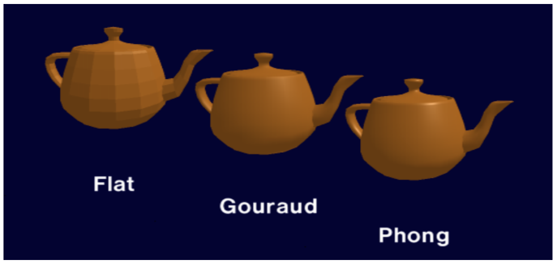
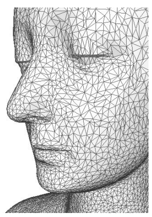
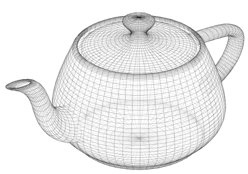
Shading Models for Polygons



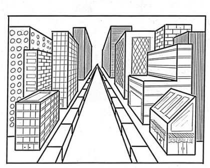
Gouraud Shading



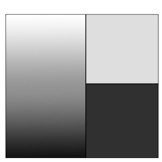
- compute vertex normals
  - average of polygons around vertex
  - directly from model during tessellation
- perform lighting operation at vertex
- linearly interpolate resulting vertex color
  (linear interpolation not correct)



Problems with Interpolated Shading

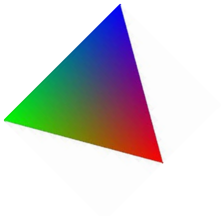


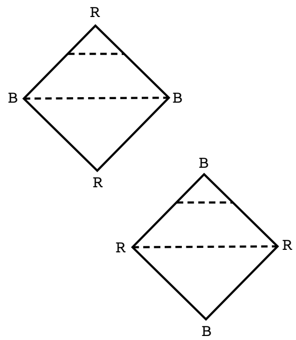
- linear interpolation of vertex values
  i.e. perspective distortion
- problems with shared vertices, T-vertices
- bad average of surface normal



Problems with Interpolated Shading

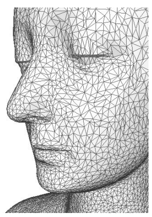
- orientation dependence

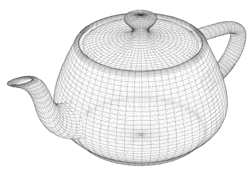




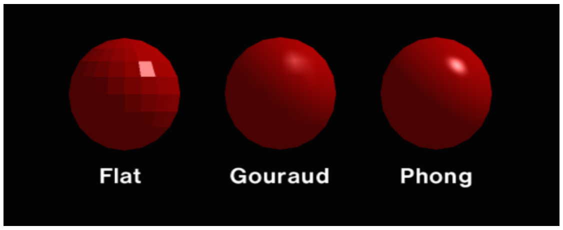
Phong Shading

- compute vertex normals
- linearly interpolate vertex normals
  (linear interpolation not correct)
- perform lighting operation per pixel

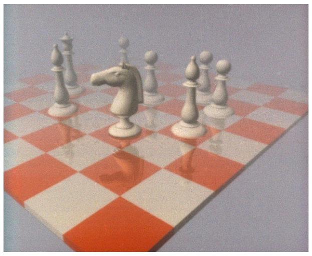




Shading Models for Polygons

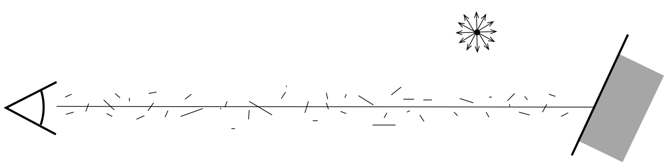


Atmospheric Attenuation/Scattering



Atmospheric Attenuation/Scattering

- some reflected light gets scattered away from line of sight
- some ambient light gets scattered into line of sight



