Computer Graphics

- BRDFs & Texturing -

Hendrik Lensch

Overview

- **Last time**
  - Radiance
  - Light sources
  - Rendering Equation & Formal Solutions
- **Today**
  - Bidirectional Reflectance Distribution Function (BRDF)
  - Reflection models
  - Projection onto spherical basis functions
  - Shading
- **Next lecture**
  - Varying (reflection) properties over object surface: texturing

Reflection Equation - Reflectance

- **Reflection equation**

$$L_o(\underline{x},\underline{\omega}_o) = \int_{\Omega_+} f_r(\underline{\omega}_i,\underline{x},\underline{\omega}_o)\, L_i(\underline{x},\underline{\omega}_i) \cos\theta_i \, d\underline{\omega}_i$$

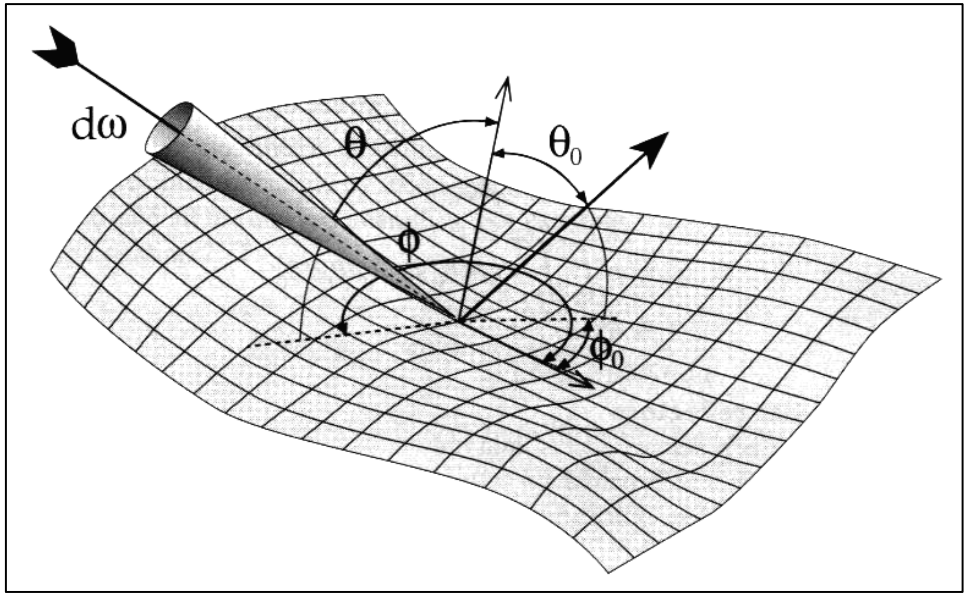
- **BRDF**
  – Ratio of reflected radiance to incident irradiance

$$f_r(\omega_o, x, \omega_i) = \frac{dL_o(x,\omega_o)}{dE_i(x,\omega_i)}$$

Bidirectional Reflectance Distribution Function

- **BRDF describes surface reflection for light incident from direction** (θ_i, φ_i) **observed from direction** (θ_o, φ_o)
- **Bidirectional**
  - Depends on two directions and position (6-D function)
- **Distribution function**
  - Can be infinite
- **Unit [1/sr]**

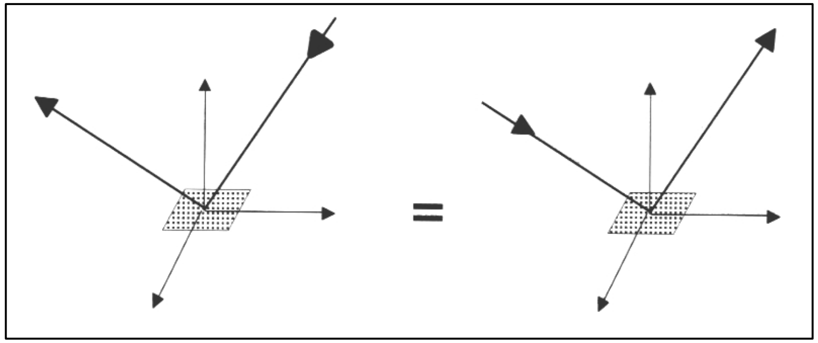
$$f_r(\underline{\omega}_o, \underline{x}, \underline{\omega}_i) = \frac{dL_o(\underline{x}, \underline{\omega}_o)}{dE_i(\underline{x}, \underline{\omega}_i)} = \frac{dL_o(\underline{x}, \underline{\omega}_o)}{dL_i(\underline{x}, \underline{\omega}_i)\cos\theta_i \, d\underline{\omega}_i}$$



BRDF Properties

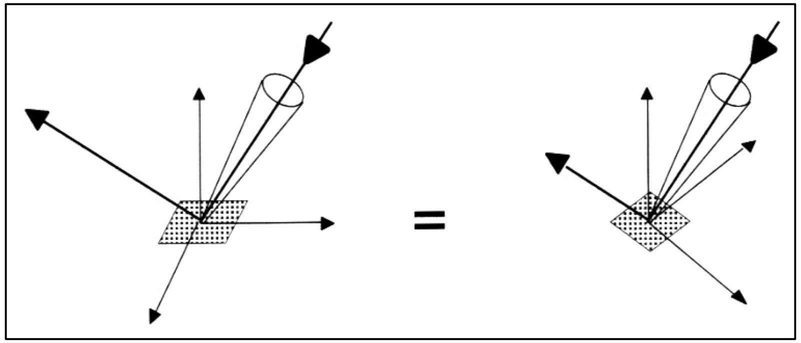
- **Helmholtz reciprocity principle**
  - BRDF remains unchanged if incident and reflected directions are interchanged

$$f_r(\omega_o, \omega_i) = f_r(\omega_i, \omega_o)$$



- **Smooth surface: isotropic BRDF**
  - reflectivity independent of rotation around surface normal
  - BRDF has only 3 instead of 4 directional degrees of freedom

$$f_r(\underline{x}, \theta_i, \theta_o, \varphi_o - \varphi_i)$$



BRDF Properties

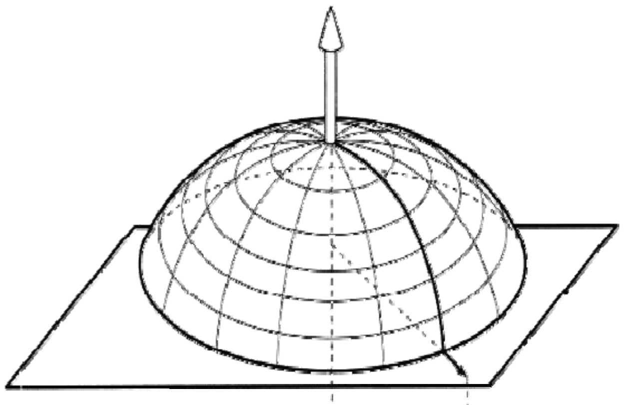
- **Characteristics**
  - BRDF units [sr^{-1}]
    - Not intuitive
  - Range of values:
    - From 0 (absorption) to ∞ (reflection, δ-function)
  - Energy conservation law
    - No self-emission
    - Possible absorption

$$\int_{\Omega} f_r(\underline{\omega}_o, \underline{x}, \underline{\omega}_i) \cos\theta_o d\omega_o \leq 1 \quad \forall\, \theta, \varphi$$

  - Reflection only at the point of entry ($x_i = x_o$)
    - No subsurface scattering

BRDF Measurement

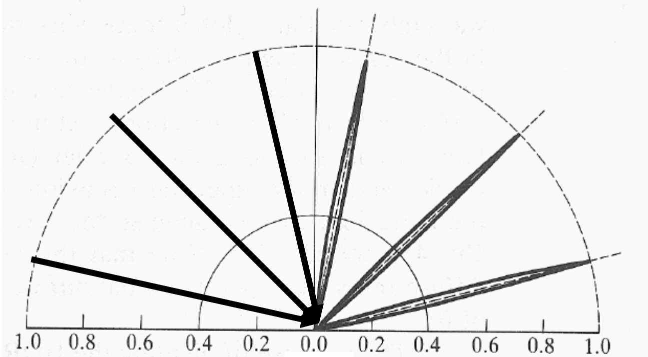
- **Gonio-Reflectometer**
- **BRDF measurement**
  - point light source position (θ, φ)
  - light detector position (θ_o, φ_o)
- **4 directional degrees of freedom**
- **BRDF representation**
  - m incident direction samples (θ, φ)
  - n outgoing direction samples (θ_o, φ_o)
  - $m*n$ reflectance values (large!!!)



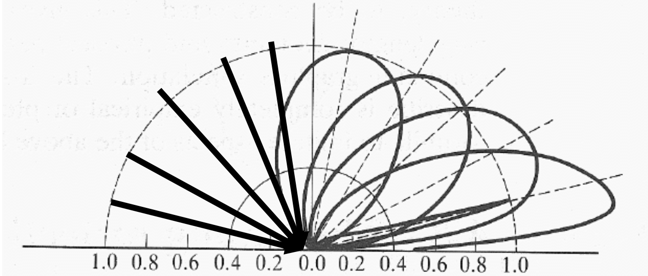
Stanford light gantry

Reflectance

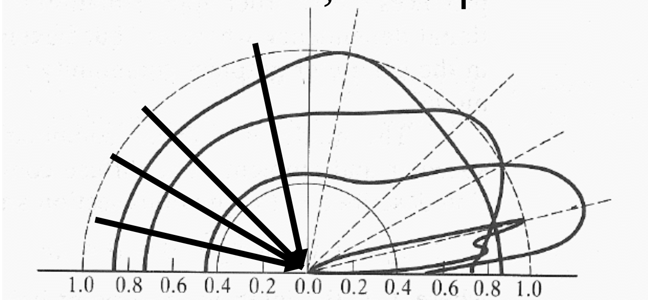
- **Reflectance may vary with**
  - Illumination angle
  - Viewing angle
  - Wavelength
  - (Polarization, ...)
- **Variations due to**
  - Absorption
  - Surface micro-geometry
  - Index of refraction / dielectric constant
  - Scattering



