

An interactive ray tracing system based on Nvidia OptiX

Michael Größler | Martin Zettwitz

14. Oktober 2015

- 1 Wer sind wir?
- 2 Ziel des Projekts
- 3 Rendering Equation
- 4 BRDFs
 - Lambert
 - Phong
 - Blinn Phong
 - Cook Torrance
 - Ward
 - Ashikhmin Shirley
 - Glas
- 5 Ergebnisse
- 6 Demo

Michael Größler
5. Semester CV
groessle@st.ovgu.de

Martin Zettwitz
5. Semester CV
martin.zettwitz@st.ovgu.de

- Projekt- und Programmiererfahrung erweitern
- interaktives Ray Tracing entwickeln
- mit Bestnote abschließen

'THE RENDERING EQUATION' by James T. Kajiya [KAJ86]

$$L_o(x, \vec{\omega}_o) = L_e(x, \vec{\omega}_o) + \int_{\Omega} f_r(\vec{\omega}_i, \vec{\omega}_o) \cdot \cos \theta \cdot L_i(x, \vec{\omega}_i) \cdot d\vec{\omega}_i \quad (1)$$

$$L_o(x, \vec{\omega}_o) = L_e(x, \vec{\omega}_o) + \int_{\Omega} f_r(\vec{\omega}_i, \vec{\omega}_o) \cdot \cos \theta \cdot L_i(x, \vec{\omega}_i) \cdot d\vec{\omega}_i$$

L_o	out-going radiance
L_e	self emission
L_i	in-going radiance
$\vec{\omega}_i$	incoming light direction
$\vec{\omega}_o$	vector towards camera
f_r	BRDF
Ω	Hemisphere over point x
$\cos \theta$	angle between normal and $\vec{\omega}_i$
x	point on surface

Unser Ansatz ohne Eigenemission und zusätzlicher Reflexion (CG)

$$L_o(x, \vec{\omega}_o) = \delta L_i(x, \vec{\omega}_r) + (1 - \delta) \sum_{i=1}^k f_r(\vec{\omega}_i, \vec{\omega}_o) \cdot E \quad (2)$$

δ	reflection coefficient
E	approximated irradiance
I_o	light intensity of a point light (independent from $\vec{\omega}_i$)

- bi-directional distribution function
- beschreibt Lichtverhalten auf Oberflächen
- betrachtet Ein- und Ausfallvektor
- unterteilt in diffusen(K_d) und spekularen(K_s) Anteil

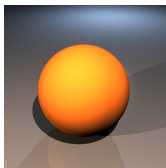
$$f_r = K_d + K_s \quad (3)$$

\vec{V}	ray from hit point to camera
\vec{N}	normal on hit point
\vec{L}	ray from hit point to light source

- nach Lambert's Emissions Gesetz aus 'Photometria' von Johann Heinrich Lambert in 1760
- einfachste Darstellung perfekt diffuser Oberflächen

$$K_d = \frac{1}{\pi} \quad (4)$$

$$K_s = 0$$



- nach [Pho75]
- einfaches und erstes empirisches standard Modell für glänzende Oberflächen(Kunststoff)

$$K_d = \frac{\rho_d}{\pi} \quad (5)$$

$$K_s = \rho_s \cdot \frac{s+2}{2\pi} \cdot \cos^s \psi \quad (6)$$

ρ_d	Parameter : diffuse coefficient
ρ_s	Parameter : specular coefficient
s	Parameter : shiny exponent, controls roughness
$\cos \psi$	angle between outgoing and reflected ray

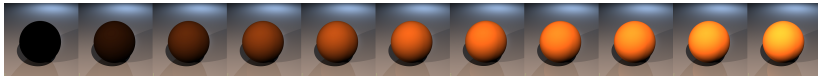


Abbildung : Parameter ρ_d from 0.0 to 1.0

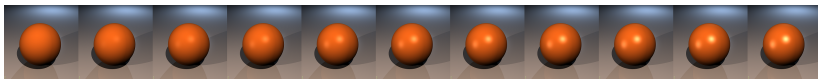


Abbildung : Parameter ρ_s from 0.0 to 1.0

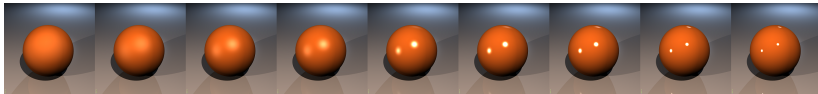


Abbildung : Parameter s from 1 to 1000

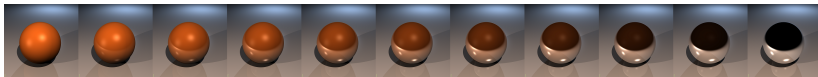


Abbildung : Parameter δ from 0.0 to 1.0

- nach [Bli77]
- erweitert das standard Phong-Modell um den Halbvektor \vec{H}

$$K_d = \frac{\rho_d}{\pi} \quad (7)$$

$$K_s = \rho_s \cdot \frac{s + 8}{8\pi} \cdot \cos^s \psi \quad (8)$$

$\cos \psi$	angle between outgoing ray and half vector \vec{H}
-------------	--

$$\vec{H} = \frac{\vec{L} + \vec{V}}{\text{length}(\vec{L} + \vec{V})} \quad (9)$$



Abbildung : Parameter s from 1 to 1000

