SUNG-EUI YOON, KAIST

RENDERING

FREELY AVAILABLE ON THE INTERNET

`http://sglab.kaist.ac.kr/~sungeui/render`

First printing, July 2018

8

Illumination and Shading

In this chapter, we look into basic concepts of illumination and shading. These topics can have different meanings depending on the context. In this chapter, we focus on computing effects of lights for illumination per vertex, followed by applying those illumination results to fill triangles. Before we talk about these concepts, let us first think about how we see things.

8.1 How can we see objects?

Each one of us might have thought about how we can see objects at one point in the past, since seeing things is a part of our daily activity. To fully explain the whole process is beyond the scope of this book. Instead, we would like to point out main components of this process.

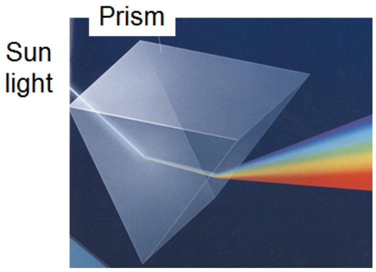

Figure 8.1: This illustrates that the sun light is composed of different wavelengths, which are perceived in different colors. The image is excerpted from the Newton magazine.

At a high level, seeing objects means that we receive the light energy in our eye, which is in the end transferred to our brain. Let us first talk about the light. The light is electromagnetic waves, and our human eyes see only a portion of the spectrum of those electromagnetic waves, commonly called visible light (Fig. 8.1). Visible light refers to wavelengths in a range of 400 and 700 nm.

Our eye has multiple layers and one of them is retina, which contains photoreceptor cells sensing the visible light. There are mainly two types of such cells: rod and cones. The rod cell is extremely sensitive to photons and can be even triggered by even a single photon. The rod cells give information mainly about intensity of the light, while cone cells are about the color information. There are also three types of cones, each of which responds to different wavelengths, which we call red, green, and blue colors (Fig. 8.2). In reality, color does not exist, but based on response levels from these three cone types, our brain reconstructs the color.

Color does not exist in reality. Instead, our brain reconstructs based on response levels of red, green, blue cones.

Now let's consider how the light interacts with materials. This process can be explained in different levels including quantum physics, but in this chapter, we give only a high level idea on the process.

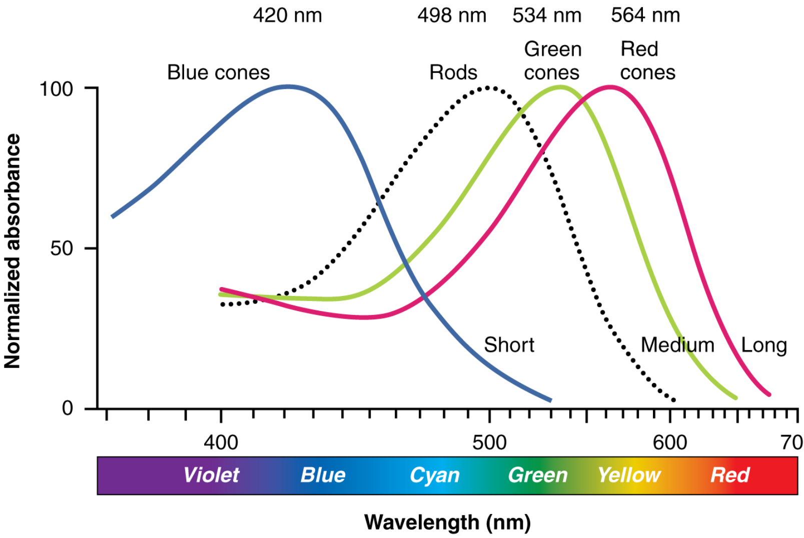

Figure 8.2: This figure shows the response level of rod and cone cells as a function of wavelengths. This figure is available from Anatomy and Physiology under the Creative Commons Attribution 3.0 license.

Once a material or an atom receives a photon, the atom enters into its excited sate. It then returns back to its normal state, while emitting its energy into the space. The energy can be interpreted into another photon or wave, thanks to the duality of the light. The key factor that we need to know is directions of the emitted photon and their wavelengths that determine the perceived color.

