

3D Transforms, Viewing, Lighting, and Shading

Due: February 21, 2011 at 11:59PM

Objective: Create a 3D virtual world that tells a story. Put in an avatar/character/vehicle etc that can navigate around this environment and explore it.

Template: [assignment2.zip](#)

Scene requirements (20 points):

- Avatar: Provide a 3D hierarchical model that will interact with your environment, such as a helicopter flying around taking pictures, or a robot that walks, etc.
 - Only the top level translation and rotation need be user controlled.
 - The hierarchy should be at least 3 tiers deep.
- In the rendering window, display 'frames per second'
- In addition to the hierarchical model, the environment should be populated with 4 other objects that tell your story. The requirements for obtaining these objects are the same as assignment 1, except that at least one object should be programmed in OpenGL using the glVertex, glNormal, and glMaterial commands – this object can be very simple, like a cube or a tetrahedron. Also, include at least a quadrilateral for the floor – either obj or programmed.

Rendering requirements (30 points):

- Place both omni-directional and spotlights in the scene to appropriately light the environment. The lights should be of changeable colors (either automatically or user controlled).
- Allow the user to toggle spotlights in the scene. Also, the spotlights focus should be movable (via key presses). One of the spot lights should be attached to your avatar and can be seen in 1st person mode (like a flashlight on your head).
- Allow the user to toggle shading (FLAT and SMOOTH) using the 's' key.

Viewing and Interaction requirements (40 points):

Multiple camera views: The user should be able to toggle between a first person camera mode, a 3rd person camera follow mode, and a bird's eye view of the entire scene.

- Bird's eye view: Allow the user to manipulate the view using the mouse. This will be similar to how many interactive 3D views work in modeling software.

A. The camera's view should be centered at the center of the environment

B. If the user holds down the LEFT mouse button and moves the mouse, the camera should move about a sphere centered at a focus point (or since there really is no camera, all the objects in the scene should rotate around this point).

C. If the user uses the MOUSE WHEEL or RIGHT mouse button, the camera should zoom in and out

- First person camera mode and a 3rd person camera follow mode: Align the camera in relation to the user's avatar position. This is similar to many 3D video games (e.g., fallout 3).
 - A. The user should be able to look around the scene using the mouse and move using WASD. It is acceptable to limit the angle that you can look up or down to less than +/- 90 degrees. You can ignore the roll angle.
 - B. The user should be able to toggle between first person and third person
 - C. the user should be able to zoom out in third person view when using either the MOUSE WHEEL or hold RIGHT mouse button and drag
- Enable the user to switch between orthographic and perspective projections by pressing 'p'.

Grads: Use quaternions to compute the rotations of your hierarchical model. By pressing 'r', give the user the ability to switch between using quaternions to compute the rotation or using euler angles.

Create report: a webpage/worddoc/pdf that shows off your assignment including screenshots (10 points)

- A. Explain how to run and control your program (a list of directions/controls)
- B. Any online models you used.
- C. Screenshots and explain your project
- D. Explain the visual effect of the shading, and explain the tradeoffs between the two approaches. – 5 points
- E. Explain the technical difference between orthographic and perspective projections and relate this to how graphics are displayed differently with each approach. – 5 points

Submit to blackboard a .zip file of all the relevant parts of your assignment. Also include a precompiled .exe in your .zip file.