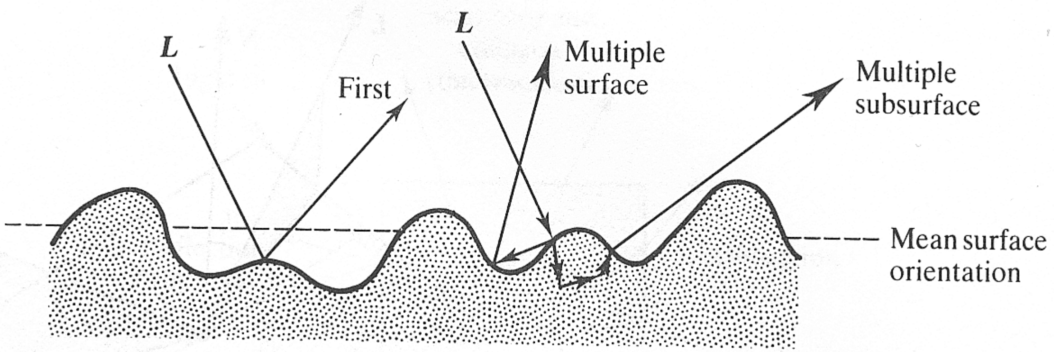
Aluminum; λ=2.0μm



Aluminum; λ=0.5μm

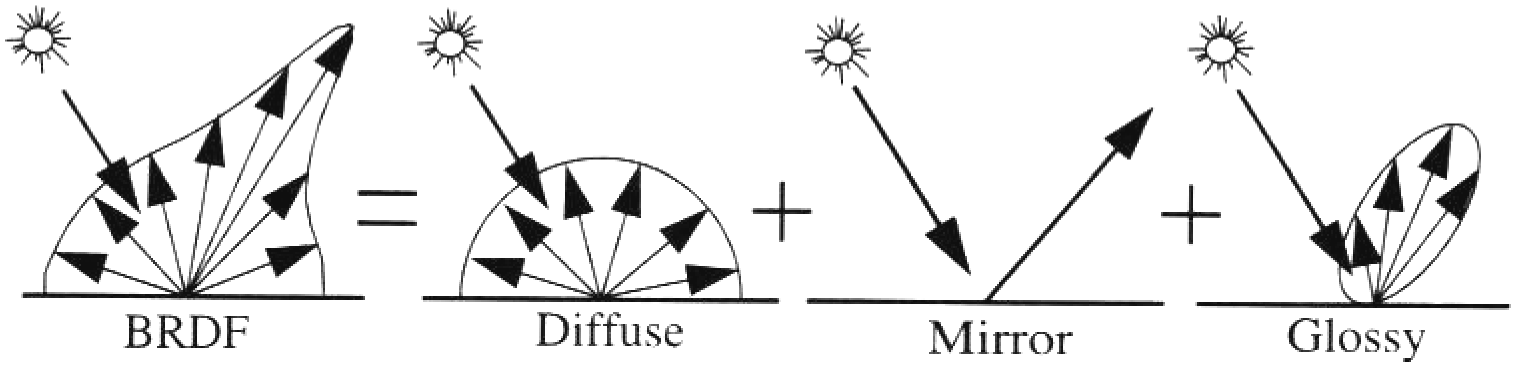


Magnesium;
λ=0.5μm



BRDF Modeling

- **Phenomenological approach**
  - Description of visual surface appearance
- **Ideal specular reflection**
  - Reflection law
  - Mirror
- **Glossy reflection**
  - Directional diffuse
  - Shiny surfaces
- **Ideal diffuse reflection**
  - Lambert's law
  - Matte surfaces

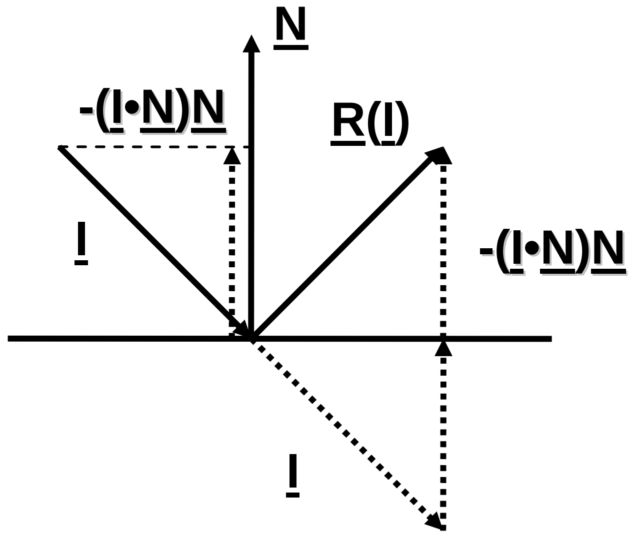


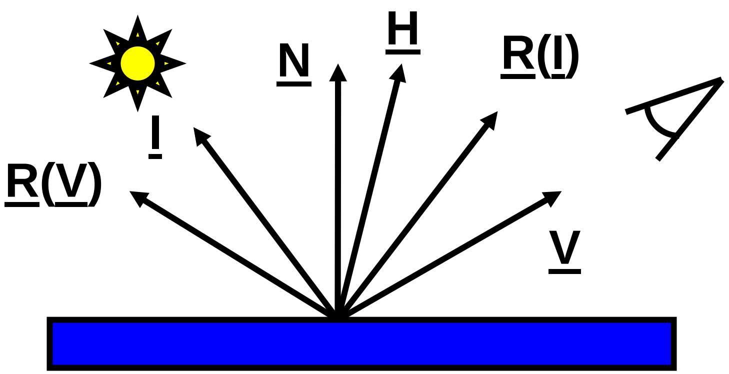
Reflection Geometry

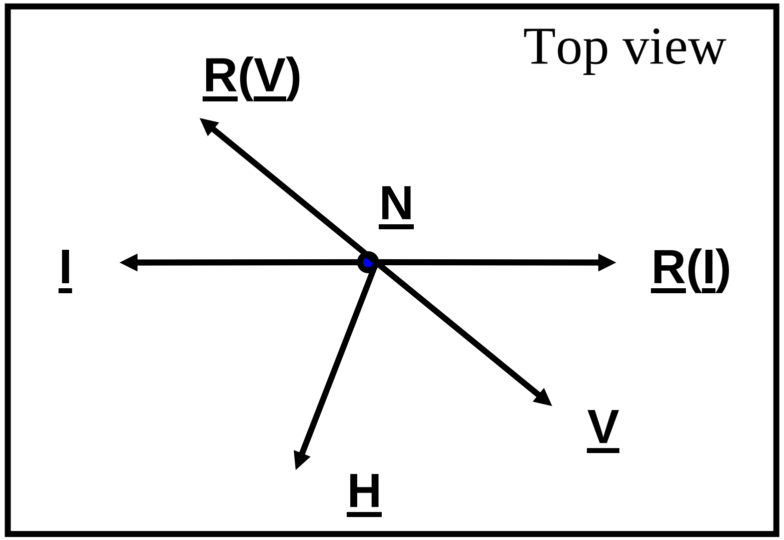
- **Direction vectors (normalize):**
  - N: surface normal
  - I: vector to the light source
  - V: viewpoint direction vector
  - H: halfway vector

    H= (I + V) / |I + V|
  - R(I): reflection vector

    R(I)= I - 2(I•N)N
  - Tangential surface: local plane



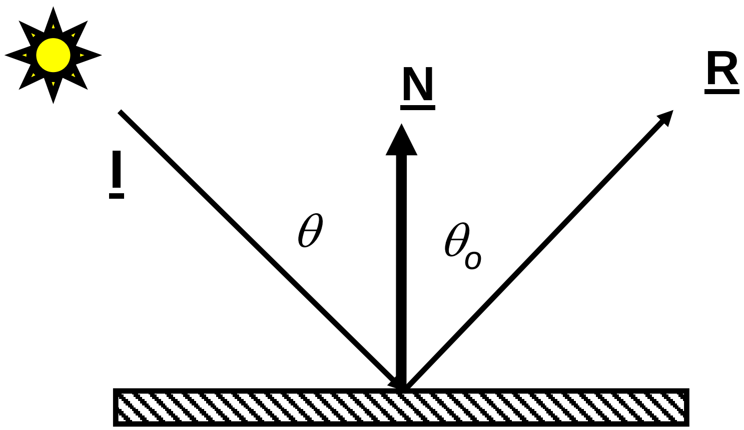




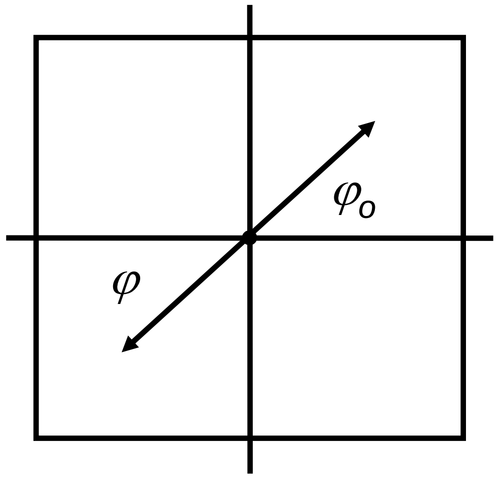
Ideal Specular Reflection

- **Angle of reflectance equal to angle of incidence**
- **Reflected vector in a plane with incident ray and surface normal vector**

$$\underline{\mathbf{R}}+(-\underline{\mathbf{I}}) = 2\cos\theta\,\underline{\mathbf{N}} = -2(\underline{\mathbf{I}}\bullet\underline{\mathbf{N}})\,\underline{\mathbf{N}}$$
$$\underline{\mathbf{R}}(\underline{\mathbf{I}}) = \underline{\mathbf{I}} - 2(\underline{\mathbf{I}}\bullet\underline{\mathbf{N}})\,\underline{\mathbf{N}}$$



$\theta = \theta_o$



$\varphi = \varphi_o + 180°$

Mirror BRDF

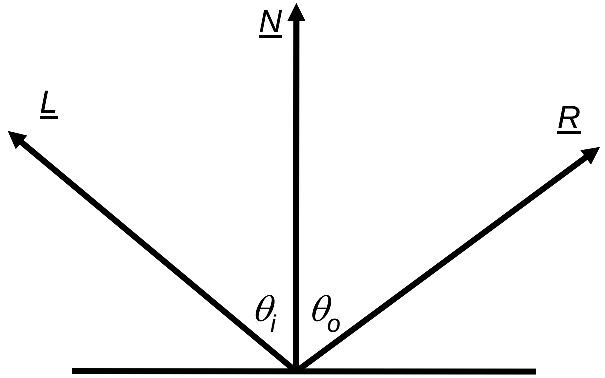
- **Dirac Delta function $\delta(x)$**
  - $\delta(x)$: zero everywhere except at $x=0$
  - Unit integral iff integration domain contains zero (zero otherwise)

$$f_{r,m}(\omega_o, x, \omega_i) = \rho_s(\theta_i) \cdot \frac{\delta(\cos\theta_i - \cos\theta_o)}{\cos\theta_i} \cdot \delta(\varphi_i - \varphi_o \pm \pi)$$

$$L_o(x, \omega_o) = \int_{\Omega_+} f_{r,m}(\omega_o, x, \omega_i) L_i(\theta_i, \varphi_i) \cos\theta_i \, d\underline{\omega}_i = \rho_s(\theta_i) L_i(\theta_o, \varphi_o \pm \pi)$$