As an concrete example, please consider a leaf shown in Fig. 8.3. The incoming sun light is a mix of various electromagnetic waves with a set of different wavelengths, and thus can be perceived as a white light. Once the sun light hits with the leaf, the leaf receives its energy, which is used for its photosynthesis and dissipated as heat. Nonetheless, some of its received energy is emitted into waves (and particles) with different wavelengths. In this leaf case, the emitted energy has the wavelengths corresponding to the green color. As a result, we see the green color to the leaf. Furthermore, the emitted energy is distributed into all the possible directions, and thus we can see the leaf in any directions towards it.

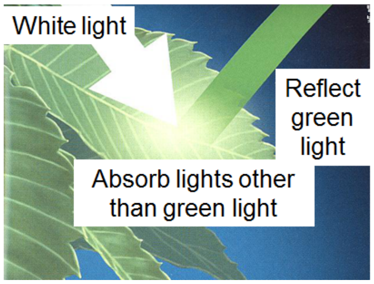

Figure 8.3: This illustrates how the leaf interacts with the coming light. This image is excerpted from the Newton magazine.

8.2 *Bi-Directional Reflectance Distribution Function*

Depending on materials, they have different reflectance behaviors. For example, the chalk is diffuse, meaning that it reflects the light in all possible directions. We thus see the chalk in any view directions. On the other hand, when we look at the surface of an apple, there can be a highlight, a bright spot, when we have a particular view direction. We call such materials to be glossy.

BRDF is used to explain the reflection behavior of a material.

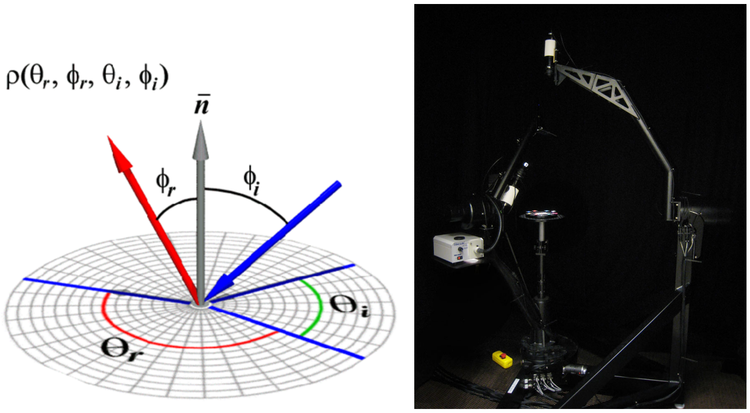

Figure 8.4: The left image shows incoming and outgoing directions, which are four parameters of a BRDF. The right image shows a gonioreflectormeter measuring the BRDF; it is from Univ. of Virginia.

Since materials have different reflectance distributions, we need a function to encode such behaviors. Bi-directional reflectance distribution function (BRDF) is introduced to meet the requirement. BRDF, $f(\cdot)$, is defined over incoming light direction and outgoing light directions. Each direction in the 3D space is encoded with two parameters, θ and ϕ. As a result, BRDF is a four dimensional function (Fig. 8.4). Detailed explanation on BRDF is available at the chapter on radiometric quantities (Ch. 12).

Gonioreflectormeter is used to measure BRDF, by rotating a light source and sensor location (Fig. 8.4). This approach takes very long time, and thus it is one of active research areas to efficiently measure the BRDF.

Measured BRDF itself can be very large in terms of memory footprint. It is thus common to encode and use them in a compact representation. In the following section, we discuss one of most simple illumination models.

8.3 *Phong Illumination Model*

The Phong model is an empirical model, but is used a lot for its simplicity.

The Phong illumination model is a simple and classical illumination model that is adopted in early versions of OpenGL. This model is just empirical, not based on physics, and does not even preserve basic physical assumption such preserving energy. Nonetheless, it has been commonly used thanks to its simplicity.

