

Lesson 10 – Physically-based rendering Microfacets

PV227 – GPU Rendering

Jiří Chmelík, Jan Čejka
Fakulta informatiky Masarykovy univerzity

20. 11. 2017

Physically-based rendering (PBR)

- Think about the physics behind everything:
 - ▶ Light
 - ▶ Lights
 - ▶ **Materials**
 - ▶ Sensors / Eyes
- In practice, still approximations
- More and more popular in real-time rendering, in rendering engines

Light propagation – Homogeneous media

- Light interacts with the material it travels through
- In homogeneous materials, the light is absorbed
 - ▶ Loses some of its energy
 - ▶ Clean water, glass, air, oil, . . .



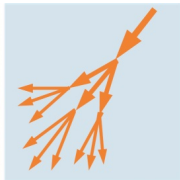
Low Absorbtion



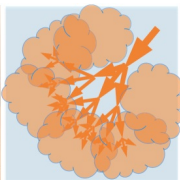
High Absorbtion

Light propagation – Heterogeneous media

- Light interacts with the material it travels through
- In heterogeneous materials, the light is scattered
 - ▶ Scatters the energy without losses
 - ▶ Milk, skin, wood, (dirty water, air with fog), . . .



Low Scattering



High Scattering

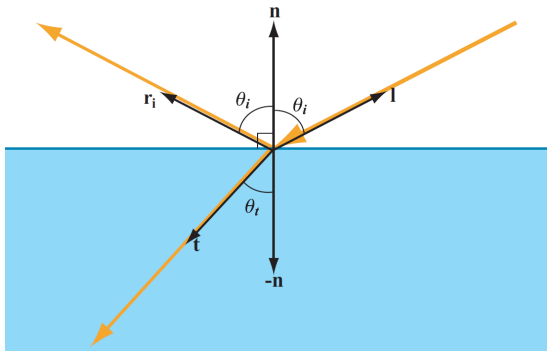
Light propagation – Absorption vs. Scattering



Absorption vs. Scattering

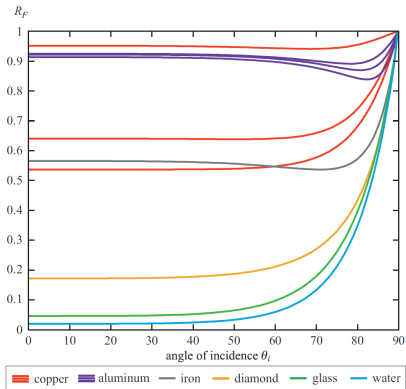
Light interaction – Materials

- Light changes its direction at the boundary between two materials
 - ▶ Reflection
 - ▶ Refraction
 - ▶ Without losses of energy



Reflection

- Perfect reflection: $\theta_i = \theta_o$
- Amount of reflected light depends on θ_i and on the wavelength
- Described by Fresnel equations













Fresnel reflection

- Depends on the wavelength, reflection has a color (!)
- Metals have usually higher values
- Dielectrics have usually lower values
- Mostly without any change until 50° , then goes straight to one
- In practice: Schlick's approximation:

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

where F_0 is Fresnel reflection at 0° , L is direction to light, N is surface normal

Fresnel reflection at 0°

Material	$F(0^\circ)$ (Linear)	$F(0^\circ)$ (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	
Aluminum	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	

Fresnel reflections at 0° for some materials

- Snell's law

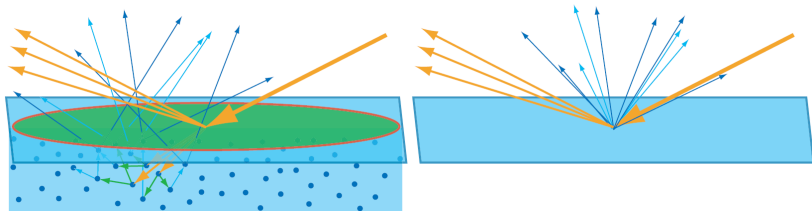
$$\frac{\sin \theta_i}{\sin \theta_t} = \frac{v_1}{v_2}$$

