Computer Viewing

Prof. George Wolberg
Dept. of Computer Science
City College of New York
Objectives

• Introduce the mathematics of projection
• Introduce OpenGL viewing functions
• Look at alternate viewing APIs
Viewing Process

camera

tripod

view volume

model
Computer Viewing

• There are three aspects of the viewing process, all of which are implemented in a pipeline,
  - Positioning the camera
    • Setting the model-view matrix
  - Selecting a lens
    • Setting the projection matrix
  - Clipping
    • Setting the view volume
Transformation Pipeline

• Transformations take us from one “space” to another
  - All of our transforms are $4 \times 4$ matrices
Transformations

• Modeling transformations
  - move models into world coordinate system

• Viewing transformations
  - define position and orientation of the camera

• Projection transformations
  - adjust the lens of the camera; define view volume

• Viewport transformations
  - enlarge or reduce the physical photograph
The Model Coordinate System

The X,Y,Z coordinates of the model's vertices are defined relative to the object’s center, where (0,0,0) is the center of the object.
The model is moved to a new position, and possibly included with other models, in the world coordinate system.
The View Coordinate System

The vertices expressed in the world coordinate system must be transformed into the view coordinate system since they are now relative to the camera.
Model-View Transformation

- A 4x4 matrix transforms vertices from the model to the world coordinate system.
- A second 4x4 matrix maps the world to the view coordinate system.
- The product of these two matrices is called the model-view matrix.
- It maps the object from the original model coordinate system directly to the camera’s (viewer’s) coordinate system.
The World and Camera Frames

- Changes in frame are defined by 4 x 4 matrices
- In OpenGL, we start with the world frame
- We move models from the world frame to the camera frame by using the model-view matrix $M$
- Initially these frames are the same ($M = I$)
- If you want to move the camera three units to the right ($+x$), this is achieved by moving the objects three units to the left ($-x$).
- Camera always stays at the origin and points in the negative z direction
Moving the Objects

Move objects back (along \(-z\) direction) to view it in front of camera, which is at origin.

\[
M = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & -d \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]
Viewing Transformations

• Position the camera/eye in the scene
  - place the tripod down; aim camera

• To “fly through” a scene
  - change viewing transformation and redraw scene

• Simple interface:

\[
\text{LookAt}( \text{eyex, eyey, eyez, atx, aty, atz, upx, upy, upz})
\]

  - up vector determines unique orientation
  - careful of degenerate positions
LookAt

LookAt(\( \text{eye}, \text{at}, \text{up} \))
Creating the LookAt Matrix

\[ \hat{n} = \frac{\vec{at-\text{eye}}}{\lVert \vec{at-\text{eye}} \rVert} \]
\[ \hat{u} = \frac{\hat{n} \times \vec{up}}{\lVert \hat{n} \times \vec{up} \rVert} \]
\[ \hat{v} = \hat{u} \times \hat{n} \]

\[ \begin{pmatrix}
    u_x & u_y & u_z & - (\text{eye} \cdot \hat{u}) \\
    v_x & v_y & v_z & - (\text{eye} \cdot \hat{v}) \\
    -n_x & -n_y & -n_z & - (\text{eye} \cdot \hat{n}) \\
    0 & 0 & 0 & 1
\end{pmatrix} \]
Specifying What You Can See (1)

- Once camera is positioned in scene, we must set up a viewing frustum (view volume) to specify how much of the world we can see.
- Done in two steps:
  - Specify the size of the frustum (projection transform).
  - Specify its location in space (model-view transform).
- Anything outside of viewing frustum is clipped:
  - Primitive is either modified or discarded (if entirely outside frustum).
Specifying What You Can See (2)

• OpenGL projection model uses eye coordinates
  - the “eye” is located at the origin
  - looking down the -z axis

• Projection matrices use a six-plane model:
  - near (image) plane and far (infinite) plane
    • both are distances from the eye (positive values)
  - enclosing planes
    • top & bottom, left & right
Specifying What You Can See (3)

Orthographic View

Perspective View

\[ O = \begin{pmatrix} \frac{2}{r-l} & 0 & 0 & \frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & \frac{t+b}{t-b} \\ 0 & 0 & -\frac{2}{f-n} & \frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{pmatrix} \]

\[ P = \begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & -\frac{(f+n)}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix} \]
Perspective Projection