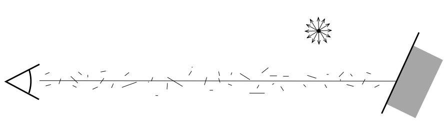
Atmospheric Attenuation/Scattering

- **transmissivity** what fraction remains after unit distance

eg: let t = 0.9 -> for distance d remaining is $(0.9)^d$

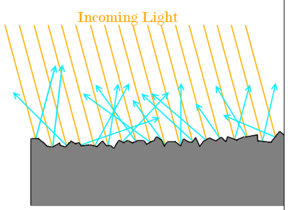
Let t = (transmissivity)^d

$$I_{out} = I_{surface} \cdot t + (1-t) \cdot I_{background}$$



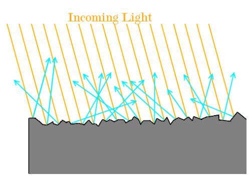
Torrance-Sparrow Reflectance Function

- Phong's $\cos^n(\alpha)$ is easy to compute, not accurate
- two physicists came up with more accurate model
- surface is made up of micro-facets
- micro-facets perfectly specular



Torrance-Sparrow Reflectance Function

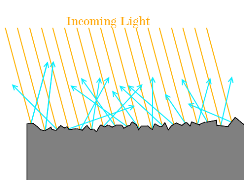
$$\rho_s = \frac{F_\lambda D G}{\pi(\vec{N}\cdot\vec{E})(\vec{N}\cdot\vec{L})}$$



D= microfacet distribution function
G= geometrical attenuation function
F_λ= Fresnel function

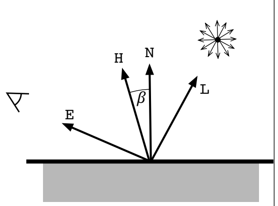
Microfacet Distribution F^n

- Phong's $\cos^n(\alpha)$ is a very simple D
  (G & F = 0)



Beckman:

$$D = \frac{1}{m^2 \cos^4(\beta)} e^{-\left\{\frac{\tan^2(\beta)}{m^2}\right\}}$$

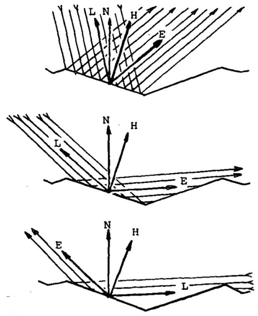


m = rms slope

Geometrical Attenuation F^n

G: self-shadowing (0…1)

$$G = \min\left\{1, \frac{2(\vec{N}\cdot\vec{H})(\vec{N}\cdot\vec{E})}{(\vec{E}\cdot\vec{H})}, \frac{2(\vec{N}\cdot\vec{H})(\vec{N}\cdot\vec{L})}{(\vec{E}\cdot\vec{H})}\right\}$$



Fresnel Reflectance F^n

- reflectance as a function of incidence angle θ

$$F = \frac{1}{2}\frac{(g-c)^2}{(g+c)^2}\left\{1+\frac{(c(g+c)-1)^2}{(c(g-c)+1)^2}\right\}$$

$c = dot(\vec{E},\vec{H})$

$g = \sqrt{n^2 + c^2 - 1}$

n = index of refraction

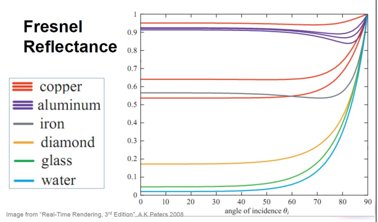


Image from "Real-Time Rendering, 3rd Edition", A.K.Peters 2008

Cook-Torrance

reflectance (Fresnel) is also a function of wavelength λ

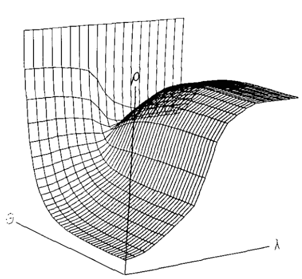


Figure 5a. Reflectance (ρ) of a copper mirror as a function of wavelength (λ) and incidence angle (θ).