Microsoft DIRECTX

NAUGHTY DOG

Reflectance Models

- Blinn-Phong
- Cook-Torrance
- Banks
- Ward
- Ashikhmin-Shirley
- Lafortune
- Oren-Nayar
- Thin-Film Interference
- Diffraction (simplified Stam)
- Hand-Painted

Microsoft DIRECTX NAUGHTY DOG

Blinn-Phong

- **Commonly used in games as:**

$$f(\mathbf{V},\mathbf{L}) = k_d\,dot(\mathbf{N},\mathbf{L}) + k_s\,dot(\mathbf{N},\mathbf{H})^n$$

- **Recast as BRDF with our notation:**

$$f_r(\omega_i,\omega_e) = k_d + \frac{k_s}{\cos\theta_i}(\cos\theta_h)^n$$

The top version is the one commonly used in games. It is not in the form of a BRDF (among other things, it includes the clamped cosine factor), so we will rewrite the same function as a BRDF using the previous notation from this talk.

Blinn-Phong

$$f_r(\omega_i, \omega_e) = k_d + \frac{k_s}{\cos\theta_i}(\cos\theta_h)^n$$

- Not reciprocal
- No shadowing / masking
- No specular/diffuse tradeoff at glancing angles
- Not normalized!
  - Hard to ensure energy conservation
  - Hard to ensure material is bright enough

The absence of reciprocity, Fresnel, etc. can be minor issues depending on the material we are trying to simulate. The lack of normalization is more of a problem.

Microsoft DIRECTX® NAUGHTY DOG

BRDF Normalization

- Why is this important?
  - Needed for global illumination algorithms to converge, usually not a concern for games
- Important for proper "HDR BRDFs"
- Normalized terms make it possible to directly control a surface's reflectance
  - Range of valid parameters is clear
- With non-normalized terms, risk of
  - Making the surface 'glow'
  - Making the surface too dark

Making the surface too dark is what commonly happens. In the Phong example, normalized parameters enable controlling smoothness and brightness separately – the non-normalized Phong mixes the two together.

Microsoft DIRECTX

NAUGHTY DOG

BRDF Normalization

- For each un-normalized BRDF term (diffuse, specular)
  - Find $C_\mathbf{R}$ such that $R(\omega_i) \le C_\mathbf{R}$ for all ω_i
  - Divide term by $C_\mathbf{R}$
  - And multiply it by a reflectance parameter like ρ_d, $R_F(0)$, or $R_F(\alpha_h)$

If the term already has a reflectance parameter included, you may need to remove it before computing the upper bound CR.

Microsoft DIRECTX® NAUGHTY DOG

BRDF Normalization

- For each un-normalized BRDF term (diffuse, specular)
  - Find $C_{\mathbf{R}}$ such that $R(\omega_i) \leq C_{\mathbf{R}}$ for all ω_i
  - Divide term by $C_{\mathbf{R}}$
  - And multiply it by a reflectance parameter like ρ_d, $R_F(0)$, or $R_F(\alpha_h)$

The upper bound should be conservative, to enforce conservation of energy. But it also needs to be as tight, to ensure that the total reflectance of the material is as close to the user-set parameters as possible. There usually isn't an analytical form for an exact bound, so some experimentation and judgment is needed.

Microsoft DIRECTX NAUGHTY DOG

Normalized Blinn-Phong

$$f_r(\omega_i, \omega_e) = \frac{\rho_d}{\pi} + \frac{(n+4)R_F(0)}{8\pi\cos\theta_i}(\cos\theta_h)^n$$

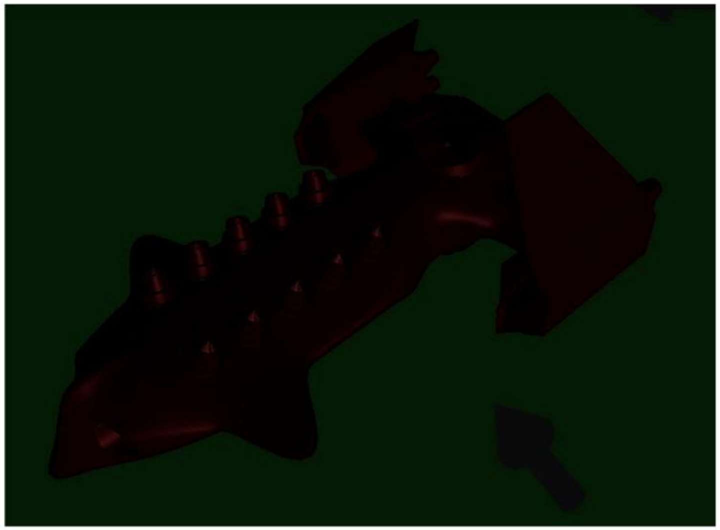
- **Now diffuse and specular reflectance can be set directly**
- **Conservation of energy:** $\boldsymbol{\rho_d + R_F(0) \leq 1}$

Actual reflectance will be just a little bit lower than the reflectance parameter due to the conservative normalization factor, but it will be close.