The Phong illumination model has the following three components:

- **Ambient term.** The ambient term represents a kind of background illumination, and works as a constant value (Fig. 8.5). Specifically, for computing the reflected ambient illumination, $I_{r,a}$, it multiplies an ambient reflectance coefficient, k_a, to an incoming ambient illumination, $I_{i,a}$, i.e., $I_{r,a} = k_a I_{i,a}$. Intuitively, this is a drastic

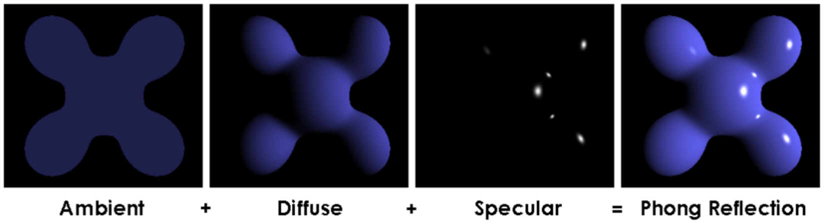

Figure 8.5: This shows different terms of the Phong model. This figure is made by Brad Smith.

simplification of complex inter-reflection between lights and materials. Simulating this term well is a critical component of global illumination, while it is drastically ignored in the Phong model.

- **Diffuse term.** Most objects can be seen in any view directions, and this indicates that they are diffuse. The diffuse term aims to support this visual phenomenon.

- **Specular term.** Certain objects such as metals show strong highlights in a particular viewing direction. The specular term simulates this feature.

Before we discuss diffuse and specular terms in a detailed manner, let's first discuss light sources, which are also mentioned when we explain the ambient term. For the ambient term, we use an ambient light that virtually emits light energy to every location of triangles. We discuss point and direction light sources, followed by briefly mentioning other types of light sources.

Point and directions light sources. The light direction plays an important role on computing illumination. A point light source emits light energy from a single point, p_l. The point light may seem too crude approximation compared to light sources that we encounter in real life. Nonetheless, we can approximate them by using a set of point light sources.

The light direction, $\vec{L}$, on a point, p, on a surface is then computed as the following:

$$\vec{L} = \frac{p_l - p}{|p_l - p|}. \tag{8.1}$$

Note that the light direction varies depending on the location of the surface p.

Unlike the point light, the directional light is located far away from the observer, and thus the light direction is considered as a constant, irrespective of observing locations. The directional light can be thought as a point light source whose location is set at infinity.

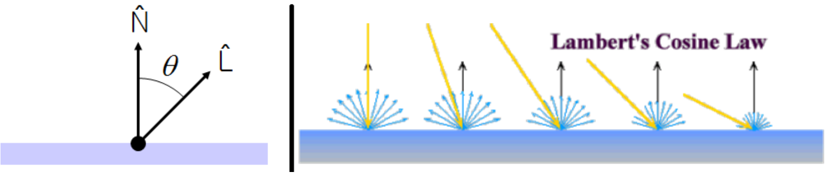

Figure 8.7: The left shows the configuration of the Lambert's cosine law, and its effects are shown in the right.

For example, sun is located far away from us, and thus we use the directional light to represent the sun light source.

Area light sources are a common type of light sources. The area light has a certain light shape with an area and thus can generate soft shadows (Fig. 8.6). Directly considering area lights is more complex than working with point lights. A simple approximation to an area light is to generate a set of point lights on the area light. The number of generated point lights defines illumination levels of soft shadow.

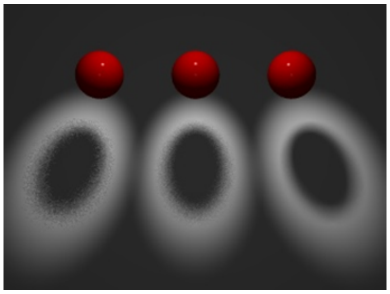

Figure 8.6: Area lights.

Diffuse term. Many objects have the diffuse property, and that is why we can see them! Here we assume the ideal diffuse material; the chalk is close to such a material. The ideal diffuse material reflects an incoming energy into all the possible directions with the same amount of energy. As a result, the reflection becomes view-independent. This diffuse property is caused by a rough surface in a microscopic level, and is perceived as the uniform distribution in the space in the macroscopic level.

The diffuse term is explained by the Lambert's cosine law.

The Lambert's cosine law explains how an incoming energy is reflected depending on the configuration between the surface and the light direction. Suppose that we want to compute the reflected energy I_r on a surface having a normal $\vec{N}$ given the light direction $\vec{L}$. The reflected energy I_r is then computed as the following:

$$\begin{aligned} I_r &= I_i \cos\theta \\ &= I_i(\vec{N}\cdot\vec{L}), \end{aligned} \tag{8.2}$$

where θ is the angle between two vectors of $\vec{L}$ and $\vec{N}$. Fig. 8.7 shows the configuration of these vectors.

