initialisation et ensuite réutilisés à chaque affichage.

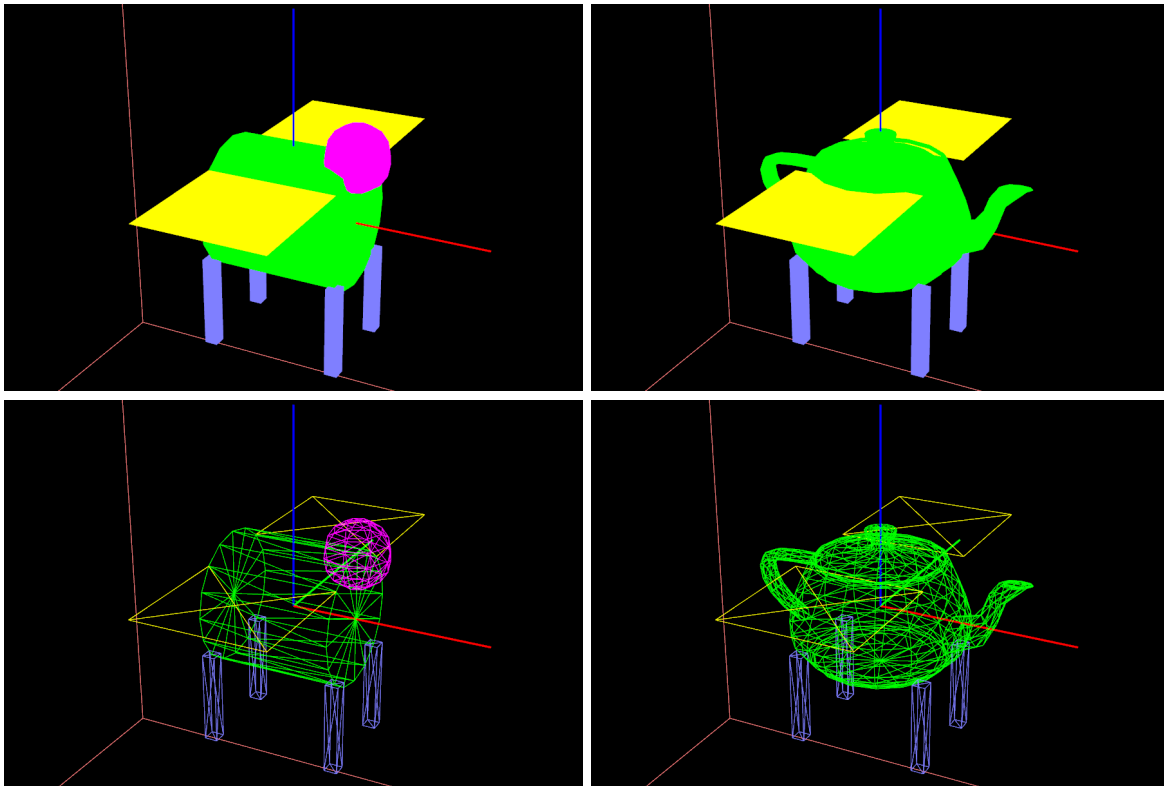


FIGURE 1 – Bestiole sous la forme d'un cylindre ou d'une théière

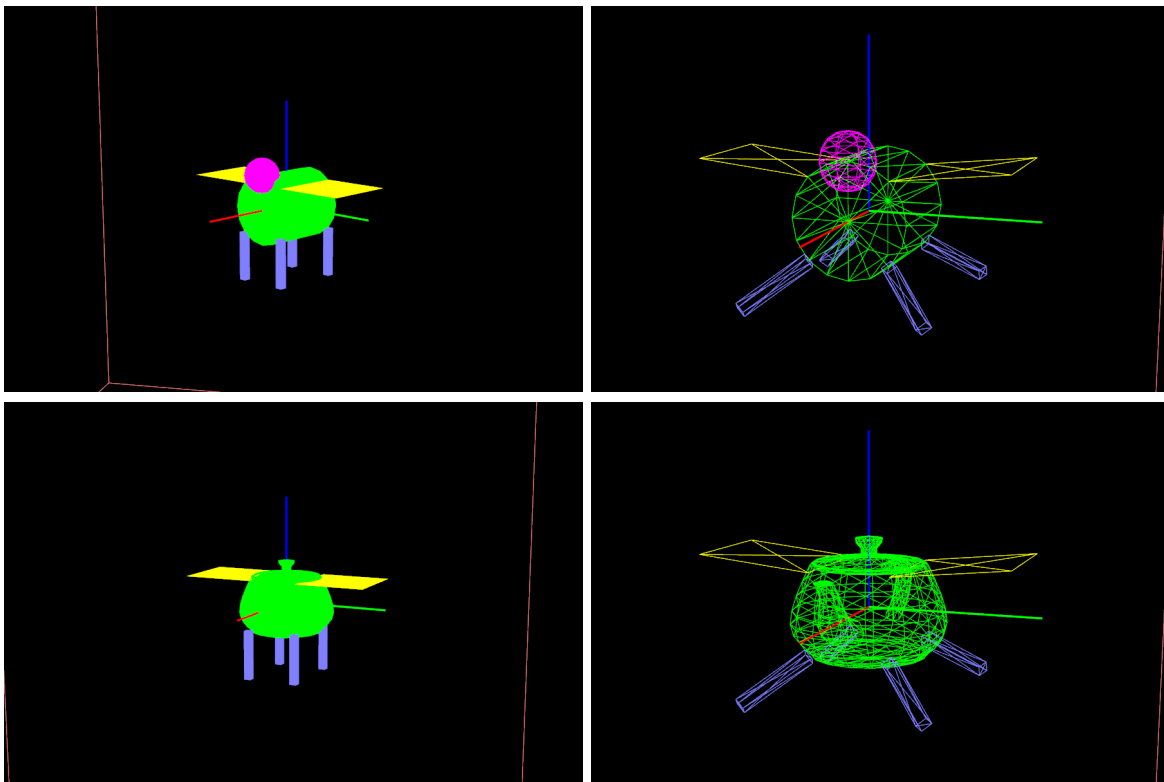


FIGURE 2 – Articulation des pattes et des ailes

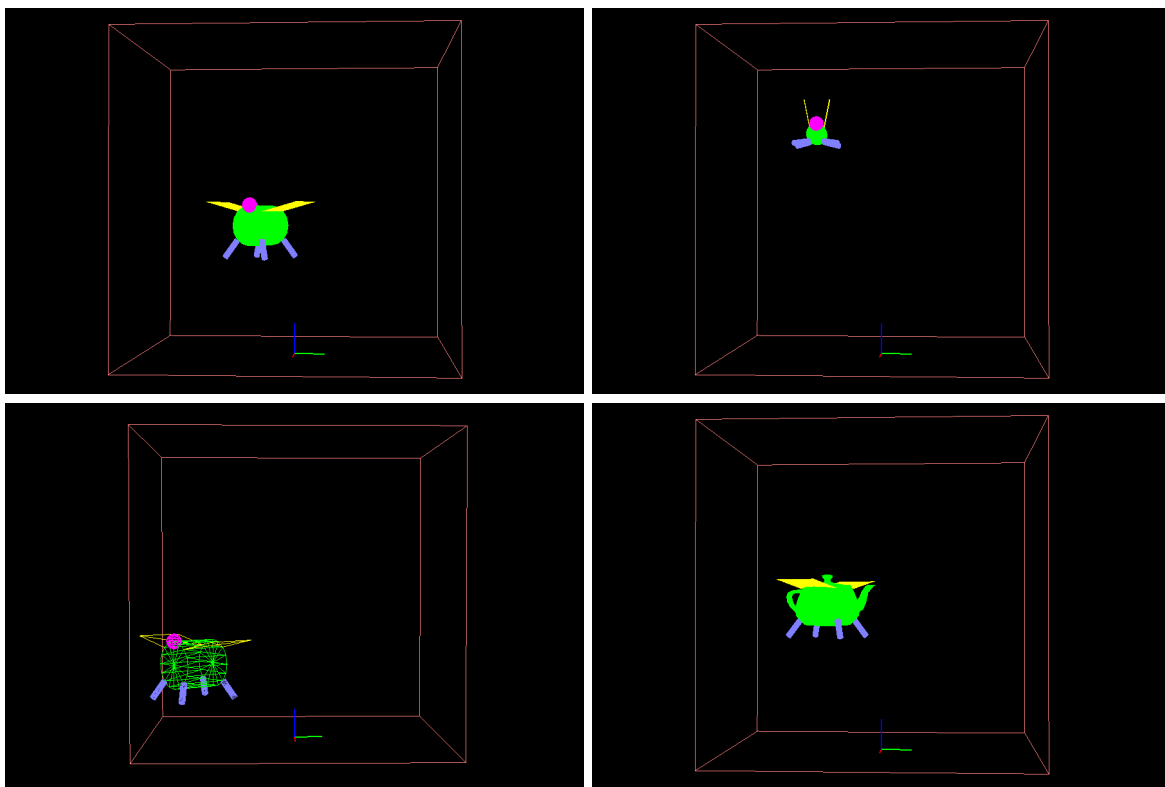


FIGURE 3 – Déplacement, rotation et articulation des pattes et des ailes en mode animation

3 Exigences

3.1 Exigences fonctionnelles

Partie 1 :

- E1. Le corps, la tête, les ailes et la pattes sont dessinés en utilisant les fonctions déjà présentes `afficherCylindre()`, `afficherSphere()` et `afficherQuad()`.
- E2. Les fonctions `Rotate()`, `Translate()` et `Scale()` sont correctement utilisées pour les transformations géométriques nécessaires au dessin de chaque partie de la bestiole.
- E3. Les fonctions `PushMatrix()` et `PopMatrix()` sont correctement utilisées pour sauvegarder l'état des matrices pour le dessin de chaque patte.
- E4. La bestiole est positionnée selon `bestiole.position` et la taille de son corps est donnée par `bestiole.taille`.
- E5. La tête de la bestiole est bien positionnée au milieu de l'arête.
- E6. La rotation du corps de la bestiole suit `angleBestiole`.
- E7. Les pattes de la bestiole sont positionnées aux arêtes du cylindre et chaque patte est de rayon `bestiole.largPatte` et de longueur `bestiole.longPatte`.
- E8. Les ailes de la bestiole sont positionnées aux arêtes du cylindre et chaque aile est carrée, d'arête `bestiole.taille`.