Microsoft DIRECTX NAUGHTY DOG

Blinn-Phong

Non-normalized
5% specular color, 32 power



Normalized
5% specular color, 32 power

Plastic spaceship; RF(0)=5%. Non-normalized version is too dark. Increasing RF(0) would change Fresnel behavior, and the value would have to be changed again if surface smoothness (specular power) is changed. In the normalized version the material reflectance and smoothness can be adjusted separately.

Microsoft DIRECTX NAUGHTY DOG

Cook-Torrance

- **Microfacet BRDF**

$$f_r(\omega_i, \omega_e) = (1 - s)\frac{\rho_d}{\pi} + s\,\frac{p(\omega_h)G(\omega_i, \omega_e)R_F(\alpha_h)}{\pi \cos\theta_i \cos\theta_e}$$

- **Reciprocal**
- **Shadowing / masking**
- **Specular/diffuse tradeoff**
  - **Specular reflectance increases at glancing angles, but diffuse reflectance doesn't decrease**
  - **Not energy-conserving**
- **Not well-normalized**

We can see this is in the same form as the microfacet BRDF we saw earlier. S is a factor between 0 and 1 that controls the relative intensity of the specular and diffuse reflection. Quite a bit of energy is lost via the geometry factor – the actual reflectance is quite a bit lower than the parameters would indicate.

Microsoft DIRECTX®   NAUGHTY DOG

Cook-Torrance

- **The Cook-Torrance paper recommends using the Beckmann NDF:**

$$p(\omega_h) = \frac{1}{m^2 \cos^4 \theta_h} e^{-\left\{\frac{\tan^2 \theta_h}{m^2}\right\}}$$

M is a parameter which controls roughness of the surface.

But it is suggested that a variety of other NDFs can work.

Microsoft DIRECTX®   NAUGHTY DOG

Normal Distribution Functions

- **Gaussian**
  - **(not normalized)** $$p(\omega_h) = \mathrm{e}^{-(c_G \theta_h)^2}$$
- **Phong** $$p(\omega_h) = \frac{n+1}{2\pi} \cos^n \theta_h$$
- **Trowbridge-Reitz**
  - **(not normalized)** $$p(\omega_h) = \left(\frac{{c_{TR}}^2}{\cos^2 \theta_h \left({c_{TR}}^2 - 1\right) + 1} \right)^2$$

Phong can be seen as a normal distribution function. These functions have subtle differences – worth trying out, especially the cheap ones. Normalization factors need to be added for the Gaussian and T-R NDFs; note that a normalized NDF doesn't guarantee a normalized BRDF.

Microsoft DIRECTX®

NAUGHTY DOG

Cook-Torrance

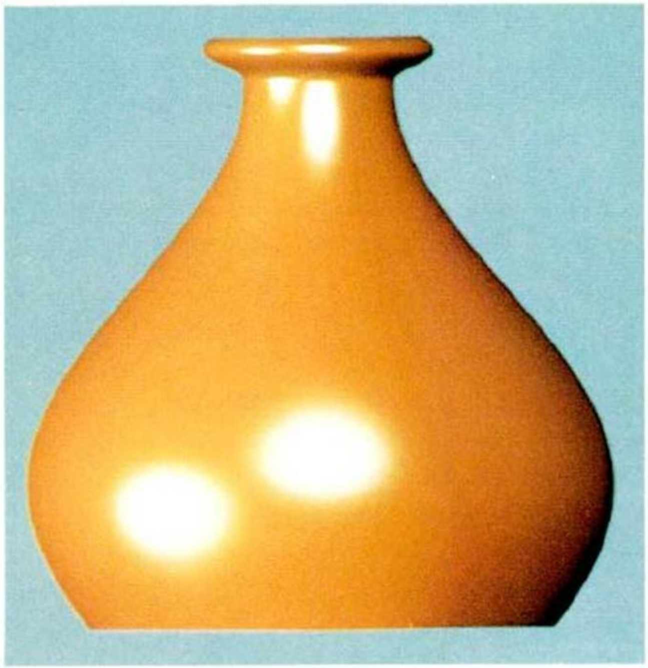
- **Geometry Term:**

$$G(\omega_i, \omega_e) = \min\left\{1, \frac{2\cos\theta_h \cos\theta_e}{\cos\alpha_h}, \frac{2\cos\theta_h \cos\theta_i}{\cos\alpha_h}\right\}$$

