2

Fundamentals of Light Transport

"He who loves practice without theory is like the sailor who boards ship without a rudder and compass and never knows where he may cast."

—Leonardo Da Vinci, 1452–1519

THE goal of rendering algorithms is to synthesize images of virtual scenes. Global illumination algorithms solve this problem by mimicking the physical behavior of light as it is emitted from light sources, scattered by elements in the scene, and ultimately detected by a virtual sensor or "camera." In this chapter we first explore the necessary background and develop the terminology needed to understand the physical properties of light. We then explain the expressions that govern the local and global behavior of light as it interacts with surfaces in a vacuum. We conclude the chapter with a short overview of techniques that have been developed in the computer graphics literature to simulate these effects, and summarize our notation in Table 2.2.

2.1 Assumptions About the Nature of Light

Our current understanding of the behavior of light relies on a progression of increasingly complete yet complicated models of light. These are: ray optics, wave optics, electromagnetic optics, and quantum optics (see Figure 2.1) [Saleh and Teich, 2007]. Computer graphics typically relies on the simplest of these models, ray optics (also called geometric optics). This model makes several simplifying assumptions about the behavior of light that limit the types of phenomena that can be simulated. In essence, in this model light can only be emitted, reflected, and transmitted. Additionally, light is assumed to travel in straight lines and at infinite speed. This means that

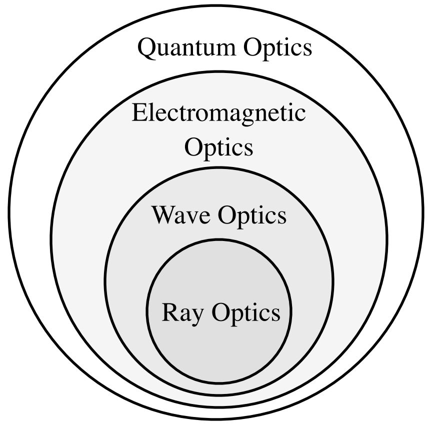

Figure 2.1: The theory of light is described by a series of increasingly complete optical models, where each successive model is able to account for more optical phenomena. In computer graphics and this dissertation, we will restrict ourselves to the simplest model, ray optics.

effects explained by the higher-level models cannot (easily) be incorporated into our simulations. In ray optics, effects such as diffraction and interference (wave optics), polarization and dispersion (electromagnetic optics), and fluorescence and phosphorescence (quantum optics) are completely ignored. In spite of these limitations, we are still able to correctly simulate a wide range of physical phenomena.

2.2 Radiometry

Radiometry is the study of the physical measurement of electromagnetic radiation, including visible light. It defines a common terminology for the physical quantities and units that are used by global illumination algorithms. In this section we review the basic radiometric quantities and illustrate them in Figure 2.2. All of these radiometric quantities in reality depend on the wavelength of light; however, we omit this in our notation.

2.2.1 Radiometric Quantities

Flux. The most fundamental radiometric quantity is radiant power, or flux. Flux, denoted Φ, expresses the amount of energy flowing across a surface over time. This is the same quantity that is used to describe the power of light bulbs and is expressed in terms of watts $[W = J \cdot s^{-1}]$.

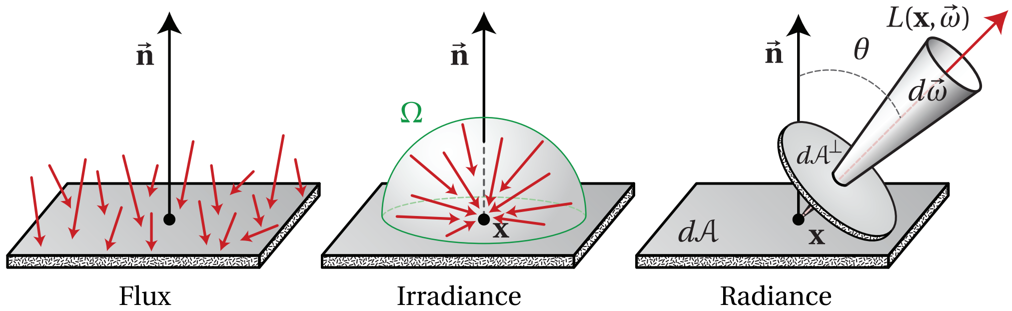

Figure 2.2: Flux measures the amount of light that hits a surface over a finite area from all directions, irradiance integrates the light arriving at a single point over the whole hemisphere, and radiance expresses the amount of light arriving at a single point from a differential solid angle.

