Computer Graphics

by Ruen-Rone Lee
ICL/ITRI

Assignment #1

Geometrical Transformations

Viewing Transformation

Projection Transforms

Requirements

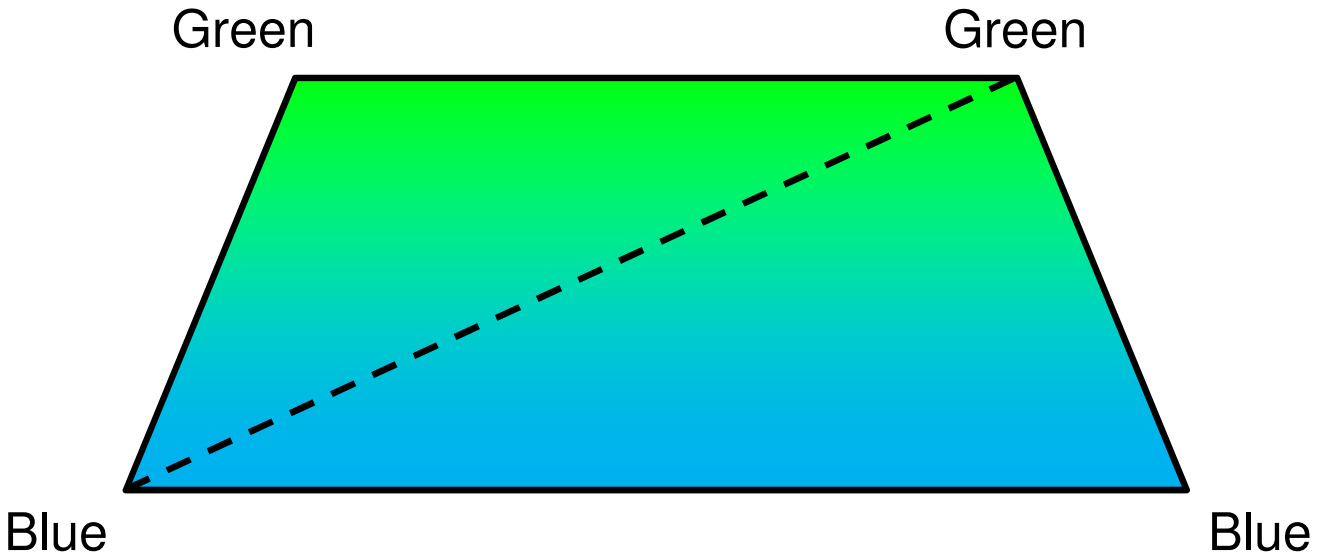
- You are required to use the framework that TA provided to do some transformations, such as geometrical, viewing, and projection, on the input 3D models.
- Interactive control is required. That is, response (re-display) immediately after any mouse or keyboard action.

Requirements

- **All the transformations (geometrical, viewing, projection) should be implemented**
  - **Geometrical transformation – translation, scaling, rotation**
  - **Viewing transformation – similar to gluLookAt function**
  - **Projection – parallel and perspective projection, similar to glOrtho and glFrustum (or gluPerspective)**
  - ***Notice: all fixed function OpenGL transformation APIs are not allowed***

Requirements

- Design a "world" that can load a specific model
- The "world" should contain a default base floor (a square or just two triangles)
  - Set the vertex colors so that the floor is displayed with the color you assigned. For example,



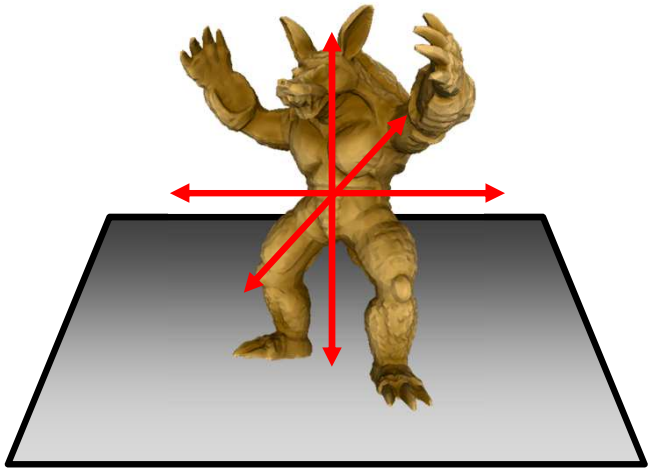
Requirements

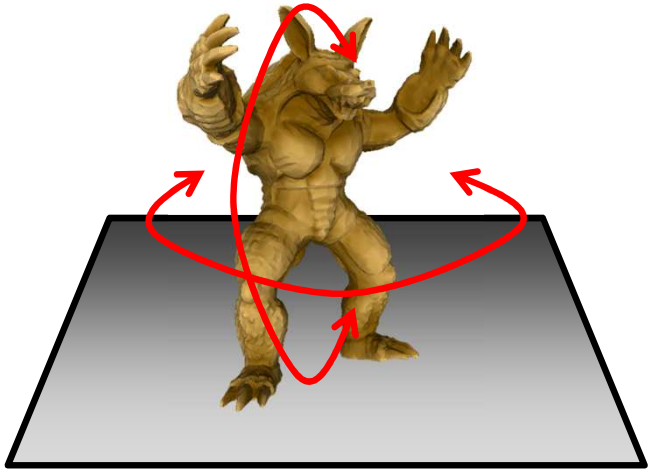
- Load a specific 3D model and display it with default viewing
  - Load a specific model as provided by TA's codes
  - Place the model properly "above" the base floor

