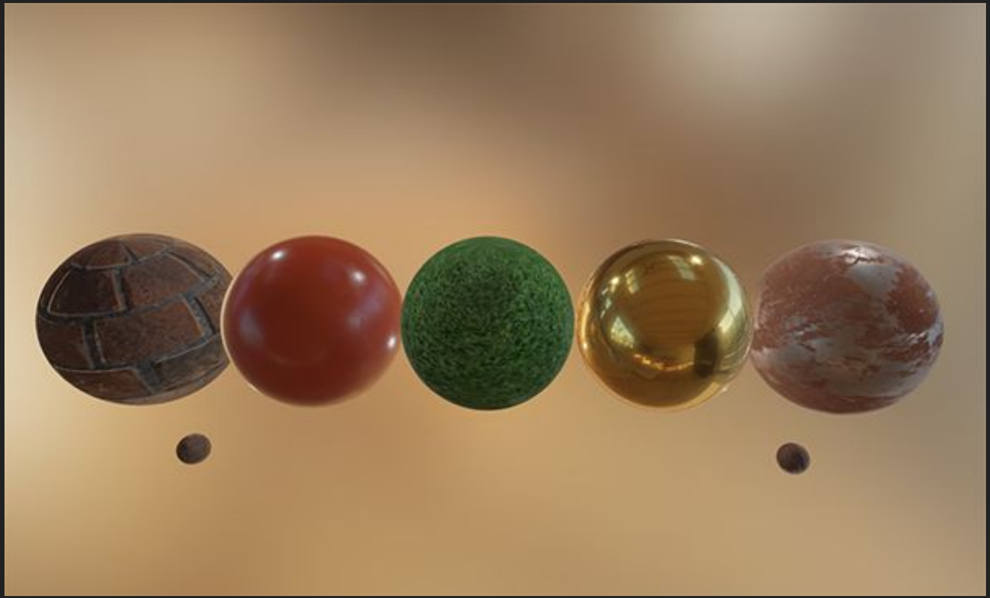
Introduction to Physically Based Rendering

Walt O’Connor

Overview

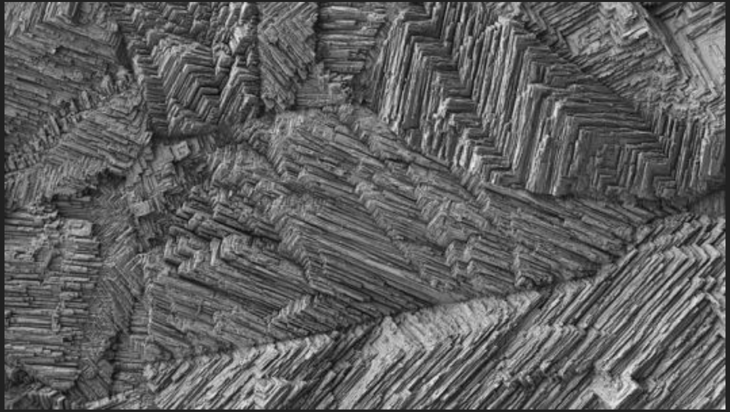
1. Microfacet Model
2. Energy Conservation
3. Metallicity
4. Reflectance Equation
    a. Cook-Torrance BRDF
    b. Cook-Torrance Specular
    c. Trowbridge-Reitz GGX
    d. Schlick Approximation
    e. Smith's Method
    f. Fresnel Equation
    g. Putting it All Together
5. Only half the picture? (normalmaps/cubemaps)