Note that as the reflected energy becomes the highest, when the light direction is aligned with the surface normal. Fig. 8.7 also shows how the reflected energy behaves as we have different light directions. When we have a material-dependent, diffuse coefficient, k_d, the reflected energy of the diffuse term is $I_{r,d} = k_d I_i(\vec{N}\cdot\vec{L})$.

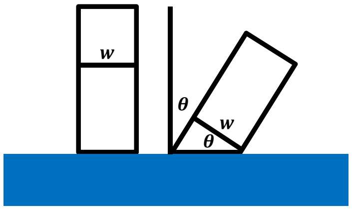

Figure 8.8: The configuration for the Lambert's cosine law.

Proving the cosine law. Let us see how to prove the Lambert's cosine law. Suppose that we have a beam of light with a width of w and energy of I (Fig. 8.8). In this case, the light density per unit area

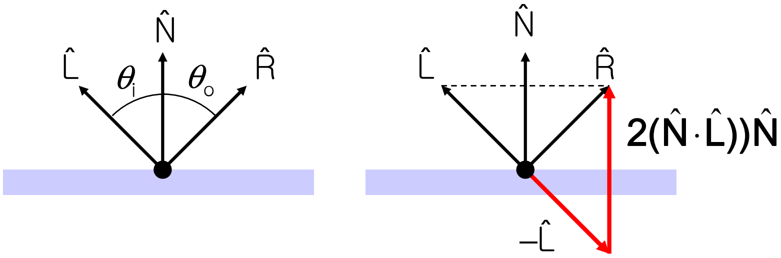

Figure 8.9: This shows how the reflected light direction is computed for the perfect specular object.

is $\frac{I}{w}$. We then lean the beam as an amount of θ. The area receiving the light energy become larger, and takes $\frac{w}{\cos\theta}$. Its light density is then $\frac{I\cos\theta}{w}$. As a result, we can see that the light density reduces as an amount of $\cos\theta$.

Specular term. Let us first consider a perfect mirror-like object. In this case, the reflected light angle is same to the incoming light angle (Fig. 8.9). This is explained by the Snell's law, which is described in a detailed manner in Ch. 10.1. Under the ideal specular material, the reflected light direction, $\vec{R}$ is computed as $2(\vec{N}\cdot\vec{L})\vec{N} - \vec{L}$.

The ideal specular material rarely exists in practice. More common objects are glossy materials, which have highlight along a particular direction and spreads its energy out from the direction. Specifically, when the viewing direction, $\vec{V}$, is on the ideal reflected direction $\vec{R}$, the viewer sees the highest illumination. We then reduce the energy as the viewing direction is away from $\vec{R}$. To capture this observation, the Phong illumination uses the following specular term:

$$\begin{aligned} I_{r,s} &= k_s I_s (\cos\phi)^{n_s} \\ &= k_s I_s (\vec{V}\cdot\vec{R})^{n_s}, \end{aligned} \tag{8.3}$$

where k_s, I_s, n_s are material-dependent specular coefficient, intensity for the specular component of a light, and specular exponent, respectively. Fig. 8.10 shows example results of the specular term.

The Phong model describes materials by treating them to have ambient, diffuse, and specular properties together.

The final Phong illumination is computed by summing these different terms, ambient, diffuse, and specular terms, of different lights (Fig. 8.5). Note that most common objects are described by combining these terms, not a single term.

Local and global illumination. While the Phong illumination is not a physically-based model, it has been widely used for its simplicity and efficiency. Nonetheless, it has a fundamental issue, a local illumination model. The Phong illumination achieves its

