Materials

Steve Rotenberg
CSE168: Rendering Algorithms
UCSD, Spring 2014

Materials

- In the subject of rendering, the term *material* usually refers to the properties of how light reflects off of a surface
- There are a lot of similar terms in computer graphics to refer to this concept such as *shader*, *reflection model*, *BRDF*, *local illumination model*, etc.
- We will usually just use the term *material*, but later in the course, we will define a more precise concept called a *BRDF*
- For the purposes of this course, the concept of a *material* will contain all of the properties to define how an incoming beam of light is scattered (reflected) by the surface, including color, shininess, transparency, and more
- Example materials include glass, metal, plastic, car paint, cloth, skin, etc.

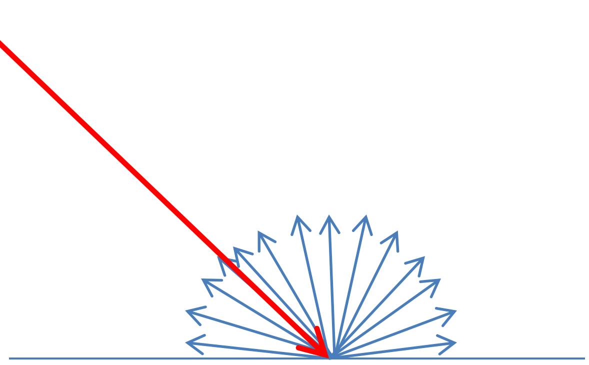
Materials

- In the last lecture, we looked at how light reflects and refracts when it hits smooth metal and dielectric surfaces
- We saw that the incident beam of light was either reflected as a single beam or split into a single reflection and a single refraction (transmission) beam
- Today we will look at some more complex examples, where light is scattered off of the material into many directions

Diffuse Materials

Diffuse Materials

- Diffuse materials are often described as having a dull or matte appearance
- An *ideal diffuse reflector* scatters incoming light equally in all directions
- One important result of this is that the surface color appears the same from any viewing direction
- Paper and smooth plaster are reasonable examples of real-world materials that behave similarly to ideal diffuse reflectors



Lambert

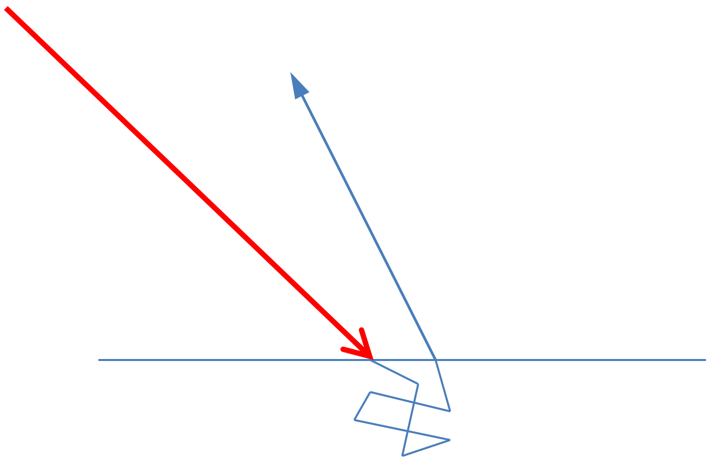
- In the computer graphics community, ideal diffuse materials are often called *Lambert* materials, or *Lambertian* materials
- Johann Heinrich Lambert was a Swiss mathematician and physicist (1728-1777)
- Lambert made a lot of contributions to science and optics, many of which are useful in computer graphics
- He is perhaps best known for being the first to prove that π is an irrational number

Albedo

- Lambert introduced a term called *albedo* which refers to the ratio of total amount of light reflected off of a surface relative to the light incident on the surface
- The albedo of a material ranges from 0 (no reflectance) to 1 (100% reflectance)
- Albedo is usually used to describe diffuse materials, but can also be useful for other types
- Examples of albedos for some materials:
  - Snow 0.8 - 0.9
  - Concrete 0.55
  - Desert sand 0.45
  - Green grass 0.25
  - Moon 0.12
  - Charcoal 0.04