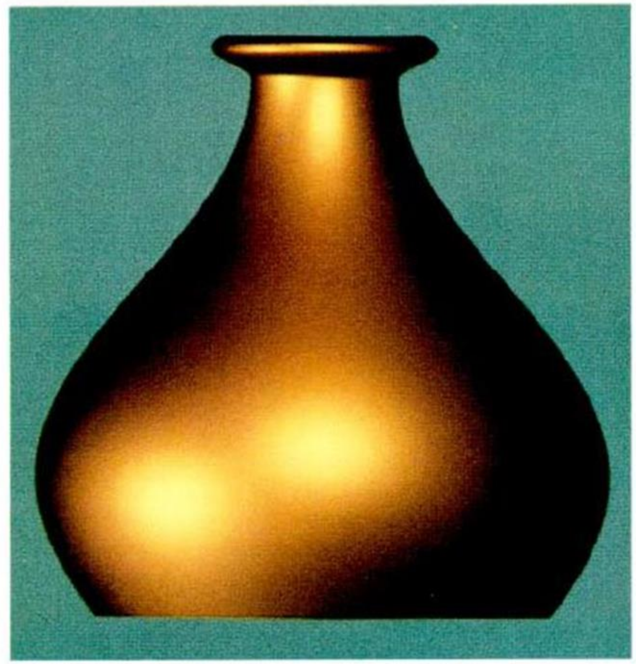
This geometry term is based on a shadowing/masking model of long thin V-shaped grooves. It is not very consistent given that it is supposed to be used on isotropic surfaces.

Cook-Torrance

Plastic



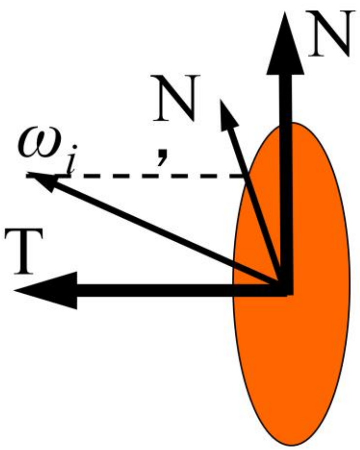
Metal

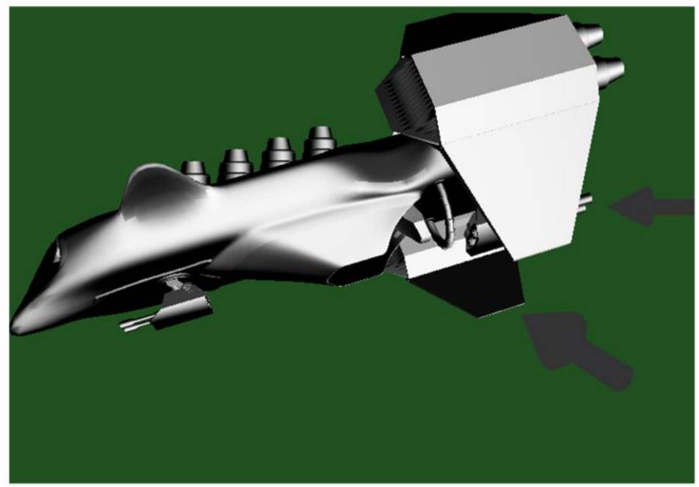


IMAGES BY R. COOK AND K. TORRANCE

Banks

- Anisotropic version of Blinn-Phong
  - Surface model composed of threads
  - Uses projection of light vector onto tangent plane instead of surface normal





Aside from the new normal vector, this is almost exactly the same as Blinn-Phong. One more difference is that when the light is not in the hemisphere about the original surface normal, the lighting is set to 0 (“self-shadowing” term). This is the only effect the surface normal has on this model.

Microsoft DIRECTX® NAUGHTY DOG

Ward - Isotropic

$$f_r(\omega_i, \omega_e) = \frac{\rho_d}{\pi} + R_F(0) \frac{e^{-\left\{\frac{\tan^2\theta_h}{m^2}\right\}}}{4\pi\, m^2 \sqrt{\cos\theta_i \cos\theta_e}}$$

- **No shadowing / masking**
- **No specular/diffuse tradeoff at glancing angles**
- **Reciprocal, conserves energy, well normalized**

This is another “pseudo-microfacet model”. It uses a Beckmann NDF, but no explicit Fresnel or masking / shadowing. It is well normalized however.

Ward - Anisotropic

$$f_r(\omega_i, \omega_e) = \frac{\rho_d}{\pi} + R_F(0) \frac{e^{-\left\{\tan^2\theta_h \left(\frac{\cos^2\phi}{m_u^2} + \frac{\sin^2\phi}{m_v^2}\right)\right\}}}{4\pi\, m_u m_v \sqrt{\cos\theta_i \cos\theta_e}}$$

The only change is in the NDF, which is now anisotropic.

