

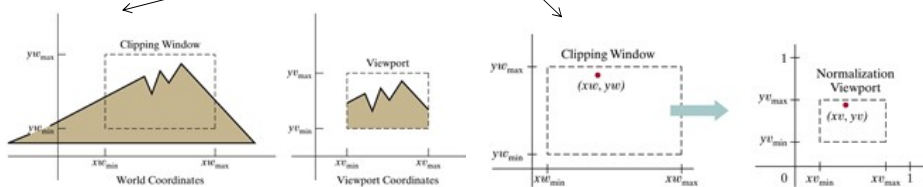
## 2D Viewing

- ❑ We specify a rectangular area (clipping window) in the **world coordinates** and a viewport in the **device coordinates** on the display
  - window defines what to appear
  - viewport defines where to display
- ❑ The mapping of the window (world coordinates) to viewport (device coordinates) is a 2D **viewing transformation**

1

## 2D Viewing

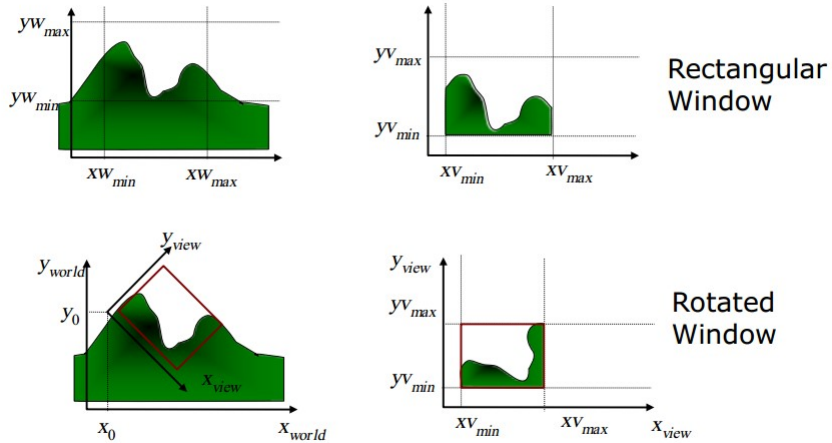
`gluOrtho2D(xwmin, xwmax, ywmin, ywmax)`



`glViewport(xvmin, yvmin, xvwidth, yvheight)`

2

## The clipping window

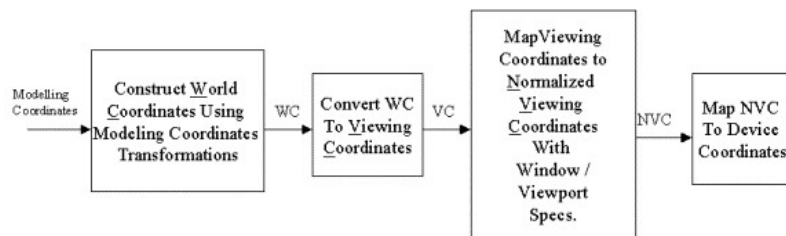


3

## 2D Viewing Pipeline

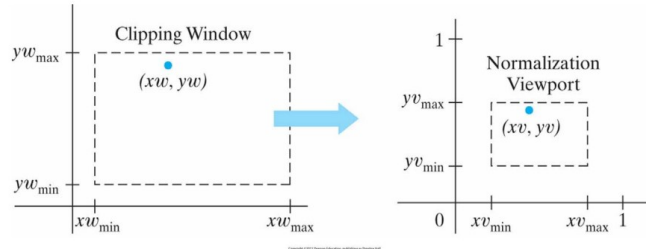
### □ 2D viewing pipeline

- Construct world-coordinate scene using modeling-coordinate transformations
- Convert world-coordinates to viewing coordinates
- Transform viewing-coordinates to normalized-coordinates (ex: between 0 and 1, or between -1 and 1)
- Map normalized-coordinates to device-coordinates.



4

**Figure 8-6** A point  $(xw, yw)$  in a world-coordinate clipping window is mapped to viewport coordinates  $(xv, yv)$ , within a unit square, so that the relative positions of the two points in their respective rectangles are the same.



