CS**4620** Spring 2018
Lecture notes: **Shading surfaces in a ray tracer**

Once we know what point on what surface is visible at a particular pixel, we need to determine what color that pixel should be. This is “shading” the surface. In a realistic context, you can think of shading as computing the answer to a physics problem: given this kind of surface, illuminated by that kind of light, how much light reflects from the surface around the shading point into the eye (or camera)?

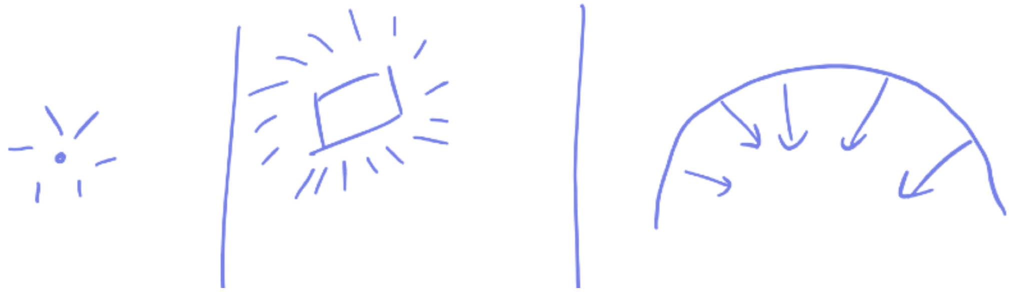
How you should compute shading depends on what the goals of your rendering are and on what kind of surface and lighting condition you are trying to depict. If the rendering is meant to be illustrative, for instance a scientific visualization or a real-time rendering of a surface the user is designing in a CAD system, visual clarity and simplicity may be the main goal. If the rendering is meant to be realistic, as in a product image to be used in advertising, or a visual effects image destined to be merged with real film footage, accurate resemblance to a real photograph may be the main goal. Or your goals might be in between, as in an animated film where a not-quite-realistic look is desired.

In any case the basic starting point is the basic physics of light reflection; from there we might pursue elaborate reflection models to replicate real optical effects, or we might use simple models, and even simplify the physics unrealistically in pursuit of visual clarity. But today we look at the physical framework for light reflection, as well as the most important surface reflection models, simple but useful approximations of real surfaces.

Light power falling on a surface

Light is a form of energy and can be modeled in various ways; for this lecture we will think of it as particles flowing through space, each carrying a little bit of energy. We will ignore color for the most part, so these particles are all the same and we just need to worry about how many there are in any particular place, and we’ll measure that quantity in terms of power: the total energy flowing per unit time. The physical unit for this is Watts. (There are other units that get used in practical situations that worry about distinguishing visible light from other energy, and separating light out by wavelength for color computations, but we will simplify all this away for now.)

Light originates at light sources, which can take many forms. The simplest, which we’ll stick to for today, is a point source: an object that is very small relative to its distance from anything that matters, so that we don’t care about its physical extent and just treat it as a point. Some examples of sources that we can often treat as points are the LED flashlight in your phone, or the headlights in a car. Other useful kinds of light sources include directional sources, which are like faraway points (e.g. the sun), area sources (like a fluorescent tube or a computer monitor), and environment sources (like the sky).



If a point source sends light equally in all directions it's called "isotropic." Suppose we have an isotropic point source with power P. How much of that power arrives at a surface we are shading? Let me argue with two simple thought experiments that the amount is going to be

$$\frac{P}{4\pi}\frac{A\cos\theta_i}{r^2}$$

where A is the area of the (assumed small and flat) surface, and r is the distance from the source to the surface.

Suppose we surround our source with a big sphere that collects all the light coming out. The whole surface receives power P, and the power is uniformly distributed over it. Draw a circle around any part of that big sphere, and call the area of that smaller piece of surface A. Since the area of the whole sphere is $4\pi r^2$, and the power is uniformly distributed everywhere, our smaller piece of surface receives power

$$\frac{P}{4\pi r^2}A$$

Of course this same thing is true if the surface is not part of a big sphere. If I have a flat surface facing the source, and it's small enough that the curvature of a sphere of radius r is negligible, then we can think of it as part of a hypothetical sphere and use this expression to compute the amount of power it receives.

