

15 - shading

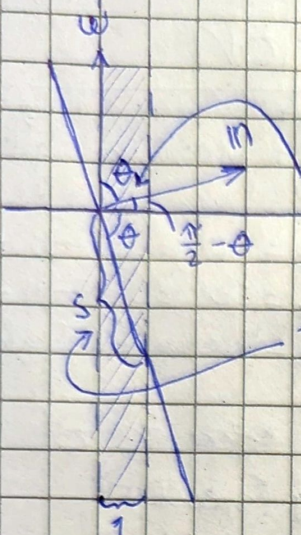
shading = lighting

w = light direction

m = surface normal

θ = angle between w and m

L = light



The amount of light hitting the surface here is $L/1$.

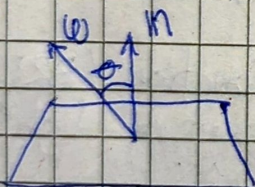
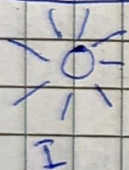
The amount of light hitting the surface here is L/s .

From geometry this gives:

$$\cos \theta = 1/s \Rightarrow 1/s = \cos \theta$$

Geometry term

Lambertian (diffuse) material (looks flat/not polished)



I = light intensity

w = light direction (unit vector)

m = surface normal (unit vector)

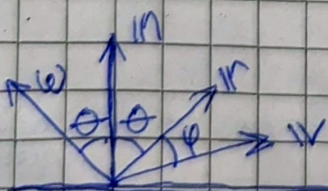
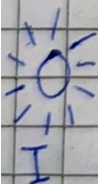
K_d = surface color (color/texture) (diffuse)

$$\text{Pixel color } C = I \cos \theta K_d = I (m \cdot w) K_d$$

Phong

Specular reflections

Diffuse is attached to material, specular ~~not~~ is reflection of light source.



I = light intensity

w = light direction

m = surface normal

r = light reflection direction

v = viewing direction

Material properties $\left\{ \begin{array}{l} K_s = \text{specular coefficient} \\ \alpha = \text{power factor} \end{array} \right.$

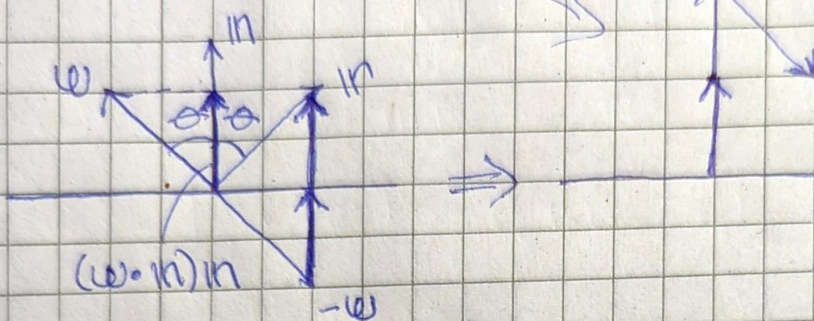
$$\text{specular reflection component} = I K_s (\cos \phi)^\alpha = I K_s (r \cdot v)^\alpha$$

$$\text{Phong material model } C = \underbrace{I \cos \theta K_d}_{\text{diffuse}} + \underbrace{I K_s (\cos \phi)^\alpha}_{\text{specular}} \Rightarrow$$

$$I(\max(0, \cos\theta) K_d + K_s (\cos\phi)^\alpha) \Rightarrow I \max(0, \cos\theta) (K_d + K_s \frac{(\max(0, \cos\theta))^\alpha}{\cos\theta})$$

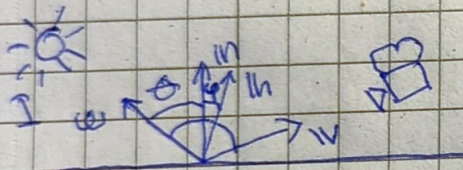
↑ should not be negative ↑ should be black if negative

$$r = 2(w \cdot n)n - w$$



Blinn material model

$$C = I(\cos\theta K_d + K_s (\cos\phi)^\alpha)$$



$$h = \frac{w + n}{|w + n|}$$

$$\cos\theta = n \cdot h$$

Ambient light

$$C = I(\underbrace{\cos\theta K_d}_{\text{diffuse}} + \underbrace{K_s (\cos\phi)^\alpha}_{\text{specular}}) + \underbrace{I_a K_a}_{\text{ambient}}$$

K_a = ambient color
= K_d often times

K_a, K_d, K_s, α = material properties

Lights

- Directional lights - Intensity, direction
- Point/spot light - Intensity, position
- Area light - Rectangle, disk, sphere, mesh...
- Sky light (environment light)

Hard to compute because direction based on shape

Image-based lighting