Irradiance. A related quantity, irradiance, expresses the amount of incident power hitting a surface per unit surface area. Hence, it has units of $[W \cdot m^{-2}]$ and can be expressed in terms of flux as:

$$E(\mathbf{x}) = \frac{d\Phi(\mathbf{x})}{dA(\mathbf{x})}. \tag{2.1}$$

Irradiance is always measured at some surface position $\mathbf{x}$ with a surface normal $\vec{\mathbf{n}}$. The term irradiance implies a measure of the flux *arriving* at a surface location $\mathbf{x}$. The term *radiant exitance* (M) or *radiosity* (B) is used instead if the flux is leaving a surface.

Radiance. Perhaps the most important quantity in global illumination is radiance. Intuitively, radiance expresses how much light arrives from a differential direction $d\vec{\omega}$ onto a hypothetical differential area *perpendicular* to that direction $dA^{\perp}$. Radiance has units of $[W \cdot sr \cdot m^{-2}]$. This five-dimensional function of position and direction can be expressed as

$$L(\mathbf{x}, \vec{\omega}) = \frac{d^2\Phi(\mathbf{x}, \vec{\omega})}{d\vec{\omega}\, dA^{\perp}(\mathbf{x})}. \tag{2.2}$$

In practice we are typically interested in measuring radiance at an actual surface instead of a hypothetical surface perpendicular to the flow of light. In this case the relationship $dA^{\perp} =$

$(\vec{\mathbf{n}} \cdot \vec{\omega})\ dA$ can be used to obtain:

$$L(\mathbf{x}, \vec{\omega}) = \frac{d^2\Phi(\mathbf{x}, \vec{\omega})}{(\vec{\mathbf{n}} \cdot \vec{\omega})\ d\vec{\omega}\ dA(\mathbf{x})}. \tag{2.3}$$

The foreshortening cosine term $(\vec{\mathbf{n}} \cdot \vec{\omega})$, which returns the cosine of the angle between $\vec{\omega}$ and the surface normal $\vec{\mathbf{n}}$, takes into account the spreading of light at glancing angles.

It is also possible to interpret radiance as a density over projected solid angle $d\vec{\omega}^{\perp} = (\vec{\mathbf{n}} \cdot \vec{\omega})\ d\vec{\omega}$, instead of projected surface area $dA^{\perp}$:

$$L(\mathbf{x}, \vec{\omega}) = \frac{d^2\Phi(\mathbf{x}, \vec{\omega})}{d\vec{\omega}^{\perp}\ dA(\mathbf{x})}. \tag{2.4}$$

This can often be a more convenient way of describing radiance since it uses the natural differential surface area dA of the surface at $\mathbf{x}$.

Radiance is most directly related to the perceived brightness of colors to the human eye and is the quantity that needs to be computed for each pixel in a rendered image.

We summarize the radiometric quantities, their symbols, and units in Table 2.1.

2.2.2 Radiometric Relationships

Radiance is so instrumental in rendering that it is useful to express the other radiometric quantities in terms of radiance. Equation 2.2 expresses radiance in terms of flux, but it is also possible to invert this relationship by integrating both sides over the hemisphere of directions Ω

Table 2.1: Definitions of the fundamental radiometric quantities with their associated symbols and units.

Symbol	**Units**	**Description**
Φ	W	Radiance flux, or power
E	$W \cdot m^{-2}$	Irradiance
M	$W \cdot m^{-2}$	Radiant exitance (outgoing)
B	$W \cdot m^{-2}$	Radiosity (outgoing)
L	$W \cdot m^{-2} \cdot sr^{-1}$	Radiance

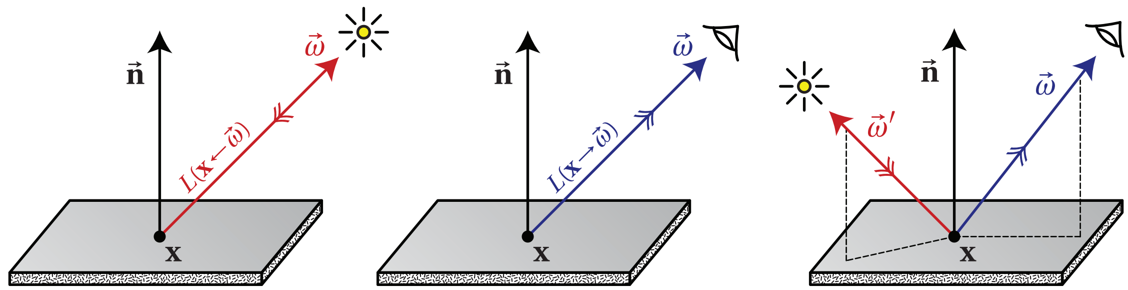

Figure 2.3: We distinguish between two types of radiance at a point $\mathbf{x}$. $L(\mathbf{x}\leftarrow\vec{\omega})$ represents incident radiance at $\mathbf{x}$ from direction $\vec{\omega}$ (left), whereas $L(\mathbf{x}\rightarrow\vec{\omega})$ represents exitant radiance at $\mathbf{x}$ in direction $\vec{\omega}$ (middle). The bidirectional reflectance distribution function (right) describes the relationship between these two quantities and is a function of both an incoming and an outgoing direction vector. In our notation, direction vectors always point out of $\mathbf{x}$ regardless of the actual direction of light flow.