  - Switch between different models

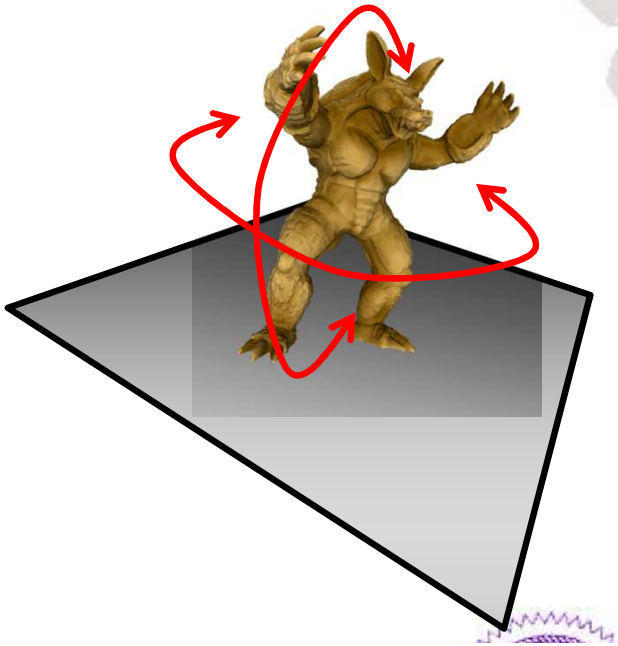
Requirements

- Set the default viewing direction to view from positive Z to origin (toward negative Z direction)
- Set the default projection to perspective projection
- Use keyboard and mouse to do all the required transformations





Geometrical Transformation



Viewing Transformation

Keyboard and Mouse Control

- **Please follow TA's instruction for the detail keyboard and mouse control specification**
- **The following 4 slides are just an example to illustrate the possible setting**
- **For example, use the following keys to switch modes and activate operations**
  - **Mode switch**
    - **"g"/"v": geometrical / viewing transformations**
    - **"o"/"p": parallel (orthographic) / perspective projection**
  - **In geometrical transformation mode**
    - **"s": scaling factors input (s_x, s_y, s_z)**
    - **"t": Translation offsets input (t_x, t_y, t_z)**
    - **"r": Rotation angles input (θ_x, θ_y, θ_z), in degrees**

Keyboard and Mouse Control (cont.)

- In viewing transformation mode
  - "e": eye coordinates input (e_x, e_y, e_z)
  - "c": center coordinates input (c_x, c_y, c_z)
  - "u": up vector input (u_x, u_y, u_z), in degrees
- In parallel or perspective projection modes
  - input (*left*, *right*, *bottom*, *top*, *near*, *far*)
- "i": Display information such as model name, mode (transformation/projection), active operation (e.g. changing scale factors (1.2, 1.2, 1.2)), etc., in the console window

Keyboard and Mouse Control (cont.)

- Use mouse buttons and wheel to adjust the values
  - Eg., in geometrical transformation mode, set transformation to translation (by pressing the key “t” first to activate the operation)
    - left mouse button down: drag horizontally for x offset; drag vertically for y offset
    - Middle wheel for z offset
  - Eg., in viewing transformation mode, switch to eye coordinates input (by pressing the key “e” to active the operation)
    - left mouse button down: drag horizontally for eye x coordinate; drag vertically for eye y coordinate
    - Middle wheel for eye z coordinate

Keyboard and Mouse Control (cont.)

- Use mouse buttons and wheel to adjust the values
  - Eg., in projection transformation mode,
    - left mouse button down: drag horizontally for left-right boundary scaling; drag vertically for bottom-top boundary scaling
    - Right mouse button down: drag horizontally for moving near clipping plane; drag vertically for moving far clipping plane

Hints for the Interactive Control

- **Follow the guidelines that TA provided to write the required transformations codes such as geometrical, viewing, and projection, based on the input controls (keyboard, mouse).**
- **Apply those generated matrices in the vertex shader codes to perform the transformation with respect to the input control**

Input Model Format

- **Wavefront 3D Graphics color models provided by TA**
  - **There are some validation 3D models released to validate your design during your code development**
  - **There are also some testing 3D models that will not be released but is used for TA to test and grade your work**
  - ***Check the TA's template for how to provide a list of 3D models in running the program***
    - **TA might change the testing sets randomly**

Hints

- Use TA's sample codes as the basic foundation to revise and add the functions required in assignment #1
- Set the viewing direction always from the eye position to the origin (assume the model has been normalized and placed at the origin)

Hints

- The base floor will be seen only when the viewing direction is not aligned with the z-axis
- For geometrical transformation, it should apply to the model only
- For viewing transformation, it should apply to both the model and the base floor

Hints

- You have to illustrate your control clearly so that TA can justified the correctness of your implementation
- You are required to demonstrate the implementation to TAs if there is a need
  - Operations incorrect
  - Insufficient documentation for the operations
  - Book the time with TAs first and show up on time

Hints

- Check with TAs if you would like to demonstrate your work personally
  - Any fancy operations or functions regarding the assignment
  - Any doubt to the score that TAs have graded
  - Again, check the time with TAs first and show up on time
    - Before class, break, or after class
    - At the CGV Lab

Due Date

- Three weeks after announcement (should be 5/9)
- Late submission is allowed with less score
- No score if you did not submit you assignment
- Plagiary is strictly forbidden
  - If you copy from others, your score will become zero
  - The score to the one who provide the original copy will also be downgraded

Submission Guide

- Please submit to course webpage at NTHU iLMS system
  - *Notice: E-mail submission will not be accepted*
- Submission should include
  - Source codes (including solution and project files)
  - Executable binary (can be run on PC/windows)
  - Documentation (explain how you did it and how to operate it)
  - *Notice: please do not submit any 3D models to save the disk space*
- Contact with TAs if you have problem in submission

Q&A