PBRT: Materials

Tuesday, January 5, 2021 10:27 AM

9 Materials

BRDFs and BTDFs describe how light is ***scattered at a point***. PBRT's ***procedural shading mechanism*** determines what BRDFs and BTDFs are found at a point and what its parameters are.

A ***shader surface*** represented by the Material class is bound to each primitive.

The BSDF class holds a set of BxDFs (up to 8). The BSDF function returns the sum of the spectra of all these BxDFs: the sum of the BRDFs when ω_o and ω_i are on the same hemisphere; the sum of the BTDFs when they are on opposite hemispheres. These ***sets*** of BxDFs per BSDF is what allows us to ***model the appearances of several different materials***: for example, a plastic material combines a Lambertian diffuse BxDF and a glossy specular BxDF.

A Material takes points and returns the corresponding BSDFs. Material uses a Texture to determine the properties of the surface at those points.

Integrators ask the Material of the surface hit by a ray for the BSDF of the point.

9.1 BSDFs

Shading normals n_s may not correspond to ***geometric surface normals*** n_g. Shading normals are either found at the vertices of triangle meshes or are created by bump mapping.

Obviously, their hemispheres of incident light directions are different. Which is why n_s can't be used directly to evaluate the scattering equation $L(\omega_o, \omega_i)$ or the reflectance equations ρ. Here, in (a), the surface is not transmissive and ω_i is not in the hemisphere of n_g, but it is in the hemisphere of n_s: if the equations used n_s, p would be shaded with a contribution from ω_i; (b) is the converse case.

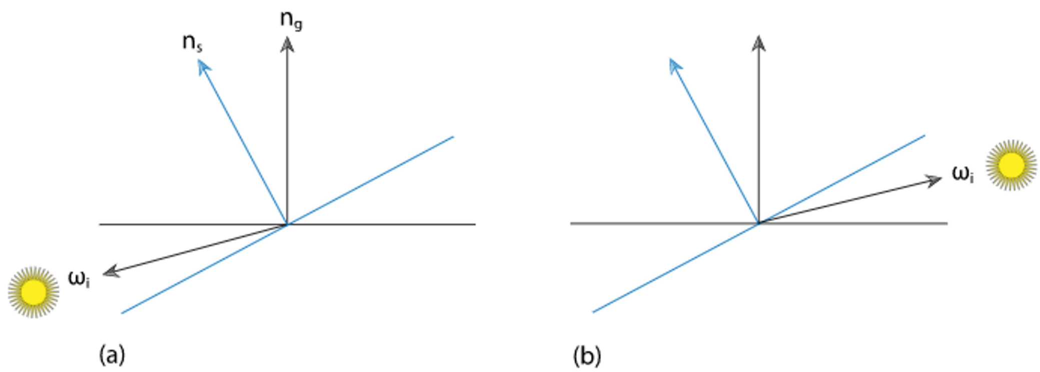

Figure 9.2: The Two Types of Errors That Result from Using Shading Normals. (a) A light leak: the geometric normal indicates that the light is on the back side of the surface, but the shading normal indicates the light is visible (assuming a reflective and not transmissive surface). (b) A dark spot: the geometric normal indicates that the surface is illuminated, but the shading normal indicates that the viewer is behind the lit side of the surface.

The solution is this:

- Use n_g when evaluating the BSDFs: if ω_o and ω_i are in the same hemisphere, evaluate only the BRDFs of the BSDF; otherwise evaluate the BTDFs.
- Use n_s when evaluating the scattering equation.

- [ ] TODO: why is it OK to use n_s in the scattering equation $L(\omega_o, \omega_i)$?

The use of shading normals is not strictly physically-based, but they can improve the visual richness. They are handled by the BSDF class.

The BSDF stores the differential geometry of a [Surface Interaction]() and also the orthonormal basis (s_s, t_s, n_s) of the [shading coordinate system](). (s_s, t_s, n_s) are all coordinate vectors relative to the standard basis, which is the world space basis.

The change-of-coordinates matrix from world space to ***shading space*** $P_{S\leftarrow W}$:

$$P_{S\leftarrow W} = \begin{bmatrix} s_x & t_x & n_x \\ s_y & t_y & n_y \\ s_z & t_z & n_z \end{bmatrix}$$

- [ ] ***TODO: NO, the book says that the change of coordinates matrix is actually the transpose of that one. But why? I didn't understand this part of the section very well. I know that the inverse of an orthogonal matrix is its transpose, but I don't understand other things said here.***

9.1.1 BSDF memory management

Objects of the BSDF class should be allocated using ARENA_ALLOC, not *new*.

9.2 Material interface and implementations

The SurfaceInteraction initiates the creation of the BSDF of the surface-ray intersection point: ComputeScatteringFunctions. SurfaceInteraction delegates the setup to its Primitive, which in turn delegates the creation to its Material.

An ***<u>interesting and important thing</u>*** to notice is that it is ***<u>textures</u>*** that encode the values of the parameters of the BSDFs that the Materials use to initialize them.

9.2.1 Matte material

Simplest in PBRT. Purely diffuse.

Parameters: ***<u>Kd</u>***, a diffuse reflectance spectrum; σ ***<u>sigma</u>***, a scalar roughness value.

These parameters are passed in textures, so that there is one set per (u, v).

Roughness sigma controls specularity: if 0, pure Lambertian reflection takes place; otherwise, the Oren-Nayar microfacet-based diffuse reflection model is used (which adds specularity; Oren and Nayar noted that ***<u>no surface has perfect Lambertian diffuse reflection</u>*** and, instead, diffuse surfaces appear brighter as the incident direction approaches the viewing direction; σ is then the ***<u>standard deviation of the microfacet orientation angle</u>***; see section 8.4.1).

Lambertian diffuse reflection, matte material with $\sigma = 0$

Oren-Nayar diffuse reflection, matte material with $\sigma > 0$

9.2.2 Plastic material

Mixture of diffuse and glossy.

Parameters: ***<u>Kd</u>***, a diffuse reflectance spectrum; ***<u>Ks</u>***, a glossy reflectance spectrum;

roughness+remapRoughess that determines the size of the specular highlight: if remapRoughness is true, roughness is from $[0,1]$; if false, it is the value of the α parameter of the ***microfacet distribution*** (section 8.4.2).

These parameters are passed in textures, so that there is one set per (u, v).

9.2.3 Mix material

- [ ] TODO

9.2.4 Fourier material

- [ ] TODO

9.2.5 Additional materials

- [ ] TODO: GlassMaterial, MetalMaterial, MirrorMaterial, SubstrateMaterial, SubsurfaceMaterial, KdSubsurfaceMaterial, TranslucentMaterial, UberMaterial.

9.3 Bump mapping

- [ ] TODO