Module III

Window to viewport transformation

Total Slide

Ankit Garg
Assistant Professor
Amity University, Haryana
Window to viewport transformation

- Window port
- Viewport
- Window to Viewport mapping is required
- Some transformations are required
- Transformation can Translation, rotation, scaling and shearing
Window to viewport transformation

• **Window-to-Viewport mapping** is the process of mapping or transforming a two-dimensional, world-coordinate scene to device coordinates. In particular, objects inside the world or clipping window are mapped to the viewport. The viewport is rectangular area on screen where world coordinates are mapped to be displayed.

• In other words, the clipping window is used to select the part of the scene that is to be displayed. The viewport is used to display selected portion of window on the output device.
Window to viewport transformation
Window to viewport transformation

• **Window port:** A world coordinate area selected for display.

• **View port:** This is a rectangular region of the screen which is selected for displaying the object. In other words we can say that view port is part of computer screen.

• **Window to Viewport Mapping:** Mapping of a part of a world coordinate scene to device coordinate is referred to as a viewing transformation.

• **Window to viewport transformation** requires more than one transformation.
Concept of window to viewport transformation

- It may be possible that size of viewport is smaller than size of window or greater than size of window.
- In this case we have to expand or decrease size of window according to the size of viewport.
- In this concept same mapping is required to convert size of window in size of viewport.
- Some mathematical computations are required to map window and viewport.
Window to Viewport Mapping

\[(x, y)\]

\[(wxmin, wymin)\]

\[(vxmin, vymin)\]

\[(wxmax, wymax)\]

\[(vxmax, vymax)\]

\[(u, v)\]
Steps for window to viewport transformation

• **Step 1:** Translate window towards origin

To shift window towards origin, lower left or upper left corner of window will become (-). Hence translation factor will become negative (-tx,-ty).

(-wxL,-wyL) – When origin is lower left corner of the screen.

(-wxL, -wyH)- When origin is upper left corner of window.
Steps for window to viewport transformation (cont..)

- **Step 2:** Resize window to the size of view port.

To convert window size into view port size following computation is required.

\[
S_x = \frac{X_V \text{ max} - X_V \text{ min}}{X_W \text{ max} - X_W \text{ min}}
\]

\[
S_y = \frac{Y_V \text{ max} - Y_V \text{ min}}{Y_W \text{ max} - Y_W \text{ min}}
\]
Steps for window to viewport transformation (cont..)

• **Step 3:** Translate window (position of window must be same as position of view port.

• If lower left corner of viewport is (0,0) we don’t need to take step 3 because window lower left corner is already shifted on origin after taking first step.

• If lower left corner is not (0,0) we have to take translation factor (+).
Steps for window to viewport transformation

1. Window in world coordinates
2. Window translated to origin
3. Window scaled to size of viewport
4. Translated by \((u_{\text{min}}, v_{\text{min}})\) to final position
Steps for window to viewport transformation

\[ T(-x_{\text{min}}, -y_{\text{min}}) \quad S\left(\frac{u_{\text{max}} - u_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}, \frac{v_{\text{max}} - v_{\text{min}}}{y_{\text{max}} - y_{\text{min}}}\right) \quad T(u_{\text{min}}, v_{\text{min}}) \]

\[ M_{\text{wv}} = T(u_{\text{min}}, v_{\text{min}}) \cdot S\left(\frac{u_{\text{max}} - u_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}, \frac{v_{\text{max}} - v_{\text{min}}}{y_{\text{max}} - y_{\text{min}}}\right) \cdot T(-x_{\text{min}}, -y_{\text{min}}) \]

\[
\begin{bmatrix}
1 & 0 & u_{\text{min}} \\
0 & 1 & v_{\text{min}} \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
\frac{u_{\text{max}} - u_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} & 0 & 0 \\
0 & \frac{v_{\text{max}} - v_{\text{min}}}{y_{\text{max}} - y_{\text{min}}} & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
1 & 0 & -x_{\text{min}} \\
0 & 1 & -y_{\text{min}} \\
0 & 0 & 1
\end{bmatrix}
\]
Example

- Let us consider an example of view transformation.

- If our window has coordinates (10,10), (20,10), (20,20), (10,20)
- Let our viewport coordinates are (.5,.5), (1,.5),(1,1), (.5,1)
- **Take step 1**: Translate window to origin
- In this case translation matrix will become

\[
\begin{pmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
-10 & -10 & 1 \\
\end{pmatrix}
\]

- **Take step 2**: convert size of window to viewport size
- \(sx=(1-0.5)/(11-1) \Rightarrow 0.5/10 \Rightarrow 0.05\)
- \(sy=(1-0.5)/(11-1) \Rightarrow 0.5/10 \Rightarrow 0.05\)
Example

- so scaling transformation matrix will be
  \[
  \begin{bmatrix}
  0.05 & 0 & 0 \\
  0 & 0.05 & 0 \\
  0 & 0 & 1 \\
  \end{bmatrix}
  \]

- Step 3: Finally window to position of view port
  \[
  \begin{bmatrix}
  1 & 0 & 0 \\
  0 & 1 & 0 \\
  0.5 & 0.5 & 1 \\
  \end{bmatrix}
  \]
Composition of transformations

\[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
-1 & -1 & 1
\end{bmatrix}
\begin{bmatrix}
.05 & 0 & 0 \\
0 & .05 & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
.5 & .5 & 1
\end{bmatrix}
\]

- Translation toward origin
- Scaling
- Translation further from origin
QUESTION

• Window coordinates are (0,0), (2,0), (2,2), (0,2).
• View port coordinates are (0,0), (4,0), (4,4), (0,4).

1. Find out relative position of coordinate (1,1) of window in viewport.

2. Calculate scaling factor for this transformation.

3. Write down all three transformation matrix for this viewing transformation.
Formula to find out scaling factor

\[ S_x = \frac{X_V \text{ max} - X_V \text{ min}}{X_W \text{ max} - X_W \text{ min}} \]
\[ S_y = \frac{Y_V \text{ max} - Y_V \text{ min}}{Y_W \text{ max} - Y_W \text{ min}} \]

Sx = width of viewport / width of window
Sy = height of viewport / height of window

Formula to find relative position

\[ x_v = x_{v \text{ min}} + (x_w - x_{w \text{ min}})s_x \]
\[ y_v = y_{v \text{ min}} + (y_w - y_{w \text{ min}})s_y \]
Program on Window to Viewport Transformation
THANKS