

# Contents

---

<b>1</b>	<b>Introduction (draft)</b>	<b>1</b>
<b>2</b>	<b>Related Work</b>	<b>5</b>
2.1	Analytical techniques . . . . .	5
2.1.1	Models . . . . .	6
2.1.2	Implementations . . . . .	6
2.2	Numerical techniques . . . . .	8
<b>3</b>	<b>Theory</b>	<b>9</b>
3.1	Light and Radiometry . . . . .	9
3.2	Radiometric quantities . . . . .	11
3.2.1	Radiant flux . . . . .	11
3.2.2	Radiant energy . . . . .	11
3.2.3	Irradiance . . . . .	11
3.2.4	Intensity . . . . .	12
3.2.5	Radiance . . . . .	12
3.2.6	Radiometric quantities for simple lights . . . . .	14
3.3	Reflectance Functions . . . . .	15
3.3.1	BRDF functions . . . . .	15
3.3.2	Examples of BRDF functions . . . . .	16
3.3.3	The rendering equation . . . . .	18
3.3.4	Fresnel equations . . . . .	19
3.4	Light transport and subsurface scattering . . . . .	19
3.4.1	Emission . . . . .	20
3.4.2	Absorption . . . . .	21
3.4.3	Out-scattering . . . . .	21
3.4.4	In-scattering . . . . .	21
3.4.5	Final formulation of the volume rendering equation . . . .	23

3.4.6	BSSRDF functions and generalized rendering equation . .	23
3.4.7	Standard dipole model . . . . .	23
3.4.8	Directional dipole model . . . . .	23
4	Results	25
	Bibliography	27

## CHAPTER 1

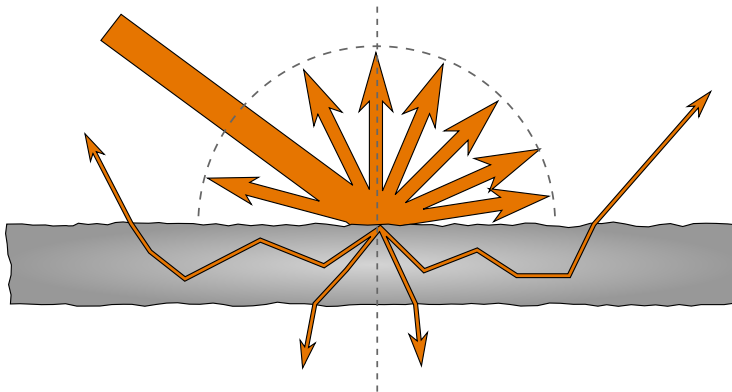
# Introduction (draft)

---

*Subsurface scattering* (SS) is a physical phenomenon that naturally occurs in a wide range of natural materials. Some of the materials that exhibit a strong SS effect in everyday life are milk, human skin and marble. Subsurface scattering is that phenomenon that occurs when light is partially absorbed by an object, bounces inside ("scatters") and finally exits the surface on another point of the material (see Figure 1.1). The phenomenon that results is generally known as *translucency*. We can see some examples of translucency in Figure 1.2

Since the beginning of computer graphics, various attempts have been performed in order to physically model subsurface scattering. Some of these models involve Monte Carlo simulations of the light entering the medium [Pharr and Hanrahan, 2000], other focus on approximating the diffusion of light within the material using an analytical approach [Jensen et al., 2001].

