Path Tracing

Rendering function:

$$L_o(p,\omega_o) = L_e(p,\omega_o) + \int_{\Omega} L_i(p,\omega_i) f(p,\omega_i,\omega_o) cos\theta_i d\omega_i$$

$$Reflected\ Light(Output\ Image)[Unknown] = Emission[Known]$$

$$+ \int Reflected\ Light[Unknown] * BRDF[Known] * Cosine\ of\ incident\ angle[Known]$$

1. Iterate each pixel, cast ray in one pixel multiple times and take the mean of these rays' color.

Following progress is included in one single ray casting:

2. If the ray hit the light source at depth 0, get Emission value L_emit, otherwise set it to be 0.
3. Seperate global illumination into **direct illumination**(L_dir) and **indirect illumination**(L_indir)

(Rigidly, Only the emission directly from light sources and direct illumination on surfaces is direct illumination)

> For direct illumination

In function $shade(p, \omega_o)$, randomly generate N rays of ω_i in $pdf(w_i)$.

$$L\_dir(p,\omega_o) = \frac{1}{N}\sum_{i=1}^{N} \frac{L_{light} * f(p,\omega_i,\omega_o) * cos\theta_i}{pdf(\omega_i)}$$

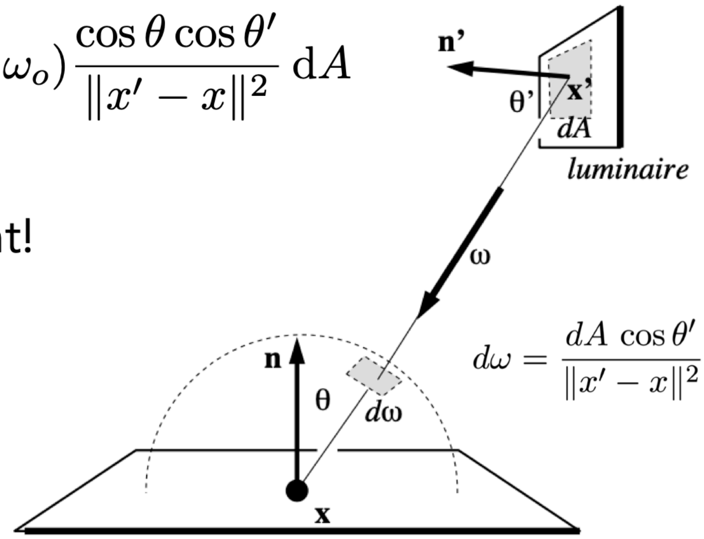
But sample randomly in space is not efficient sampling method, we can restrict only sampling the ray to the light source.

We transform the differential of incident solid angle($d\omega_i$) to light surface's differential space(dA), now $pdf = \frac{1}{A}$.

$$d\omega = \frac{dAcos\theta'}{||x' - x||^2}$$

Then we can rewrite the rendering equation as

$$
\begin{aligned}
L_o(x, \omega_o) &= \int_{\Omega+} L_i(x, \omega_i) f_r(x, \omega_i, \omega_o) \cos\theta \,\mathrm{d}\omega_i \\
&= \int_A L_i(x, \omega_i) f_r(x, \omega_i, \omega_o) \frac{\cos\theta \cos\theta'}{\|x' - x\|^2} \,\mathrm{d}A
\end{aligned}
$$



Now an integration on the light!

Monte Carlo integration:

“f(x)”: everything inside

Pdf: 1 / A

After sample point from the light source, we also need to check whether there's obstacle between the reflecting object and light source.

This is the direct illumination part.

> For indirect illumination

In function $shade(p, \omega_o)$, we also simply randomly generate N rays of ω_i in $pdf(w_i)$.(Sample the solid angle directly)

$$L\_indir(p, \omega_o) = \frac{1}{N} \sum_{i=1}^{N} \frac{shade(q, -\omega_i) * f(p, \omega_i, \omega_o) * cos\theta_i}{pdf(\omega_i)}$$

Since this progress is recursively, so we only sample one ray in both two kinds of illumination(N=1). And we stop the recursion by RussianRoulette.

$$E = P * (L\_indir/P) + (1 - P) * 0 = L\_indir$$

Noticed: this sampled ray shouldn't hit the light source.

4. Sum up two kinds of illumination

$$L = L\_emit + L\_dir + L\_indir$$

Microfacet

BRDF

This is a kind of new material kind.

It differs in BRDF calculation.