and area $\mathcal{A}$ to arrive at:

$$\Phi = \int_{\mathcal{A}}\int_{\Omega} L(\mathbf{x}\rightarrow\vec{\omega})\,(\vec{\mathbf{n}}\cdot\vec{\omega})\;d\vec{\omega}\;d\mathcal{A}(\mathbf{x}). \tag{2.5}$$

Combining this with Equation 2.1 allows us to also express irradiance and radiosity in terms of radiance:

$$E(\mathbf{x}) = \int_{\Omega} L(\mathbf{x}\leftarrow\vec{\omega})\,(\vec{\mathbf{n}}\cdot\vec{\omega})\;d\vec{\omega}, \tag{2.6}$$

$$M(\mathbf{x}) = B(\mathbf{x}) = \int_{\Omega} L(\mathbf{x}\rightarrow\vec{\omega})\,(\vec{\mathbf{n}}\cdot\vec{\omega})\;d\vec{\omega}. \tag{2.7}$$

2.2.3 Incident and Exitant Radiance Functions

In this dissertation we use the convention that $L(\mathbf{x}\rightarrow\vec{\omega})$ denotes *exitant* radiance *leaving* $\mathbf{x}$ in direction $\vec{\omega}$, and $L(\mathbf{x}\leftarrow\vec{\omega})$ represents *incident* radiance *arriving* at $\mathbf{x}$ from direction $\vec{\omega}$. In both situations the direction vector $\vec{\omega}$ points *out* of the surface. This is illustrated in Figure 2.3. In addition to representing radiance flowing in different directions, this distinction is more fundamental since these two quantities measure different sets of photon events. Incident radiance measures photons just before they arrive at a surface, whereas exitant radiance measures the

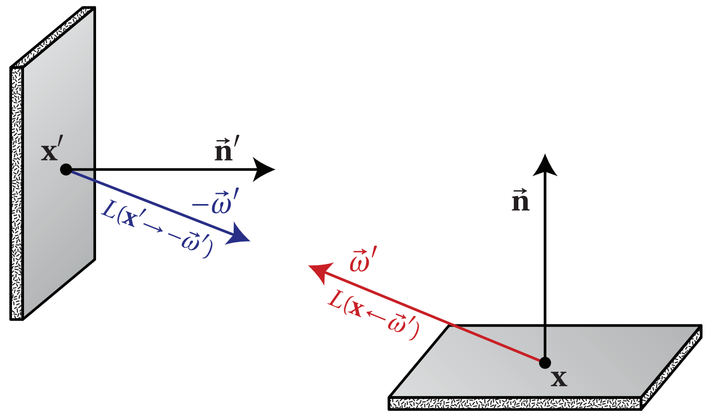

Figure 2.4: In a vacuum, the incident radiance at $\mathbf{x}$ from direction $\vec{\omega}$ is equal to the exitant radiance from the nearest visible surface in that direction.

photons departing. Hence, in general

$$L(\mathbf{x}\!\rightarrow\!\vec{\omega}) \neq L(\mathbf{x}\!\leftarrow\!\vec{\omega}). \tag{2.8}$$

Within a vacuum, photons propagate unobstructed. This means that all the radiance incident at a point from direction $\vec{\omega}$ will continue on as exitant radiance in direction $-\vec{\omega}$

$$L(\mathbf{x}\!\leftarrow\!\vec{\omega}) = L(\mathbf{x}\!\rightarrow\!-\vec{\omega}). \tag{2.9}$$

Hence, radiance remains constant until it interacts with a surface. This observation allows us to form a simple relation between the exitant and incident radiance functions between two distinct points. To accomplish this we introduce the *ray casting function* $\mathbf{r}(\mathbf{x},\vec{\omega}) = \mathbf{x}'$, which returns $\mathbf{x}'$, the point on the closest surface from $\mathbf{x}$ in the direction $\vec{\omega}$. In the case that the space between surfaces is a vacuum, radiance remains constant along straight lines. This property means that the incident radiance at a point $\mathbf{x}$ from direction $\vec{\omega}$ is equal to the outgoing radiance from the closest visible point in that direction. This is illustrated in Figure 2.4 and can be expressed as:

$$L(\mathbf{x}\!\leftarrow\!\vec{\omega}) = L(\mathbf{x}'\!\rightarrow\!-\vec{\omega}). \tag{2.10}$$

It is important to remember that Equations 2.9 and 2.10 are only valid within a vacuum.