- **Specular reflectance ρ_s**
  - Ratio of reflected radiance in specular direction and incoming radiance
  - Dimensionless quantity between *0* and *1*

$$\rho_s(\theta_i) = \frac{\Phi_o(\theta_o)}{\Phi_i(\theta_i)}$$



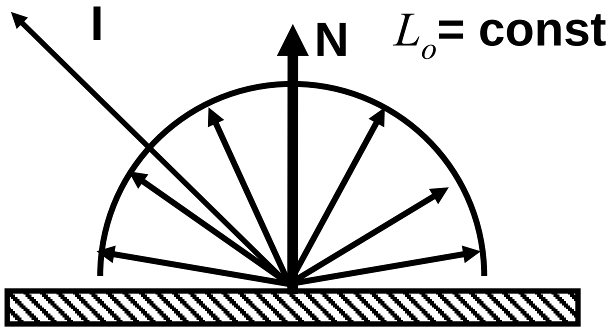
Diffuse Reflection

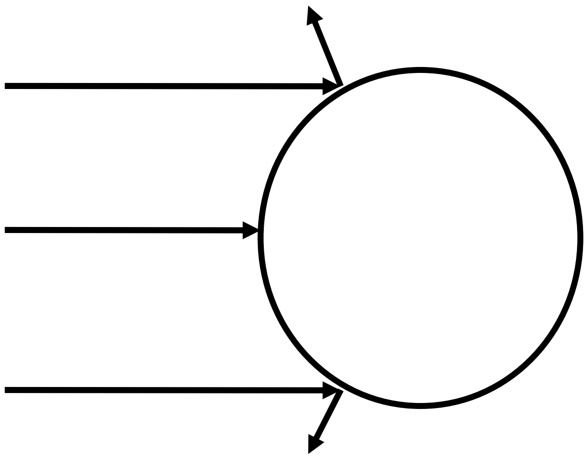
- **Light equally likely to be reflected in any output direction (independent of input direction)**
- **Constant BRDF**

$$f_{r,d}(\underline{\omega}_o, \underline{x}, \underline{\omega}_i) = k_d = \text{const}$$

$$L_o(\underline{x}, \underline{\omega}_o) = \int_\Omega k_d L_i(\underline{x}, \underline{\omega}_i) \cos\theta_i \, d\underline{\omega}_i = k_d \int_\Omega L_i(\underline{x}, \underline{\omega}_i) \cos\theta_i \, d\underline{\omega}_i = k_d E$$

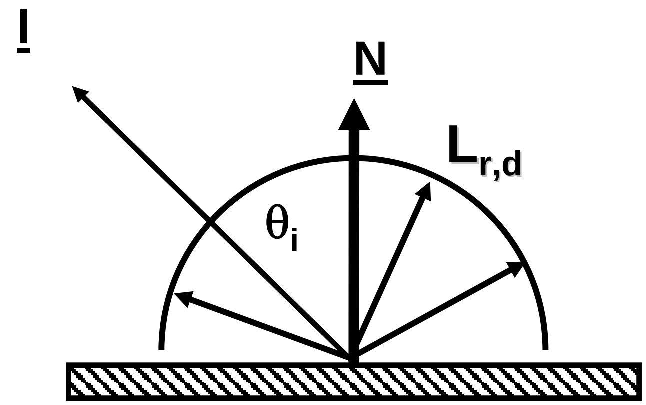
  - k_d: diffuse coefficient, material property [1/sr]





Lambertian Diffuse Reflection

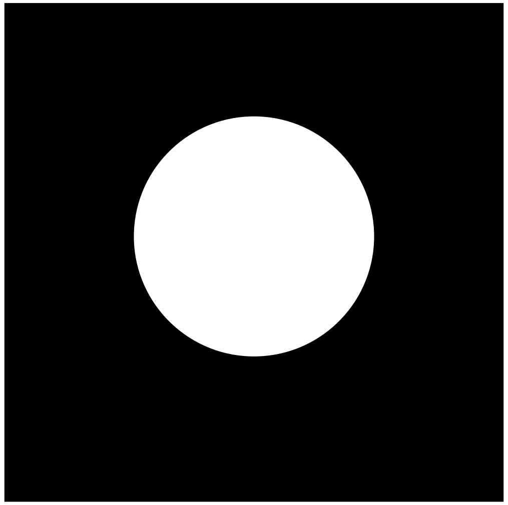
- **Radiosity** $$B = \int_\Omega L_o(\underline{x}, \underline{\omega}_o) \cos\theta_o \, d\underline{\omega}_o = L_o \int_\Omega \cos\theta_o \, d\underline{\omega}_o = \pi \, L_o$$
- **Diffuse Reflectance** $$\rho_d = \frac{B}{E} = \pi \, k_d$$
- **Lambert's Cosine Law** $$B = \rho_d E = \rho_d E_i \cos\theta_i$$
- **For each light source**
  - $L_{r,d} = k_d \, L_i \cos\theta_i = k_d \, L_i \, (\underline{l} \bullet \underline{N})$

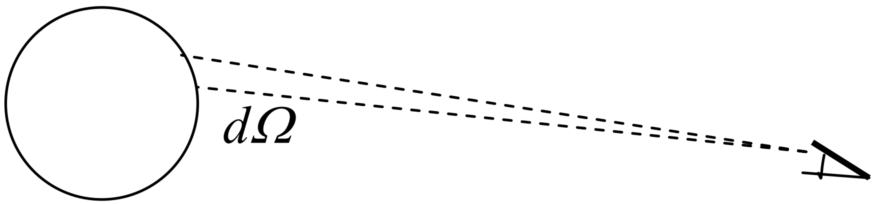


Lambertian Objects

Self-Luminous
spherical Lambertian Light Source

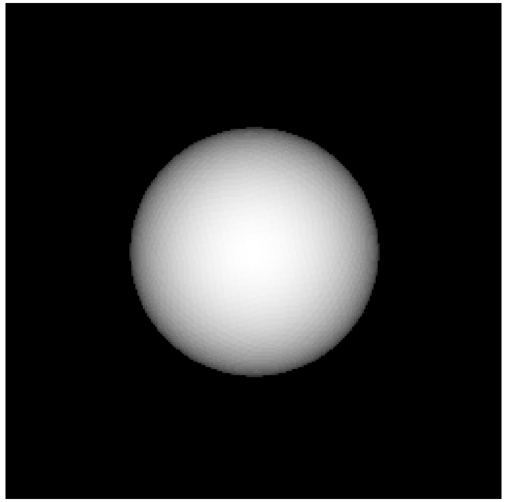
$$\Phi_0 \propto L_0 \cdot d\Omega$$

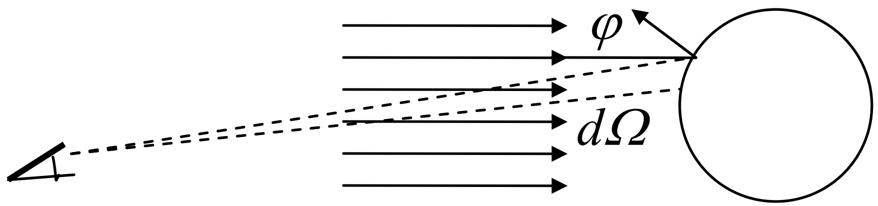




Eye-light illuminated
Spherical Lambertian Reflector

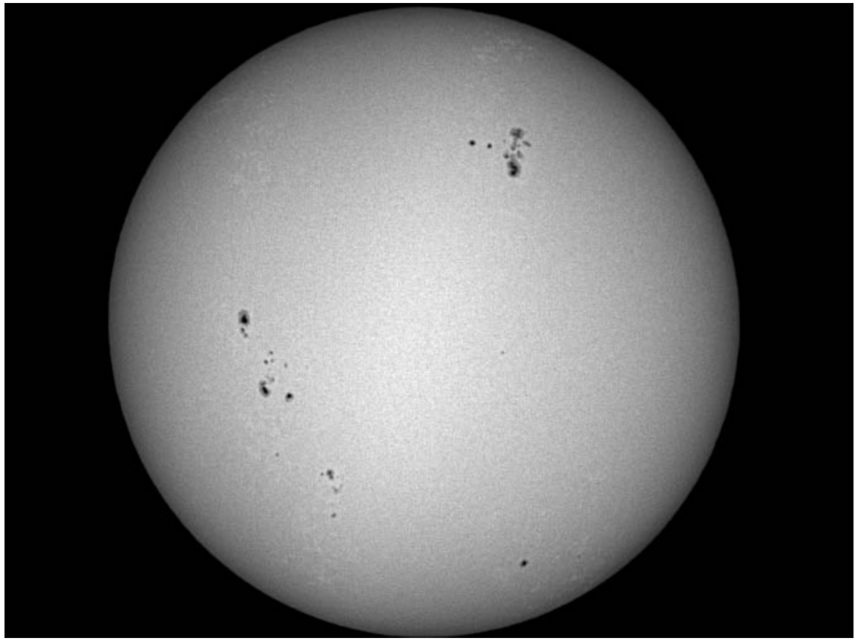
$$\Phi_1 \propto L_0 \cdot \cos\varphi \cdot d\Omega$$





Lambertian Objects II

The Sun



- Absorption in photosphere
- Path length through photosphere longer from the Sun’s rim

The Moon

- Surface covered with fine dust
- Dust on TV visible best from slanted viewing angle

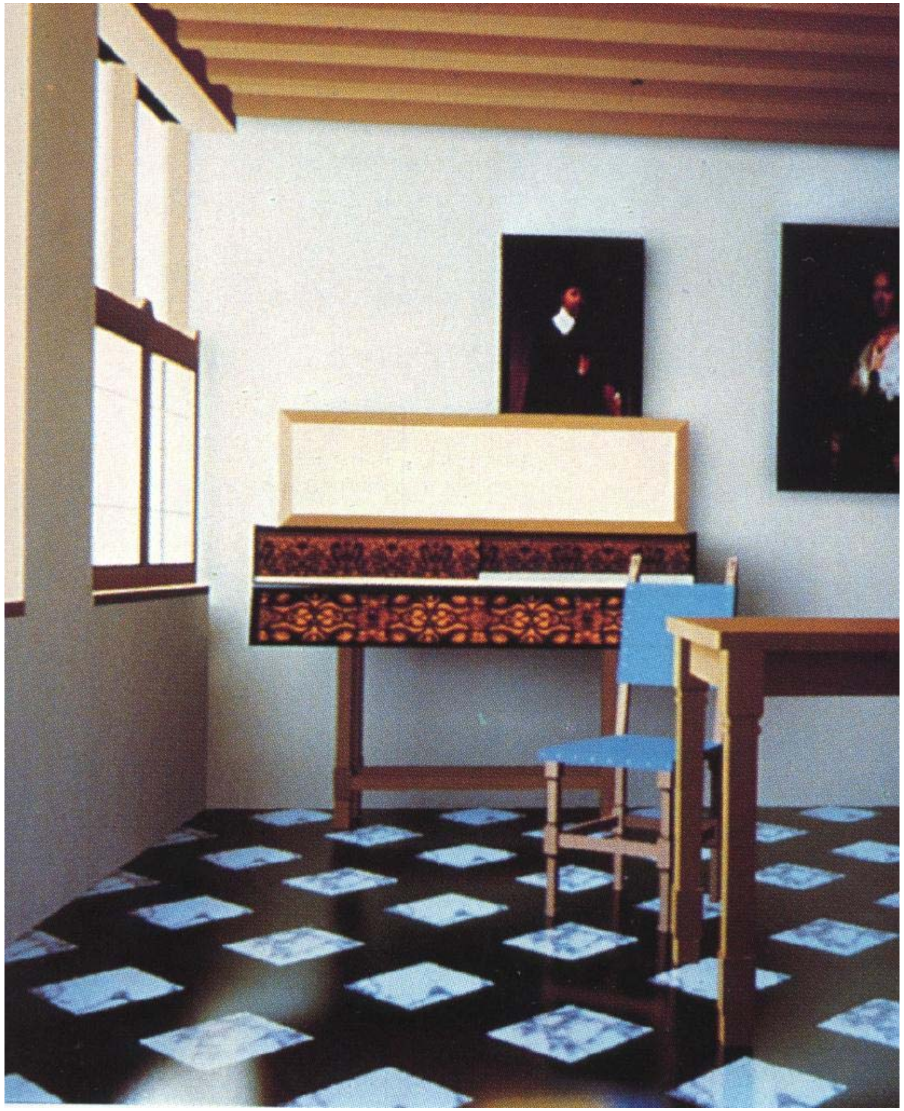
⇒ Neither the Sun nor the Moon are Lambertian

