

Blending & Transparency

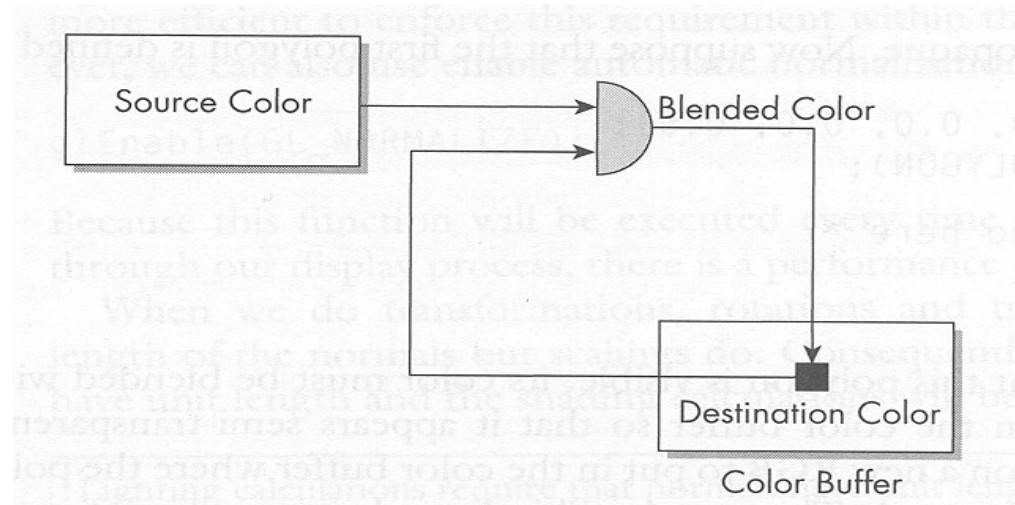
**CSCI 4229/5229
Computer Graphics
Summer 2025**

Blending Pixels

- Color (R,G,B, α) (4x8 bits = 32 bit color)
 - α blending
 - $R_c = \alpha R_a + (1-\alpha)R_b$
 - $G_c = \alpha G_a + (1-\alpha)G_b$
 - $B_c = \alpha B_a + (1-\alpha)B_b$
- Uses
 - Transparency (1=opaque, 0=invisible)
 - Anti-aliasing
 - Transitions

Blending in OpenGL

- glEnable(GL_BLEND)
- glBlendFunc(source,destination)
 - source (what we're drawing) β
 - destination (what's there already) γ
 - $R_c = \beta R_a + \gamma R_b$
 - $G_c = \beta G_a + \gamma G_b$
 - $B_c = \beta B_a + \gamma B_b$
 - $\alpha_c = \beta \alpha_a + \gamma \alpha_b$
- In general $\beta + \gamma \neq 1$



Source Factors (β)

- GL_ZERO
- GL_ONE
- GL_DST_COLOR
- GL_ONE_MINUS_DST_COLOR
- GL_SRC_ALPHA
- GL_ONE_MINUS_SRC_COLOR
- GL_DST_ALPHA
- GL_ONE_MINUS_DST_ALPHA
- GL_SRC_ALPHA_SATURATE

Destination Factors (γ)

- GL_ZERO
- GL_ONE
- GL_SRC_COLOR
- GL_ONE_MINUS_SRC_COLOR
- GL_SRC_ALPHA
- GL_ONE_MINUS_SRC_ALPHA
- GL_DST_ALPHA
- GL_ONE_MINUS_DST_ALPHA

Blending Operations

- $GL_ZERO = (0,0,0,0)$
- $GL_ONE = (1,1,1,1)$
- $GL_SRC_COLOR = (R,G,B,\alpha)_S$
- $GL_DST_COLOR = (R,G,B,\alpha)_D$
- $GL_ONE_MINUS_SRC_COLOR = (1, 1,1,1) - (R,G,B,\alpha)_S$
- $GL_ONE_MINUS_DST_COLOR = (1,1,1,1) - (R,G,B,\alpha)_D$
- $GL_ONE_MINUS_SRC_ALPHA = (1,1,1,1) - (\alpha,\alpha,\alpha,\alpha)_S$
- $GL_ONE_MINUS_DST_ALPHA = (1,1,1,1) - (\alpha,\alpha,\alpha,\alpha)_D$
- $GL_SRC_ALPHA = (\alpha,\alpha,\alpha,\alpha)_S$
- $GL_DST_ALPHA = (\alpha,\alpha,\alpha,\alpha)_D$
- $GL_SRC_ALPHA_SATURATE = (f,f,f,1) \text{ } f = \min(\alpha_S, 1 - \alpha_D)$

Mixing Objects

- First draw opaque objects
 - Make Z-buffer writable (`glDepthMask(1)`)
 - Set $\alpha=1$ (but may not matter)
- Next draw translucent objects
 - Make Z-buffer readonly (`glDepthMask(0)`)
 - Set $\alpha<1$
 - `glBlendFunction(GL_SRC_ALPHA, GL_ONE)`
- Order (mostly) doesn't matter