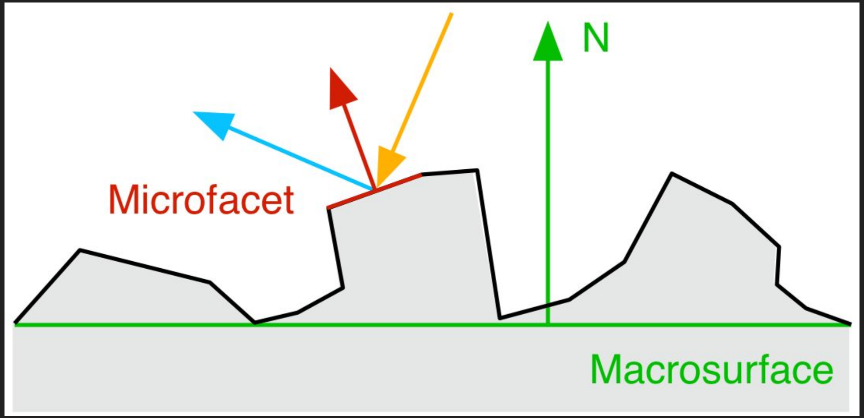


Microfacet Model

Model surfaces as rough collection of infinitely small mirrors.

New models have microfacet be Lambertian surfaces, end up at almost exactly the same end result.

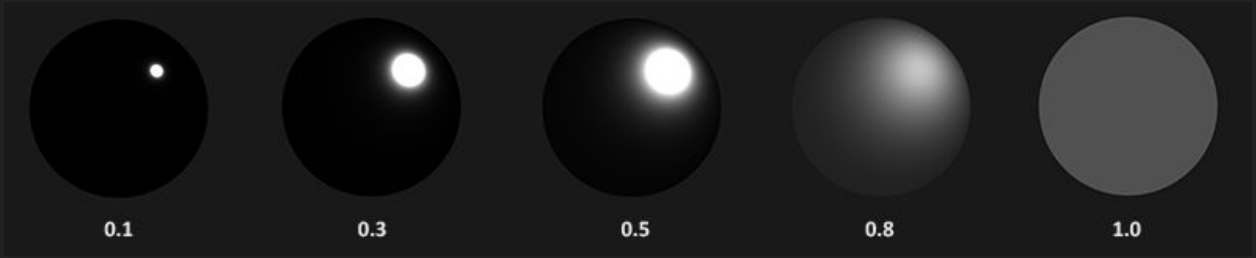




Consequences of Microfacet Model

Only modeling microfacet reflection:

Anyone know why the entire ball is illuminated at roughness = 1.0?



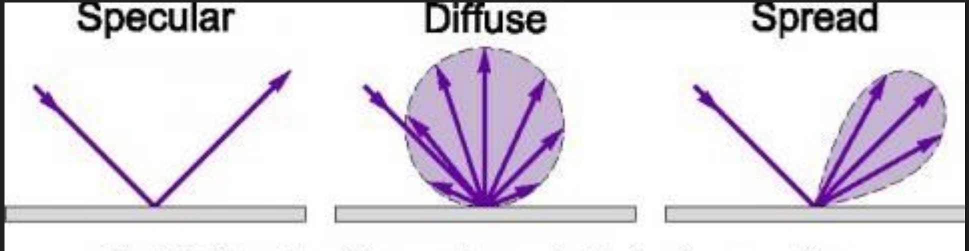


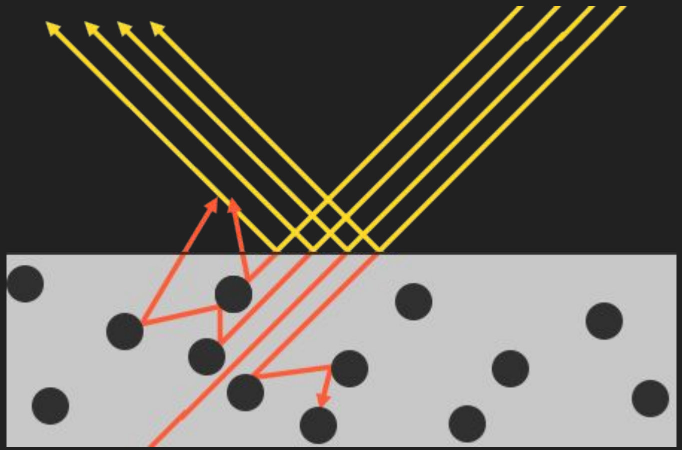
Fig. 3.2 Specular, diffuse, and spread reflection from a surface.