Here is the reflection(specular) part of microfacet BRDF:

$f\_cooktorrance = \frac{DFG}{4(\omega_o \cdot n)(\omega_i \cdot n)}$

D: **Normal distribution function**(approximates the amount the surface's microfacets are aligned to the halfway vector)

G: **Geometry function**(describes the self-shadowing property)

F: **Fresnel equation**(the ratio of surface reflection at different surface angles)

D and G is controlled by a roughness parameter α, here I take Trowbridge-Reitz GGX as D and Schlick-GGX as G.

$$NDF_{GGXTR}(n,h,\alpha) = \frac{\alpha^2}{\pi((n \cdot h)^2(\alpha^2-1)+1)^2}$$

$$G_{SchlickGGX}(n,v,k) = \frac{n \cdot v}{(n \cdot v)(1-k)+k}$$

$$k_{direct} = \frac{(\alpha+1)^2}{8}$$

Self shadowing considers both view and light:

$$G(n,v,l,k) = G_{sub}(n,v,k)G_{sub}(n,l,k)$$

Fresnel equation just take Fresnel-Schlick approximation here, instead of complex equation:

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

F_0 represents the base reflectivity of the surface when looking straight at its surface.

The refracted light plays an effect like diffuse light.

$$1 = k_d + k_s$$

The sum of ratio of reflected and refracted light is 100%.

The final BRDF becomes:

$$f_{microfacet} = k_d \frac{color}{\pi} + k_s \frac{DFG}{4(\omega_o \cdot n)(\omega_i \cdot n)}$$

Noticed: k_s is included in Fresnel.

$$f_{microfacet} = k_d \frac{color}{\pi} + \frac{DFG}{4(\omega_o \cdot n)(\omega_i \cdot n)}$$

Importance Sampling

In Monte Carlo integration, if we take pdf(x) = f(x), integrated function will be a constant function:

$$\frac{1}{N}\sum_{i=0}^{N-1}\frac{f(x)}{pdf(x)}$$

So a good approximation to integrated function is good for pdf(x) for sampled x.

When we consider the common diffuse material, We think the irradiance is reflected evenly in the whole hemisphere solid angle.

$$f(\omega_i) = L_i(p,\omega_i)f(p,\omega_i,\omega_o)cos\theta_i d\omega_i$$
$$f_{diffuse} = f(p,\omega_i,\omega_o) = \frac{color}{\pi}$$

The variance triggered by uniform sample over solid angle ω is acceptable.

But for microfacet, the NDF is usually the dominant one(it has an extremely high peak when $cos\theta = 1$). So we sample the light with $pdf = NDF$.

Instead of sampling solid angle directly, we usually use spherical coordinate to sample it. So it is not the pdf respecting the solid angle that we are interested, it is the pdf respecting the spherical coordinates.

$$pdf(\omega) = \frac{\alpha^2 cos\theta}{\pi(cos^2\theta(\alpha^2-1)+1)^2}$$
$$\Rightarrow pdf(\theta,\phi) = \frac{\alpha^2 cos\theta sin\theta}{\pi(cos^2\theta^2(\alpha^2-1)+1)^2}$$
$$\Rightarrow pdf(\theta) = \int_0^{2\pi}\frac{\alpha^2 cos\theta sin\theta}{\pi(cos^2\theta(\alpha^2-1)+1)^2}d\phi = \frac{2\alpha^2 cos\theta sin\theta}{(cos^2\theta(\alpha^2-1)+1)^2}$$
$$\Rightarrow \epsilon = cdf(\theta<\mu) = \int_0^{\mu}\frac{2\alpha^2 cos\theta sin\theta}{(cos^2\theta(\alpha^2-1)+1)^2}d\theta = \frac{\alpha^2}{cos^2\theta(\alpha^2-1)^2+(\alpha^2-1)} - \frac{1}{\alpha^2-1}$$
$$\Rightarrow \theta = arccos\sqrt{\frac{1-\epsilon}{\epsilon(\alpha^2-1)+1}}$$

All we need to do is randomly sample probability ϵ in uniform distribution and get the actual sampled θ, which denotes the sampled light.

Don't forget to consider this new $pdf(\omega)$ in Monte Carlo integration.

Reference

[1] GAMES101

[2] Monte Carlo Methods in Practice

[3] Sampling Microfacet BRDF

[4] learnopengl PBR Theory

[5] jackysunhz/MyPathTracer

[6] Ubpa/RenderLab