1. Scale the clipping window to the size of viewport using a fixed point  $(xw_{min}, yw_{min})$

$$S = \begin{bmatrix} s_x & 0 & xw_{min}(1 - s_x) \\ 0 & s_y & yw_{min}(1 - s_y) \\ 0 & 0 & 1 \end{bmatrix}$$

2. Translate  $(xw_{min}, yw_{min})$  to  $(xv_{min}, yv_{min})$

$$T = \begin{bmatrix} 1 & 0 & xv_{min} - xw_{min} \\ 0 & 1 & yv_{min} - yw_{min} \\ 0 & 0 & 1 \end{bmatrix}$$

$$M = T \cdot S = \begin{bmatrix} s_x & 0 & \frac{xw_{max}xv_{min} - xw_{min}xv_{max}}{xw_{max} - xw_{min}} \\ 0 & s_y & \frac{yw_{max}yv_{min} - yw_{min}yv_{max}}{yw_{max} - yw_{min}} \\ 0 & 0 & 1 \end{bmatrix}$$

## OpenGL 2D viewing functions

### ❑ Projection mode

- glMatrixMode (GL\_PROJECTION)
- glLoadIdentity()

### ❑ GLU Clipping window function

- Orthogonal projection  
gluOrtho2D  $(xw_{min}, xw_{max}, yw_{min}, yw_{max})$
- If we do not specify a clipping window, the default coordinates are  
 $(xw_{min}, yw_{min}) = (-1.0, -1.0)$  and  $(xw_{max}, yw_{max}) = (1.0, 1.0)$

# OpenGL 2D viewing functions

## □ Viewport

➤ `glViewport (xvmin, yvmin, vpwidth, vpHeight)`

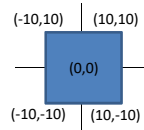
default is size of the display window

- ❖  $x_{vmin}$  and  $y_{vmin}$  are the positions of the lower-left corner of the viewport
- ❖  $vp_{width}$ ,  $vp_{Height}$  are the pixel width and height of the viewport
- ❖ If we do not use `glViewport`, the default viewport size and position are the same as the display window

7

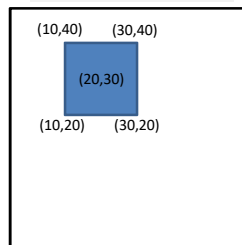
## Example: 2D Viewing

Modeling Coordinates



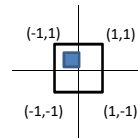
`glTranslatef (20, 30, 0);`  
`DrawSquare ();`

World Coordinates

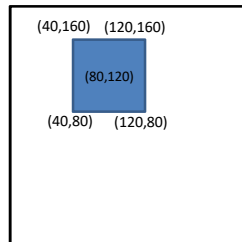


`gluOrth2D (0, 50, 0, 50)`

Normalized Coordinates



Device Coordinates

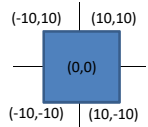


`glViewport (0, 0, 200, 200)`

8

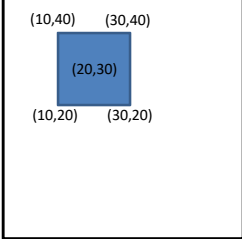
# Example: 2D Viewing

Modeling Coordinates



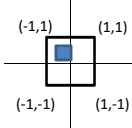
```
glTranslatef (20, 30, 0);  
DrawSquare ();
```

World Coordinates

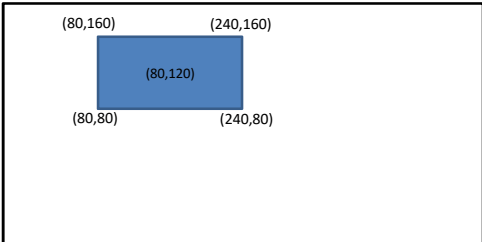


```
gluOrth2D (0, 50, 0, 50)
```

Normalized Coordinates



Device Coordinates



```
glViewport (0, 0, 400, 200)
```