Diffuse Scattering

- Diffuse scattering is actually caused by light entering the material and then bouncing around off many particles just below the surface
- After several bounces, the light may make it back out of the surface, but will be in an essentially random direction



Lambert's Cosine Law

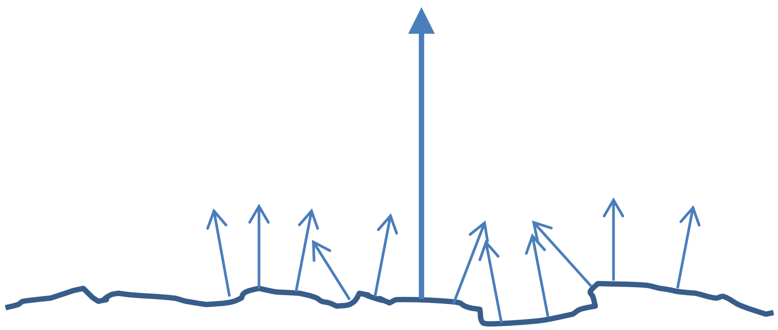
- Lambert's law of diffuse reflection says that the intensity of light reflected off of a surface is proportional to the cosine of the angle between the incident light direction and the normal

$$L_r = \frac{\rho}{\pi} \cdot L_i \cdot \cos\theta_i$$

L_r is the intensity of the reflected light
L_i is the intensity of the incident light
ρ is the albedo (essentially the 'color' of the material)
θ_i is the angle between the incident light and the normal

Microgeometry

- Many of the macroscopic optical properties of materials at due to the microscopic geometry of the surface
- Many materials are not smooth at a small scale- they have lots of little bumps
- We can think of a surface as being made up of microfacets, whose normals are described by some sort of *distribution function* relative to the average surface normal

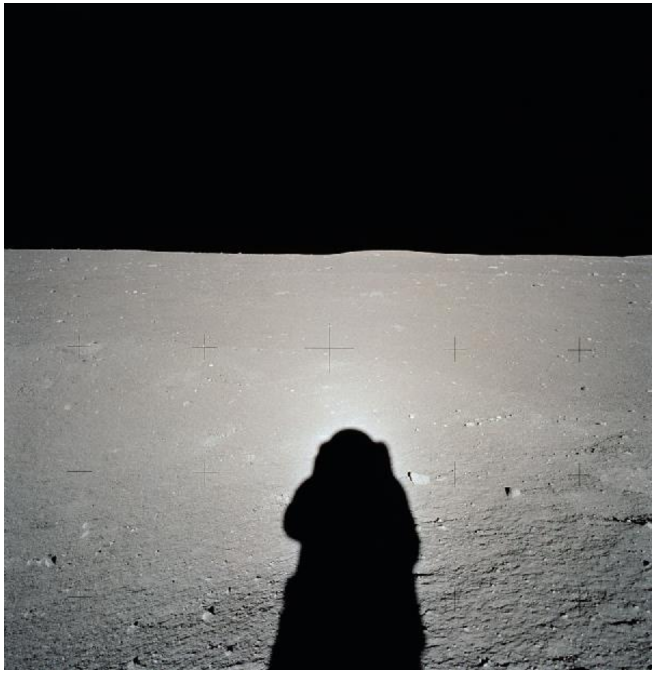


Microgeometry

- When we consider *surface roughness*, we can expect that some microfacets will shadow others from the light *(shadowing),* some will block others from view *(masking)*, and some will reflect onto others *(interreflection)*
- When seen from a macroscopic point of view, we get the aggregate effect of all of these combined
- The result is a complex distribution of reflected rays coming from a single incident ray
- Many advanced material models are derived from assumptions about the distribution of microfacets, and how light interacts between these facets