“Diffuse” Reflection

- **Theoretical explanation**
  - Multiple scattering
- **Experimental realization**
  - Pressed magnesium oxide powder
  - Almost never valid at high angles of incidence

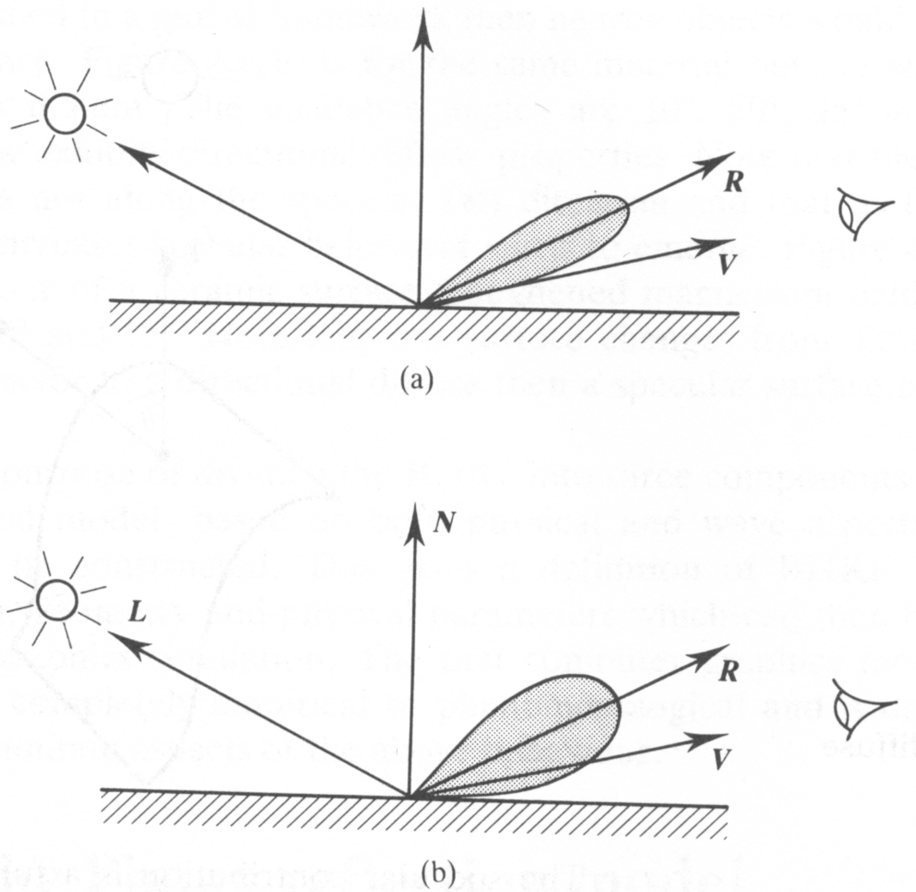
Paint manufacturers attempt to create ideal diffuse paints

Glossy Reflection



Glossy Reflection

- **Due to surface roughness**
- **Empirical models**
  - Phong
  - Blinn-Phong
- **Physical models**
  - Blinn
  - Cook & Torrance

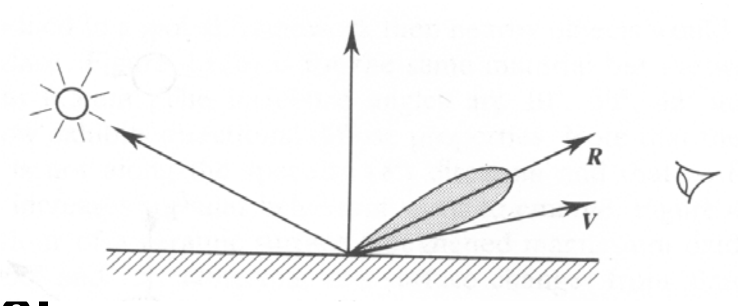


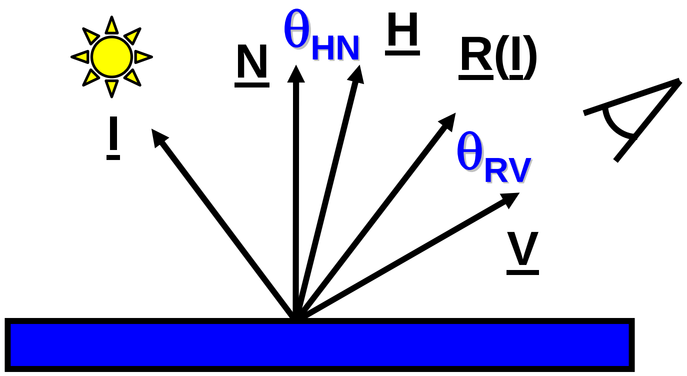
Phong Reflection Model

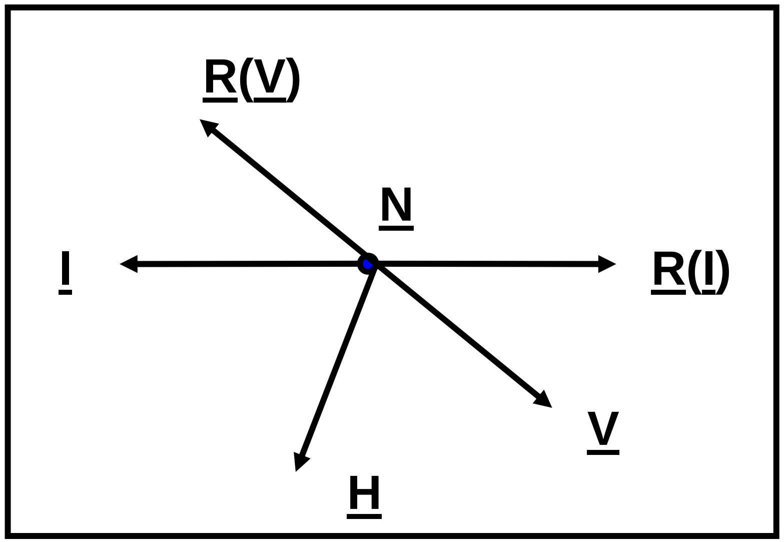
- **Cosine power lobe**

$$f_r(\omega_o, x, \omega_i) = k_s \left(\underline{R}(\underline{I}) \cdot \underline{V} \right)^{k_e}$$

  – $L_{r,s} = L_i\, k_s \cos^{ke} \theta_{RV}$
- **Dot product & power**
- **Not energy conserving/reciprocal**
- **Plastic-like appearance**



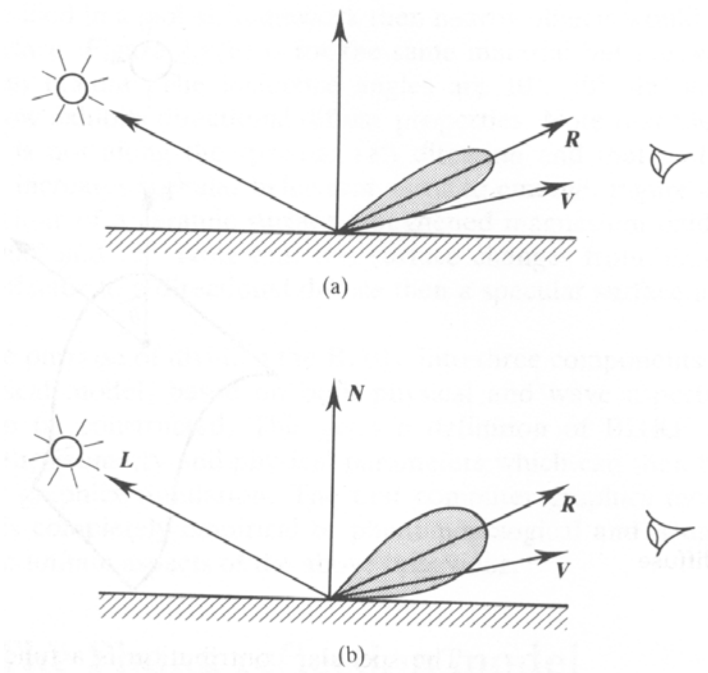


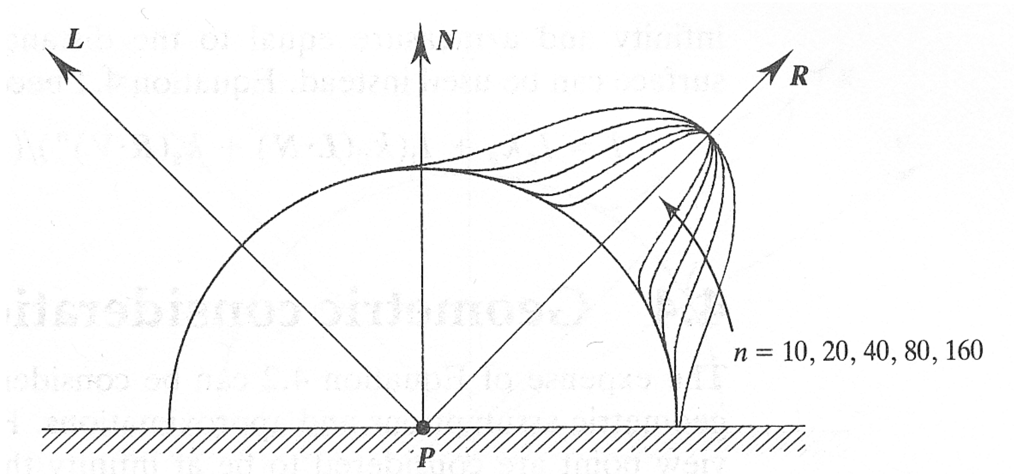


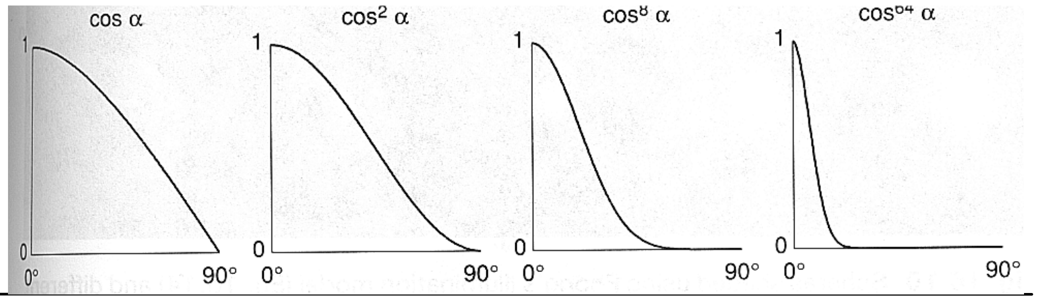
Phong Exponent k_e

$$f_r(\omega_o, x, \omega_i) = k_s\left(\underline{R}(\underline{I}) \cdot \underline{V}\right)^{k_e}$$

