Preparation for Quiz2
Classification of Projections

Perspective Projection

Parallel Projection
Oblique Projections

• Projectors **not** orthogonal to projection plane
  – The projection plane is typically parallel to a face of the object

• Classified by the angle between projector and plane
  • $\pi/4$: **Cavalier type**
    – **Preserves** the lengths of edges orthogonal to projection plane
  • Arctan(2): **Cabinet type**
    – **Halves** the lengths of edges orthogonal to projection plane

Cavalier: 45 degree

Cabinet: $\arctan(2) = 63.4$ degree
Examples of Parallel Projections

- **cavalier**
- **multiview orthographic**
- **cabinet**
Perspective Projection

eye, or Center of Projection (COP)

projectors

picture plane

A'

A

projectors
Vanishing Points Examples

Frankfurt airport [ Wikimedia ]

[ Shutterstock ]
Types of Perspective Projections

• Based on number of vanishing points for lines parallel to the three coordinate axes
  – Determined by # of axes parallel to the viewing plane

A unit cube:

One-point Perspective
(view plane parallel to 2 axes)

Two-point Perspective
(view plane parallel to 1 axis)

Three-point Perspective
(view plane not parallel to any axis)
Camera Coordinate System

- Summary
  - Three axes, computed from look vector \( L \) and up vector \( U \):

  \[
  n = \frac{-L}{|L|} \\
  v = \frac{U - (U \cdot n)n}{|U - (U \cdot n)n|} \\
  u = v \times n
  \]

  - \( u, v, n \) form a camera coordinate system
Step 1
Step 1
Step 1

[Diagram showing a computer projection process with axes and clipping planes.]
Step 2

• Some preparations
  – First, make width/height angles to be $\pi/2$
  – Non-uniform scaling in X,Y coordinates

A look down the X axis
Step 2 (cont)

- Where we are now:

A look down the X axis (same picture when looking down Y)
Putting Together

- Translation: \( T \)
- Rotation: \( R \)
- Scaling: \( S_{xy}, S_{xyz} \)
- Perspective transformation: \( D \)

\[
\begin{align*}
(T, R, S_{xy}, S_{xyz}, D)
\end{align*}
\]
Putting Together

- Complete viewing transformation to bring a point $\mathbf{q}$ to the canonical volume:

$$ q' = D S_{xyz} S_{xy} R T q $$
Cohen-Sutherland Line Clipping in 2D

- Divide plane into 9 regions
- Compute the sign bit of 4 comparisons between a vertex and an edge
  - $y_{\text{max}} - y; \ y - y_{\text{min}}; \ x_{\text{max}} - x; \ x - x_{\text{min}}$
  - point lies inside only if all four sign bits are 0, otherwise exceeds edge
- 4 bit outcode records results of four bounds tests:
  - First bit: above top edge
  - Second bit: below bottom edge
  - Third bit: to the right of right edge
  - Fourth bit: to the left of left edge
- Compute outcodes for both vertices of each line (denoted OC$_0$ and OC$_1$)
- Lines with OC$_0$ = 0 (i.e., 0000) and OC$_1$ = 0 can be trivially accepted.
- Lines lying entirely in a half plane outside an edge can be trivially rejected if (OC$_0$ AND OC$_1$) $\neq$ 0 (i.e., they share an “outside” bit)
Cyrus-Beck/Liang-Barsky Parametric Line Clipping (3/3)

- Eliminate $t$’s outside $[0,1]$ on the line
- Which remaining $t$’s produce interior intersections?
- Can’t just take the innermost $t$ values!
- Move from $P_0$ to $P_1$; for a given edge, just before crossing:
  - if $N_i \cdot D < 0 \Rightarrow$ Potentially Entering (PE);
  - if $N_i \cdot D > 0 \Rightarrow$ Potentially Leaving (PL)
- Pick inner PE/PL pair: $t_E$ for $P_{PE}$ with
  max $t$, $t_L$ for $P_{PL}$ with min $t$, and
  $t_E > 0, t_L < 1$
- If $t_L < t_E$, no intersection