



Rémy Malgouyres
Université Clermont 1
<http://laic.u-clermont1.fr/~mr/>

Synthèse d'images 3D avec OpenGL

TP n° 2 Simulation d'un robot humanoïde

Objectifs :

Le but de ce TP est de parvenir à maîtriser totalement les fonctions `glRotatef`, `glTranslatef` et `glScalef` dans le mode `GL_MODELVIEW`. On pratiquera aussi les fonctions `glPushMatrix` et `glPopMatrix`.

Le but du *TP* est de simuler un robot humanoïde avec toutes ses articulations. Nous dessinerons d'abord un doigt, puis une main, puis un avant bras, etc...

Copiez les fichiers `robotopengl.c`, `robotopengl.h`, `main.c` et `makefile` du répertoire `/users/public/M1 Info/OpenGL/DataTP2.zip`. Ajouter au fichier `main.c` le contenu de votre fonction d'affichage, les événements liés à la souris, le redimensionnement, et le traitement des caractères spéciaux (prenez ces fonctions du TP2). Testez.

Dans la fonction d'affichage, remplacez le dessin d'une théière par un appel à la fonction `DessinObjet`. Observez la fonction `main`. Un menu est chargé qui permet d'appeler les fonctions `DessineDoigt`, `DessineMain`, `DessineAvantBras`, etc.. Ces options sont testées dans `DessinObjet`, qui appelle la bonne fonction.

Observez la fonction `Clavier`. La pression de certaines touches permet de modifier les angles des articulations du robot. On appuie sur 0 pour sélectionner la partie gauche du robot, et sur 1 pour sélectionner la partie droite.

Les fonctions de dessin proprement dites sont dans le fichier `robotopengl.c`. Elles ont été vidées de leur contenu. À vous de les programmer.

Exercice 1 Dessin d'un doigt Dans la fonction `DessinePhalange`, on dessine un parallépipède rectangle (un cube avec des redimensionnement des axes). Les dimensions sont de 1 dans les directions y et z , et `longueur` dans la direction des x . On utilisera les fonction `glPushMatrix` et `glPopMatrix` pour que les redimensionnements n'affectent pas tous les dessins faits par la suite. La phalange doit être translatée de `longueur/2-0.5` dans la direction des x pour ne pas être centrée sur l'origine.

La fonction `DessineDoigt` dessine trois phalanges (voir la figure 1) :

- La première de longueur 3.
- La deuxième de longueur 2.
- La troisième de longueur 2.

La troisième translatée de 2 dans la direction des x , et est tournée d'un angle `param.angle_doigt` autour de l'axe des z (voir la structure définie dans `robotopengl.h`) par rapport à la deuxième phalange.