Of course not all surfaces face directly towards the light. A surface facing exactly sideways receives no light, and generally a surface whose normal makes an angle θ with the direction to the light receives a fraction $\cos\theta$ of what the same surface area receives when facing the light.

So, in the end, a small surface of area A, with surface normal $\mathbf{n}$, illuminated by a point source from direction ω_i, receives power

$$P(A) = \frac{P}{4\pi}\frac{\cos\theta_i}{r^2}A = I\,\frac{\cos\theta_i}{r^2}A = I\,\frac{\mathbf{n}\cdot\omega_i}{r^2}A$$

In this formula, the first factor (*P*/4*π*) is called the light source intensity, *I*; the second is a geometry factor.

Irradiance

You'll notice this discussion is all in terms of a finite flat surface of area *A*. But going back to the problem of shading in a ray tracer, we want to shade at a single point on a potentially curved surface. The way to reconcile these two notions is to think of the light arriving at a point **x** as the limit of light-per-unit-area arriving at ever smaller areas containing **x**. This is called the "irradiance" at **x**; it is a density of power with respect to area on the surface. In the case of a point source it is simply the power derived just above, divided by the area, which cancels the area, leaving:

$$E(\mathbf{x}) = I\frac{\mathbf{n}\cdot\omega_i}{r^2}$$

You can think of it analogously to a density of rainfall; we measure that in terms of the volume of water that falls on a small area of surface. In that case the units work out to, say, cubic centimeters per square centimeter, or just centimeters. In the case of light, irradiance is measured in Watts per square meter.

Irradiance is the starting point for surface reflectance calculations; the cosine factor and inverse-squared-distance factor each produce important lighting effects. The cosine factor makes brightness fall off smoothly as surfaces tilt away from the light, and the inverse distance factor means that objects far from the light source are lit much more dimly than surfaces close to the light.

Diffuse reflection

So far we only know about how much light falls on the surface; we wanted to know how much reflects. The simplest model for this is diffuse reflection, or Lambertian reflection, named after Lambert, who figured out how irradiance works back around 1760. It's very simple: it says the irradiance that falls on the surface is reflected equally in all directions, so the observed color of the surface is the same, no matter where you are looking from. Some examples of surfaces that are modeled pretty well by Lambertian reflection are a field of snow, a piece of chalk, or a piece of drawing paper.

In Lambertian reflection, the reflected light is directly proportional to the irradiance:

$$L = k_d E.$$

The factor k_d can be called the diffuse coefficient. But a more useful notion is diffuse reflectance, which is the fraction of light energy reflected, as opposed to absorbed. Reflectance is easy to reason about; it has a valid range from 0 (perfectly absorbing black surface) to 1 (perfectly reflecting white surface). In typical practice a very black paint might have R_d = 0.03, and a bright white surface like paper might have R_d = 0.85 or even 0.9. Brighter and darker surfaces exist in the optics laboratory but they are expensive and have to be kept scrupulously clean to maintain their properties.

A useful reference value to have in mind is a reflectance of 18%, 0.18, which is what photographers define as “middle gray”. This is a good value to code as a default for diffuse reflectance.

The relationship between R_d and k_d, which for now I will just state, is

$$k_d = \frac{R_d}{\pi}.$$

The unit for L, the reflected light from a surface, is called “radiance.” More on that a bit later also.

So in total, if you have a point **x** on a diffuse surface of reflectance R_d, which has surface normal n, illuminated from direction ω_i by a point source of intensity I that is r units away, you can compute the color (radiance) of a pixel that sees **x** as

$$L = E\frac{R_d}{\pi} = I\frac{\mathbf{n}\cdot\omega_i}{r^2}\frac{R_d}{\pi}.$$

This is the simplest shading model. You will find many traditional graphics sources that omit the π, which leads to diffuse colors that are too bright; this is not a problem when everything is basically heuristic and can be adjusted to match, but when physics-based models that predict correct brightness are used, it will make things wrong. Also, some traditional models omit the inverse-square factor. This is very wrong from a physics perspective but can be useful in contexts where realism is not important, because it makes it easier to get all the colors into a reasonable range of brightness.