Microsoft DIRECTX®

NAUGHTY DOG®

Ward - Anisotropic

$$f_r(\omega_i, \omega_e) = \frac{\rho_d}{\pi} + R_F(0)\frac{e^{-2\left\{\frac{\left(\frac{\cos\alpha_u}{m_u}\right)^2 + \left(\frac{\cos\alpha_v}{m_v}\right)^2}{1+\cos\theta_h}\right\}}}{4\pi\, m_u m_v \sqrt{\cos\theta_i \cos\theta_e}}$$

This approximation is much faster to compute since it uses mostly dot-products.

Ward - Anisotropic

IMAGE BY G. WARD

Although it is missing some of the features of the physically-based BRDFs, the images still look quite nice.

Ashikhmin-Shirley

- Specular and diffuse terms

$$f_r(\omega_i,\omega_e)= f_d(\omega_i,\omega_e)+ f_s(\omega_i,\omega_e)$$

- No shadowing / masking
- Specular/diffuse tradeoff at glancing angles
- Reciprocal, conserves energy, well normalized

Another ‘pseudo-microfacet’ model

Microsoft DIRECTX® NAUGHTY DOG®

Ashikhmin-Shirley

- **Diffuse term**

$$f_d(\omega_i,\omega_e)=\frac{28\rho_d}{23\pi}\left(1-R_F(0)\right)\left(1-\left(1-\frac{\cos\theta_i}{2}\right)^5\right)\left(1-\left(1-\frac{\cos\theta_e}{2}\right)^5\right)$$

- **Trades off reflectance with the specular term at glancing angles**
- **Without losing reciprocity or energy conservation**

Ashikhmin-Shirley

- **Specular term**

$$f_s(\omega_i,\omega_e)=\frac{\sqrt{(n_u+1)(n_v+1)}}{8\pi}\frac{\cos\theta_h^{\,n_u\cos^2\phi+n_v\cos^2\phi}}{\cos\alpha_h\max(\cos\theta_i,\cos\theta_e)}R_F(\alpha_h)$$

Note max(cos, cos) term in denominator

Ashikhmin-Shirley

- **Specular term**

$$f_s(\omega_i,\omega_e)=\frac{\sqrt{(n_u+1)(n_v+1)}}{8\pi}\frac{\cos\theta_h^{\frac{n_u\cos^2\alpha_u+n_v\cos^2\alpha_v}{1+\cos^2\theta_h}}}{\cos\alpha_h\max(\cos\theta_i,\cos\theta_e)}R_F(\alpha_h)$$