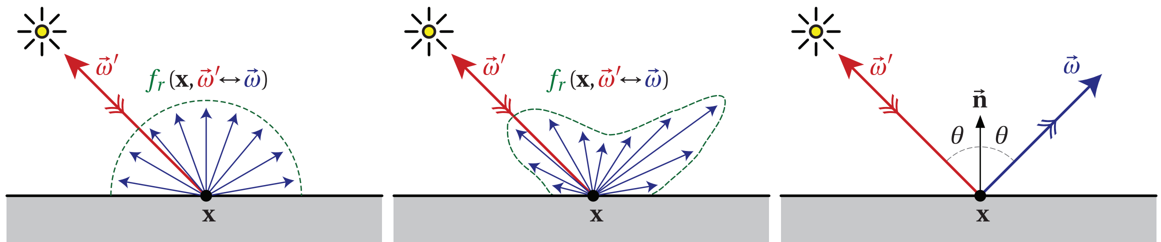

Figure 2.5: The ideal diffuse, or Lambertian, BRDF (left) reflects light equally in all directions. On the other extreme, perfectly specular materials (right) reflect all light in the mirror direction. General BRDFs (center) define a 2D reflectance distribution function for each incoming direction $\vec{\omega}'$.

We will extend this model in Chapter 4 to include effects from participating media.

2.3 Interaction of Light with Surfaces

In the remainder of this chapter we will explore the interaction of light with surfaces.

2.3.1 The BRDF

When light encounters objects in the scene, it interacts with the surfaces by being reflected, refracted, or absorbed. If we make the simplifying assumption that light striking a surface location will reflect at the same location, then the interaction between the light and the surface can be described using a six-dimensional function called the *bidirectional reflectance distribution function*, or BRDF. At a high level, the BRDF describes how "bright" a surface will look from a particular direction $\vec{\omega}$ when being illuminated by a light from another direction $\vec{\omega}'$. More precisely, the BRDF expresses the relationship between irradiance and reflected radiance at a point $\mathbf{x}$:

$$f_r(\mathbf{x}, \vec{\omega}' \to \vec{\omega}) \equiv \frac{dL(\mathbf{x} \to \vec{\omega})}{dE(\mathbf{x} \leftarrow \vec{\omega}')} = \frac{dL(\mathbf{x} \to \vec{\omega})}{L(\mathbf{x} \leftarrow \vec{\omega}')\,(\vec{\mathbf{n}} \cdot \vec{\omega}')\, d\vec{\omega}'}. \tag{2.11}$$

The last step results from substituting Equation 2.6 into the denominator. Three example BRDFs are illustrated in Figure 2.5.

Properties of the BRDF

1. **Domain.** For a particular surface point $\mathbf{x}$, the BRDF is a four-dimensional function: two dimensions to specify the incoming direction, and two dimensions to specify the outgoing

direction. Furthermore, if the BRDF is allowed to vary spatially over an object's surface, this leads to an additional two dimensions.

2. **Range.** The BRDF can take on any positive value.

3. **Reciprocity.** The value of the BRDF remains unchanged if the incident and outgoing directions are swapped. Mathematically, Helmholtz's law of reciprocity states that:

$$f_r(\mathbf{x}, \vec{\omega}', \vec{\omega}) = f_r(\mathbf{x}, \vec{\omega}, \vec{\omega}'). \tag{2.12}$$

Because of this property, we use the following notation for the BRDF to indicate that the directions can be interchanged:

$$f_r(\mathbf{x}, \vec{\omega}' \leftrightarrow \vec{\omega}). \tag{2.13}$$

From a practical point of view, reciprocity means that surface reflection is invariant to the direction of light flow, i.e, the reflected radiance remains unchanged if the light and camera positions are swapped. This property is essential for many global illumination algorithms and allows light to be traced either in the forward or backward direction.