Diffuse reflection generally arises from surfaces that have “messy” 3D structure—intuitively, interacting with messy, random stuff causes light to become randomized and “forget” which way it came from. The examples above all have this property: snow is a loose pile of ice crystals with complex shapes; chalk is a packed powder consisting of many irregular particles; paper is a compressed mass of randomly arranged wood fibers. It is not a perfect model for any of these materials; the reflected light does depend on the direction, and even for pretty diffuse surfaces the Lambertian model can easily be wrong by factors of 2 to 5 or even more, particularly in backward (light is near the viewer) or forward (light is opposite the viewer) configurations. But it is a very useful baseline, and widely used.

Specular reflection (caution: notes under construction starting here!)

In many materials, some or all of the light reflects from the top surface of the material, rather than from a messy 3D subsurface. The extreme case is a very smooth surface (arising because of surface tension, e.g. a calm pond or float glass used in windows, due to crystal cleavage as in some minerals, or due to deliberate polishing as in optical lenses or polished mirrors. When a surface is very smooth it produces a mirror reflection, where light arriving from a particular direction is reflected into exactly one direction (and possibly refracted into a second direction inside the material). When it is somewhat imperfect, with little irregularities on the surface (called surface roughness) that are small compared to the scale of the scene, the result is a surface that reflects light in a mirror-like way but spreads it out a bit.

Examples of surfaces like this abound: paints, plastics, coated paper, ceramic tile, skin, leaves, unpolished metal, … all kinds of materials that are shiny to some extent (which is due to surface reflection) but don't reflect a sharp mirror image (which is due to surface roughness). Reflection from a shiny surface is known as "specular reflection." Most of these materials (except metal) exhibit both diffuse and specular reflection, because light that doesn't reflect from the surface goes into the interior of the material and scatters diffusely.

Unlike diffuse reflection, specular reflection *does* depend on where the viewer is as well as where the light is. Visually, the characteristic of specular reflection is that you see reflections of light sources, which are called "specular highlights," that move around on the surface as you change your view.

If you play around with some specular surfaces and a flashlight, you can immediately see a few features of specular reflections. They are usually white (except on colored metals); they can range quite a bit in size and brightness depending on the material (small and bright for smoother finishes, broad and dimmer for rougher finishes); and they are a lot brighter at grazing angles, when the light source is opposite the viewer and the viewing and illumination directions are close to tangent to the surface.

Thus the second most useful type of reflection model (after diffuse) is one that describes specular reflection from rough surfaces.

Bidirectional Reflectance Distribution Function (BRDF)

To model specular reflection, surface shading has to depend on both the light direction, ω_i, *and* the viewing (reflected) direction, ω_r. A mathematical model for how shading varies is called a "bidirectional" reflectance model because it depends on two directions. We encapsulate this bidirectional dependence in a function

$$f_r(\omega_i, \omega_r)$$

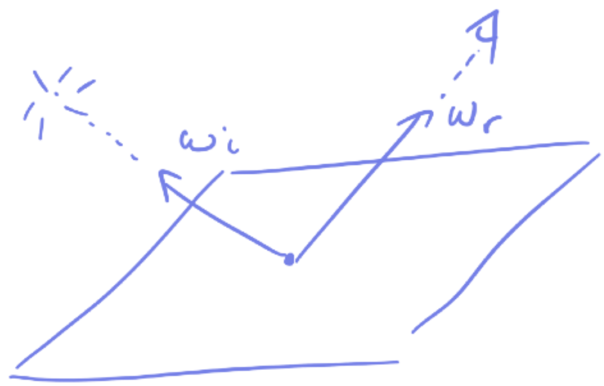
that plays the same role as the diffuse reflection coefficient but varies with viewing and illumination direction. By this I mean that you compute the radiance seen by the camera, when a surface is illuminated by a point light, as:

$$L = f_r(\omega_i, \omega_r)E = f_r(\omega_i, \omega_r)\frac{\mathbf{n} \cdot \omega_i}{r^2}I \, .$$