Partie 2 :

- E9. On peut remplacer l'utilisation de `LookAt()`, en gardant le même point de vue, par une série de transformations utilisant `Translate()` et `Rotate()`.
- E10. Les fonctions `glGenBuffers()`, `glBindBuffers()`, `glBufferData()` et `glDrawElements()` sont correctement utilisées afin d'utiliser deux VBOs (sommets et indices) pour afficher la théière. Les VBOs sont définis à l'initialisation et réutilisés à chaque affichage.
- E11. Le corps de la bestiole peut être affiché en utilisant cette théière tel qu'illustré à la Figure 1.
- E12. (Le logiciel utilise correctement les touches listées à l'annexe A pour faire varier les divers paramètres.)

3.2 Exigences non fonctionnelles

Pour la partie 1, des modifications sont principalement à faire dans les fonctions `afficher*()`. Pour la partie 2, des modifications sont principalement à faire dans les fonctions `initiliaser()`, `afficherTheiere()` et `Camera::definir()`.

3.3 Rapport

Vous devez répondre aux questions dans le fichier `Rapport.txt` qui sera inclus dans la remise. Vos réponses doivent être complètes et suffisamment détaillées. (Quelqu'un pourrait suivre les instructions que vous avez écrites sans avoir à ajouter quoi que ce soit.)

ANNEXES

A Liste des commandes

Touche	Description
q	Quitter l'application
x	Activer/désactiver l'affichage des axes
i	Réinitialiser le point de vue
g	Permuter l'affichage en fil de fer ou plein
c	Permuter l'affichage des faces arrières
l	Utiliser LookAt ou Translate+Rotate pour placer la caméra
m	Choisir le modèle affiché : cube, théière
MOINS	Reculer la caméra
PLUS	Avancer la caméra
DROITE	Déplacer la bestiole vers +X
GAUCHE	Déplacer la bestiole vers -X
PAGEPREC	Déplacer la bestiole vers +Y
PAGESUIV	Déplacer la bestiole vers -Y
BAS	Déplacer la bestiole vers +Z
HAUT	Déplacer la bestiole vers -Z
f	Diminuer la taille du corps
r	Augmenter la taille du corps
VIRGULE	Tourner la bestiole dans le sens anti-horaire
POINT	Tourner la bestiole dans le sens horaire
o	Diminuer l'angle des pattes
p	Augmenter l'angle des pattes
j	Diminuer l'angle des ailes
u	Augmenter l'angle des ailes
b	Incrémenter la dimension de la boîte
h	Décrémenter la dimension de la boîte
ESPACE	Mettre en pause ou reprendre l'animation
BOUTON GAUCHE	Déplacer (modifier angles) la caméra

B Figures supplémentaires

Allez voir la théière bien connue en infographie sur Internet :

http://www.sjbaker.org/wiki/?title=The_History_of_The_Teapot

http://en.wikipedia.org/wiki/Utah_teapot.

FIGURE 4 – La théière utilisée dans *Toy Story*FIGURE 5 – La théière utilisée dans l'épisode *Treehouse of Horror VI*



FIGURE 6 – La théière utilisée dans un écran de veille (Windows)

C Apprentissage supplémentaire

1. Quel est le nombre minimal de `PushMatrix()`/`PopMatrix()` à utiliser ? Pourquoi faut-il éviter d'en ajouter inutilement ?
2. Allonger les pattes selon la taille du corps.
3. Ajouter des ailes supplémentaires à la bestiole.
4. Utiliser un octaèdre régulier au lieu d'un cylindre.
5. Utiliser un cube au lieu d'un cylindre pour le corps.
6. Utiliser un cylindre au lieu d'un cube pour les pattes.
7. Remplacer la `LookAt()` qui positionne la caméra en orbite autour de la scène. Le VecteurUp devra être modifié en fonction de la position de l'observateur afin d'avoir l'impression que l'observateur est en orbite.
8. Si vous voulez que la tête de la bestiole se rende à une certaine position au-dessus du plan (p.e. pour y manger une miette), comment allez-vous vous y prendre pour déterminer les valeurs appropriées des angles afin de positionner la tête à cet endroit ? (C'est ce qu'on nomme la « cinématique inverse ».)
Y-a-t-il plusieurs solutions possibles ?
Y-a-t-il des solutions qu'il faut éviter ? (Faites attention à ce que la bestiole ne sorte pas de sa cage !)