- **Determines size of highlight**







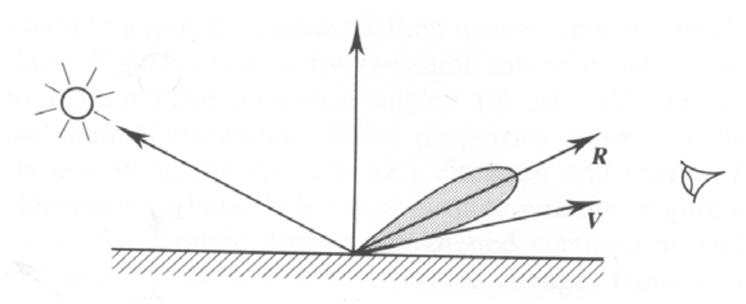
Blinn-Phong Reflection Model

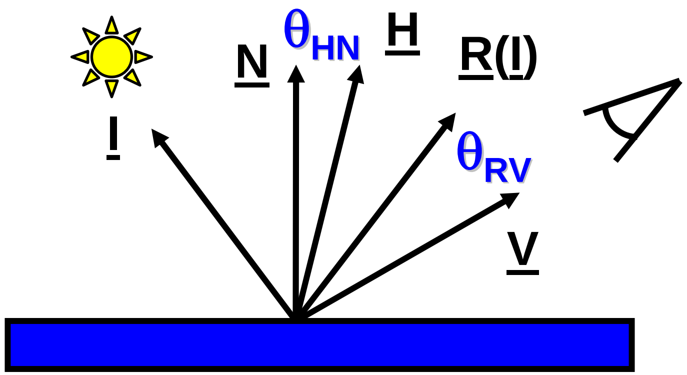
- **Blinn-Phong reflection model**

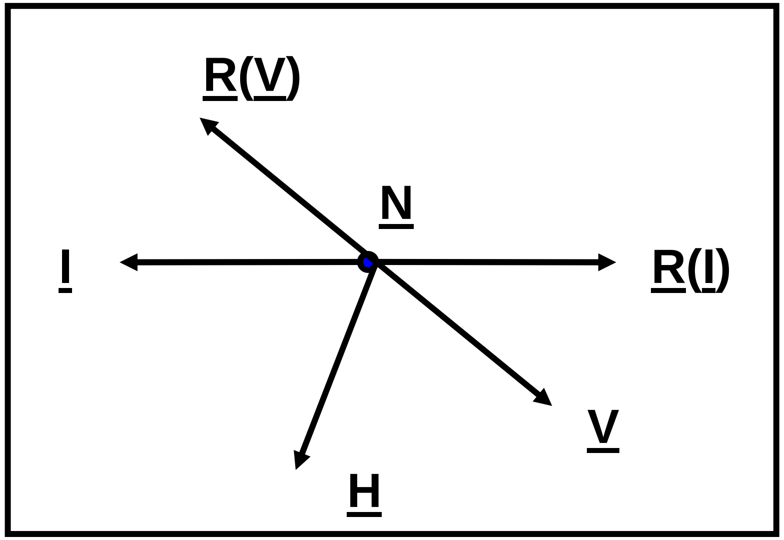
$$f_r(\omega_o, x, \omega_i) = k_s \left(H \cdot N\right)^{k_e}$$

  - $L_{r,s} = L_i\, k_s \cos^{ke} \theta_{HN}$
  - $\theta_{RV} \Rightarrow \theta_{HN}$
  - Light source, viewer far away
  - I, R constant: H constant

    θ_{HN} less expensive to compute







Phong Illumination Model

- **Extended light sources: *l* point light sources**

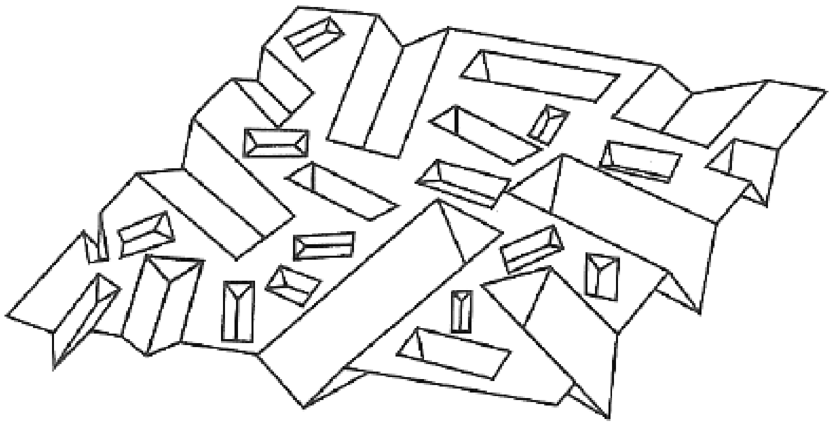
$$\mathrm{L_r} = k_a L_{i,a} + k_d \sum_l L_l (I_l \cdot N) + k_s \sum_l L_l (R(I_l) \cdot V)^{k_e} \quad \text{(Phong)}$$

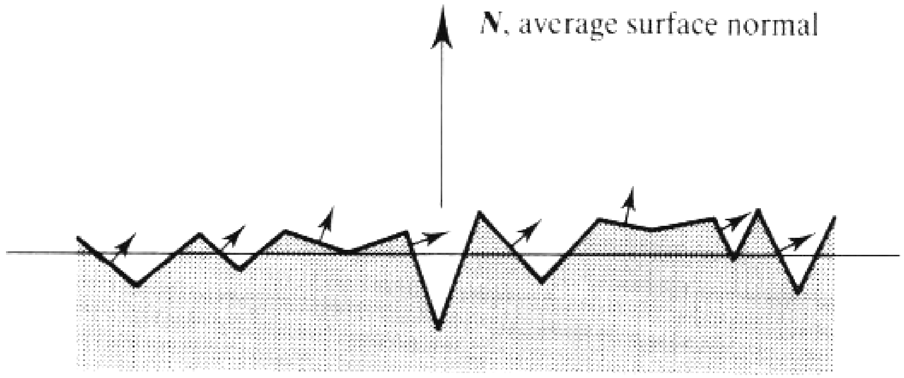
$$\mathrm{L_r} = k_a L_{i,a} + k_d \sum_l L_l (I_l \cdot N) + k_s \sum_l L_l (H_l \cdot N)^{k_e} \quad \text{(Blinn)}$$

- **Color of specular reflection equal to light source**
- **Heuristic model**
  - Contradicts physics
  - Purely local illumination
    - Only direct light from the light sources
    - No further reflection on other surfaces
    - Constant ambient term
- **Often: light sources & viewer assumed to be far away**

Microfacet Model

- **Isotropic microfacet collection**
- **Microfacets assumed as perfectly smooth reflectors**
- **BRDF**
  - Distribution of microfacets
    - Often probabilistic distribution of orientation or V-groove assumption
  - Planar reflection properties
  - Self-masking, shadowing



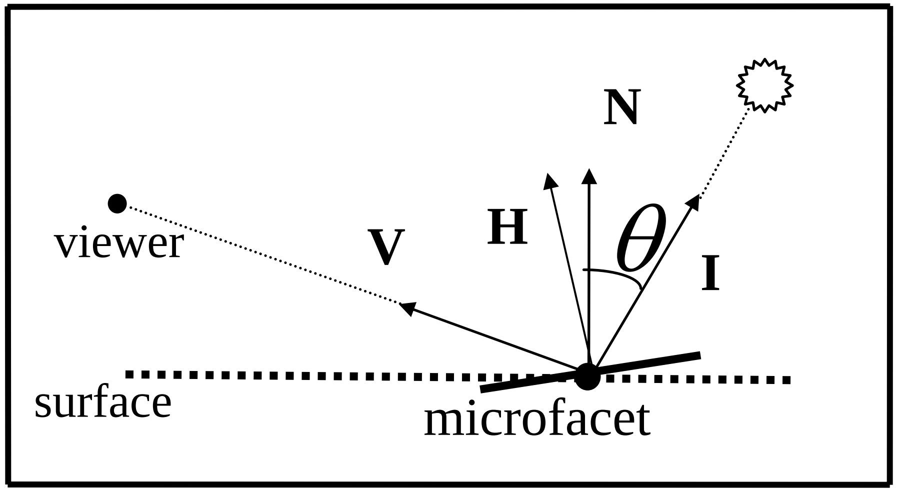


Ward Reflection Model

- **BRDF**

$$f_r = \frac{\rho_d}{\pi} + \rho_s \frac{1}{\sqrt{(I \bullet N)(V \bullet N)}} \bullet \frac{\exp(-\tan^2 \angle(H, N) / \sigma^2)}{4\pi\sigma^2}$$

  - σ standard deviation (RMS) of surface slope
  - Simple expansion to anisotropic model (σ_x, σ_y)
  - Empirical, not physics-based
  - Inspired by notion of reflecting microfacets
  - Convincing results
  - Good match to measured data



Physics-inspired BRDFs

- **Notion of reflecting microfacet**
- **Specular reflectivity of the form**

$$f_r = \frac{D \cdot G \cdot F_\lambda(\lambda, \theta_i)}{\pi \, \underline{N} \cdot \underline{V}}$$

  - D : statistical microfacet distribution
  - G : geometric attenuation, self-shadowing
  - F : Fresnel term, wavelength, angle dependency of reflection along mirror direction
  - $\underline{N} \bullet \underline{V}$: flaring effect at low angle of incidence

- **Cook-Torrance model**
  - F : wavelength- and angle-dependent reflection
  - Metal surfaces

Cook-Torrance Reflection Model

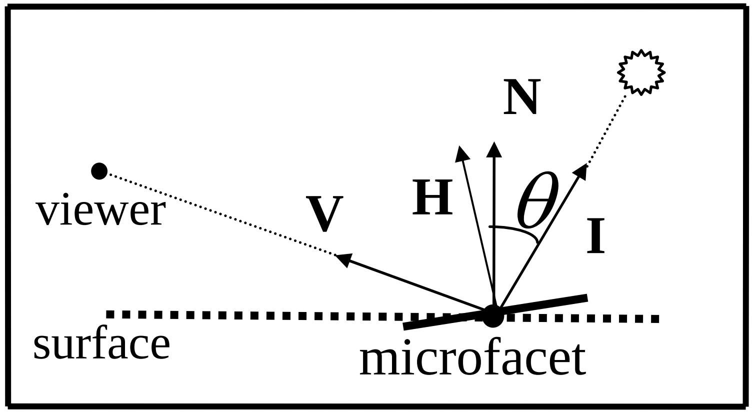
- Cook-Torrance reflectance model is based on the *microfacet* model. The BRDF is defined as the sum of a diffuse and specular components:

$$f_r = k_d \rho_d + k_s \rho_s; \qquad k_d + k_s \leq 1$$

  where k_s and k_d are the specular and diffuse coefficients.
- Derivation of the specular component ρ_s is based on a **physically derived** theoretical reflectance model