The function f_r is known as the **b**idirectional **r**eflectance **d**istribution **f**unction of the surface, abbreviated BRDF. This is simply a generalization of the formula for computing diffuse shading, where a diffuse surface simply has a BRDF that happens to be constant:

$$f_r(\omega_i, \omega_r) = \frac{R_d}{\pi} \text{ (for diffuse surfaces only).}$$

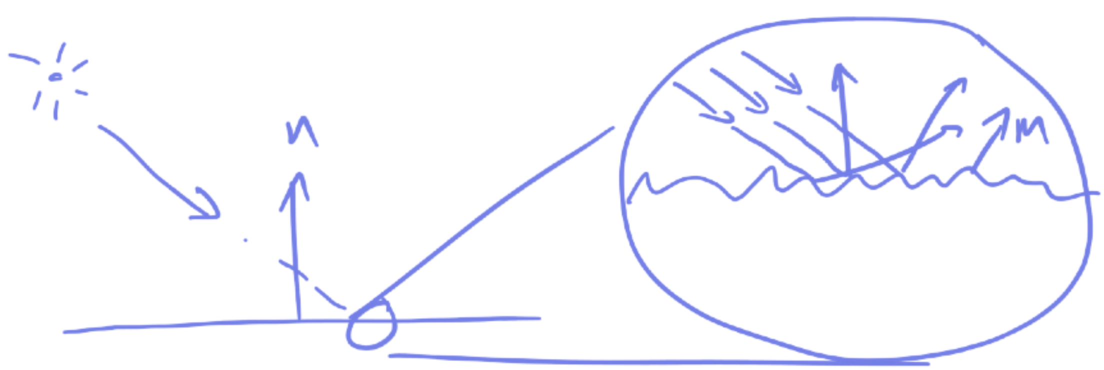
The BRDF answers the question "If I illuminate the surface with a unit of radiance from direction ω_i, how many units of radiance will I observe in direction ω_r?



Once we write shading this way, knowing what a particular material looks like boils down to knowing the BRDF. By "knowing," I mean that we have a procedure in hand that we can use to compute the BRDF for any given values of ω_i and ω_r.

The microfacet reflection model

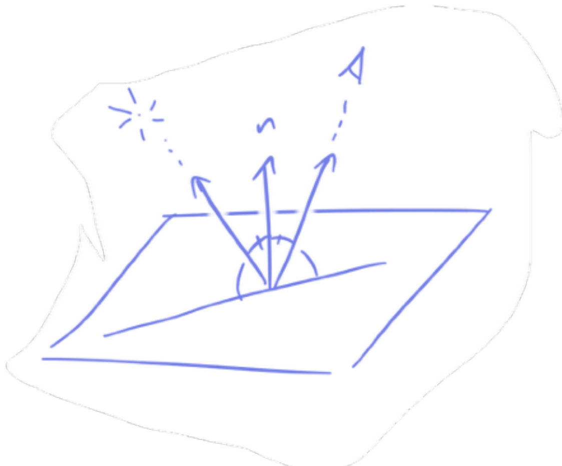
The most widely used type of model for specular reflection that serves as a decent approximation for real surfaces is a "microfacet" model. There are several variants of this idea in use, but they all share a simple basic idea: we think of the surface as being covered with tiny bumps (much smaller than a pixel), but if we zoom in on these bumps, their surfaces are smooth and mirrorlike. When light encounters the surface, each ray locally reflects from this mirror *microsurface* and since the surface is bumpy each ray goes in a somewhat different direction.



This mental model provides a framework to write down an approximation for The components that go into a microfacet model describe how light reflects from this microsurface, locally, and then how often the bumpy geometry directs light towards the viewer.

Fresnel reflection

The road to a model for rough surface reflection starts with mirror reflection. The best-known way to describe the geometry of mirror reflection is that it happens when the angle of incidence equals the angle of reflection, which is to say the incident and reflected directions ω_i and ω_r and the surface normal **n** are coplanar and ω_i and ω_r both make the same angle with **n**:



but another way to say the same thing, which is particularly convenient to work with, is that the “half vector,” **h**, the vector that bisects the angle between ω_i and ω_r, coincides with **n**:

The half vector is simple to compute because it’s simply the average of ω_i and ω_r:

$$\mathbf{h}(\omega_i, \omega_o) \;=\; \frac{\omega_i + \omega_o}{|\omega_i + \omega_o|}$$

where the denominator is there to scale **h** back to a unit vector.