- In metals, all energy is absorbed
- In homogeneous materials, the light continues in different direction
- In heterogeneous materials (including skin, wood, plastic, . . .), the light is scattered and absorbed

Refraction – diffuse light

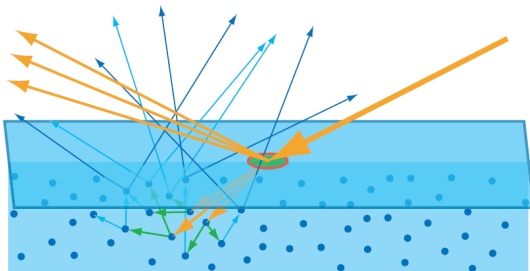
- Diffuse lighting

- ▶ When all the (non-absorbed) light exits the surface at approximately the same point as the light enters.



Refraction – sub-surface scattering

- Sub-surface scattering (SSS)
 - ▶ When all the (non-absorbed) light exits the surface at different places.



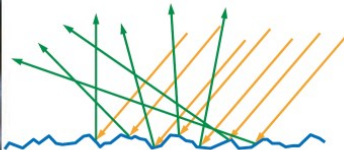
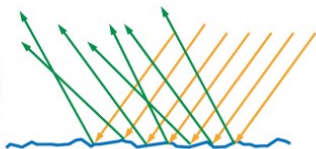
Ambient lighting?

- “Does not exist in PBR”
- Average of lighting coming from all directions.

Microfacets

- Only some objects are flat (mirrors, water surface, . . .), others are not
- With microfacets, the surface is represented with very small facets
 - ▶ Smaller than 'pixel', not for displacement in geometry, not for normal mapping
 - ▶ (Larger than light's wavelength)
 - ▶ Each microfacet is a flat surface
- Many microfacets models, for different materials
 - ▶ Different distribution of orientation of facets
 - ▶ Different shadowing between facets
 - ▶ Different approximation of the model

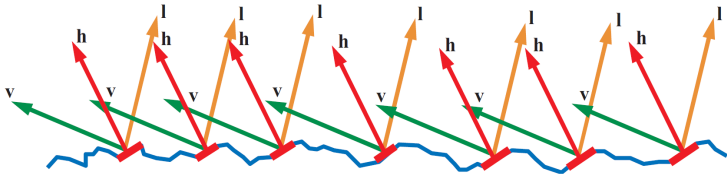
Microfacets



Top: smooth surface, Bottom: rough surface

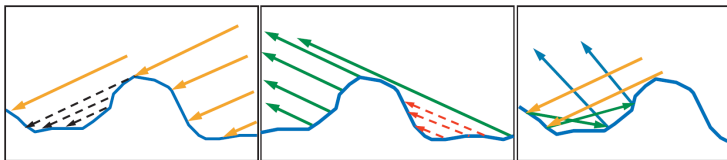
Distribution of microfacets

- We are usually interested in facets which are oriented in the proper direction to give us perfect reflection.
 - ▶ i.e. facets that are oriented in the half-vector direction
- Gaussian distribution, ...



Geometrical attenuation

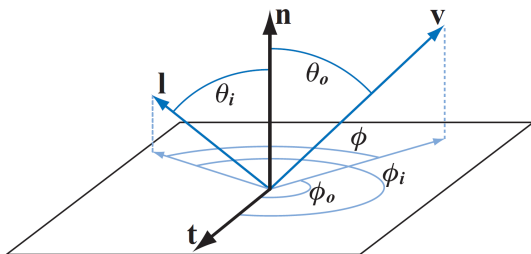
- Shadowing: Facets occlude the light for other facets
- Masking: Facets cannot be seen due to other facets
- Interreflection: Facets reflect the light to other facets, and then the light is reflected to the viewer