implementation-friendly form

Ashikhmin-Shirley

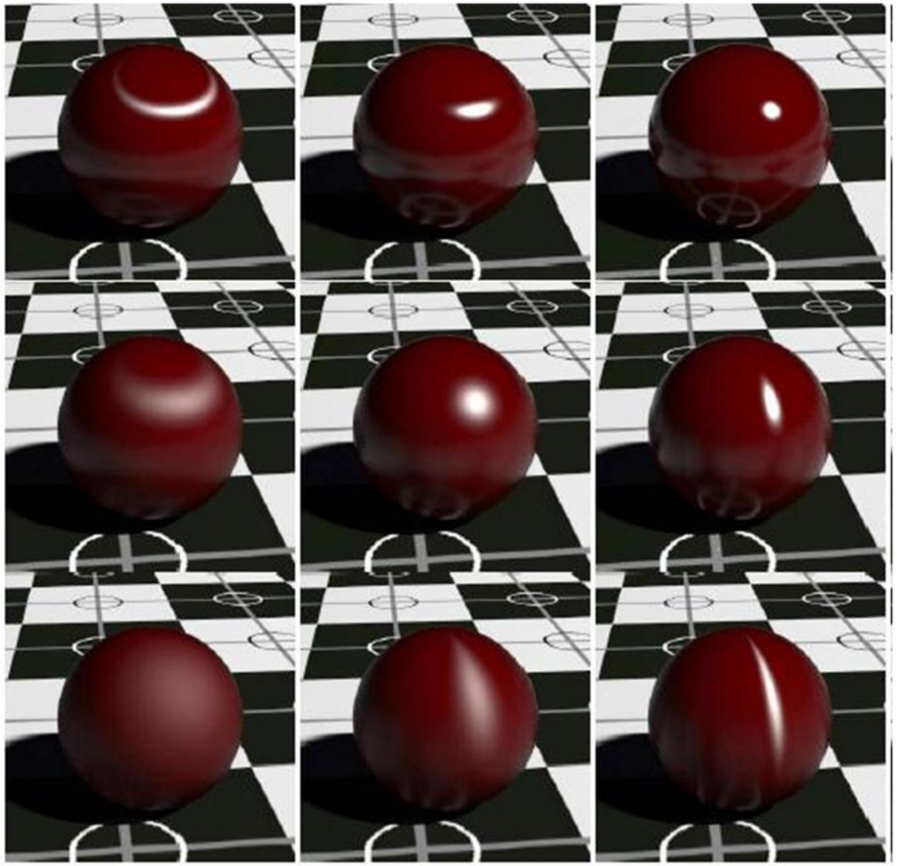


IMAGE BY M. ASHIKHMIN AND P. SHIRLEY

Ashikhmin-Shirley

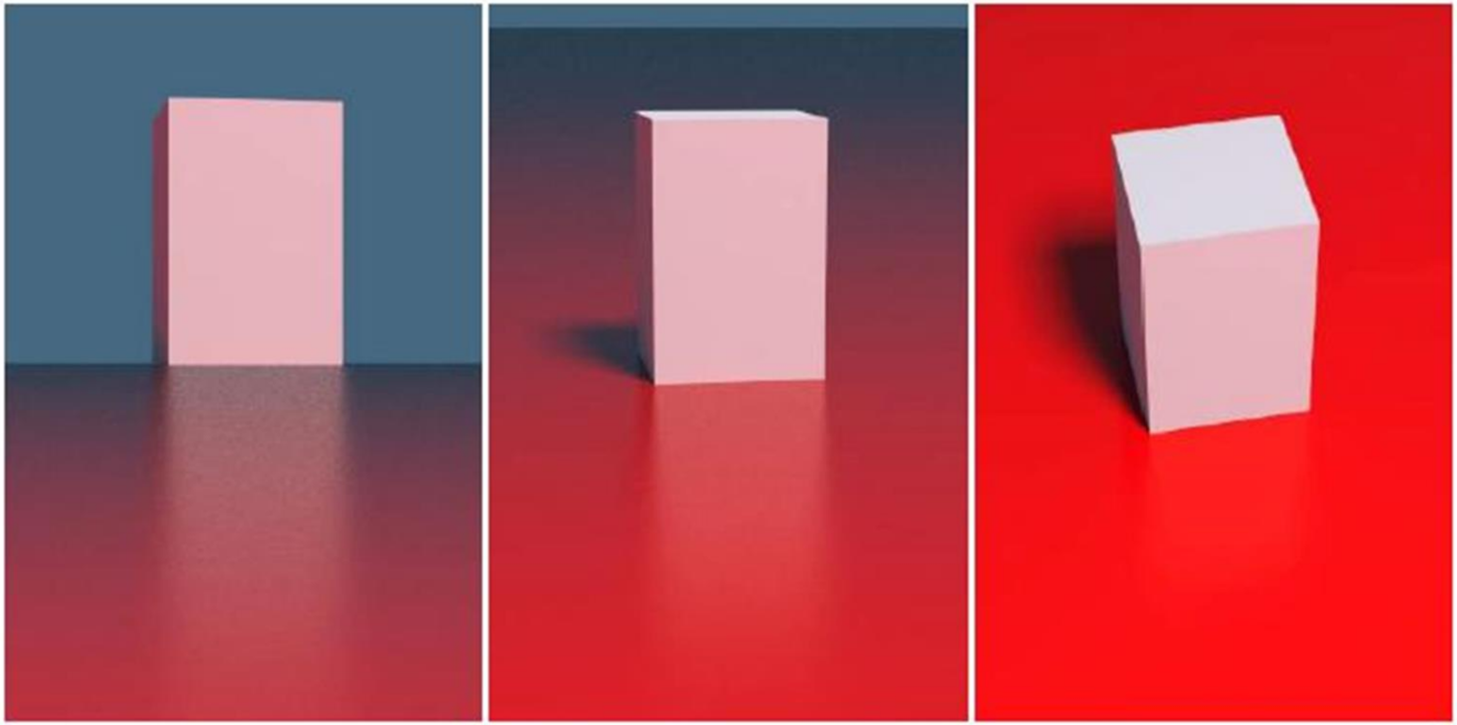


IMAGE BY M. ASHIKHMIN AND P. SHIRLEY

Lafortune

- **Generalization of reciprocal version of original Phong (not Blinn-Phong) specular term:**

$$f_s(\omega_i, \omega_e) = \frac{(n+1)R_F(0)}{2\pi} dot(\mathbf{V}, \mathbf{R})^n$$

Lafortune

- **In local frame, reflection operator is just multiplying x and y components by -1:**

$$f_s(\omega_i, \omega_e) = \frac{(n+1)R_F(0)}{2\pi}\left((-1)\mathbf{V}_x\mathbf{L}_x + (-1)\mathbf{V}_y\mathbf{L}_y + 1\mathbf{V}_z\mathbf{L}_z\right)^n$$

This requires that the light and view direction are in the local frame of the surface (which the BRDF definition assumes anyway, it's just that with many BRDFs you can get away with not actually transforming them into that space).