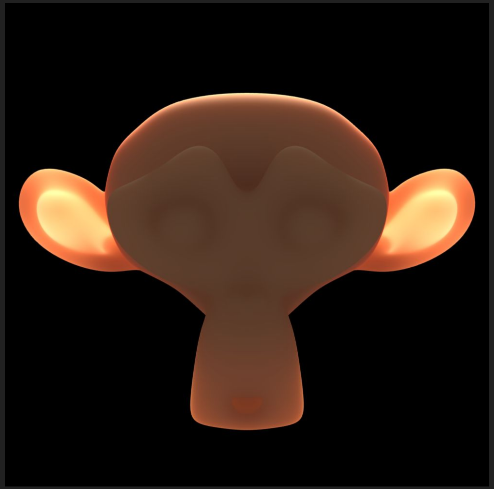
Energy Conservation

Light is either reflected or refracted.

Refracted light enters the object and exits nearby.

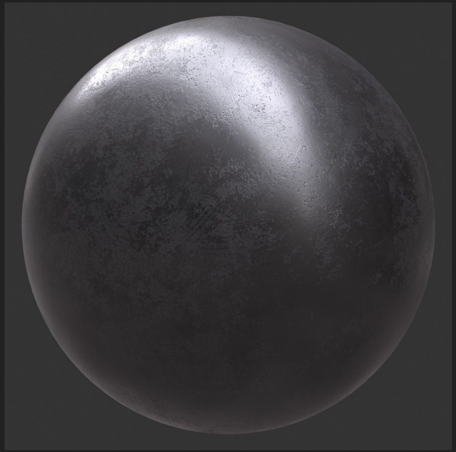
Assume opaque objects since transparent objects need refractance equations and complex surfaces like skin need subsurface scattering.





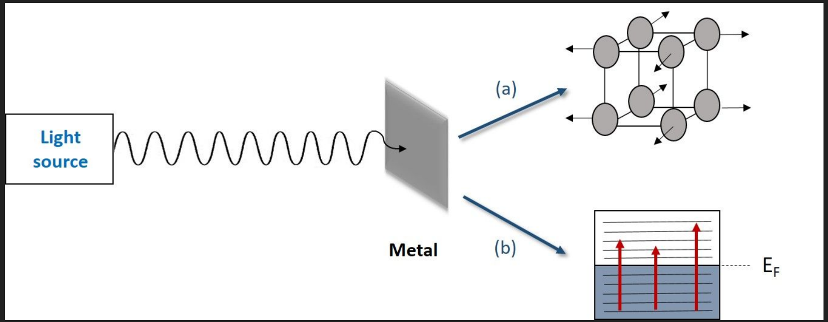
Metallicity

Why do metals look different than plastic?



Metal's can't refract light, only reflect it. They absorb ALL light that is not reflected.

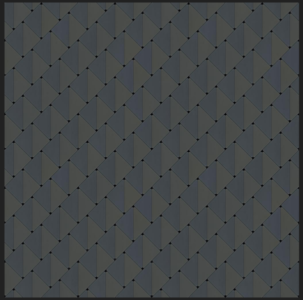
Why?

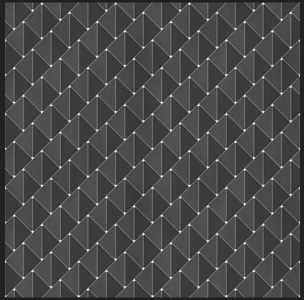


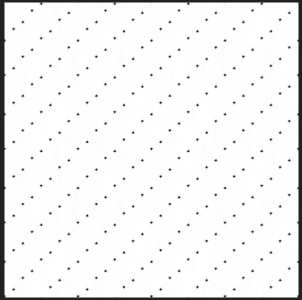
(hint: Non metallics are called dielectrics in graphics programming for a reason)

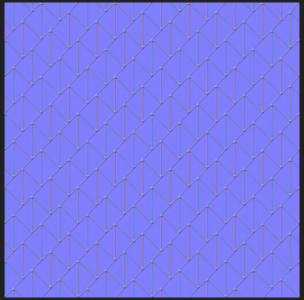
Inputs to PBR Shader

(how are these generated)