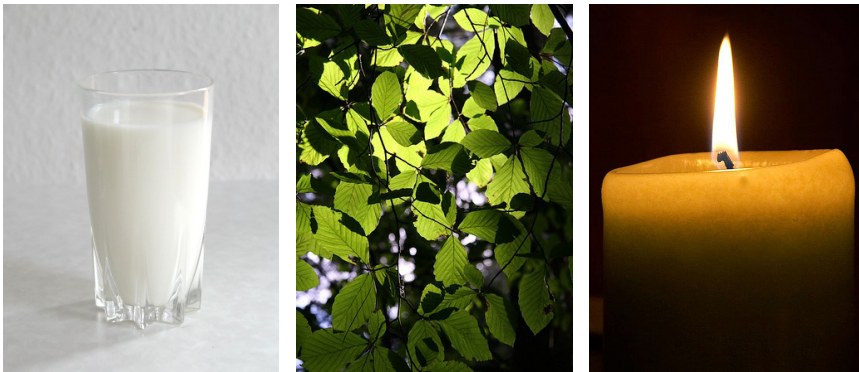
The first model that proposed an analytical approach was the one by Jensen et al. [2001], as an approximation of the radiative transfer equation. This approximation has then been exploited by different authors, in order to account for multi-layered materials [Donner and Jensen, 2005], heterogeneous materials [Wang et al., 2010] and thin surfaces [Wang et al., 2010]. A recent analytical approximation, proposed by Frisvad et al. [2013], extends the approximation in order to account for the directionality of the incoming light.



**Figure 1.1:** Diagram of subsurface scattering. Most of the incoming light gets reflected, but some of it enters the material and leaves it at a different point.

In recent years, with the advent of programmable graphics cards (GPU), it has become possible to exploit these algorithms and bring them to interactive frame rates, and in some cases even to real time rendering. Jensen and Buhler [2002] were the first to propose an efficient implementation (though not real time and on CPU) for rendering subsurface scattering using an octree. More recently, several methods have been proposed, including image-based splats, sum-of-Gaussians filtering, and grid-propagation based methods.

In this thesis we want to employ some cutting edge GPU techniques, with the aid of the programmable pipeline, to implement Frisvad et al. directional model in a real-time fashion. This method should achieve real time results (i.e. in the range of 30 to 60 frames per second) for a wide range of natural materials.



**Figure 1.2:** Some examples of translucent materials: milk, leaves and a candle. Images courtesy of Wikimedia Commons.



## CHAPTER 2

# Related Work

---

In rendering of subsurface scattering, all approaches rely on approximating correctly the *Radiative Transport Equation* (RTE). We identified two main approaches to the problem in literature:

**Analytical** One class of solutions consists of approximating the RTE or one of its approximations via an analytical model. These model can have different level of complexity and computation times, and are often adaptable to a wide range of materials. However, often they rely on assumptions on the scattering parameters that limit their applicability.

**Numerical** In this other class of solutions, a numerical solution for the RTE is actually computed. While providing an exact solution, the computation times are longer. When interactivity is needed, generally some heavy pre computation must be used.

## 2.1 Analytical techniques

In the analytical techniques, two different areas of research must be distinguished. The first area is the research on the actual models, while the second is

research on how the actual models can be implemented efficiently. Each model is usually represented by a specific function called BSSRDF (*Bidirectional Sub-surface Scattering Reflectance Distribution Function*), that describes how light propagates between two points on the surface. This function in the general case must be calculated between all the couple of points on the surface and then integrated over each point. Implementation techniques focus on efficiently implementing this integration step, often making assumptions for which points the computation can be avoided.

### 2.1.1 Models

Regarding the models, the first and most important is the dipole developed by Jensen et al. [2001]. The model relies on an approximation of the RTE called the *diffusion approximation*, that relies on the assumption on highly scattering materials. In this case, a BSSRDF for a planar surface in a semi-infinite medium can be obtained. The BSSRDF needs only the distance between two points to be calculated, and with some precautions can be also extended to arbitrary geometry. This model does not include any single scattering term, that needs to be evaluated separately. The model was then further extended in order to account for multi-layered materials [Donner and Jensen, 2005].

A significant improvement on the model was later given by D'Eon [2012], that improved the model to better fit path traced simulations without any extra computation cost. A more advanced model based on quantization was proposed by D'Eon and Irving [2011], that introduced a new physical foundation in order to improve the accuracy of the original diffusion approximation. Finally, some higher order approximation exist [Frisvad et al., 2013], in order to account for the directionality of the incoming light and single scattering. This allows a more faithful representation of the model at the price of extended computation times.