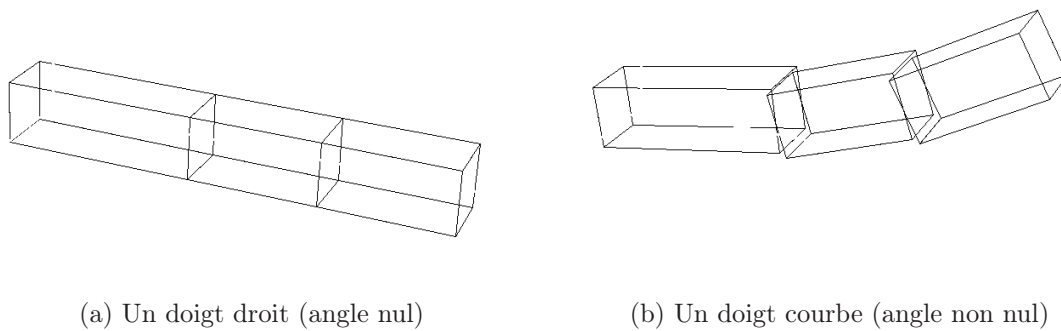


FIG. 1: Dessiner un doigt

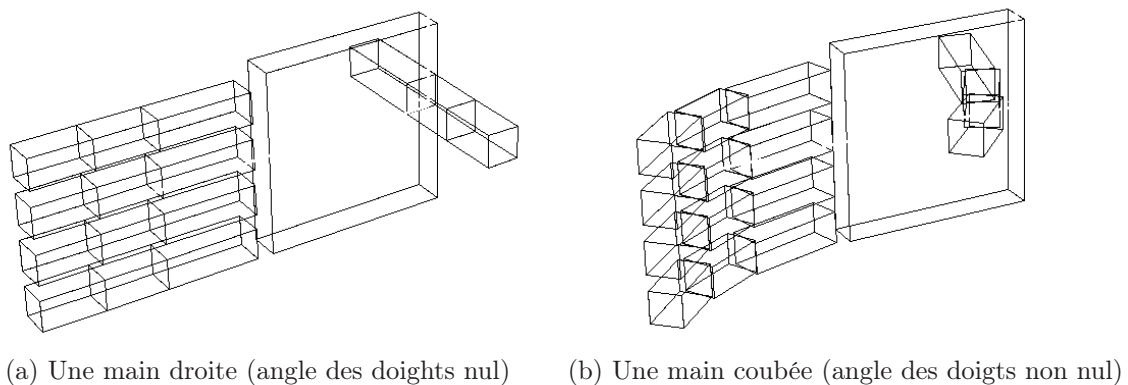


FIG. 2: La main

L'ensemble de la deuxième et la troisième phalange est translaté de 3 et tournée de `param.angle_doigt` par rapport à la première phalange.

L'ensemble des trois phalanges est tourné d'un angle `param.angle_doigt`.

Pour tester, compilez, lancez le programme, click droit, sélectionner "doigt main gauche", et observez. Pour mettre les angles à zéro, appuyez sur 'I'. Les angles du doigt sont modifiés par pression des touches 'd' et 'D'.

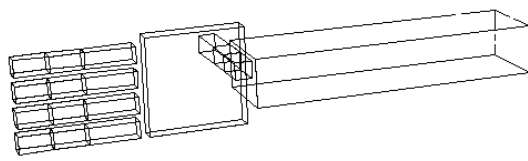
Exercice 2 Pour dessiner la main, on dessine 5 doigts et une paume (voir la figure 2).

La paume sera un parallélépipède rectangle de longueur 6 en x , 1 en y , et 6 en z . On la translatera de 3 en x pour qu'elle ne soit pas centrée en 0.

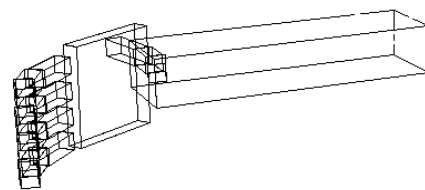
On translatera tous les doigts sauf le pouce de 4 en x , puis on les disposera en z par intervalle régulier avec des intervalles de 1,5.

Le pouce sera translaté convenablement et tourné de -90 degrés autour de l'axe des z .

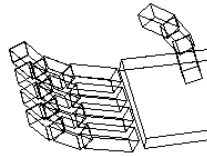
Exercice 3 L'avant bras sera composé d'un parallélépipède rectangle (de dimensions 18 en x , 3 en y et 5 en z) et de la main (voir la figure 3). La main subira une rotation d'angle `param.angle_twistmain+180` (touches 'u' et 'U') autour de l'axe des x (pour le twist de la main, et une rotation d'angle `param.angle_poignet` (touches 'p' et 'P') autour de l'axe des z . On translatera l'avant bras de 9 en x pour que le coude soit à l'origine.



(a) Un avant bras droit (angle du poignet et twist nul)

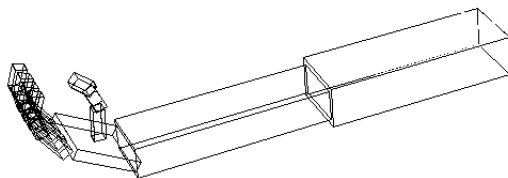


(b) Un avant bras avec un angle non nul

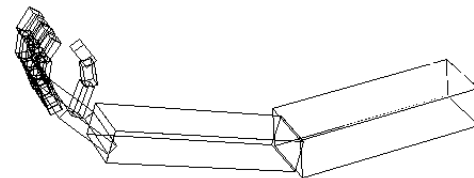


(c) Un avant br

FIG. 3: L'avant bras



(a) Un bras droit (angle du coude nul)