Opposition Effect

- The *opposition effect* is the visible increase in brightness when one views a rough surface from the same direction as the light source



Opposition Effect

Opposition Effect

- The opposition effect is mainly due to a phenomenon called *shadow hiding*
- When light hits a rough, bumpy surface, the lower crevices of the surface can be shadowed by the higher bumps
- These shadows are visible from most angles except when the light in coming from the viewing direction
- In this case, the shadows disappear, resulting in a visibly brighter appearance
- A second phenomenon called *coherent backscatter* is also partly responsible in certain situations when the bumps are roughly the same size as the wavelength of light, however this effect is limited to a much smaller angle

Oren-Nayar Reflectance Model

- Michael Oren and Shree Nayar developed a computer graphics reflectance model in 1993 that attempts to capture the more complex behavior of real diffuse surfaces
- It is a generalization of Lambert diffuse reflectance for surfaces with bumpy microgeometry
- It assumes that each microfacet is a pure Lambertian reflector, and it considers both shadow hiding as well as diffuse interreflection between microfacets
- As a result, it can produce the opposition effect, resulting in more realistic diffuse materials

Oren-Nayar Reflectance Model

$$L_r = \frac{\rho}{\pi} \cdot L_i \cdot \cos\theta_i \cdot \Big(A + \big(B \cdot \max\big(0, \cos(\varphi_i - \varphi_r)\big)\sin\alpha \cdot \tan\beta\big)\Big)$$

where

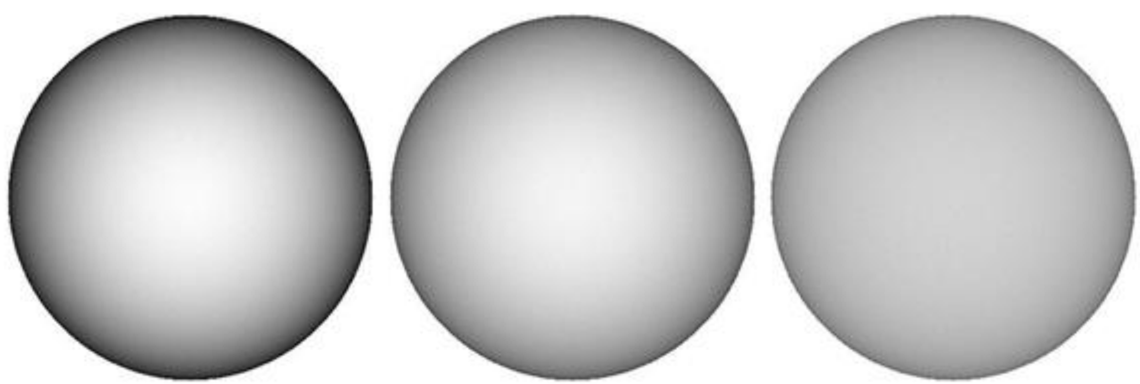
$$A = 1 - 0.5\,\frac{\sigma^2}{\sigma^2 + 0.33}$$

$$B = 0.45\,\frac{\sigma^2}{\sigma^2 + 0.09}$$

$\alpha = \max(\theta_i, \theta_r)$
$\beta = \min(\theta_i, \theta_r)$
$\sigma = \text{roughness}$ (ranges from 0 to around 0.5)
$\varphi_i - \varphi_r =$angle between incident and reflected rays projected onto the plane

Oren-Nayar Reflectance Model

Real Image | Lambertian Model | Oren-Nayar Model



$\sigma = 0$ | $\sigma = 0.1$ | $\sigma = 0.3$

Oren-Nayar Reflectance Model

- When the surface roughness is equal to 0 (perfectly smooth), the model reduces to the ideal Lambertian diffuse reflection model
- By the way, the model shown on the previous slide is their ‘qualitative model’, which makes a few simplifying assumptions and reduces the computational cost
- The original paper also proposed some more elaborate models
- The original paper is called ‘Generalization of Lambert’s Reflection Model’