Finally, for real-time critical applications (such as games), translucency is often estimated as a function of the thickness of the material, that is used to modify a lambertian term [Tomaszewska and Stefanowski, 2012]. While not physically accurate, this technique allows to have a fast translucency effect that can be easily added to existing deferred pipelines.

### 2.1.2 Implementations

Most research on efficient implementations of a subsurface scattering analytical model has been made on the original model by Jensen et al. [2001]. The first



efficient implementation was proposed by Jensen and Buhler [2002], based on a two-pass hierarchical integration approach. Samples on the model are organized in an octree data structure, that then is used to render the object. In the first step, the radiance from the light is stored in the points. In the second pass, using the octree, the contribution from neighboring points is computed, clustering far points in order to speed up calculations. In the original paper, the single scattering term is approximated with as a simple BRDF approximation.

Lensch et al. [2002] approached the problem by subdividing the subsurface scattering contribution into two: a direct illumination part and a global illumination part (i.e. the light shining through the object). The global illumination part is pre-computed as vertex-to-vertex throughput and then summed to the direct illumination term in real-time. Translucent shadow maps [Dachsbacher and Stamminger, 2003] use an approach similar to standard shadow maps: they render the scene from the light point of view, and then calculate the dipole contribution in one point only from a selected set of points, according to a specified sampling pattern. As in Lensch et al. [2002], the contribution is split into global and local to permit faster computations. Mertens et al. [2003b] propose a fast technique based on radiosity hierarchical integration techniques, that unlike the previous implementation can handle deformable geometry.

Another important category of methods is screen space methods. Mertens et al. [2003a] propose an image space GPU technique that pre-computes a set of sample points for the area integration and then performs the integral over multiple GPU passes. d'Eon et al. [2007] proposes a method in image-space, interpreting subsurface scattering as a sum of images to which a gaussian filter has been applied. The gaussians are then summed with weights that make them fit the diffusion approximation. Jimenez et al. [2009] improves further the technique, giving more precise results in case of skin. Shah et al. [2009] present a fast technique that render the object as a series of splats, using GPU blending to sum over the various contributions.

Regarding more advanced models, the better and the quantized dipole can be applied to any of the previous implementations, since they do not require additional information that the standard dipole. On the other hand, the directional dipole requires the direction of the incoming light as part of its calculations, so it is generally not applicable to the mentioned implementations.

## 2.2 Numerical techniques

Numerical techniques for subsurface scattering are often not specific, but come for free or as an extension of a global illumination numerical approximation, since the governing equations are essentially the same. Given their generality, they are usually slower than their analytical counterpart, and often rely on heavy pre-computation steps in order to achieve interactive framerates. Jensen's Photon Mapping [Jensen and Christensen, 1998] was originally developed to render anisotropic subsurface scattering. Classical approaches as a full Monte-Carlo simulation implementation of the light-material interaction [Dorsey et al., 1999], and finite-difference methods exist in literature [Stam, 1995].

Some less general methods have been introduced in order to devise more efficient approximations when it comes to the specific problem of subsurface scattering. Stam [1995] uses the diffusion approximation with the finite difference method on the object discretized on a 3D grid. Fattal [2009] uses as well a 3D grid, that is swept with a structure called light propagation map, that stores the intermediate results until the simulation is complete.

Wang et al. [2010], instead of performing the simulation on a discretized 3D grid, makes the propagation directly in the mesh, converting it into a connected grid of tetrahedrons called *QuadGraph*. This grid can be optimized to be GPU cache friendly, and provide a real-time rendering of deformable heterogeneous objects. The problem in this method is that the *QuadGraph* is slow to compute (20 minutes for very complex meshes) and has heavy memory requirements for the GPU.