Task: Implement microfacets models

- Cook-Torrance (1982)
- Oren-Nayar (1994)
- Ashikhmin-Shirley (2000)
- Normalized Blinn-Phong (2008)

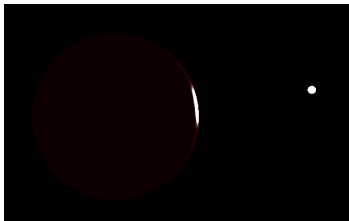
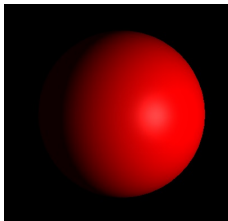
Legend to the following equations



- \vec{N} , \vec{T} , \vec{B} are surface normal, tangent, and bitangent
- \vec{L} is direction to the light, \vec{V} is direction to the viewer
- \vec{H} is half-vector, vector between the light and the viewer
- All dot products are non-negative, e.g.: $\max(0, \vec{N} \cdot \vec{L})$
- All vectors are normalized
- All results must be multiplied by light's intensity and color
- $Fresnel(\vec{V} \cdot \vec{H}) = F_0 + (1 - F_0)(1 - \vec{V} \cdot \vec{H})^5$

Cook-Torrance

- Useful for most surfaces, metals,
- All microfacets are perfect mirrors
- Single parameter m (roughness), usually in range (0, 1)



Cook-Torrance cont.

- Diffuse:

$$I_{diff} = Color_{diff} \cdot (\vec{N} \cdot \vec{L})$$

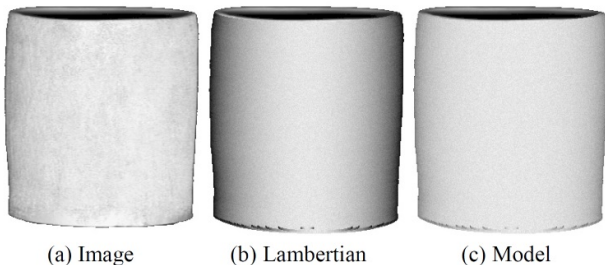
- Specular:

$$I_{spe} = \frac{F \cdot G \cdot D}{4 \cdot (\vec{N} \cdot \vec{V})}$$

where

- ▶ Fresnel $F = Fresnel(\vec{V} \cdot \vec{H})$
 - ▶ Geom. atten. $G = \min\left(1, \frac{2 \cdot (\vec{N} \cdot \vec{H}) \cdot (\vec{N} \cdot \vec{V})}{(\vec{V} \cdot \vec{H})}, \frac{2 \cdot (\vec{N} \cdot \vec{H}) \cdot (\vec{N} \cdot \vec{L})}{(\vec{V} \cdot \vec{H})}\right)$
 - ▶ Beckmann distribution $D = \frac{e^{\frac{(\vec{N} \cdot \vec{H})^2 - 1}{m^2 \cdot (\vec{N} \cdot \vec{H})^2}}}{m^2 \cdot (\vec{N} \cdot \vec{H})^4}$
 - ▶ (m is roughness of the material)
- Diffuse is energy-conserving, specular is energy-conserving, but not together

- For non-shiny objects like concrete, flowerpots, bricks, Moon
- All microfacets are Lambertian (diffuse) surfaces
- No specular highlights
- Retroreflections at boundaries
- Single parameter m (roughness)



Oren-Nayar cont.