Specular Materials

Specular Materials

- The term *specular* refers to mirror-like reflection
- It isn't limited to perfectly smooth mirror surfaces however
- Rough metallic surfaces appear shiny, although they don't act like perfect mirrors

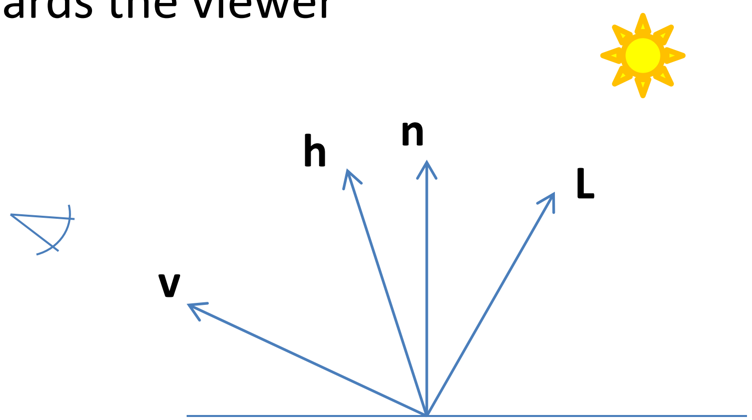
Cook-Torrance Reflectance Model

- The Cook-Torrance reflection model is based on the assumption that the surface is made up of microfacets- each of which is an ideal Fresnel metal reflector
- It was proposed by Michael Cook and Kenneth Torrance in 1981
- The Oren-Nayar model was inspired by this model
- The Cook-Torrance model was based on an earlier model by Torrance and Sparrow from 1967 that evolved from research in radar reflections

Cook-Torrance Reflection Model

- They use a vector called **v**, which is the *view vector*, a vector pointing towards the viewer (this is generally going to be –**d** vector if **d** is the ray direction coming from the camera)
- They also use a vector **L**, which points towards the light (I'm using a capital L because the lower case l looks like an I)
- They introduce a vector **h**, called the *halfway vector* which lies halfway between **v** and **L**
- **h** refers to the normal of a hypothetical microfacet that would reflect the light directly towards the viewer

$$\mathbf{h} = \frac{\mathbf{v} + \mathbf{L}}{|\mathbf{v} + \mathbf{L}|}$$



Cook-Torrance Model

$$L_r = L_i \cdot \frac{F \cdot G \cdot D}{\pi(\mathbf{n} \cdot \mathbf{L})(\mathbf{n} \cdot \mathbf{v})}$$

F =Fresnel term

G =Geometric attenuation term

D =Microfacet distribution function

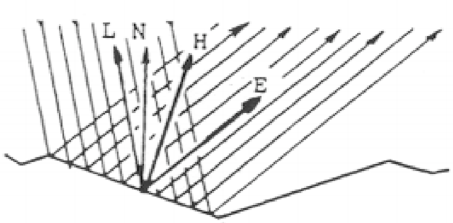
Fresnel Term

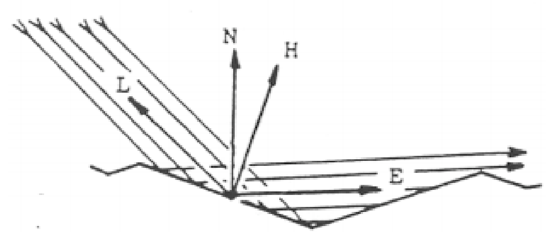
- The Fresnel term *F* can be the Fresnel equation for metals that we looked at in the previous lecture
- There are also various simplifications that have been proposed