Precomputed radiance transfer methods is another class of general global illumination methods, that generally pre-compute part of the lighting and store it in tables [Donner et al., 2009], allowing to retrieve it efficiently with an additional memory cost.

A recent method called SSLPV - Subsurface Scattering Light Propagation Volumes [Børllum et al., 2011] extends a technique originally developed by Kaplanyan and Dachsbacher [2010] to propagate light efficiently in a scene using a set of discretized directions on a 3D grid. The method allows real-time execution times and deformable meshes with no added pre-computation step, with the drawback of not being physically accurate.

## CHAPTER 3

# Theory

---

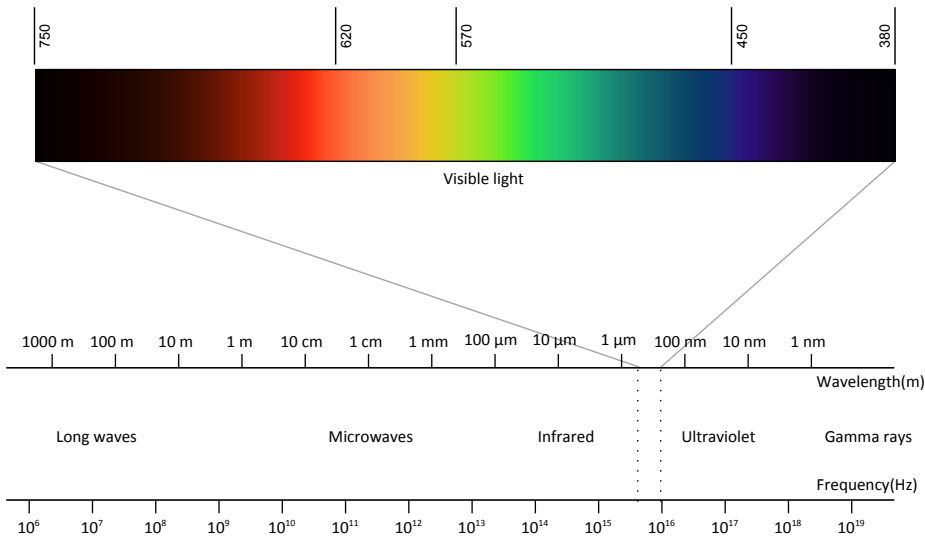
In this chapter, we give a theoretical introduction to the topic dealt with in this thesis. The ultimate goal of this chapter is to introduce and describe a analytical model for subsurface scattering. First, we will give a brief introduction to what light is, and how we physically describe it. Secondly, we will introduce the basic radiometric quantities that will be used throughout the chapter. Then, we will describe how this quantities are related and can be used to describe light-material interaction, using reflectance functions, of which BSSRDF functions are a special case. Finally, we will introduce subsurface scattering and the diffusion approximation, concluding with a description of two BSSRDF functions actually used to describe it, by Jensen et al. [2001] and Frisvad et al. [2013].

### 3.1 Light and Radiometry

Light is a form of electromagnetic radiation, that propagates through space as a sinusoidal wave. Usually by "light" we usually refer to *visible light*, the small part of the electromagnetic spectrum the human eye is sensible to (see Figure 3.3). This small window is between the 380 nm of the infrared and 750 nm of the ultraviolet, but the precise boundaries vary according to the environment and the observer. Instead explicitly noted, we will use the terms light and visible

light interchangeably in this report.

The study of light is usually referred as optics. In computer aided image synthesis, we are interested in representing faithfully how visible light propagates though the scene and how interacts with materials. In addition, we are usually interested on lighting effects that are noticeable at human scales (1 mm - 1 km). For example, we are interested in subsurface scattering, absorption and emission phenomena but not in diffraction, interference and quantum effects, that for visible light happen on a microscopic scale (1 nm - 1  $\mu$ m).



