PBRT: Light Sources

Monday, January 11, 2021 6:19 PM

12 Light sources

Physical processes that lead to photon emission.

The Light interface allows light transport algorithms to be agnostic of the type of light source.

Some implementations of Light methods need Monte Carlo integration, because some can't be computed in closed form.

12.1 Light emission

All objects with temperature above absolute zero have moving atoms and the motion of atomic particles that hold electrical charges causes objects to emit electromagnetic radiation over a range of wavelengths (Maxwell).

Objects at room temperature also emit electromagnetic radiation, but in the invisible, infrared wavelength range. Only objects that are warm enough emit radiation in the visible wavelengths.

Light emission is a result of ***incandescence***, ***electroluminescence***, ***chemiluminescence***, ***sonoluminescence***, or ***mechanoluminescence***.

We are concerned with ***incandescence*** and ***electroluminescence*** only: electrons from an electric current colliding with the atoms of a material and pushing their outer electrons to a higher energy level. When such an electron returns to a lower energy level, a photon is emitted.

The general principle behind incandescence is that electric current flowing through a material heats it, causing it to emit electromagnetic radiation. Electroluminescence seems to be similar, but the material doesn't get warmer (I don't understand it).

Luminous efficacy measures:

- How effectively a light source converts power to visible illumination. The more the emitted spectrum concentrates energy in the visible wavelengths, the more effective the source is.
- How effectively a light source converts power to electromagnetic radiation (as opposed to other forms of energy, like heat?).

And is computed as the ratio of one photometric quantity to its corresponding radiometric quantity: luminous flux to radiant flux Φ, or luminous exitance (outgoing illuminance) to radiant exitance M (outgoing irradiance E) at a point, or exitant luminance to radiance L at a point in a particular direction.

$$\frac{\int \Phi_e(\lambda)\, V(\lambda)\, d\lambda}{\int \Phi_i(\lambda)\, d\lambda}$$

where $V(\lambda)$ is the response curve used in conversions from radiometric to photometric quantities.

12.1.1 Blackbody emitters

A blackbody has perfect luminous efficacy.

A blackbody absorbs all incident power, reflecting none of it, appearing perfectly black in the presence of light.

A blackbody is perfectly diffuse, emitting radiance equally in all directions.

The radiance emitted by a blackbody is given by ***Planck's law***:

$$L_e(\lambda, T) = \frac{2hc^2}{\lambda^5(e^{\frac{hc}{\lambda k_b T}} - 1)}$$

where T is temperature in Kelvins, c is the spped of light in the medium, h is Planck's constant, and k_b is Boltzmann constant.

As temperature increases, more light is emitted in the visible frequencies, tending to blue.

Doubling the temperature increases radiant exitance M (outgoing irradiance at a point) by a factor of 16 according to the ***Stefan-Boltzmann law***:

$$M(p) = \sigma T^4$$

where σ Stefan-Boltzmann constant.

Since it grows so fast and to avoid using large numbers, we normalize radiant exitance by finding the radiant exitance L_{max} of the wavelength where it has maximum value for the given temperature according to ***Wien's displacement law***:

$$\lambda_{max} = \frac{b}{T}$$

where b is Wien's displacement constant. Normalization yields $L_e = 1$ at λ_{max} and L_e for the other λs is in $[0,1)$.

- [ ] TODO: Kirchhoff's law of non-blackbody emission.
- [ ] TODO: Color temperature.

12.1.2 Standard illuminants

The CIE computed and plotted the SPDs of various light sources and called them ***standard illuminants***.

The standard illuminant series A models blackbodies. Series B, C and D model daylight. Series E is constant. Series F is fluorescent.

TODO: Files cie.stdillum.*h in scenes/spds, but they don't seem to exist in the repository.

12.2 Light interface

Shadow rays are the ones that are traced to find occluders between the light source and a point.

Light sources with area need to get sampled.

Light sources with more power make higher contributions.

12.2.1 Visibility testing

The VisibilityTester class is mentioned always in the context of ***shadow rays***. What is the exact definition of a shadow ray? ***Camera rays*** depart from the camera. Do camera rays become shadow rays after bouncing? No, those are ***reflection rays***. Shadow rays exist between light sources and surfaces or participating media and are used to find occluders that cast shadows on the surface or participating medium at the endpoint of the ray.

VisibilityTesters get created for segments of shadow rays, between 2 Interaction points (on surfaces of participating media).