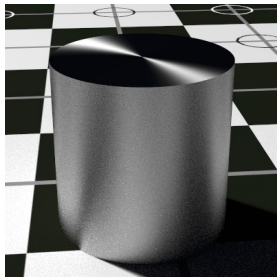
- Diffuse:

$$I_{diff} = Color_{diff} \cdot (\vec{N} \cdot \vec{L}) \cdot (A + B \cdot \max(0, \cos(\phi)) \cdot C)$$

where

- ▶ $\theta_i = \arccos(\vec{N} \cdot \vec{L})$
 - ▶ $\theta_o = \arccos(\vec{N} \cdot \vec{V})$
 - ▶ $\alpha = \max(\theta_i, \theta_o)$
 - ▶ $\beta = \min(\theta_i, \theta_o)$
 - ▶ $\cos(\phi) = \text{norm}(\vec{V} - \vec{N} \cdot (\vec{V} \cdot \vec{N})) \cdot \text{norm}(\vec{L} - \vec{N} \cdot (\vec{L} \cdot \vec{N}))$
 - ▶ $A = 1.0 - 0.5 \cdot \frac{m^2}{m^2 + 0.57}$
 - ▶ $B = 0.45 \cdot \frac{m^2}{m^2 + 0.09}$
 - ▶ $C = \sin(\alpha) \cdot \tan(\beta)$
- Diffuse is energy-conserving

- For brushed objects (metal) with anisotropic reflections
- All microfacets are perfect mirrors
- Two parameters $shin_T$, $shin_B$: shininess exponents in tangent and bitangent directions, usually greater than 1



Ashikhmin-Shirley cont.

- Diffuse (not energy-conserving with specular):

$$I_{diff} = Color_{diff} \cdot (\vec{N} \cdot \vec{L})$$

- or diffuse (energy-conserving with specular):

$$I_{diff} = Color_{diff} \cdot (\vec{N} \cdot \vec{L}) \cdot \frac{28}{23} (1 - F_0) \left(1 - \left(1 - \frac{(\vec{N} \cdot \vec{L})}{2} \right)^5 \right) \left(1 - \left(1 - \frac{(\vec{N} \cdot \vec{V})}{2} \right)^5 \right)$$

where

- ▶ F_0 is Fresnel reflection at 0°

- Specular:

$$I_{spe} = \text{Fresnel}(\vec{V} \cdot \vec{H}) \cdot (\vec{N} \cdot \vec{L}) \cdot \frac{\sqrt{(\text{shin}_T + 1)(\text{shin}_B + 1)}}{8} \cdot \frac{(\vec{N} \cdot \vec{H}) \frac{\text{shin}_T \cdot (\vec{T} \cdot \vec{H})^2 + \text{shin}_T \cdot (\vec{B} \cdot \vec{H})^2}{1 - (\vec{N} \cdot \vec{H})^2}}{(\vec{V} \cdot \vec{H}) \cdot \max((\vec{N} \cdot \vec{L}), (\vec{N} \cdot \vec{V}))}$$

Normalized Blinn-Phong

- Improvement of the original Blinn-Phong (1977)
- Specular is energy conserving, without creating or losing energy
- Diffuse is the same

$$I_{diff} = Color_{diff} \cdot (\vec{N} \cdot \vec{L})$$

- Original specular

$$I_{spe} = (\vec{N} \cdot \vec{L}) \cdot Color_{spe} \cdot (\vec{N} \cdot \vec{H})^{shin}$$

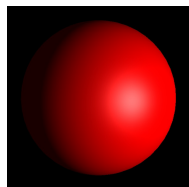
- Normalized specular

$$I_{spe} = (\vec{N} \cdot \vec{L}) \cdot Color_{spe} \cdot \frac{shin + 8}{8} (\vec{N} \cdot \vec{H})^{shin}$$

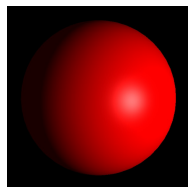
where *shin* is shininess exponent

Normalized Blinn-Phong cont.