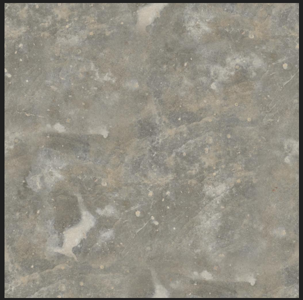
Albedo/

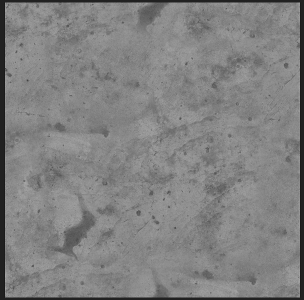
F0 (cover F0 later)

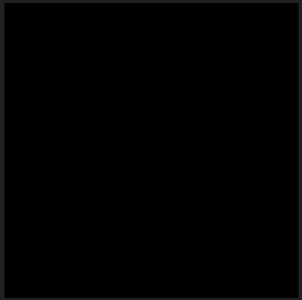
Roughness

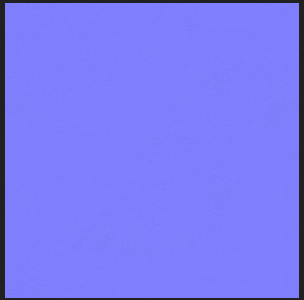
Metalness

Normalmap









Reflectance Equation

Integral needed for light sources of arbitrary shape (consider fluorescent tube light)

For point lights becomes a summation.

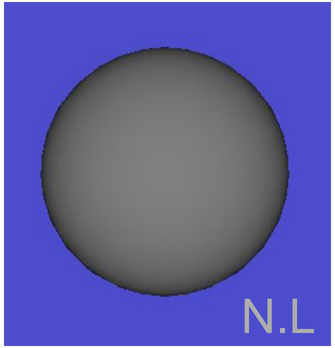
fr = reflective distribution function

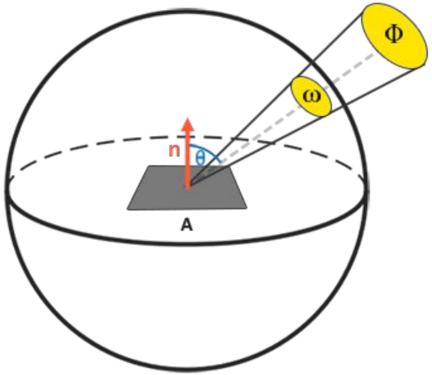
$$L_o(p, \omega_o) = \int\limits_{\Omega} f_r(p, \omega_i, \omega_o) L_i(p, \omega_i) n \cdot \omega_i d\omega_i$$

$\omega_i =$ direction to incoming light
$\omega_o =$ direction of the outgoing light
$p =$ position in space
$n =$ surface normal

$L_i(p, position_i) = \frac{intensity_i}{(p - position_i)^2}$ where

$p =$ current point (fragment) position
$position_i =$ current light position
$intensity_i =$ current light intensity

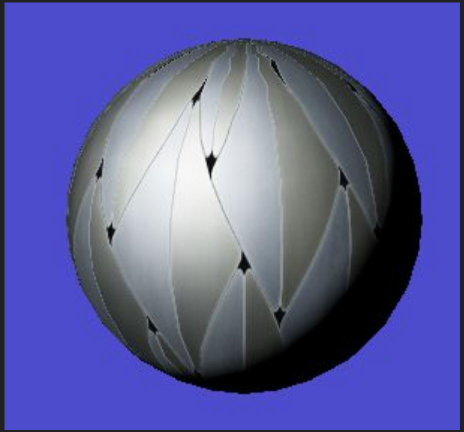




Cook Torrance Bidirectional Reflective Distribution Function

Works kind of like phong shading, except kd and ks are determined analytically from the material properties.

$$f_r = \frac{k_d * albedo}{\pi} + k_s f_{cook-torrance-specular}$$



Cook Torrance Specular

D? F? G?