Cook-Torrance Specular Term

$$\rho_s = \frac{F_\lambda DG}{\pi(\underline{N}\cdot\underline{V})(\underline{N}\cdot\underline{I})}$$



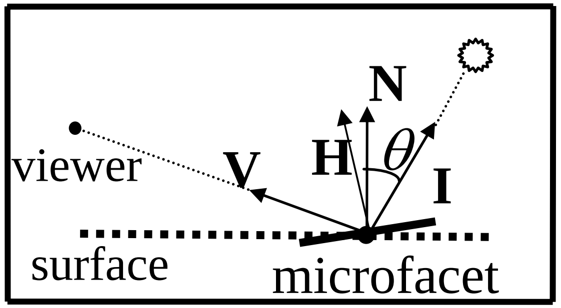
- **D : Distribution function of microfacet orientations**
- **G : Geometrical attenuation factor**
  - represents self-masking and shadowing effects of microfacets
- **F_λ : Fresnel term**
  - computed by Fresnel equation
  - relates incident light to reflected light for each planar microfacet

$$F_\lambda \approx (1+(V\cdot N))^\lambda$$

- **N·V : Proportional to visible surface area**
- **N·I : Proportional to illuminated surface area**

Microfacet Distribution Functions

- **Isotropic Distributions** $D(\underline{\omega}) \Rightarrow D(\alpha) \quad \alpha = \mathbf{N} \cdot \mathbf{H}$



  - α : angle to average normal of surface
  - Characterized by half-angle β

$$D(\beta) = \frac{1}{2}$$

- **Blinn**

$$D(\alpha) = \cos^{\frac{\ln 2}{\ln \cos \beta}} \alpha$$

- **Torrance-Sparrow**

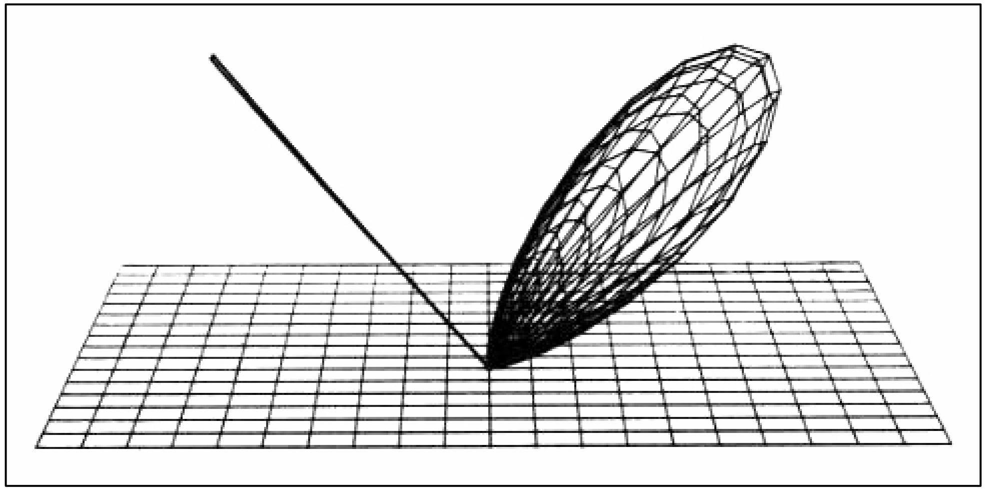
$$D(\alpha) = e^{-\left(\frac{\sqrt{2}}{\beta}\alpha\right)^2}$$

- **Beckmann**
  - m : average slope of the microfacets
  - Used by Cook-Torrance

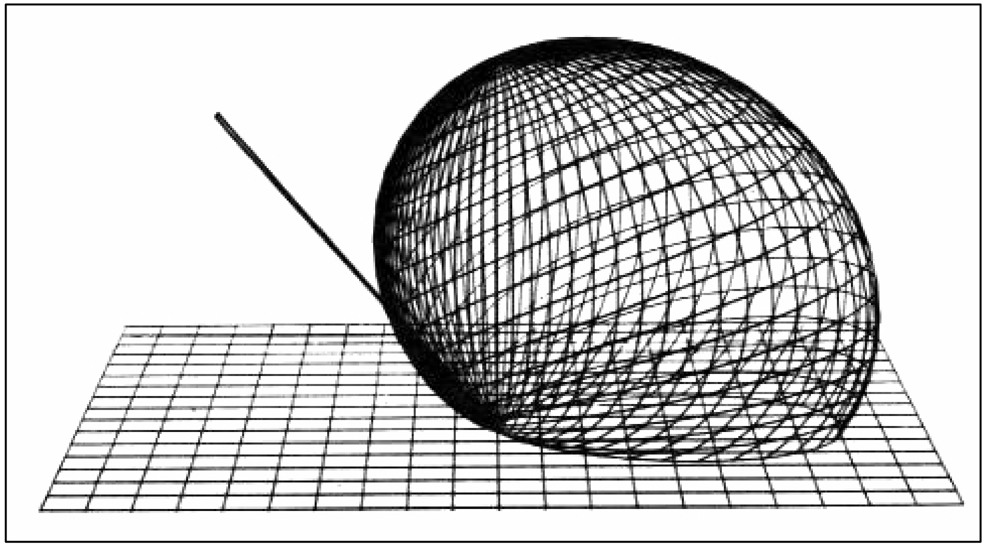
$$D(\alpha) = \frac{1}{4m^2 \cos^4 \alpha} e^{-[\tan \alpha / m]^2}$$

Beckman Microfacet Distribution Function

m=0.2



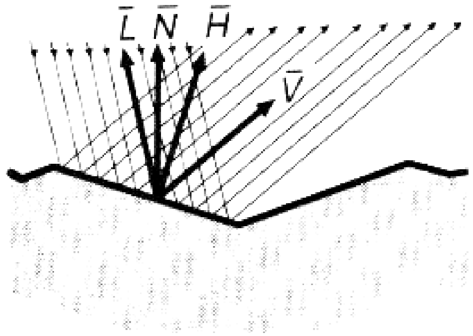
m=0.6



Geometric Attenuation Factor

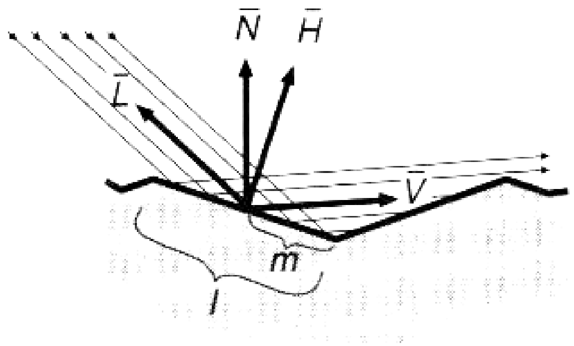
- **V-shaped grooves**
- Fully illuminated and visible

$$G = 1$$



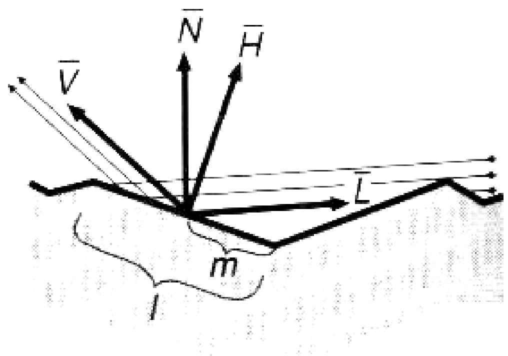
- Partial masking of reflected light

$$G = \frac{2(\underline{N} \cdot \underline{H})(\underline{N} \cdot \underline{V})}{(\underline{V} \cdot \underline{H})}$$



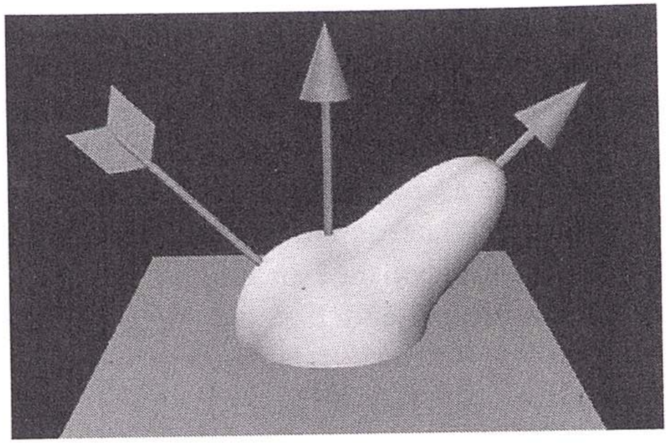
- Partial shadowing of incident light

$$G = \frac{2(\underline{N} \cdot \underline{H})(\underline{N} \cdot \underline{I})}{(\underline{V} \cdot \underline{H})}$$

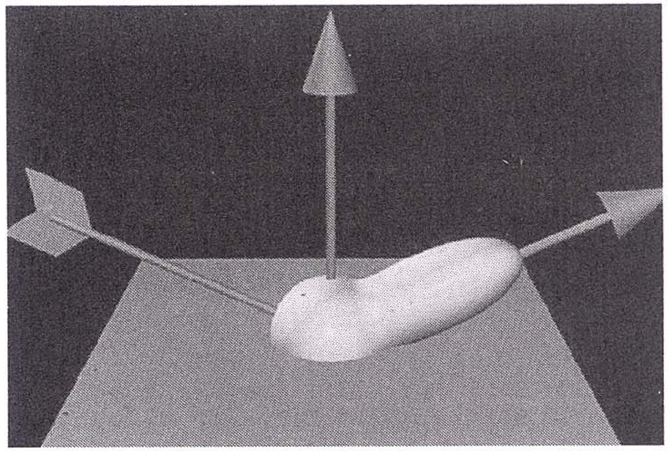


$$G = \min\left\{1, \frac{2(\underline{N} \cdot \underline{H})(\underline{N} \cdot \underline{V})}{(\underline{V} \cdot \underline{H})}, \frac{2(\underline{N} \cdot \underline{H})(\underline{N} \cdot \underline{I})}{(\underline{V} \cdot \underline{H})}\right\}$$