**Figure 3.1:** The electromagnetic spectrum.

The study of the measurement of electromagnetic radiation is called *radiometry*. The energy of light, like all the others forms of energy, is measured in *Joules* [ $J = \text{kg m s}^{-2}$ ], and its power in *Watts* [ $W = \text{kg m s}^{-3}$ ]. *Photometry*, on the other hand, measures electromagnetic radiation as it is perceived from the human eye, and limits itself only to the visible spectrum, while radiometry spans all of it. The quantities for energy and power are called respectively *talbot* [cd s] and *candela* [cd].

In image synthesis radiometry is employed, as its quantities directly derive from the electromagnetic theories, are universal, and can be easily converted to the photometric ones when necessary. The most important radiometric quantities used in computer graphics are *radiant flux*, *radiant energy*, *radiance*, *irradiance* and *intensity*.

## 3.2 Radiometric quantities

### 3.2.1 Radiant flux

The radiant flux, also known as radiant power, is the most basic quantity in radiometry. It is usually indicated with the letter  $\Phi$  and it is measured in joules per seconds [ $\text{J s}^{-1}$ ] or Watts [W]. The quantity indicates how much power the light irradiates per unit time. Given its nature, the radiant flux is independent from the distance from the light source.

### 3.2.2 Radiant energy

Radiant energy, usually indicated as  $Q$ , is the energy that the light carries in a certain amount of time. Like all the other SI units for energy, it is measured in joules [J]. Radiant energy is usually obtained integrating the radiant flux along time for an interval  $\Delta T$ :

$$Q = \int_{\Delta T} \Phi dt$$

Due to the dual nature of the light, the energy carried by the light can be derived both considering the flux of photons as particles, or considering light as a wave. We will not dig further into the topic, because for rendering purposes is not important if we characterize light as a flux of particles or as a sinusoidal wave.

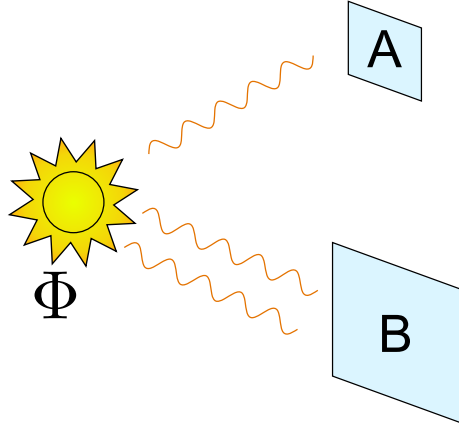
### 3.2.3 Irradiance

Irradiance, usually defined as  $E$ , is the radiometric unit that measures the radiant flux per unit area *falling* on a surface. It is measured in Watts per square meter [ $\text{W m}^{-2}$ ]. It is obtained by further deriving the radiant flux by the area:

$$E = \frac{d\Phi}{dA}$$

Irradiance is usually the term in literature used for the *incoming* power per unit area. The converse, i.e. the irradiance leaving a surface, it is usually referred as

*radiant exitance* or *radiosity*, and indicated with the letter  $B$ .



**Figure 3.2:** Irradiance versus power. For the two surfaces  $A$  and  $B$ , the received power  $\Phi$  is the same, while the two irradiances  $E_A$  and  $E_B$  are different, as the area of  $B$  is twice as the one of  $A$ .

### 3.2.4 Intensity

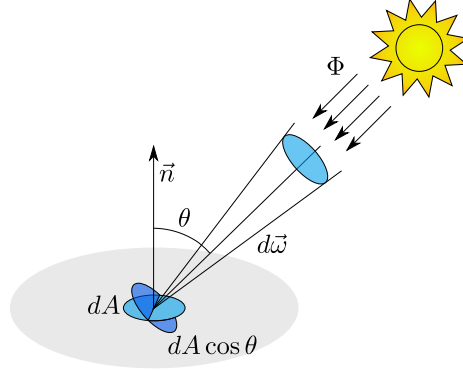
Intensity is defined as the differential radiant flux per differential solid angle:

$$I(\vec{\omega}) = \frac{d\Phi}{d\omega}$$

It is measured in Watts / steradian [ $\text{W sr}^{-1}$ ] and it is indicated with the letter  $I$ . Intensity is often a misused term in the physics community, as it is used for many different measures. Depending on the community, intensity may refer to irradiance or even to radiance (see the following section). We will refer to intensity as it is generally interpreted by the optics community, i.e. radiant intensity.