Need to model three things:

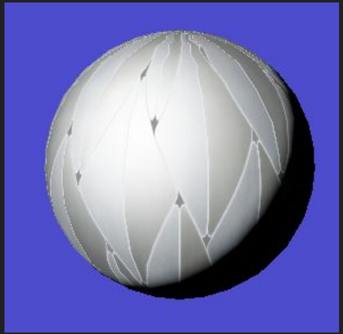
Microfacets reflecting light (D)

Microfacets blocking each other (G)

Reflection trending to infinity at the edge of the object (F)

$$f_{cook-torrance-specular} = \frac{DFG}{4(\omega_o \cdot n)(\omega_i \cdot n)}$$

ω_i = direction to incoming light
ω_o = direction of the outgoing light
p = position in space
n = surface normal



Trowbridge-Reitz GGX Normal Distribution Function (N)

Analytically determined approximation of roughness.

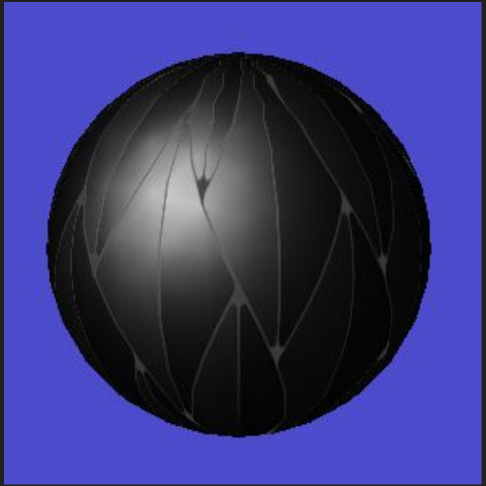
h is vector halfway between light direction and view direction (ideal angle for reflection)

If actual normal matches ideal reflection normal, minimize effect of roughness.

Can see how important per fragment roughness is just from this.

$$NDF_{TR-GGX}(n, h, roughness) = \frac{roughness^2}{\pi((n \cdot h)^2(roughness^2 - 1) + 1)^2}$$

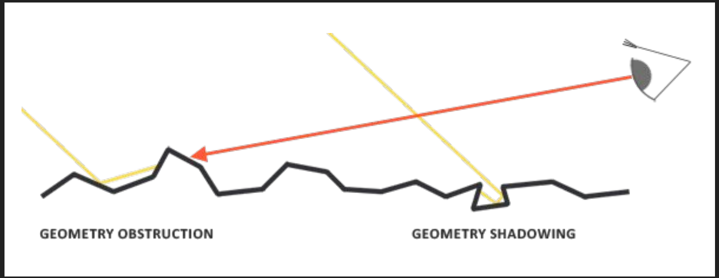
$$h = \frac{l + v}{||l + v||}$$

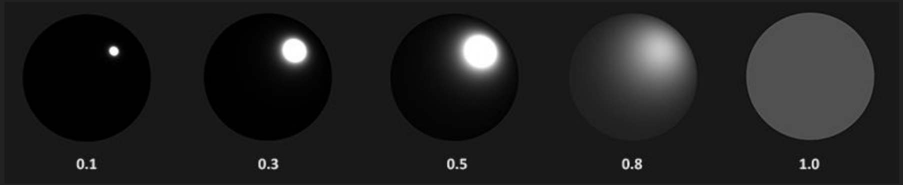


Schlick GGX Geometry Approximation (G part 1)

Models microsurface geometry blocking light being reflected.