The half vector tells us what ray the light is reflected to, but the amount of light reflected depends on the physical properties of the material: electrical conductors reflect most of the light, and absorb a bit, and electrical insulators (dielectrics) transmit most of the light, reflect some, and don’t absorb any at the surface. The behavior is well described by Fresnel’s laws, which give the reflectance (the fraction of light reflected) as a function of the angle at which the light arrives at the surface and the refractive indices of the material above and below the surface. The physics behind them is interesting and not terribly complicated, but here we’ll just state the laws. There are many ways to write them down; one is:

$$F(\omega_i, h) \;=\; \frac{1}{2}\frac{(g-c)^2}{(g+c)^2}\left(1 \;+\; \frac{(c(g+c)-1)^2}{(c(g-c)+1)^2}\right)$$

where

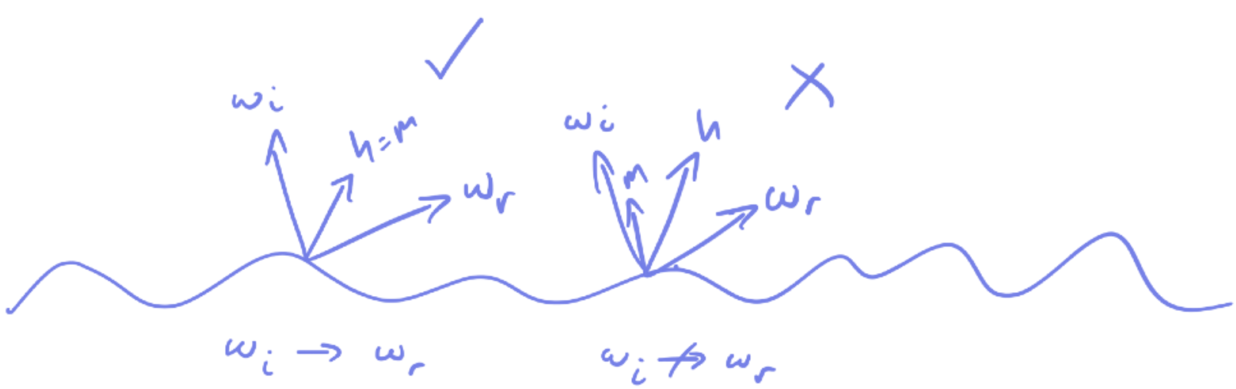
$$g \;=\; \sqrt{\frac{n_t^2}{n_i^2} - 1 + c^2}$$

$$c \;=\; |\omega_i \cdot \mathbf{h}|$$

These equations are for reflection from dielectric (insulating) materials when light is not polarized (something we'll ignore in this class). Fresnel's laws can also be used (in somewhat more general form) to predict reflectance factors for metals, but it is a bit more involved because of the losses involved in reflection from imperfect conductors, which also depend on wavelength, producing strong colors in metals like gold and copper.

Microfacet distribution

Under these assumptions, the key to answering the question "how much of the light coming from ω_i goes to ω_r?" is to answer the question "how often is the normal to the microsurface equal to **h**?" This is because a ray from ω_i will reflect to ω_r exactly when the microsurface normal it encounters when it hits the surface happens to equal **h**.



A microfacet model starts with the answer to this second question, called the "microfacet distribution function." This is a function D(**h**) that gives the probability that the surface normal at a randomly selected point on the microsurface will equal **h**. (To be more precise it's a probability density for the normal being near **h**, rather than a probability of being exactly **h**.) For a typical surface D(**h**) is largest for **h** near the overall normal of the surface (the macrosurface normal).