4. **Relationship between incident and reflected radiance.** The information in the BRDF can be used to derive the relationship between incident and reflected radiance. By multiplying both sides of Equation 2.11 by the denominator and integrating over all directions we can derive an expression for computing the reflected radiance at a point $\mathbf{x}$:

$$L(\mathbf{x} \rightarrow \vec{\omega}) = \int_{\Omega} f_r(\mathbf{x}, \vec{\omega}' \leftrightarrow \vec{\omega})\, L(\mathbf{x} \leftarrow \vec{\omega}')\, (\vec{\mathbf{n}} \cdot \vec{\omega}')\, d\vec{\omega}'. \tag{2.14}$$

In essence, the reflected radiance off of a surface can be computed by integrating all the incident radiance arriving over the hemisphere of directions. This expression describes the local behavior of light as it interacts with surfaces and is know as the *local illumination* model.

5. **Energy conservation.** Due to energy conservation, a surface cannot reflect more light than

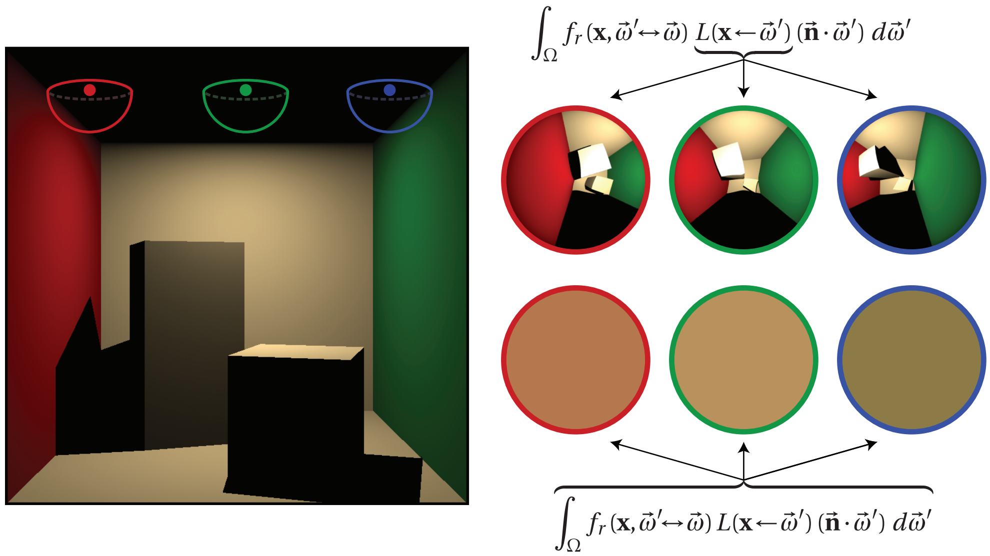

Figure 2.6: The exitant radiance at a point on a surface depends on the incident radiance over the hemisphere above the point. The incident radiance field can be visualized as a hypothetical “fisheye” view from the point (top) and the exitant radiance is the integral of this value over the whole hemisphere.

it receives. By examining Equation 2.14, this can be expressed as the following constraint:

$$\int_\Omega f_r(\mathbf{x},\vec{\omega}'\leftrightarrow\vec{\omega})\,(\vec{\mathbf{n}}\cdot\vec{\omega}')\,d\vec{\omega}' \le 1, \quad \forall\vec{\omega}. \tag{2.15}$$

2.3.2 The Rendering Equation

At a high level, the illumination at each surface point is based on the “brightness” of what that point “sees” within its hemisphere of directions. This could be light sources or other surfaces. The brightness of these other surfaces is in turn evaluated in the same way, introducing a recursive definition for calculating illumination. This intuitive explanation of lighting is illustrated in Figure 2.6. In order to compute the lighting at a surface we must solve for this recursive light energy distribution.

Hemispherical Formulation

The equation which mathematically describes this equilibrium is called the rendering equation [Kajiya, 1986]. The rendering equation expresses the outgoing radiance, $L(\mathbf{x}\rightarrow\vec{\omega})$, of any point in a scene as the sum of the emitted radiance of the underlying surface, L_e, and the reflected radiance, L_r.

$$\underbrace{L(\mathbf{x}\!\rightarrow\!\vec{\omega})}_{\text{outgoing}} = \underbrace{L_e(\mathbf{x}\!\rightarrow\!\vec{\omega})}_{\text{emitted}} + \underbrace{L_r(\mathbf{x}\!\rightarrow\!\vec{\omega})}_{\text{reflected}}. \tag{2.16}$$

