# Texture Shading

CS418 Computer Graphics

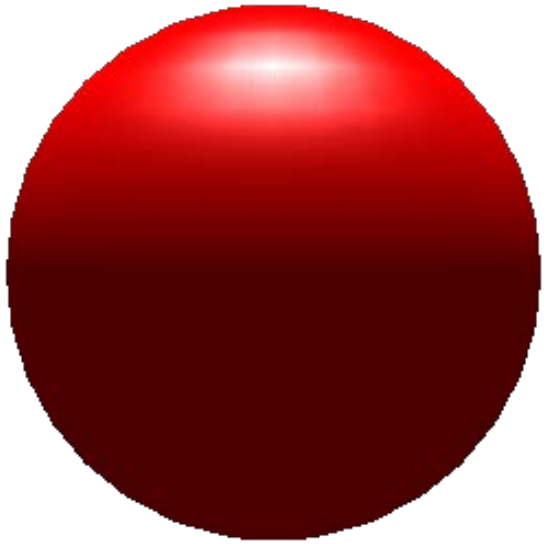
John C. Hart

# Phong Lighting Equation

$$I = k_a L_a + k_d L_d (N \cdot L) + k_s L_s (V \cdot R)^n$$

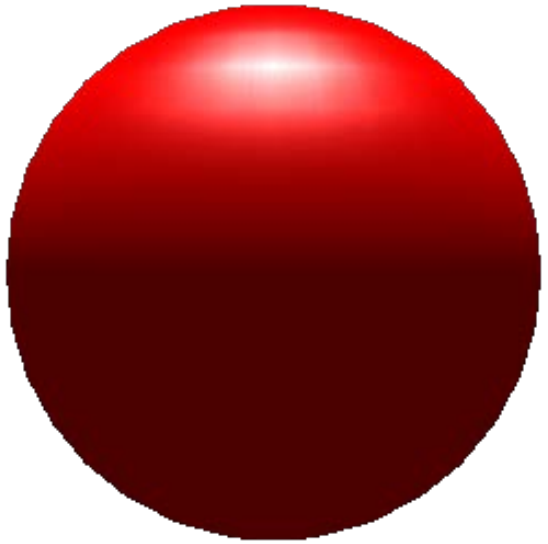
$$I = k_a L_a + k_d L_d (N \cdot L) + k_s L_s (V \cdot R)^n$$

$$I = k_a L_a + k_d L_d (N \cdot L) + k_s L_s (V \cdot R)^n$$



# Texture Coordinates

$$I = k_a L_a + k_d L_d (N \cdot L) + k_s L_s (V \cdot R)^n$$
$$I = k_a L_a + k_d L_d (N \cdot L) + k_s L_s (V \cdot R)^n$$
$$I = k_a L_a + k_d L_d (N \cdot L) + k_s L_s (V \cdot R)^n$$

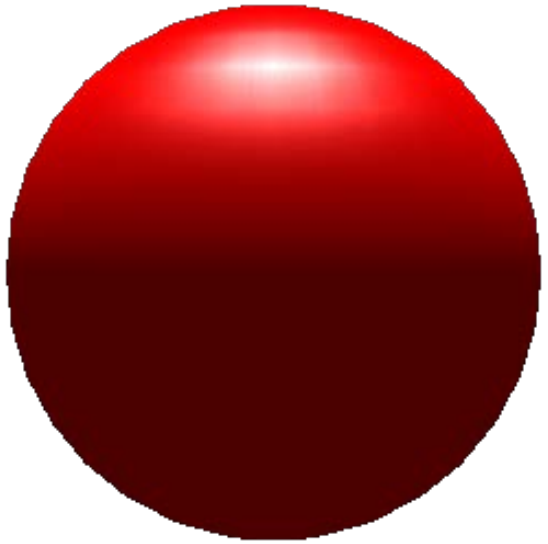


- Set texcoords of vertices to:

$$s = N \cdot L, \quad t = V \cdot R$$

# Texture Coordinates

$$I = k_a L_a + k_d L_d (N \cdot L) + k_s L_s (V \cdot R)^n$$
$$I = k_a L_a + k_d L_d (N \cdot L) + k_s L_s (V \cdot R)^n$$
$$I = k_a L_a + k_d L_d (N \cdot L) + k_s L_s (V \cdot R)^n$$