(b) Un bras courbé (angle du coude non nul)

FIG. 4: Le bras

Exercice 4 Le bras sera composé d'un parallélépipède rectangle (de dimensions 18 en x , 4 en y et 6 en z), et de l'avant bras (voir la figure 4). L'avant-bras subira une rotation d'angle `param.angle_twistcoude` (touches 'v' et 'V'), et une rotation d'angle `param.angle_coude` (touches 'c' et 'C'). L'ensemble sera changé d'échelle avec un facteur `FACTEUR_BRAS` (voir le `#define`). L'ensemble du bras subira une rotation d'angle `param.angle_twistepaule` (touches 'w' et 'W') autour de l'axe des y , et une rotation d'angle `param.angle_epaule` (touches 'e' et 'E') autour de l'axe des z (voir la figure 5).

Exercice 5 La jambe sera composée de 3 parallélépipèdes rectangles (voir la figure 6) :



FIG. 5: Une épaule avec un angle non nul

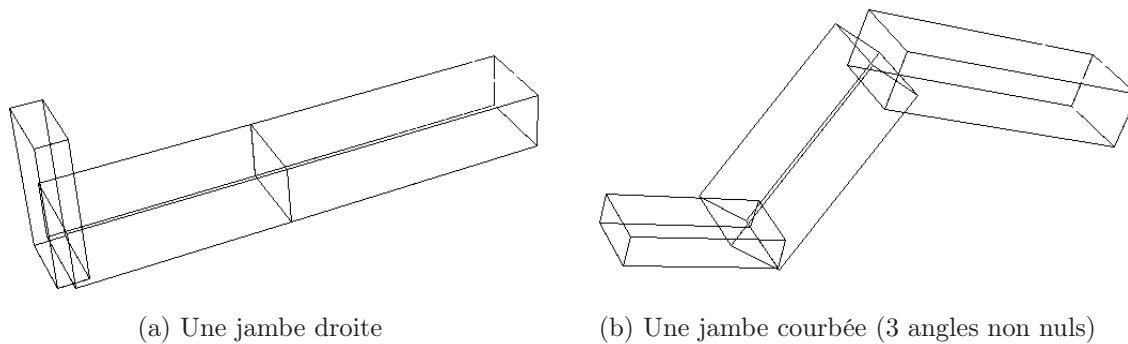


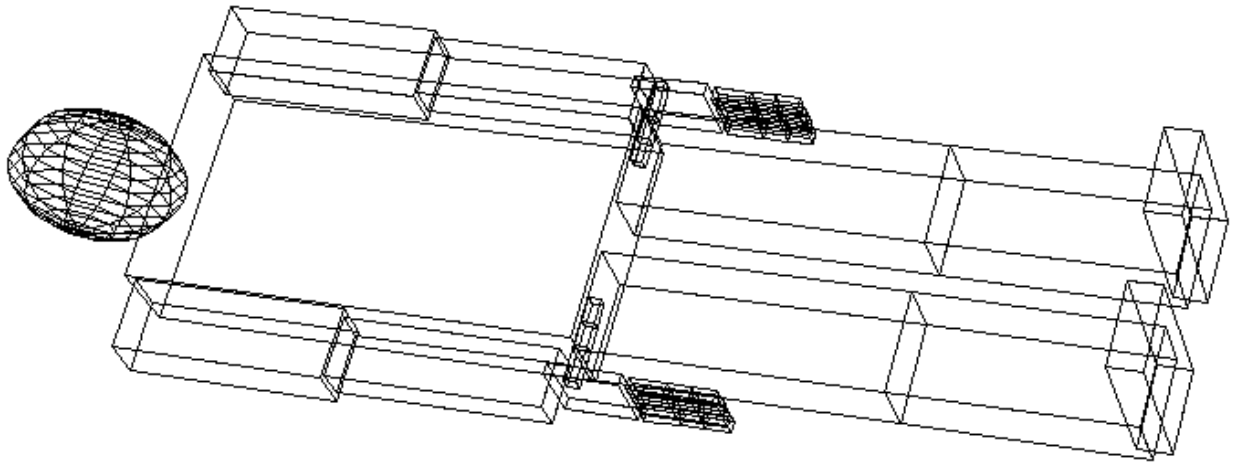
FIG. 6

- La cuisse de dimensions (18, 4, 6) ;
- Le mollet de dimensions (14, 4, 6) ;
- Le pied de dimensions (10, 2, 5).