Lafortune

- Generalize to one spectral (RGB) constant and four scalar constants per term, add several terms (lobes) :

$$f_s(\omega_i, \omega_e) = \sum R\left(C_x \mathbf{V}_x \mathbf{L}_x + C_y \mathbf{V}_y \mathbf{L}_y + C_z \mathbf{V}_z \mathbf{L}_z\right)^n$$

Lafortune

- Lambertian:
  - $R = \rho_d/\pi,\ n = 0$
- Non-Lambertian diffuse:
  - $R = \rho_d,\ C_x = C_y = 0,\ C_z = (n+2)/2\pi$
- Off-specular reflection:
  - $C_z < -C_x = -C_y$
- Retro-reflection:
  - $C_x > 0,\ C_y > 0,\ C_z > 0$
- Anisotropy:
  - $C_x \neq C_y$

Besides the standard cosine lobe, this can handle many other cases.

Lafortune

- Very general, not too expensive to compute (unless a lot of lobes are used), but has very unintuitive parameters
- Best used to fit measured data or some other model with more intuitive parameters

Lafortune

IMAGE BY D. MCALLISTER, A. LASTRA AND W. HEIDRICH

Lafortune

IMAGE BY D. MCALLISTER, A. LASTRA AND W. HEIDRICH

Microsoft DIRECTX® NAUGHTY DOG

Oren-Nayar

- **Lambertian Microfacet model**

$$f_r(\omega_i,\omega_e)=\frac{\rho}{\pi}(A+B\cos\varphi)\sin\alpha\tan\beta$$

$$A=1.0-0.5\frac{\sigma^2}{\sigma^2+0.33} \qquad B=0.45\frac{\sigma^2}{\sigma^2+0.09}$$

$$\alpha=\max(\theta_i,\theta_e)$$
$$\beta=\min(\theta_i,\theta_e)$$

Sigma is a roughness parameter. 0 is smooth (Lambertian), and increasing the number increases the roughness. Remember that φ is the relative azimuth angle between the directions of incidence and exitance.

Microsoft DIRECTX NAUGHTY DOG

Oren-Nayar

- Normalized, reciprocal, physically based

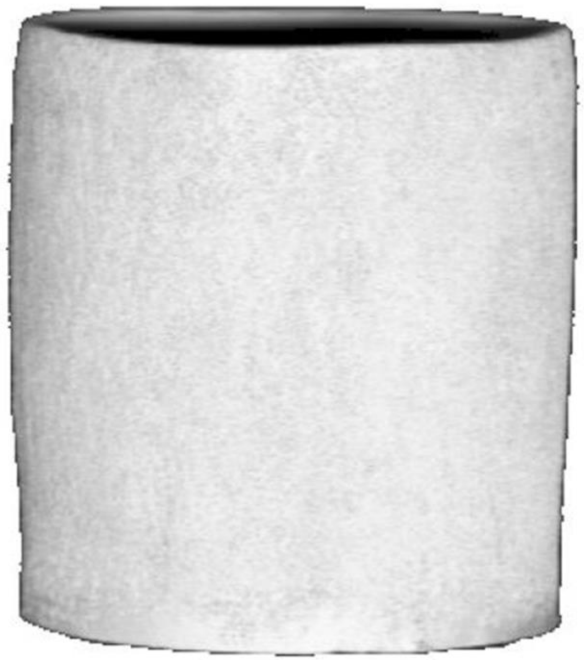
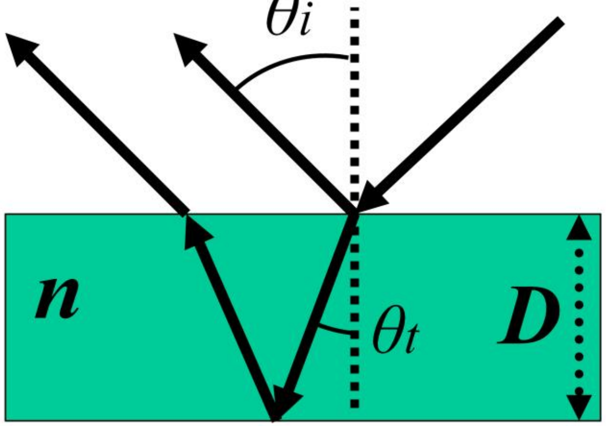