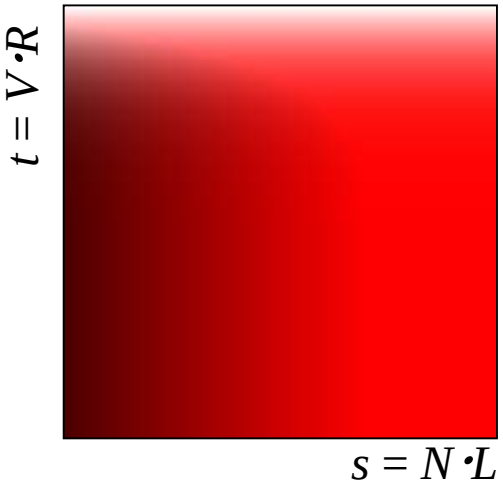


- Set texcoords of vertices to:

$$s = N \cdot L, \quad t = V \cdot R$$

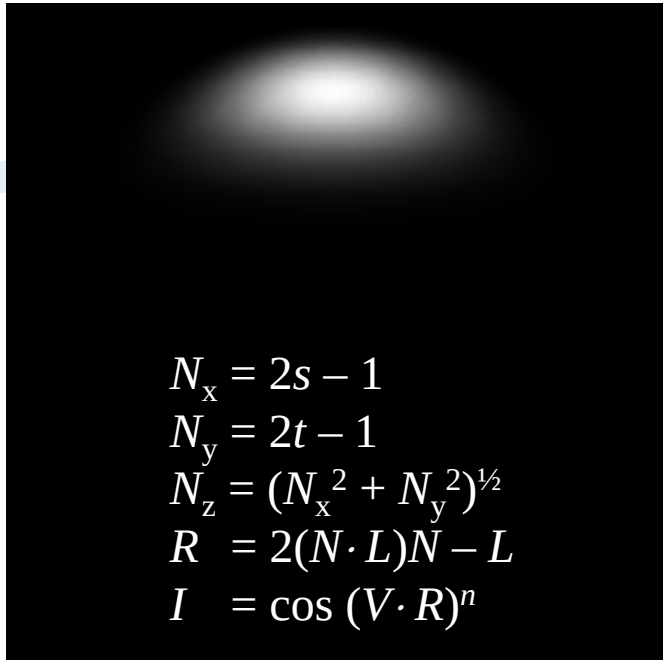
- Create a texture with colors:

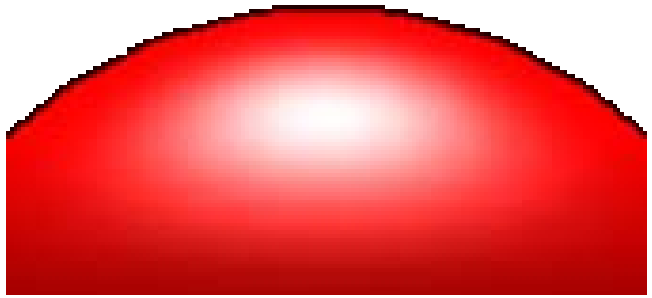
$$C(s,t) = k_a L_a + k_d L_d s + k_s L_s t^n$$
for all $0 \le s,t \le 1$

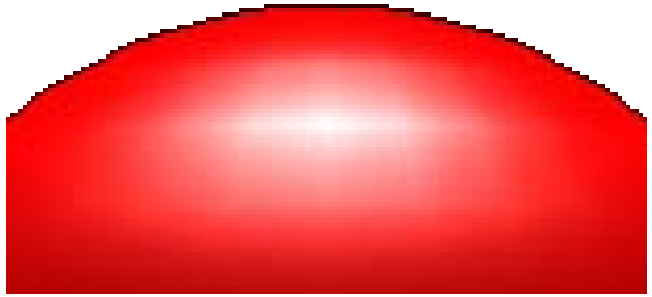


# Phong Map

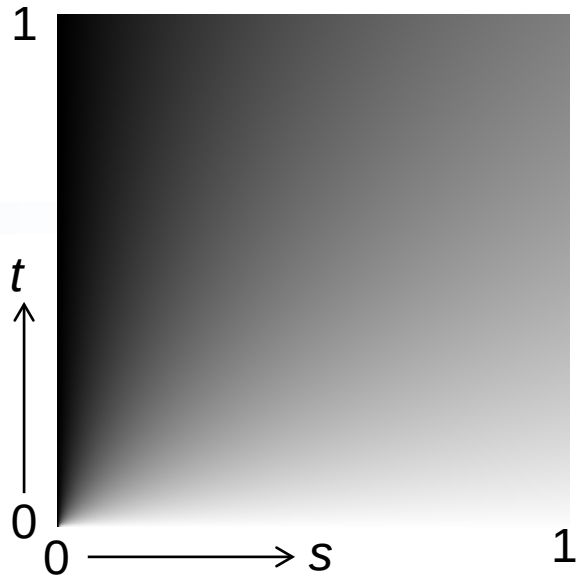
- The sphere map is an environment map stored as the reflection of a scene on sphere
- Not only stores scene, stores the *reflection* of the scene
  - Let the scene be a single point light source
  - Render a Phong specular highlight on a sphere
  - Use rendered sphere as an environment spheremap
- Texcoord interpolation samples highlight through face interior
- Texture is view/light dependent





# A Skin Texture Shader




- Skin appears softer than Lambertian reflectance because of subsurface scattering

- Seeliger lighting model

$$I = (N \cdot L) / (N \cdot L + N \cdot V)$$

- Implement as a texture shader

$$s = N \cdot L$$

$$t = N \cdot V$$

$$C = s/(s+t)$$