### 3.2.5 Radiance

Radiance is the most important quantity in image synthesis. It is defined precisely as the differential of the flux per solid angle per projected surface area,



**Figure 3.3:** Radiance. The element of area  $A$  gets projected according to the angle  $\theta = \cos^{-1} \vec{n} \cdot \vec{\omega}$ . Then the incoming flux  $\Phi$  gets divided by the projected area and by the solid angle subtended by it.

and it is measured in Watt per steradian per square meter [ $\text{W sr}^{-1} \text{m}$ ].

$$L(\vec{\omega}) = \frac{d^2 \Phi}{d\omega dA \cos \theta}$$

Where  $\theta$  is the angle between the surface normal and the incoming ray of light (so that  $\cos \theta = \vec{n} \cdot \vec{\omega}_i$ ). Radiance is important in image synthesis because it is the natural quantity to associate with a ray of light, as it remains constant along it. In addition, the sensibility of the human eye to light is directly proportional to the radiance. For a discussion on why radiance is related to the sensitivity of sensors and the human eye, see X.

All the other radiometric quantities can be derived from radiance:

$$\begin{aligned} E &= \int_{2\pi} L_i(\vec{\omega}) \cos \theta \, d\omega \\ B &= \int_{2\pi} L_o(\vec{\omega}) \cos \theta \, d\omega \\ I(\vec{\omega}) &= \int_A L(\vec{\omega}) \cos \theta \, dA \\ \Phi &= \int_A \int_{2\pi} L(\vec{\omega}) \cos \theta \, d\omega dA \end{aligned} \tag{3.1}$$

For simplicity of notation, in the previous notations the dependence from the

point of incidence  $\mathbf{x}$  has been dropped. In principle, all the quantities listed in equation 3.1 are dependent on  $\mathbf{x}$ .

### 3.2.6 Radiometric quantities for simple lights

To help with the formulas used later in the report, we derive the above mentioned radiometric quantities for the two simplest types of light, i.e. directional and point lights.

- *Directional lights* simulate very distant light sources, in which all the rays of light are parallel (e.g. the sun). They are represented by a direction  $\vec{\omega}_l$  and a constant radiance value,  $L$ .
- *Point lights* simulate lights closer to the observer. Isotropic point lights are represented by a position of the light  $\mathbf{x}_l$  and a constant intensity  $I$ . Point lights have a falloff that depends on the inverse square law, i.e. the radiance diminishes with the square of the distance.

Table 3.1 shows different radiometric quantities evaluated for point and directional lights, for a surface point  $\mathbf{x}$  with surface normal  $\vec{n}$ .

Quantity	Directional light	Point light
Cosine term	$\cos \theta = \vec{n} \cdot \vec{\omega}_l$	$\cos \theta = \frac{(\mathbf{x} - \mathbf{x}_l) \cdot \vec{n}}{ \mathbf{x} - \mathbf{x}_l }$
$\Phi(\mathbf{x})$ Flux	$\infty$	$4\pi I$
$E(\mathbf{x})$ Irradiance	$L \cos \theta$	$I \frac{\cos \theta}{ \mathbf{x}_l - \mathbf{x} ^2}$
$I(\mathbf{x}, \vec{\omega})$ Intensity	$\infty$	$I$
$L(\mathbf{x}, \vec{\omega})$ Radiance	$L$	$\frac{I}{ \mathbf{x}_l - \mathbf{x} ^2}$

**Table 3.1:** Different radiometric values for simple light sources. Note that flux and irradiance for a directional light are infinite because they are integrated over an infinite area.