IMAGE BY M. OREN AND S. NAYAR

Thin-Film Interference

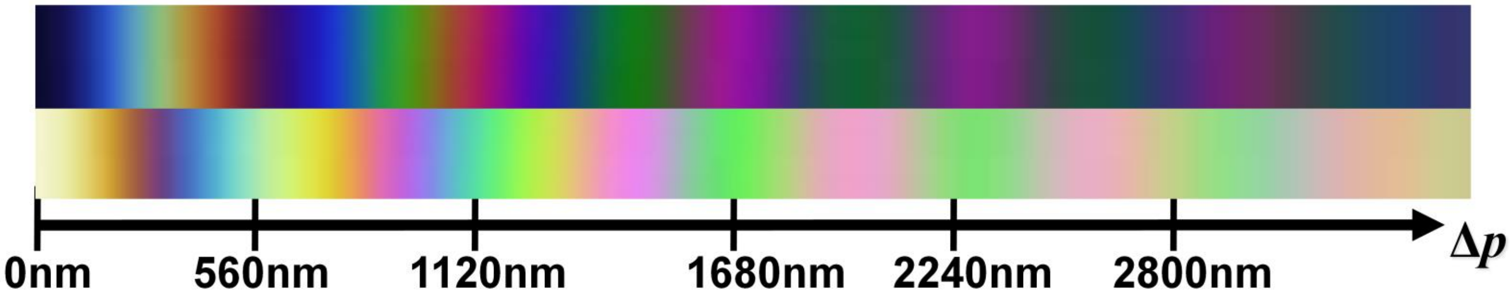
- **Factor that can modulate a specular term**
  - **For half-angle BRDF use α_h instead of θ_i**



$$\Delta p = 2nD\cos\theta_t$$

$$\Downarrow$$

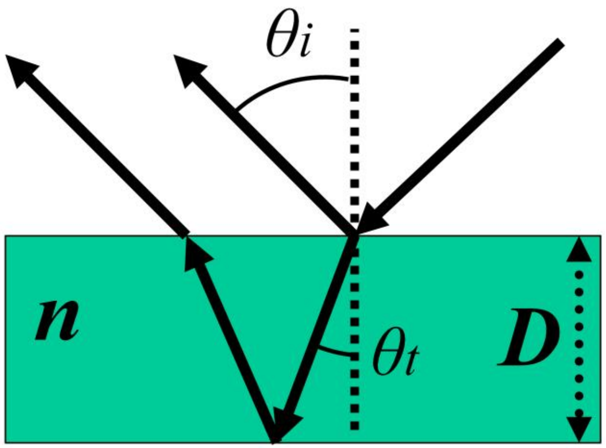
$$\Delta p = 2D\sqrt{\left(n^2 - 1\right) + \cos^2\theta_i}$$



Just calculate Δp, and lookup in a suitable 1D ramp texture based on whether the thin film is backed by a material with higher refractive index. The thickness can be constant or varying. From the second form, we can see that the larger the refractive index n, the less effect the incidence angle has.

Thin-Film Interference

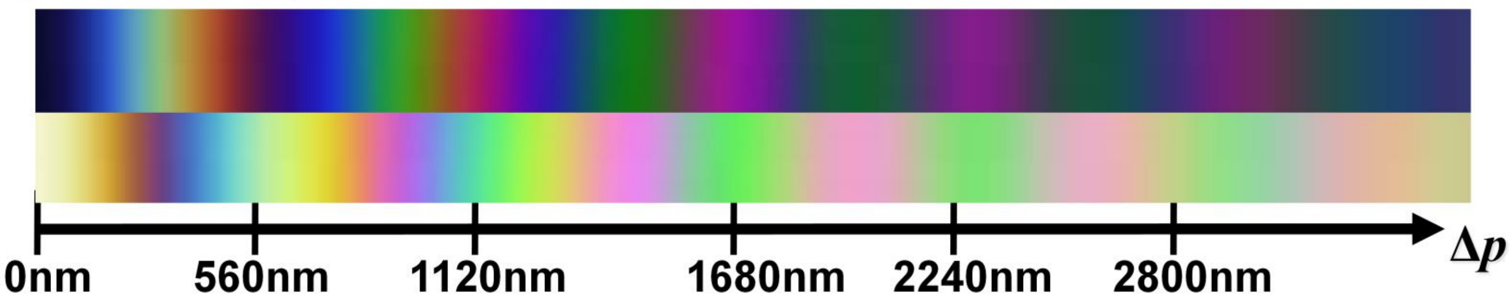
- Factor that can modulate a specular term
  - For half-angle BRDF use α_h instead of θ_i



$$\Delta p = 2nD\cos\theta_t$$

$$\Downarrow$$

$$\Delta p = 2D\sqrt{\left(n^2 - 1\right) + \cos^2\theta_i}$$



This is simplified, since it only takes the first two reflections into account and assumes they are of the same intensity for interference purposes. The approximation can be made more accurate by tweaking the 1D ramp textures, at the cost of some generality.