A popular particular form for D(**h**) is the Beckmann distribution, which is derived from the theory of gaussian processes. This distribution has a single parameter α that describes how rough the surface is, from very smooth (shiny plastic, perhaps) around $\alpha = 0.01$ to moderately smooth (a less-shiny ceramic glaze, perhaps) at $\alpha = 0.1$ to extremely rough (think MacBook aluminum) at $\alpha = 0.3$ to 0.5. The Beckmann formula is:

$$D(\mathbf{h}) \;=\; \frac{\chi^+(\mathbf{h}\cdot\mathbf{n})}{\pi\alpha^2\cos^4\theta_h}\;\exp\left(\frac{-\tan^2\theta_h}{\alpha^2}\right).$$

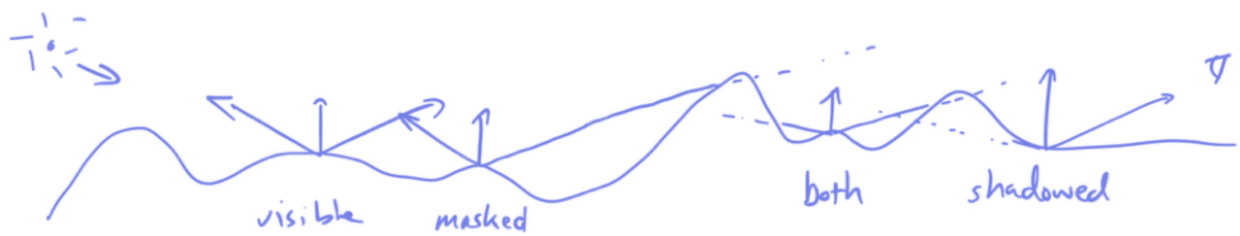
Here χ^+ is a function that returns 1 if its argument is positive and zero otherwise, and θ_h is the angle between **h** and **n**. This distribution takes the form of a peak with its maximum occurring when **h** = **n**.

Shadowing and masking

One problem with a microfacet model defined using only *F* and *D* is that it predicts too much reflection at grazing angles. The problem is that we have not yet accounted for the fact that light doesn't fall on the entire surface, and light also can't always escape from all points on the surface toward the reflection direction, because other parts of the microsurface get in the way. We sum up the effects of shadowing and masking in a function

$$G(\omega_i, \omega_r, \mathbf{h})$$

that answers the question "how likely is it that a point with microsurface normal **h** can see out both in direction ω_i and in direction ω_r?"



There are a few ways to define *G*, all of which produce a factor that is near 1 for most angles, and drops to zero as either ω_i or ω_r goes to grazing (near tangent to the surface) directions. Exact expressions are not always possible; for Beckmann, Walter recommends using an approximation in terms of a rational function:

$$G(\omega_i, \omega_r, \mathbf{h}) = G_1(\omega_i, \mathbf{h})G_1(\omega_r, \mathbf{h})$$

$$G_1(\mathbf{v}, \mathbf{h}) \;=\; \chi^+\left(\frac{\mathbf{v}\cdot\mathbf{h}}{\mathbf{v}\cdot\mathbf{n}}\right)\begin{cases}\frac{3.535a + 2.181a^2}{1 + 2.276a + 2.577a^2}, & \text{if } a < 1.6\\ 1 & \text{otherwise}\end{cases}$$

where a = $\alpha \tan\theta_v$, and θ_v is the angle between **v** and **n**.

Putting it all together

With these three functions in hand, we can write down the complete expression for the BRDF:

$$f_r(\omega_i, \omega_o) \;=\; \frac{F(\omega_i, \mathbf{h})G(\omega_i, \omega_o, \mathbf{h})D(\mathbf{h})}{4|\omega_i\cdot\mathbf{n}||\omega_o\cdot\mathbf{n}|}$$

It's simply the product of those three factors: how much surface is doing the reflecting (*D*), what fraction reflects (*F*), and what fraction is not shadowed or masked (*G*). The cosine terms in the denominator come from geometric considerations that we won't expand on here. The surface normal **n** is implicitly a parameter to this function, and the half vector **h** is implicitly a function of ω_i and ω_r.

One can create other microfacet models by choosing different functions for *D*. Roughly speaking, the shape of highlights on the surface will resemble the shape of *D*. If D is narrow, highlights are small; if it has long tails, the highlight has long tails. A popular alternative to Beckmann is a function called GGX or Trowbridge-Reitz, which is provided in the Walter et al. paper.