### 3.3 Reflectance Functions

After introducing the basic radiometric quantities, we still lack a way to describe light material interaction. More precisely, we need a way to relate the incoming and the outgoing radiance on a point of a chosen surface. As we have already discussed before, we use radiance since is the radiometric quantity that is directly proportional to what the human eye measures.

#### 3.3.1 BRDF functions

One of the possible way to describe light-material interaction is by using a BRDF function, acronym for *Bidirectional Reflectance Distribution Function*. The BRDF function  $f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o)$  is defined on one point  $\mathbf{x}$  of the surface as the differential ratio between the exiting radiance and the incoming irradiance:

$$f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) = \frac{dL_o(\mathbf{x}, \vec{\omega}_o)}{dE_i(\mathbf{x}, \vec{\omega}_i)} = \frac{dL_o(\mathbf{x}, \vec{\omega}_o)}{L_i(\mathbf{x}, \vec{\omega}_i) \cos \theta_i d\vec{\omega}_i} \quad (3.2)$$

The BRDF states that the incoming and the outgoing radiance are proportional, so that the energy hitting the material at the point  $\mathbf{x}$  is proportional to the energy coming out from the point. The BRDF function is generically reciprocal for the Hemholtz reciprocity principle ( $f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) = f(\mathbf{x}, \vec{\omega}_o, \vec{\omega}_i)$ ), and anisotropic (if the surface changes orientation and  $\vec{\omega}_i$  and  $\vec{\omega}_o$  stays the same, the resulting BRDFs are different). In addition, it is positive ( $f(\mathbf{x}, \vec{\omega}_o, \vec{\omega}_i) \geq 0$ ), and conserves energy, so that the energy of the outgoing ray is no grater that the one of the incoming one ( $\int_{2\pi} f(\mathbf{x}, \vec{\omega}_o, \vec{\omega}_i) \cos \theta_o d\vec{\omega}_o \leq 1$ ).

By inverting equation 3.2, we obtain the so-called *reflectance equation*:

$$L_o(\mathbf{x}, \vec{\omega}_o) = \int_{2\pi} f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) L_i(\mathbf{x}, \vec{\omega}_i) \cos \theta_i d\vec{\omega}_i$$

That later we will extend to obtain the rendering equation. The BRDF function has some limitations, being not able to account for all phenomena. For example, with a BRDF it is not possible to account for subsurface scattering phenomena, because it assumes the light enters and leaves the material in the same point. To model these phenomena, more complicated functions are needed, like the BSSRDF function described later in this chapter.

### 3.3.2 Examples of BRDF functions

There are many examples of BRDF functions in literature. In this section, we will introduce three of the simplest ones: for a more exhaustive description of BRDF functions, refer to X. The three BRDFs are the lambertian or diffuse BRDF, the specular or mirror BRDF and the family of glossy BRDFs.

#### 3.3.2.1 Lambertian BRDF

In the lambertian BRDF, the incoming radiance is distributed equally in all directions, regardless of the incoming direction. To do this, the BRDF must be constant:

$$f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) = k_d$$

We can check that then the radiance is scattered equally in all directions by simple integration:

$$\begin{aligned} L_o(\mathbf{x}, \vec{\omega}_o) &= \int_{2\pi} f_d L_i(\mathbf{x}, \vec{\omega}_i) \cos \theta_i d\vec{\omega}_i \\ L_o(\mathbf{x}, \vec{\omega}_o) &= k_d \int_{2\pi} L_i(\mathbf{x}, \vec{\omega}_i) \cos \theta_i d\vec{\omega}_i \\ L_o(\mathbf{x}, \vec{\omega}_o) &= k_d E(\mathbf{x}) \end{aligned}$$

The lambertian model is an ideal model, so very few material exhibit a lambertian diffusion, like unfinished wood or spectralon, a synthetic material created in order to have a lambertian diffusion.