Diffraction (Simplified Stam BRDF)

- Simplified version of a BRDF by Jos Stam in GPU Gems, itself simplified from his SIGGRAPH 1999 paper
- Represents a diffraction grating with long grooves separated by a distance D
- Grooves are perpendicular to the local frame tangent vector

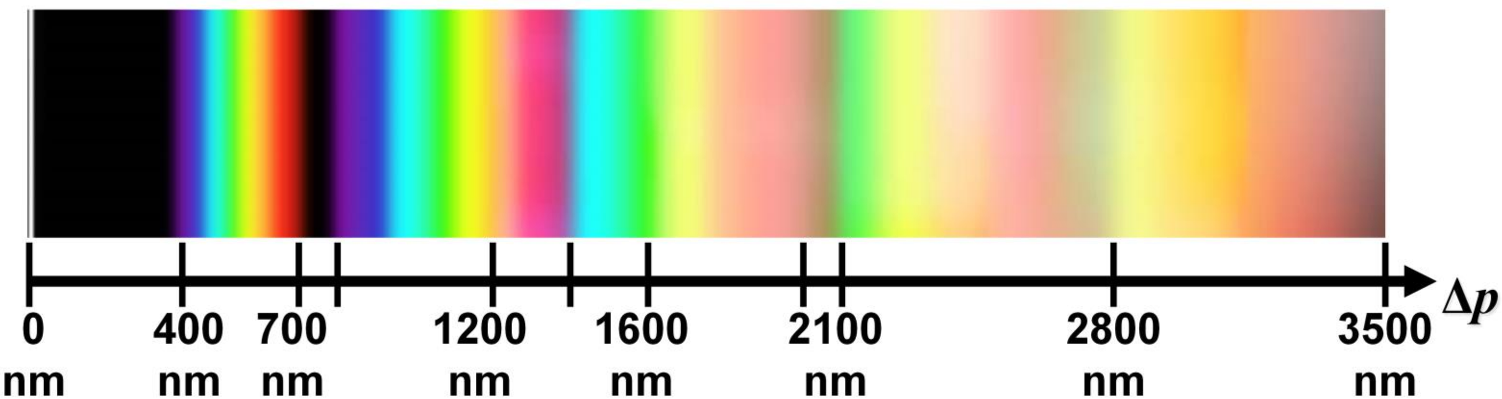
$$\Delta p = dot(\mathbf{V} + \mathbf{L}, \mathbf{T})D$$

Note that here we are using essentially a non-normalized form of the half-angle vector.

Microsoft DIRECTX® NAUGHTY DOG

Diffraction (Simplified Stam BRDF)

- Lookup the reflectance color in a 1D ramp texture using Δp, similarly to thin-film interference BRDF
- Stam recommends adding a standard anisotropic term for the "0-order" reflection



The ramp texture can be calculated by adding together suitably shifted and scaled specular "rainbow ramps".

Hand-painted BRDFs

- Textures are commonly used to accelerate evaluating BRDFs
- These textures can be calculated from the BRDF equations, but they can also be hand-painted by artists to get a BRDF with certain desired visual properties
- NDF texture
  - 1D texture (isotropic NDF)
  - 2D texture (anisotropic NDF)
- Fresnel texture (paint $R_F(\alpha)$ into a 1D texture)
- Fresnel / NDF
  - 2D texture with Fresnel on one axis and isotropic NDF on the other

Other examples are possible. In the implementation section we will discuss using textures as lookup tables for factorized BRDFs, this can always be an opportunity for artist tweaking of the BRDF.

References

- Introduction To Modern Optics, 2nd ed., Fowles (Dover)
- Principles of Digital Image Synthesis, Glassner (MK)
- The Physics and Chemistry of Color, 2nd ed., Nassau (Wiley)
- Geometric considerations and nomenclature for reflectance, Nicodemus et. al. (NBS 1977)
- A practitioners' assessment of light reflection models, Shirley et. al. (Pacific Graphics 1997)
- A survey of shading and reflectance models for computer graphics, Schlick (Computer Graphics Forum)
- Illumination for computer generated pictures, Phong (Communications of the ACM)
- Models of light reflection for computer synthesized pictures, Blinn (SIGGRAPH 1977)
- A reflectance model for computer graphics, Cook and Torrance (ACM ToG)
- Measuring and modeling anisotropic reflection, Ward (SIGGRAPH 1992)
- An anisotropic phong BRDF model, Ashikhmin and Shirley (JGT)
- Non-linear approximation of reflectance functions, Lafortune et. al. (SIGGRAPH 1977)
- Generalization of Lambert's reflectance model, Oren and Nayar (SIGGRAPH 1994)
- Predicting reflectance functions from complex surfaces, Westin et. al. (SIGGRAPH 1992)
- A microfacet-based BRDF generator, Ashikmin et. al. (SIGGRAPH 2000)
- Illumination in diverse codimensions, Banks (SIGGRAPH 1994)
- An accurate illumination model for objects coated with multilayer films, Hirayama et. al. (Eurographics 2000)
- Diffraction shaders, Stam (SIGGRAPH 1999)
- Simulating diffraction, Stam, in GPU Gems (AW)

Acknowledgements

- Michael Ashikhmin, Henrik Wann Jensen, Kazufumi Kaneda, Eric Lafortune, David McAllister, Addy Ngan, Michael Oren, Kenneth Torrance, Gregory Ward, and Stephen Westin for permission to use images