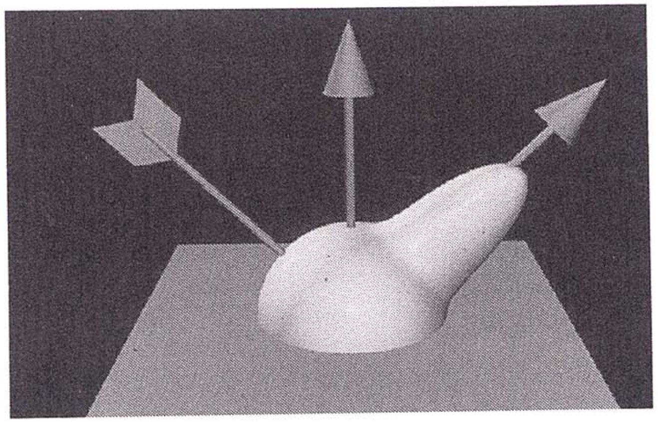
Comparison Phong vs. Torrance



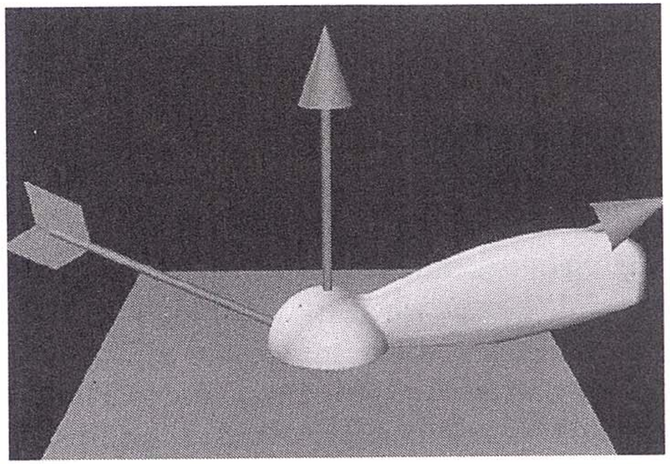
(a)



(b)



(c)

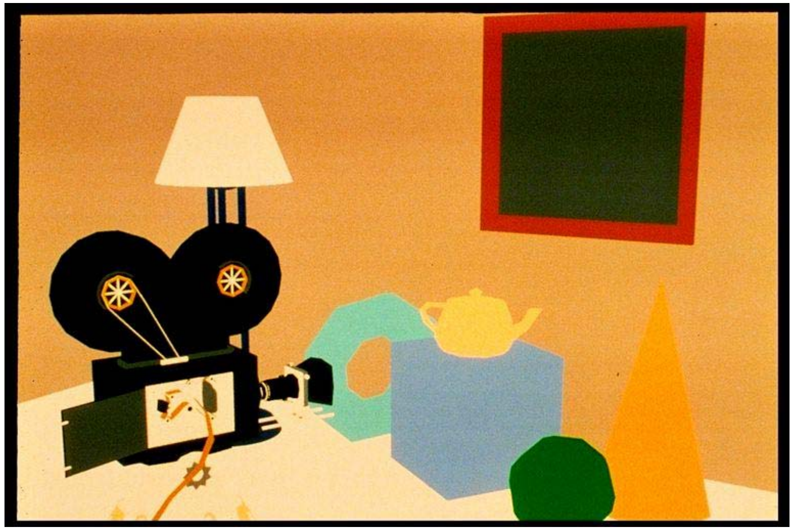


(d)

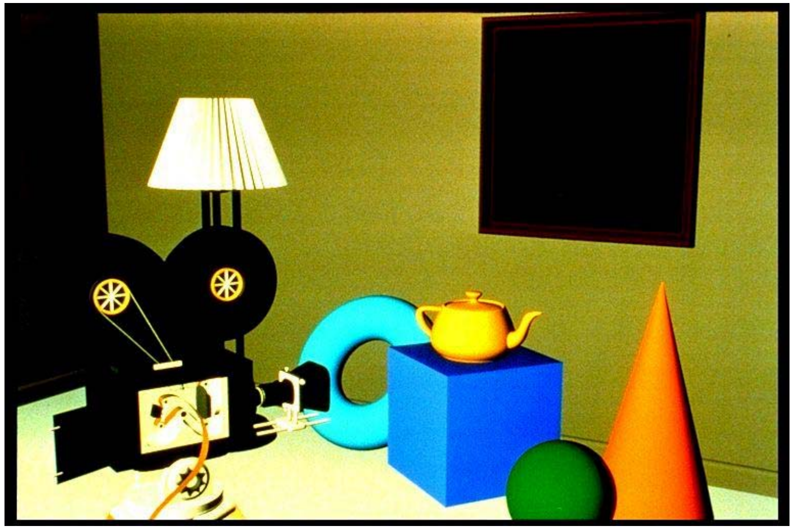
Texturing

Simple Illumination

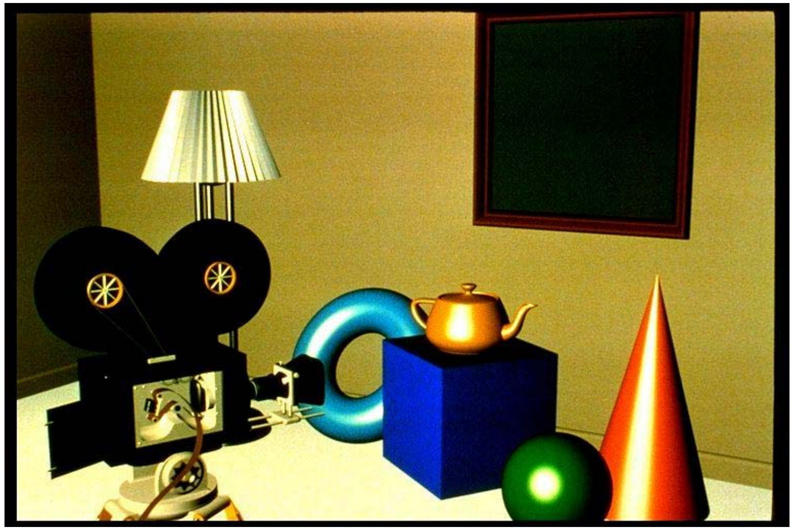
- **No illumination**
- **Constant colors**



- **Parallel light**
- **Diffuse reflection**



Standard Illumination



- **Parallel light**
- **Specular reflection**

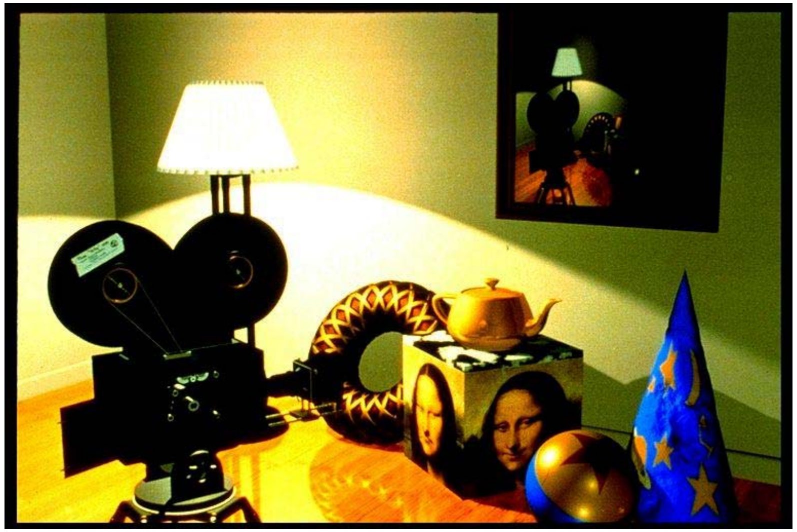
- **Multiple local light sources**
- **Different BRDFs**

Object properties constant over surface

Texturing

Locally varying object characteristics

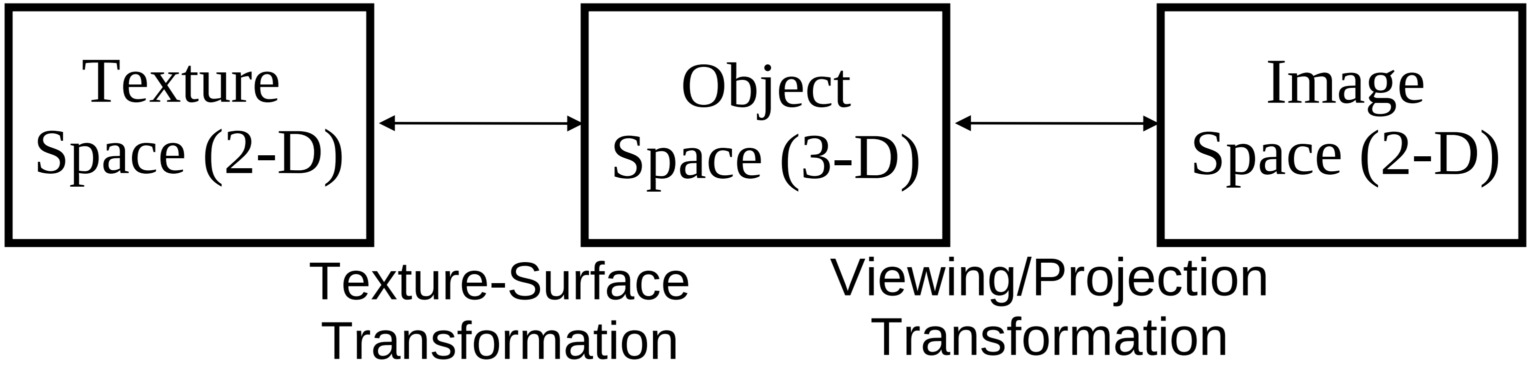
- **2D Image Textures**
- **Shadows**
- **Bump-Mapping**
- **Reflection textures**



Texture-modulated Quantities

- **Modulation of object surface properties**
- **Reflectance**
  - Color (RGB), diffuse reflection coefficient kd
  - Specular reflection coefficient ks
- **Opacity (α)**
- **Normal vector**
  - N(P)= N(P+ t N) or N= N+dN
  - „Bump mapping“ or „Normal mapping“
- **Geometry**
  - P= P + dP
  - „Displacement mapping“
- **Distant illumination**
  - “Environment mapping“, “Reflection mapping“