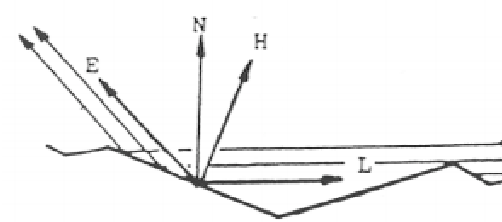
Geometric Attenuation

- *Geometric attenuation* refers to the decrease in light reflection due to both shadowing and masking

$$G = \min\left(1, \frac{2(\mathbf{n}\cdot\mathbf{h})(\mathbf{n}\cdot\mathbf{v})}{(\mathbf{v}\cdot\mathbf{h})}, \frac{2(\mathbf{n}\cdot\mathbf{h})(\mathbf{n}\cdot\mathbf{L})}{(\mathbf{v}\cdot\mathbf{h})}\right)$$







Microfacet Distribution Function

- There have been various functions proposed that describe the distribution of microfacets around the average surface normal

- Gaussian: $D = ce^{-(\alpha/m)^2}$
- Beckmann: $D = \frac{1}{m^2 \cos^4 \alpha} e^{-\left(\frac{\tan^2 \alpha}{m^2}\right)}$

where

α=acos(**n**·**h**)
m=root mean square slope of microfacets
c=an arbitrary constant (?)

Cook-Torrance Reflection Model

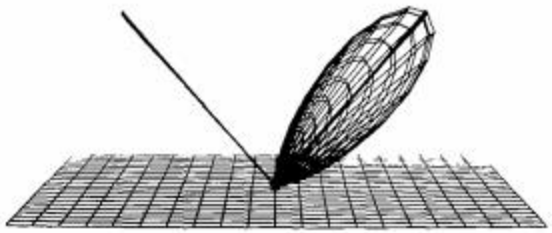


Figure 3a. Beckmann distribution for m=0.2.

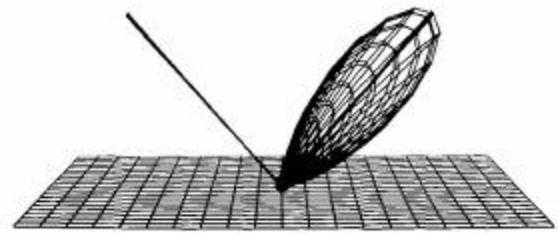


Figure 3b. Gaussian distribution for m=0.2.

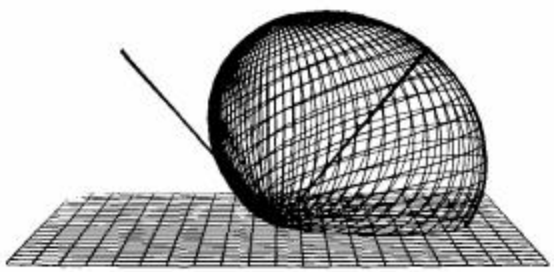


Figure 3c. Beckmann distribution for m=0.6.

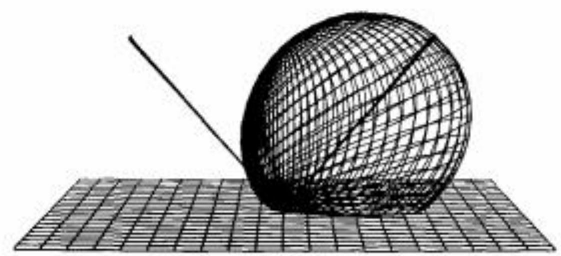


Figure 3d. Gaussian distribution for m=0.6.