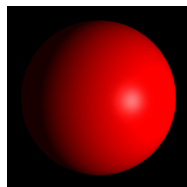
- Original Blinn-Phong, $Color_{spe} = 120/255$



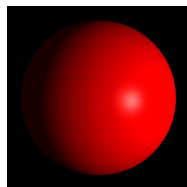
shin = 25



shin = 50

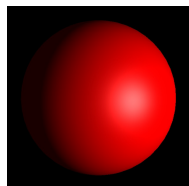


shin = 75

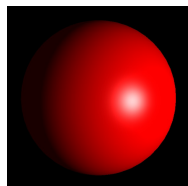


shin = 100

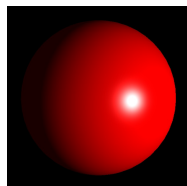
- Normalized Blinn-Phong, $Color_{spe} = 32/255$



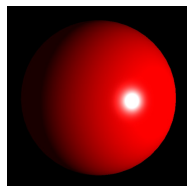
shin = 25



shin = 50



shin = 75



shin = 100

Task: Implement microfacets models

- **Task 1:** Implement Normalized Blinn-Phong

- ▶ Fragment shader *BlinnPhongNormalized_fragment.glsl*
- ▶ Parameters in variables:
 - ★ $Color_{diff}$ in *material_diffuse*
 - ★ $Color_{spe}$ in *material_specular*
 - ★ *shin* in *material_shininess*
- ▶ Compare with the original Blinn-Phong

Task: Implement microfacets models

● Task 2: Implement Cook-Torrance

- ▶ Fragment shader *CookTorrance_fragment.glsl*
- ▶ Parameters in variables:
 - ★ $Color_{diff}$ in *material_diffuse*
 - ★ F_0 in *material_fresnel*
 - ★ m in *material_roughness*
- ▶ Notice the highlight when looking from near surface angles
- ▶ Comparative with Blinn-Phong. To get approx. the same result:
 - ★ Set Specular color and Fresnel color to the same value
 - ★ Set $roughness = \sqrt{\frac{2}{2+shininess}}$

Task: Implement microfacets models

- **Task 3:** Implement Oren-Nayar

- ▶ Fragment shader *OrenNayar_fragment.glsl*
- ▶ Parameters in variables:
 - ★ $Color_{diff}$ in *material_diffuse*
 - ★ m in *material_roughness*
- ▶ Set *roughness* to 0.5 and compare with Blinn-Phong with black *specular*

Task: Implement microfacets models

● **Task 4:** Implement Ashikhmin-Shirley

- ▶ Fragment shader *AshikhminShirley_fragment.glsl*
- ▶ Parameters in variables:
 - ★ $Color_{diff}$ in *material_diffuse*
 - ★ F_0 in *material_fresnel*
 - ★ $shin_T$ in *material_shininess_tangent*
 - ★ $shin_B$ in *material_shininess_bitangent*
- ▶ Test on cylinder or teapot, set different shininess in tangent and bitangent directions (e.g. 20 and 500)
- ▶ When using simple computation for diffuse color, and setting shininess to the same values, the result should be compatible with Blinn-Phong.

- SIGGRAPH courses on PBR:

- ▶ <http://renderwonk.com/publications/s2010-shading-course/>
- ▶ <http://blog.selfshadow.com/publications/s2012-shading-course/>
- ▶ <http://blog.selfshadow.com/publications/s2013-shading-course/>
- ▶ <http://blog.selfshadow.com/publications/s2014-shading-course/>
- ▶ <http://blog.selfshadow.com/publications/s2015-shading-course/>
- ▶ <http://blog.selfshadow.com/publications/s2016-shading-course/>

Further reading

- Cook, R., Torrance, K.: *A Reflectance Model for Computer Graphics*
- Oren, M., Nayar, S.: *Generalization of Lambert's Reflectance Model*
- Ashikhmin, M., Shirley, P.: *An Anisotropic Phong BRDF Model*
- Akenine-Möller, T., et al.: *Real-Time Rendering*
- Pharr, M., et al.: *Physically Based Rendering, From Theory to Practice*