The local illumination model in Equation 2.14 expresses the reflected radiance on a surface. Substituting this into Equation 2.16 results in the hemispherical form of the rendering equation:

$$\underbrace{L(\mathbf{x}\!\rightarrow\!\vec{\omega})}_{\text{outgoing}} = \underbrace{L_e(\mathbf{x}\!\rightarrow\!\vec{\omega})}_{\text{emitted}} + \underbrace{\int_\Omega f_r(\mathbf{x},\vec{\omega}'\!\leftrightarrow\!\vec{\omega})\, L(\mathbf{x}\!\leftarrow\!\vec{\omega}')\,(\vec{\mathbf{n}}\cdot\vec{\omega}')\; d\vec{\omega}'}_{\text{reflected}}. \tag{2.17}$$

The combination of Equation 2.17 with Equation 2.10 (which relates the exitant radiance on the left hand side to the incident radiance on the right hand side) forms the recursive nature of global illumination. This arises from the fact that outgoing radiance at one point is dependent on the outgoing radiance at all other points in the scene. Hence, the radiance L appears on both sides of the equation.

Area Formulation

Sometimes it is more convenient to express the rendering equation as an integration over points on the surfaces of the scene geometry. This can be done by performing a change of variable using the relation,[1]

$$d\vec{\omega}'(\mathbf{x}) = \frac{(\vec{\mathbf{n}}'\cdot -\vec{\omega}')}{\|\mathbf{x}'-\mathbf{x}\|^2}\, d\mathcal{A}(\mathbf{x}'), \tag{2.18}$$

where $\mathbf{x}'$ is a point on a surface with surface normal $\vec{\mathbf{n}}'$.

In the hemispherical formulation visibility is implicitly accounted for using the ray casting function. In order to change the integration from the hemisphere of directions to surface area we must explicitly take into account the visibility between surface points. To accomplish this we

[1] This expression is obtained by computing the Jacobian determinant of the transformation matrix mapping directions on the hemisphere to points on the scene geometry.

introduce a visibility function V:

$$\forall \mathbf{x}, \mathbf{x}' \in \mathcal{A} : V(\mathbf{x} \leftrightarrow \mathbf{x}') = \begin{cases} 1 & \text{if } \mathbf{x} \text{ and } \mathbf{x}' \text{ are mutually visible,} \\ 0 & \text{otherwise.} \end{cases} \tag{2.19}$$

Table 2.2: Notation used in this dissertation.

Symbol	Units	Description	Page
$\mathbf{x}$		Position	
$\vec{\mathbf{n}}$		Surface normal at $\mathbf{x}$ (always normalized: $\|\vec{\mathbf{n}}\| = 1$)	
$d\mathcal{A}(\mathbf{x})$		Differential surface area at $\mathbf{x}$	
$d\mathcal{A}^{\perp}(\mathbf{x}, \vec{\omega})$		Differential surface area perpendicular to $\vec{\omega}$ at $\mathbf{x}$	10
$\vec{\omega}$		Normalized direction (away from surface)	
$d\vec{\omega}$		Differential solid angle ($d\vec{\omega} = \sin\theta \, d\theta \, d\phi$)	
$d\vec{\omega}^{\perp}$		Differential projected solid angle	11
Ω	sr	Hemisphere of directions	11
$\Omega_{4\pi}$	sr	Sphere of directions	61
$\mathcal{A}$	m^2	Surface area	131
$\mathcal{V}$	m^3	Medium volume	131
$L(\mathbf{x}, \vec{\omega})$	$W \cdot m^{-2} \cdot sr^{-1}$	Radiance at $\mathbf{x}$ in direction $\vec{\omega}$	10
$L(\mathbf{x} \leftarrow \vec{\omega})$	$W \cdot m^{-2} \cdot sr^{-1}$	Incident radiance at $\mathbf{x}$ from $\vec{\omega}$	12
$L(\mathbf{x} \rightarrow \vec{\omega})$	$W \cdot m^{-2} \cdot sr^{-1}$	Outgoing radiance leaving $\mathbf{x}$ in direction $\vec{\omega}$	12
$L(\mathbf{x} \leftarrow \mathbf{x}')$	$W \cdot m^{-2} \cdot sr^{-1}$	Incident radiance at $\mathbf{x}$ from $\mathbf{x}'$	
$L(\mathbf{x} \rightarrow \mathbf{x}')$	$W \cdot m^{-2} \cdot sr^{-1}$	Outgoing radiance leaving $\mathbf{x}$ towards $\mathbf{x}'$	
L_e	$W \cdot m^{-2} \cdot sr^{-1}$	Emitted radiance	17
L_r	$W \cdot m^{-2} \cdot sr^{-1}$	Reflected radiance	15
L_i	$W \cdot m^{-2} \cdot sr^{-1}$	In-scattered radiance	61
$f_r(\mathbf{x}, \vec{\omega} \leftrightarrow \vec{\omega}')$	sr^{-1}	BRDF	14
$p(\mathbf{x}, \vec{\omega} \leftrightarrow \vec{\omega}')$	sr^{-1}	Phase function (normalized)	61
$\sigma_s(\mathbf{x})$	m^{-1}	Scattering coefficient at $\mathbf{x}$	57
$\sigma_a(\mathbf{x})$	m^{-1}	Absorption coefficient at $\mathbf{x}$	57
$\sigma_t(\mathbf{x})$	m^{-1}	Extinction coefficient at $\mathbf{x}$	57
$\tau(\mathbf{x} \leftrightarrow \mathbf{x}')$	unitless	Optical thickness: $\int_{\mathbf{x}'}^{\mathbf{x}} \sigma_t(x) \, dx$	59
$T_r(\mathbf{x} \leftrightarrow \mathbf{x}')$	unitless	Transmittance: $e^{-\tau(\mathbf{x} \leftrightarrow \mathbf{x}')}$	58
$V(\mathbf{x} \leftrightarrow \mathbf{x}')$	unitless	Visibility function	18, 77
$G(\mathbf{x} \leftrightarrow \mathbf{x}')$	unitless	Geometric coupling term (surfaces)	19
$H(\mathbf{x} \rightarrow \mathbf{x}')$	unitless	Geometric coupling term (media)	77
$\hat{G}(\mathbf{x} \leftrightarrow \mathbf{x}')$	unitless	Generalized geometric coupling term (surfaces and media)	132
$W_e(\mathbf{x} \rightarrow \vec{\omega})$	$m^{-3} \cdot sr^{-1}$	Importance at $\mathbf{x}$ towards $\vec{\omega}$	133
$W_e(\mathbf{x} \rightarrow \mathbf{x}')$	$m^{-3} \cdot sr^{-1}$	Importance at $\mathbf{x}$ towards $\mathbf{x}'$	133
α_i	unitless	Weight for photon i	134
$\delta(x)$	unitless	Dirac delta distribution	136
δ_{ij}	unitless	Kronecker delta function	172
ξ	unitless	Uniformly distributed random number between 0 and 1	82