Anisotropic Materials

Isotropic vs. Anisotropic

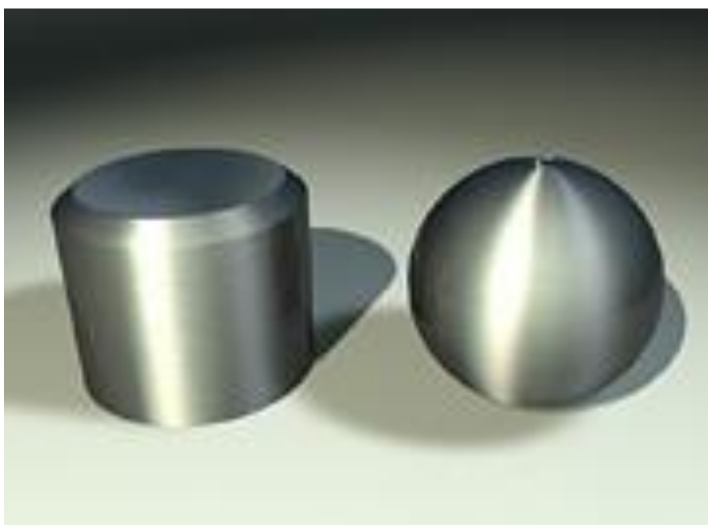
- Lets say that we place a sample of a material flat on a table in front of us, and we have a light source in the room shining at the table
- Then, without moving the light or changing our viewing angle, we rotate the material on the table
- If the reflected color we see remains constant as the material rotates, we call the material *isotropic* (isos=equal/same, tropos=turning/circle)
- If the reflected color changes as the material rotates, we call it *anisotropic* (an=not)

Isotropic Materials

- Many common materials are isotropic due to the overall random distribution of surface microgeometry combined with random distribution of pigment particles in the medium
- There is no inherent directionality at the microscopic scale which leads to no visible directionality at the macroscopic scale

Anisotropic Materials

- Some materials do have some sort of inherent directionality at the microscopic level
- A common example is brushed metals, where the metal surface is roughened along one particular direction

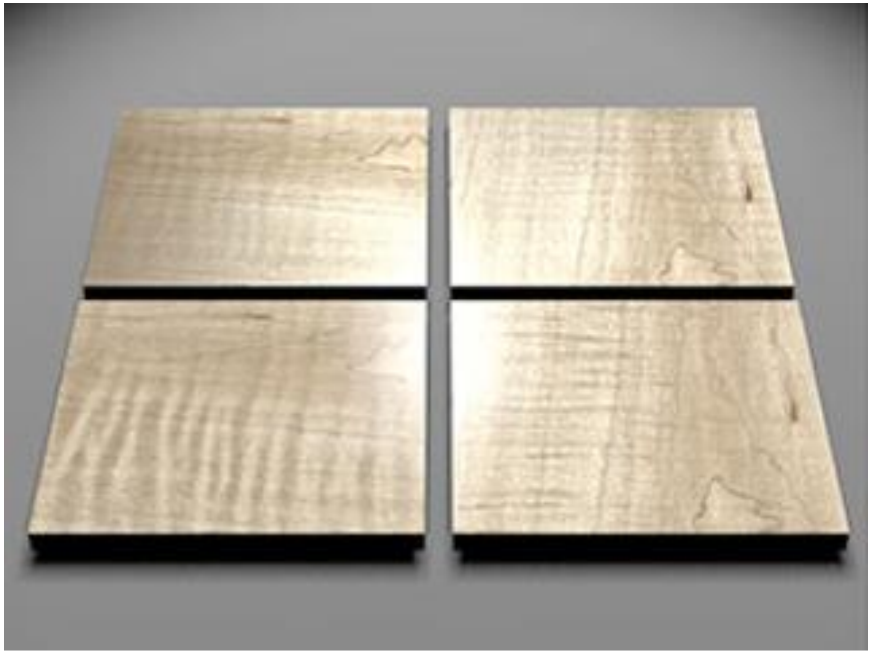


Anisotropic Materials

- Cloth is another example of an isotropic material, due to the directionality of the threads in the weave
- Satin and velvet are two good examples of complex fabrics

Anisotropic Materials

- Wood and some other natural materials sometimes have a anisotropic appearance due to the directional alignment of cells



Anisotropic Materials

- Hair and fur are also strongly anisotropic

Anisotropic Materials

- To render an anisotropic material, we need more information about a surface than just the position and normal
- We need some sort of information about the orientation of the material in the plane
- Typically, we use tangent vectors, which are in the plane and provide a reference frame for the material orientation
- We will look at this in more detail in a later lecture, as well as looking at some anisotropic reflection models