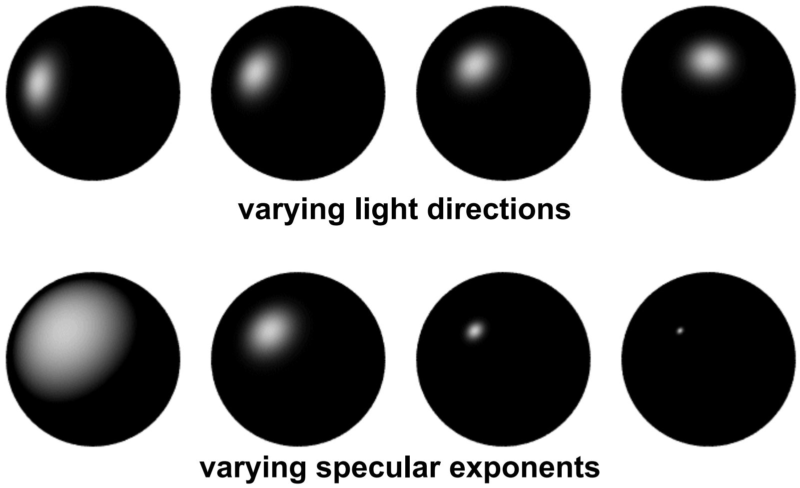

Figure 8.10: This shows how the illumination changes as a function of a viewing direction and specular exponents of the specular term of the Phong illumination model.

efficiency by considering only the local information such as the surface normal, viewing information, and the light information.

The Phong model and rasterization considers the rendering process locally for efficiency.

When there is a blocker occluding the light energy, it, however, generates shadows, which is not considered at all at the Phong illumination. To support such effects, we need to consider the reflected energy or blocked energy from other geometry. This requires us to access global information, which slows down the overall processing. Rasterization is designed in a way to reduce such global access for achieving a high performance and thus the Phong illumination has been well suited for rasterization. We discuss how to generate shadow within rasterization in Ch. 9.4.1. A more physically based approach is to use ray tracing techniques, which are explained in Part II.

8.4 *Shading*

Shading is a process of adjusting a color of a primitive based on various information such as the normal of the primitive and its angle to the light or view direction. Shading can refer to the illumination process and cover broader approaches including various effects (e.g., lens flare), which are implemented by shader programs. While shading is a broad topic, we discuss how to compute colors within the primitive (e.g., triangle) in this section.

Common shading (or interpolation) methods are as the following:

- **Flat shading.** For flat shading, we use only a single color for the primitive. As a result, each triangle in this approach looks to be constant, i.e., flat, in the image space. To perform flat shading, we use a normal of a triangle and perform the Phong illumination model or other illumination models to compute the single color. This is the simplest and fastest approach.

- **Gouraud shading.** This approach provides a smooth rendering result by computing different colors for vertices of a primitive and interpolating them within the primitive. While this approach shows smooth rendering results, it comes with performing three independent illumination for vertices and interpolation.
- **Phong shading.** This approach interpolates normals of vertices and computes colors with them within the primitive by evaluating an illumination model with interpolated normals. Since we interpolate normals of vertices, we can generate highlight within the triangle, even though we do not have such highlight in each vertex. When we use the Gouraud shading in this case, we cannot generate the highlight within the primitive, since we interpolate colors of vertices. Fig. 8.11 shows difference between Gouraud and Phong shading methods.

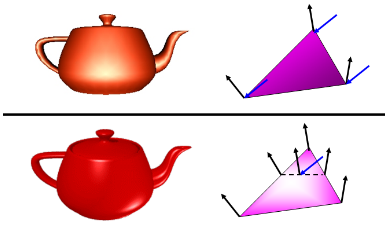

Figure 8.11: This shows Gouraud (top) and Phong (bottom) shading methods. Blue arrows indicate the locations of evaluating an illumination model.

8.5 *Common Questions*

We have learned that we compute an illumination value for each vertex. For smooth objects (like the teapot model shown in the slide), it should be okay. But, when we draw a box, then the box may look smooth, not showing different and discontinuous colors between neighboring faces of the box. If we use a normal for each vertex of the box, we may get such smooth rendering result, which is not correct one. Instead, we use multiple vertices for each point of the box. For example, we use different vertex data for each point in different faces of the box. Note that these vertices should have the same positional value, but they can have different normals, which can generate discontinuous colors between different faces. Refer to the slide of "Decoupling Vertex and Face Attributes via Indirection" in the lecture slide of "Interacting with a 3D world".

Is there any techniques that can show better quality than the Phong Illumination and can be used in interactive games? I want to know techniques that can show near physically-based illumination that can be used in games? Ambient occlusion has been proposed as an approximation for physically-based global illumination. It can be pre-computed and used quite quickly at runtime, leading to be suitable for interactive games. Moreover, in some CG movies, this technique has been used.