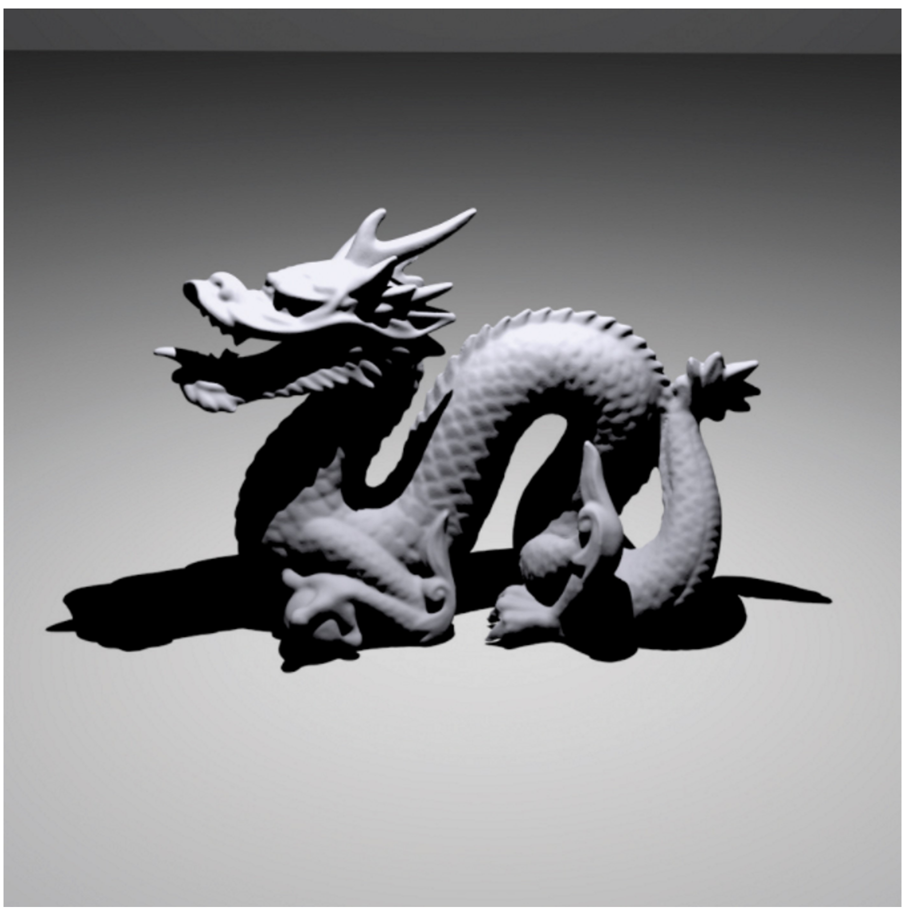
We are interested in finding occluders between the 2 Interaction points.

- [ ] TODO: Cover VisibilityTester::Tr after covering Chapter 11 Volume scattering.

12.3 Point lights

PointLight is one of many types of ***point lights***, lights with a single point of emission. Some are anisotropic, some are isotropic: their distribution of light may or may not vary with direction.

Point lights create hard shadows.

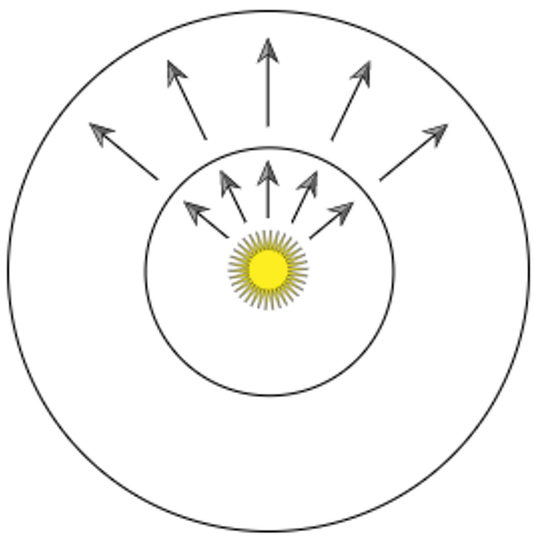


Light space is the vector space of a light. A PointLight is at the origin.

The radiometric quantity associated with point lights is [intensity](). A PointLight has constant intensity because it is ***isotropic***. Radiance is not used to describe a point light because it doesn't have area (and radiance is computed at the limit of the light's surface area projected on the surface). The ***directional distribution*** of isotropic lights is a ***delta distribution***.

PointLight implements Sample_Li to conform with the other types of lights whose radiometric quantity is indeed radiance and so that light transport algorithms don't have to have special cases. The "radiance" returned by PointLights has an ***implicit delta distribution***: a single direction. So for a given surface point, a PointLight doesn't need to be sampled at multiple incident directions.

Even when knowing that the light's "radiance" has an implicit delta distribution, simply returning the PointLight's constant intensity from Sample_Li would still be wrong, because the units are different: intensity is W/sr and radiance is $W/(m^2 sr)$. PBRT solves this by dividing intensity by the square of the distance between the point and the light, thereby adding $1/m^2$ to the units. Despite looking suspicious, this ***division*** seems to be ***physically correct***: the intensity of a point light ***falls off*** in proportion to the square of the radius of an imaginary sphere around it; in our case, the sphere's radius would extend from the point light to the point on the surface. (Intensity falls off, but flux remains constant: note that the total amount of flux measured on either of the two spheres in the next figure is the same—although less energy is passing through any local part of the large sphere than the small sphere, the greater area of the large sphere means that the total flux is the same.)



Total power emitted by a point light is obtained by integrating intensity over the entire sphere of directions $\Omega = S^2$:

$$\Phi = \int_{\Omega=S^2} I \, d\omega = I \int_{S^2} d\Omega = 4\pi I$$

12.3.1 Spotlights

- [ ] TODO

12.3.2 Texture projection lights

- [ ] TODO

12.3.3 Goniophotometric diagram lights

- [ ] TODO

12.4 Distant lights

- [ ] TODO

12.5 Area lights

- [ ] TODO

12.6 Infinite area lights

An infinitely far away area light source that surrounds the entire scene. You can imagine it as an enormous sphere that casts light into the scene from every direction.

A use case of infinite area lights is ***image-based environment lighting***.

An InfiniteAreaLight and an EnvironmentCamera can be used to produce a ***latitude-longitude radiance map*** (also known as ***equirectangular projection***) that can then be sampled by the ray tracing algorithm when a rays travel without hitting any object in the scene.

- [ ] TODO: Postponed because it's not essential for 1st implementation pass.