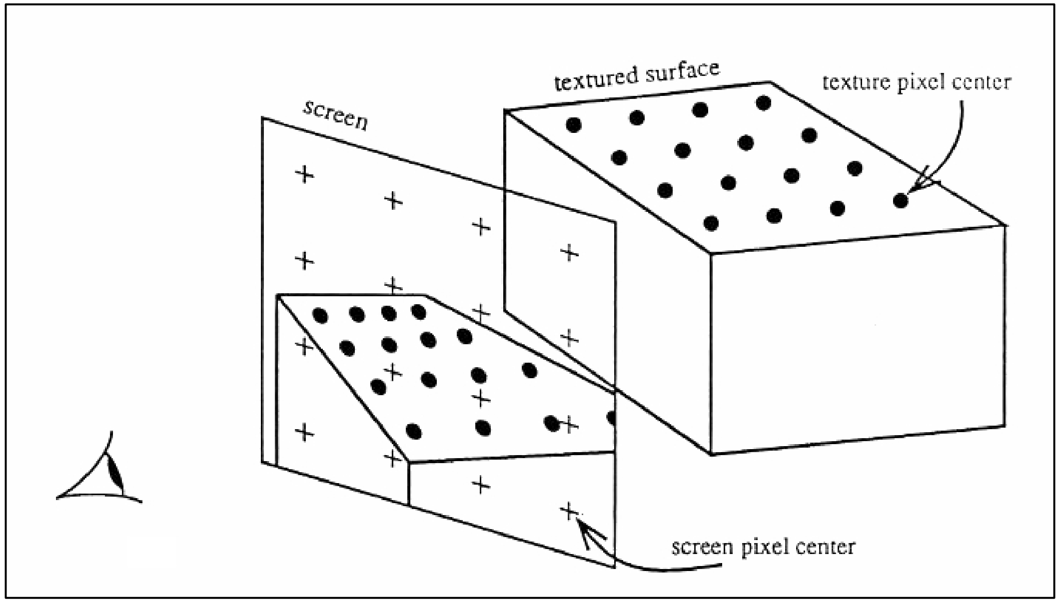
Texture Mapping Transformations

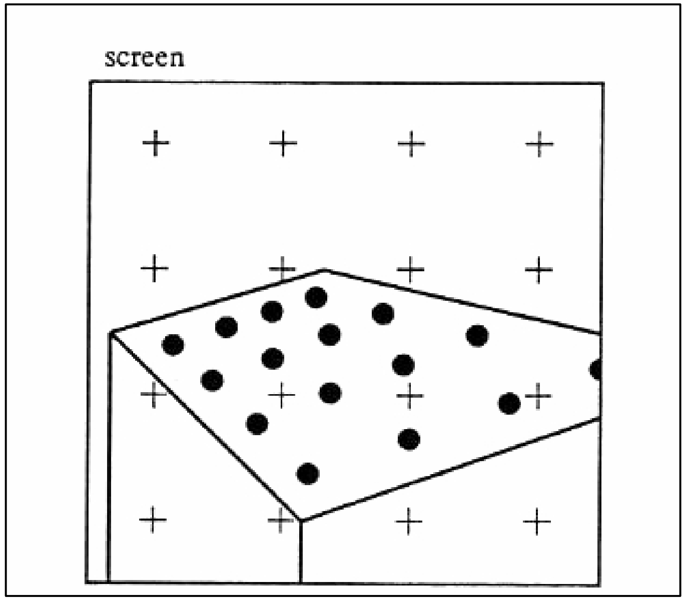


The texture is mapped onto a surface in 3-D object space, which is then mapped to the screen by the viewing projection. These two mappings are composed to find the overall 2-D texture space to 2-D image space mapping, and the intermediate 3-D space is often forgotten. This simplification suggests texture mapping's close ties with image warping and geometric distortion.

Texture space (u,v)
Object space (x_o,y_o,z_o)
Screen space (x,y)

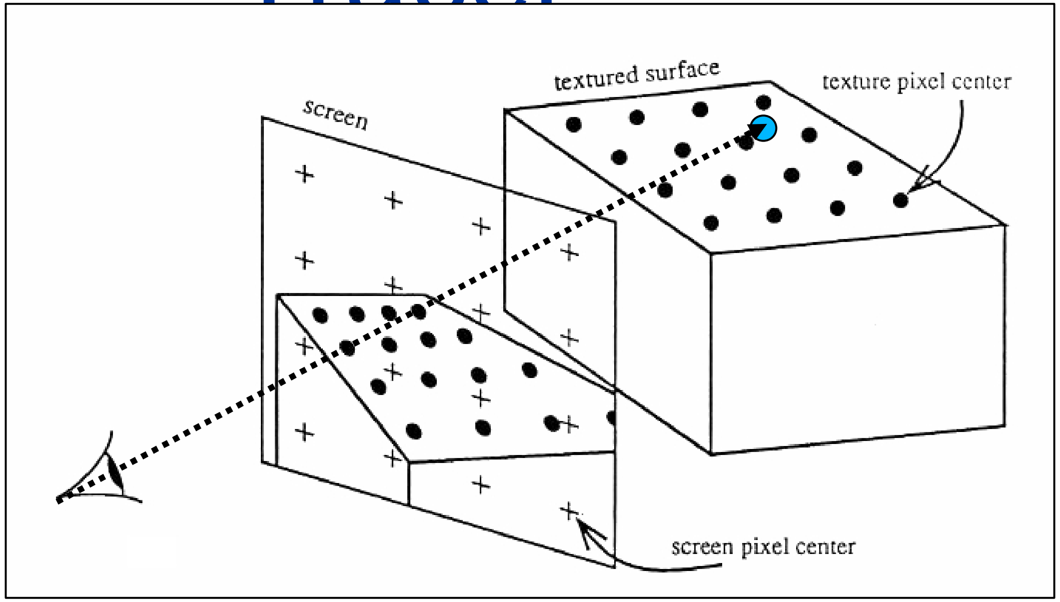
2D Texturing

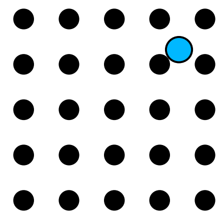




- **2D texture mapped onto object**
- **Object projected onto 2D screen**
- **2D→2D: warping operation**
- **Uniform sampling ?**
- **Hole-filling/blending ?**

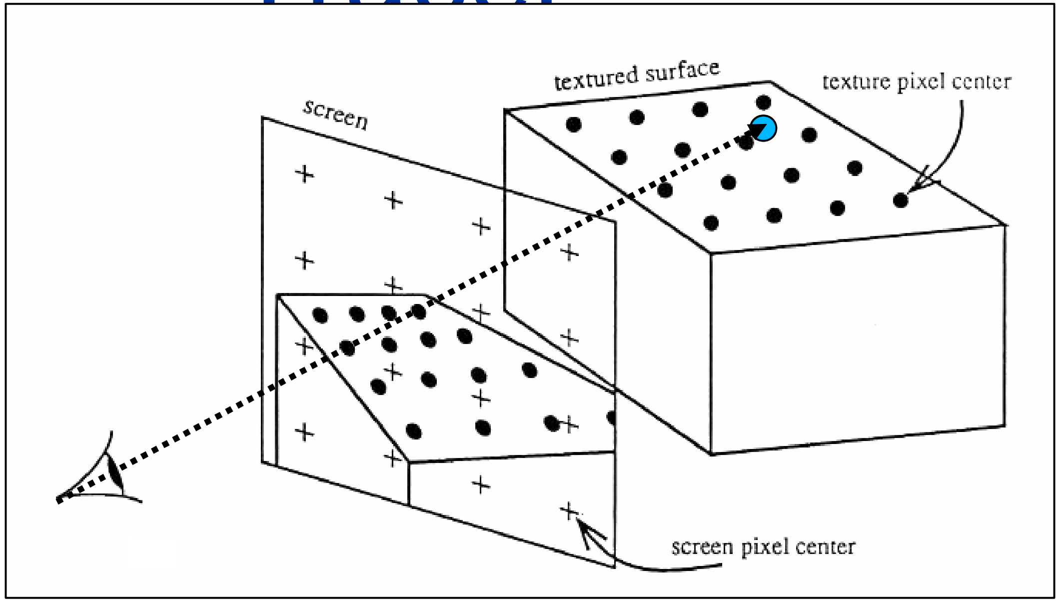
Texture Mapping in a Ray Tracer

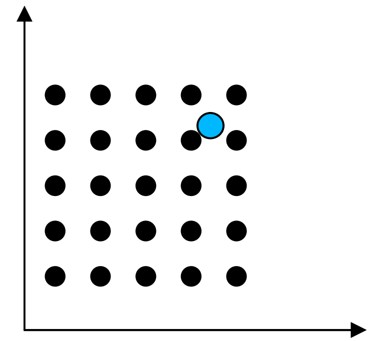




- **approximation:**
  - ray hits surface
  - surface location corresponds to coordinate inside a texture

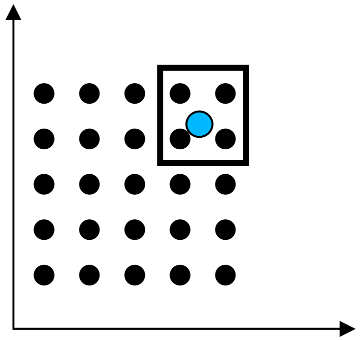
Texture Mapping in a Ray Tracer

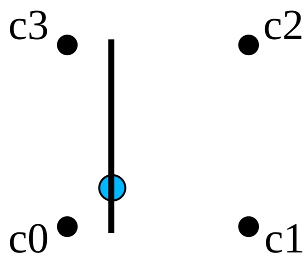




- **approximation:**
  - ray hits surface
  - surface location corresponds to coordinate inside a texture

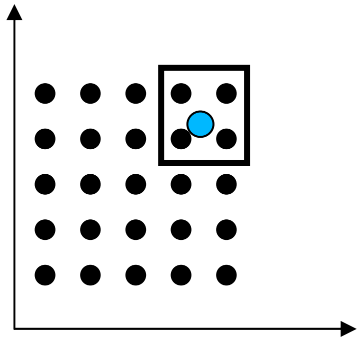
Interpolation 1D

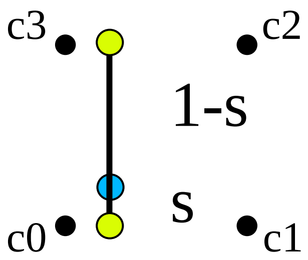




- **How to interpolate the color of the pixel?**

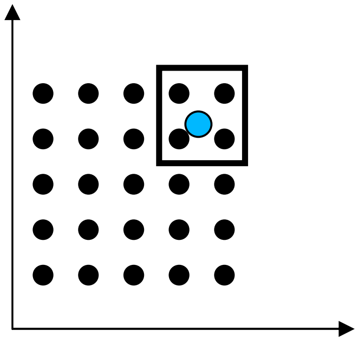
Interpolation 1D

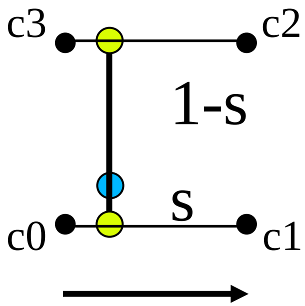




- **How to interpolate the color of the pixel?**

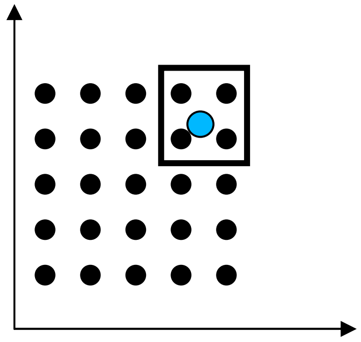
Interpolation 2D

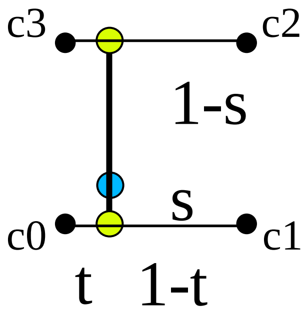




- **How to interpolate the color of the pixel?**

Interpolation 2D





- **How to interpolate the color of the pixel?**
- **1D: i0 = (1-t)c0 + tc1**
  i1 = (1-t)c3 + tc2
- **2D: c = (1-s) i0 + s i1**