Other Material Properties

Rough Dielectrics

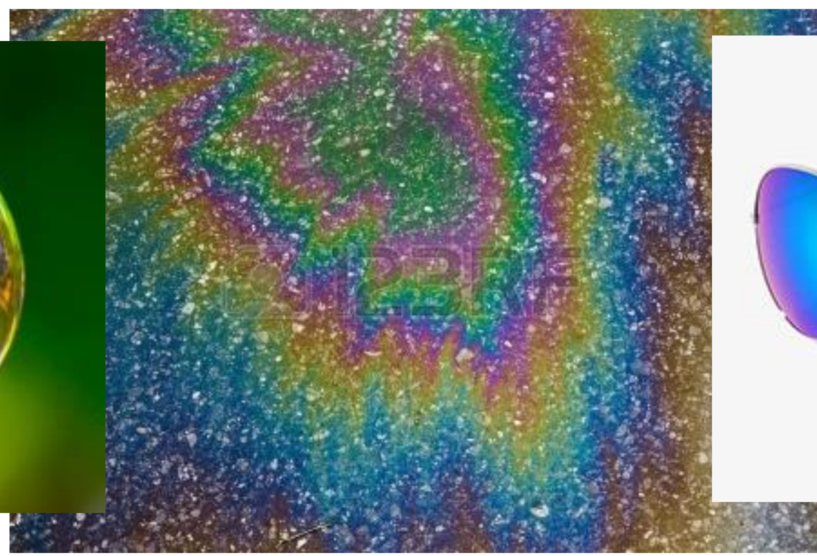
- We can derive models for rough dielectric surfaces, similar in concept to the Cook-Torrance model for rough metals

Retroreflection

- *Retroreflection* refers to specular reflection back towards the light source
- It is not the same as the opposition effect, but it is another type of *backscatter* phenomenon

Iridescence

- Iridescence refers to property of some materials changing color depending on the view direction
- This can be caused by different phenomena such as constructive and destructive interference in thin films like bubbles, oil on water, or surface coatings



Diffraction

- Diffraction of light on bumps near the wavelength of light can also cause iridescent effects

Translucency

- Translucency and subsurface scattering are other common properties that can be captured
- We'll look at these some more in a later lecture

Material Rendering

Materials

- Because there are a wide range of materials, it is nice to allow a flexible definition of materials, instead of just having one single material model
- This is a perfect place to take advantage of derived classes and virtual functions in C++
- We can create a base class Material and derive various specific material types from that

Material Class

```
class Material {
public:
    Material();
    virtual void ComputeReflectance(Color &col, Vector3 &in, Vector3 &out,
                                    Intersection &hit)=0;
};
```

Colors

- The subject of color is actually quite complex and we will discuss it in a lot more detail in a later lecture
- For now, I just want to mention that it is important to have a Color class that is used in all places where the renderer does operations on colors
- It is tempting to just use a Vector3, since we think of colors as having 3 componets (red, green, blue), and we do a lot of similar operations as vectors (addition, scaling, etc.)
- However, as we will see later, colors really should be treated as spectral distributions across all visible wavelengths (not just 3!)
- If we use a Color class and just make it simple RGB for now, then we can later swap in a more sophisticated color class that uses the same interface (Add(), Scale()…) and upgrade to a more realistic color model with minimal effort

Color Class

```
class Color {
public:
        Color();

        void Add(const Color c);
        void AddScaled(const Color c,float s);
        void Scale(float s);
        void Multiply(const Color c);

        void Exponent();              // Computes e^c
        void Gamma(float exp);        // Computes pow(c,exp);

        int ToInt();                  // Converts to 24 bit RGB
        void FromInt(int c);          // Converts from 24 bit RGB
};
```