We can now transform Equation 2.17 into an integration over surface area:

$$L(\mathbf{x}\to\vec{\omega}) = L_e(\mathbf{x}\to\vec{\omega}) + \int_{\mathcal{A}} f_r(\mathbf{x},\mathbf{x}'\to\mathbf{x},\vec{\omega})\, L(\mathbf{x}\leftarrow\mathbf{x}')\, V(\mathbf{x}\leftrightarrow\mathbf{x}')\, \frac{(\vec{\mathbf{n}}\cdot\vec{\omega}')\,(\vec{\mathbf{n}}'\cdot-\vec{\omega}')}{\|\mathbf{x}'-\mathbf{x}\|^2}\, d\mathcal{A}(\mathbf{x}'). \tag{2.20}$$

Typically, the last terms involving the surface positions and normals are folded into a symmetric geometric coupling term

$$G(\mathbf{x}\leftrightarrow\mathbf{x}') = \frac{(\vec{\mathbf{n}}\cdot\vec{\omega}')\,(\vec{\mathbf{n}}'\cdot-\vec{\omega}')}{\|\mathbf{x}'-\mathbf{x}\|^2}, \tag{2.21}$$

which results in

$$L(\mathbf{x}\to\vec{\omega}) = L_e(\mathbf{x}\to\vec{\omega}) + \int_{\mathcal{A}} f_r(\mathbf{x},\mathbf{x}'\to\mathbf{x},\vec{\omega})\, L(\mathbf{x}\leftarrow\mathbf{x}')\, V(\mathbf{x}\leftrightarrow\mathbf{x}')\, G(\mathbf{x}\leftrightarrow\mathbf{x}')\, d\mathcal{A}. \tag{2.22}$$

2.4 Methods for Solving the Rendering Equation

The rendering equation is very costly to compute and is far too complex to solve analytically in the general case. Because of this, it is typically approximated using numerical integration. Several algorithms have been proposed in the literature to fully or partially solve the rendering equation. These global illumination algorithms can generally be classified into two categories: finite element methods and Monte Carlo ray tracing methods.

2.4.1 Finite Element Methods

Finite element methods in computer graphics were originally adapted from the radiative heat transfer literature [Siegel and Howell, 2002]. In the compute graphics field, finite element techniques have collectively been called *radiosity methods*. Radiosity computes indirect illumination by discretizing the scene geometry into a collection of small patches. These patches form the basis over which the global illumination solution is computed. The lighting computation is solved by expressing the rendering equation as a linear system of equations describing the light exchange between patches in the scene.