Mostly makes an impact at the transition from light to shadow, where there's actually a good chance light is getting blocked





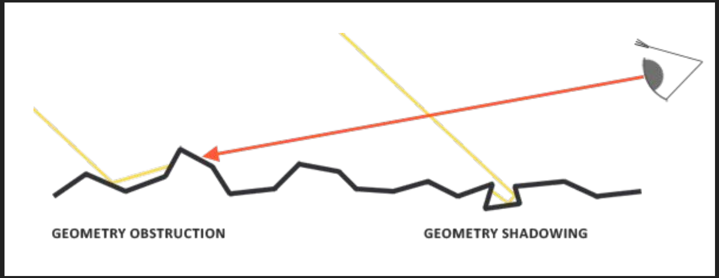
$$G_{Schlick-GGX}(n, v, roughness) = \frac{n \cdot v}{(n \cdot v)(1 - \frac{(roughness+1)^2}{8}) + \frac{(roughness+1)^2}{8}}$$

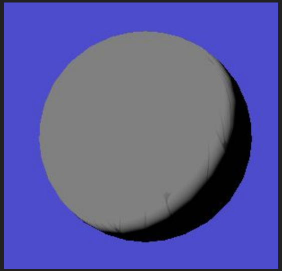
Smiths Approximation for Geometry

Problem: Microsurfaces block light rays coming in from the light AND block light rays reflecting to the camera.

Model both and multiply.

Really take a look at the transition from light to shadow on the gray sphere.





$$G_{smiths-method} = (n, v, l, roughness) = G_{Schlick-GGX}(n, v, roughness) * G_{Schlick-GGX}(n, l, roughness)$$

Fresnel Equation (Other Schlick Approximation) (F)

Surfaces become infinitely reflective when viewed perpendicularly *regardless of roughness*.

F0 computed using index or refraction. F0 normally between 0.03 and 0.06 for dielectric materials.

Problem: metals don't refract.

Solution: metal's don't refract so they don't need any albedo information. Pack F0 data for metal in to the albedo texture.

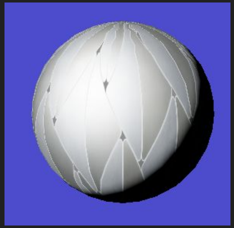
$$F_{Schlick}(h, v, F_0) = F_0 + (1 - F_0)(1 - (h \cdot v))^5$$

Material	F_0 (Linear)	F_0 (sRGB)	Color
Water	(0.02, 0.02, 0.02)	(0.15, 0.15, 0.15)	
Plastic / Glass (Low)	(0.03, 0.03, 0.03)	(0.21, 0.21, 0.21)	
Plastic High	(0.05, 0.05, 0.05)	(0.24, 0.24, 0.24)	
Glass (high) / Ruby	(0.08, 0.08, 0.08)	(0.31, 0.31, 0.31)	
Diamond	(0.17, 0.17, 0.17)	(0.45, 0.45, 0.45)	
Iron	(0.56, 0.57, 0.58)	(0.77, 0.78, 0.78)	
Copper	(0.95, 0.64, 0.54)	(0.98, 0.82, 0.76)	
Gold	(1.00, 0.71, 0.29)	(1.00, 0.86, 0.57)	
Aluminium	(0.91, 0.92, 0.92)	(0.96, 0.96, 0.97)	
Silver	(0.95, 0.93, 0.88)	(0.98, 0.97, 0.95)	

Putting it All Together

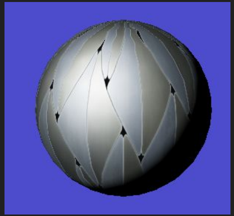
Compute the specular

$$f_{cook-torrance-specular} = \frac{DFG}{4(\omega_o \cdot n)(\omega_i \cdot n)}$$



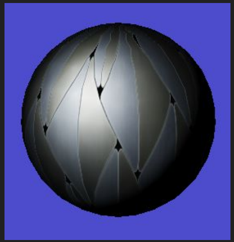
Add the lambertian diffuse

$$f_r = \frac{k_d * albedo}{\pi} + k_s f_{cook-torrance-specular}$$



Apply lighting

$$L_o(p, \omega_o) = \int_{\Omega} f_r(p, \omega_i, \omega_o) L_i(p, \omega_i) n \cdot \omega_i d\omega_i$$



… Tone Map + Gamma Correct

Only Half the Picture

You need normal maps (done by perturbing the view vector to simulate the surface being at a different angle to the camera) to make surfaces appear 3D.

You need cubemaps/other image based lighting techniques to make an object appear to reflect the world (or use raytracing).