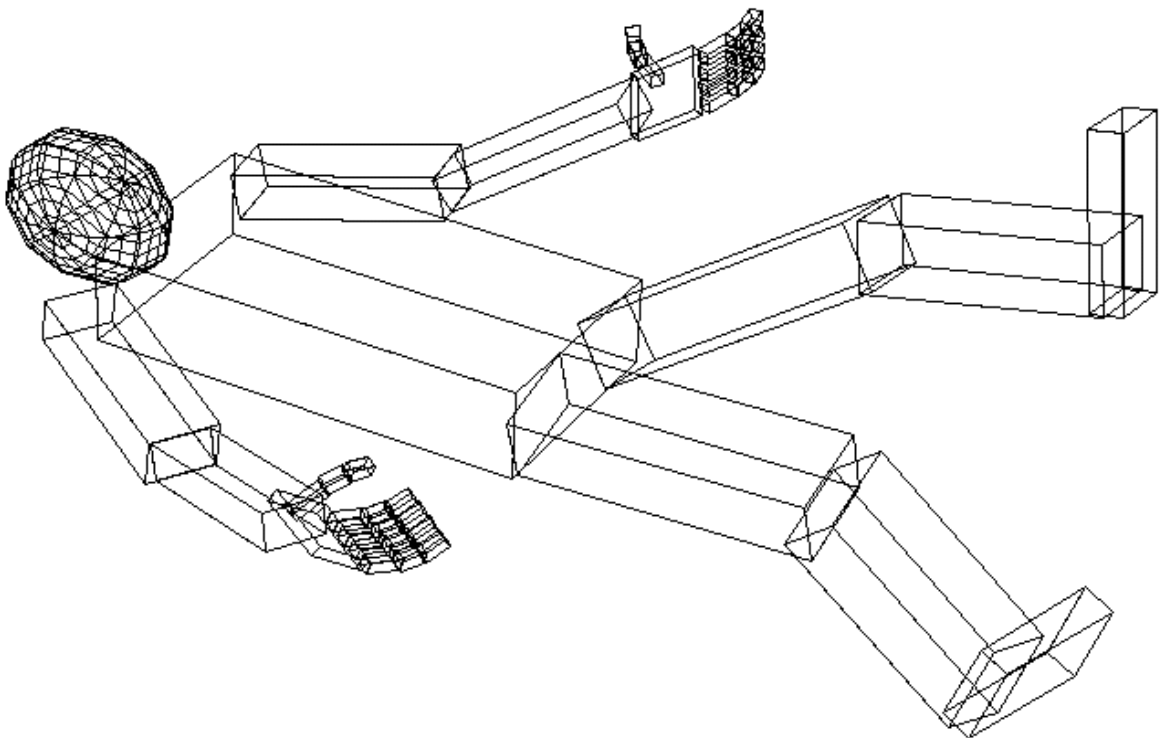
L'ensemble de la jambe subira un changement d'échelle de facteur `FACTEUR_JAMBE`. On fera subir les rotations et translations convenables à ces éléments en fonctions des trois angles `param.angle_twistjambes` (touches 'm' et 'M'), `param.angle_jambe` (touches 'j' et 'J'), `param.angle_genoux` (touches 'g' et 'G') et `param.angle_pied` (touches 'r' et 'R').

Exercice 6 L'ensemble du robot sera composé du buste (un parallélépipède rectangle de dimensions (30, 6, 16)), de deux jambes, de deux bras, et de la tête. La jambe gauche et le bras gauche subiront une symétrie (facteur d'échelle -1 sur l'axe z).

La tête sera un ellipsoïde d'axes (1.2, 1, 0.8) (facteurs d'échelle sur les axes pour une sphère). La tête sera tournée d'un angle `param.angle_tete` autour du point du cou (touches 't' et 'T').



(a) Un robot avec tous ses angle nuls



(b) Un robot avec ses angles non nuls

FIG. 7: Le robot finalisé