# Transformers - Modelling

CS 475/CS 675: Computer Graphics - Assignment 2, Part 1

Due Date: 3/10/2014

## 1 Transformers

Transformers are robots that can transform into various kinds of vehicles, like cars, planes and helicopters. They originally are a toy line by Hasbro Inc.

As an example, you can see these:

`http://www.youtube.com/watch?v=veZck1MtX5g`

The overall aim of this entire assignment is to create a short film with a tansformer of your own design. In this part of the assignment you will model the transformer.

## 2 Modelling Overview

The transformer movies have very complex transformations that you need not animate. It is much easier to animate the transformations of a toy. Instructions for such may be found in the the instructions for the toys like the one given here:

`http://tfwiki.net/wiki/File:Bumblebee-instructions.jpg`

Choose any of the transformers and create the basic robot with cuboids, or quads or any shape you want. At a minimum it must have a head, neck, torso, 2 upper arms, 2 lower arms, 2 thighs, 2 legs 2 hands and 2 feet. Other geometry is optional - though they do make your model look more realistic.

Design the heirarchical tree of transformation matrices that represent this model. Now choose the parameters that must vary to help you transform your model from a robot to a vehicle. Assign keys to vary each of these parameters interactively. Finally, assign a key that causes the entire transformation from robot to vehicle to happen, and pressing it again, causes the reverse transformation to happen.

## 3 Modeling Requirements:

1. Create a hierarchical model of a Transformer as explained above. You can choose the shape of each part to be made up of whatever primitives you want, but you have to model in OpenGL. No external tools like 3DS Max, or Maya or Blender are allowed. You will be allowed to improve your model later as you learn more things in the course, but you cannot change its basic hierarchy.

2. Learn how to use *display lists* in OpenGL for modelling. Try to encapsulate each rigid body part in a display list. Display lists are compulsory, if you are using OpenGL 2.1. If you use OpenGL 3 and above, you can use Vertex Buffers instead.

3. Your robot model should at least have the following degrees of freedom (dof):

   (a) 1 dof at the knees and elbows

   (b) 3 dof everywhere else (all the other joints specified above).

4. Remember that finally you have to animate the model - so model the hierarchy carefully and choose your parameters and structural constraints wisely. You should not change the model hierarchy from this assignment to the subsequent parts. For this reason, you have to put down (draw the tree) what is your hierarchy in your report.

5. In preparation for the animation, assign keys to vary the values of each of your model parameters, for example, you can use the *uparrow* key to make the model bend backward at the joint in the spine and the *downarrowkey* to make the model bend forward. Similarly add keys to change all joint parameters.

## 4 Rendering

1. Color the parts of the model appropriately to make it look better.

2. Also figure out how to texture the robot model with an appropriate texture.

3. Render the robot against a black background. The user should now be able to use keys and make various parts of each model move as explained above.

## 5 Use of OpenGL and GLFW

You are free to use whatever OpenGL and GLFW functions you want. No external modeler like Maya or Blender should be used. All modeling is to be done in OpenGL.

## 6 Things to avoid:

1. Do not compile and produce an *a.out*. Learn how to use a Makefile.
2. Do not write code for non-inlined functions in header files.
3. Do not write untidy code - you will lose marks if you sprinkle your code with global variables, write code that is difficult to read and is unindented or write code that is not properly structured into objects, classes and files.
4. Do not make a model that looks exactly similar to some other group's model from the class - both groups will then lose marks. This is an assignment where you have enough chance to show that all of you are original thinkers - please do not hesitate to be creative. So you are free to discuss solution strategies with your classmates but make sure that your code and your models are different.

## 7 Marking

- Modelling the robot properly as per the instructions above: 80 marks
- Using Display Lists correctly: 15 marks
- Demonstrating that various parts can be moved using the keyboard: 40 marks
- Texturing the robot: 25 marks
- Report with the hierarchical model tree : 10 marks
- Total : 170
- Deduction - I am expecting everybody to write properly formatted, indented and structured code from now on. Untidy code will be penalized.
- Late submission will follow a policy of graceful degradation with a 25% penalty for each day's delay (i.e., you get zero marks if the assignment is more than three days late after the due date.)

**TO SUBMIT:**

1. A Tar-Gzipped archive of the complete source code (and only source code). It should compile using the given Makefile on any Ubuntu system.

2. A link to a html report page on the assignment that should contain some details about what you implemented and images of some the results that you generated. Put the link in a README file in the archive you submit. Also, include all the keyboard bindings in your code that move the various parts of the robot.

3. The submission will be through the submission portal.