Initial work in radiosity placed heavy restrictions on the complexity of the light sources, geometry and materials of the scene. These early approaches only worked for scenes with large area light sources and Lambertian, or perfectly diffuse, surfaces where the BRDF is a simple constant [Goral et al., 1984; Cohen and Greenberg, 1985; Nishita and Nakamae, 1985]. With these restrictions, the radiosity algorithm is simple to implement and the lighting solutions computed are *view-independent*. This allows for a single solution to be reused for rendering images from arbitrary viewpoints. Due to this, radiosity techniques were well suited for walk-through animations of simple diffuse scenes. Unfortunately, precomputation for walk-through animations requires significant computation time as well as storage space for the final solution. Many researchers tried to address some of these drawbacks by adding view-dependent computation [Smits et al., 1992], support for more complex reflections [Wallace et al., 1987; Immel et al., 1986; Sillion et al., 1991; Christensen et al., 1996; Gortler et al., 1993], as well as clustered and hierarchical computation [Stamminger et al., 1998; Hanrahan et al., 1991; Smits et al., 1994]. Support for high-frequency lighting can be obtained through discontinuity meshing where the tessellation of the solution is carefully subdivided to better resolve sharp discontinuities such as hard shadow boundaries. These additions unfortunately make the radiosity algorithm much more complex. Furthermore, radiosity cannot efficiently handle scenes with complex geometry since the cost of the lighting simulation is directly tied to the geometric complexity of the scene.

2.4.2 Monte Carlo Ray Tracing Methods

Monte Carlo methods rely on the use of random sampling in order to compute their results. The first real use of Monte Carlo methods stems from work on the atomic bomb during the second world war in the 1940s and has been used extensively in the neutron transport field ever since [Metropolis and Ulam, 1949; Metropolis, 1987]. Monte Carlo methods developed independently in computer graphics. They were introduced to computer graphics by Appel [1968], who first suggested the use of ray casting to generate images by intersecting the scene geometry with a ray for each pixel. In 1980, Whitted suggested the recursive evaluation of illumination by tracing specular reflection and refraction rays, and additionally, shadow rays for direct Lambertian illumination. Whitted also suggested the use of random perturbation of the surface normal in order to simulate rough or glossy specular reflections.

Ray tracing methods can also naturally account for complex surfaces and light interactions. By further combining ray tracing with Monte Carlo techniques, researchers were able to simulate a vast array of effects including depth of field, motion blur, glossy reflections, and global illumination [Cook, 1986; Cook et al., 1984; Kajiya, 1986]. Monte Carlo ray tracing methods such as path tracing or bidirectional path tracing [Kajiya, 1986; Lafortune and Willems, 1993; Veach and Guibas, 1994] can simulate virtually all effects described by the rendering equation. The downside of Monte Carlo techniques is variance, which manifests itself as high-frequency noise in the rendered image. Though several researchers have explored ways to minimize this noise [Shirley, 1990, 1991; Lafortune, 1996; Veach and Guibas, 1995], these unbiased methods still typically involve tracing hundreds of rays at each pixel in order to converge to solutions of reasonable quality.

One of the main advantages of radiosity over unbiased ray tracing is that for a specified simulation quality, computation time is independent of the actual resolution of the rendered image. Radiosity can effectively reuse computation over large regions of the image. In contrast, unbiased ray tracing techniques require independent evaluation at each pixel, which precludes resolution independence. Biased Monte Carlo ray tracing relaxes this restriction and has produced some of the most efficient methods to date. Irradiance caching [Ward et al., 1988] and photon mapping [Jensen, 2001] methods are two such approaches. We review these two techniques in more detail in Chapter 3 and Chapter 7, respectively. We also provide a more detailed introduction to Monte Carlo integration in Appendix A.

2.4.3 Hybrid Methods

Several methods have also been proposed which combine features of Monte Carlo and finite element techniques in order to get the best of both worlds. These typically take a *multi-pass* approach by augmenting finite element methods with a Monte Carlo pass to account for more general reflection models [Wallace et al., 1987; Sillion and Puech, 1989; Chen et al., 1991]. Another motivation for combining the techniques is to reduce discretization artifacts in finite element methods. A common optimization technique is to precompute a low-quality solution using radiosity techniques, and then to simulate one additional bounce of illumination using a higher quality *final gather* pass [Reichert, 1992]. This final gather pass typically takes the form of a Monte Carlo evaluation and can effectively remove many of the artifacts present in the low-quality solution. In addition to being a full global illumination algorithm on its own, the irradiance caching method described in the next chapter can also be effectively used as a final gather optimization technique.