#### 3.3.2.2 Mirror BRDF

Another simple kind of BRDF is the perfectly specular BRDF, or mirror BRDF. In this function, all the incoming radiance from one direction  $\vec{\omega}_i$  is completely diffused into the reflected direction  $\vec{\omega}_r$ , defined as  $\vec{\omega}_r = \vec{\omega}_i - 2(\vec{\omega}_i \cdot \vec{n})\vec{n}$ . The resulting BRDF is defined as follows:

$$f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) = \frac{\delta(\vec{\omega}_o - \vec{\omega}_r)}{\cos \theta_i}$$

The function  $\delta(\vec{\omega})$  is a hemispheric delta function. Once integrated over a hemisphere, the function evaluates to one only for the vector  $\vec{\omega} = \mathbf{0}$ . Putting the BRDF into the reflectance equation gives the following outgoing radiance:

$$L_o(\mathbf{x}, \vec{\omega}_o) = \begin{cases} L_i(\mathbf{x}, \vec{\omega}_i) & \text{if } \vec{\omega}_o = \vec{\omega}_r \\ 0 & \text{otherwise} \end{cases}$$

that is the expected result, as all the radiance is reflected into the direction  $\vec{\omega}_r$ .

### 3.3.2.3 Glossy BRDFs

As we can see from real life experience, rarely objects are completely diffuse or completely specular. These two models are idealized models, that represent an ideal case. So, to create a realistic BRDF model, we often need to combine the two terms and add an additional one, called glossy reflection. This term is often needed to model the behavior of the surface more realistically, especially at grazing angles.

The most used BRDF model used to model glossy reflections is based on microfacet theory. In this theory, the surface of an object is modeled as composed of small mirrors. In one of its classical formulation, the BRDF is represented as:

$$f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) = \frac{DGR}{4 \cos \theta_r \cos \theta_i} = \frac{GR}{4} \frac{(\vec{n} \cdot \vec{h})^s}{(\vec{n} \cdot \vec{r})(\vec{n} \cdot \vec{\omega}_i)}$$

$D$  regulates how microfacets are distributed, and it is often modeled as  $(\vec{n} \cdot \vec{h})^s$ , where  $\vec{h}$  is the half vector between the eye and the light, and  $s$  is an attenuation parameter.  $\vec{h}$  is defined as:

$$\vec{h} = \frac{\vec{\omega}_o + \vec{\omega}_i}{\|\vec{\omega}_o + \vec{\omega}_i\|}$$

$G$  accounts for the object self shadowing, while  $R$  is the Fresnel reflection term (more details in section Y).  $\vec{r}$  is the reflection vector as defined in the previous

section. See figure Z on how the vectors for the glossy reflection -  $\vec{n}$ ,  $\vec{h}$  and  $\vec{r}$  - are defined.

Various alternative definitions exist for the  $D$  and  $G$  function, varying among the literature. Other glossy models not based on microfacet theory do exist as well.

### 3.3.3 The rendering equation

Given the reflectance equation, it is possible to generalize it in order to model all the lighting in an environment. In fact, this form of the reflectance equation does not account for two important factors.

The first factor are emissive surfaces. We need to add an emissive radiance term  $L_e(\mathbf{x}, \vec{\omega})$  that models how much radiance is a point on a surface emitting in a certain directions. This is useful to model lights as any other surface in the scene. Note that point lights have a singularity: they emit infinite radiance on the point they are placed.

The second factor is that the reflectance equation accounts only for direct illumination. In general, we want to model global illumination, i.e. to include also light that bounced onto another surface before reaching the current surface. To model this, we can replace the  $L_i$  term in the reflectance equation with another term  $L_r$  that accounts for reflected radiance. This term can be usually modeled as the product of the radiance of the light plus a visibility function  $V(\mathbf{x})$ .

Accounting for all the described factors, we reach one formulation of the rendering equation:

$$L_o(\mathbf{x}, \vec{\omega}_o) = L_e(\mathbf{x}, \vec{\omega}) + \int_{2\pi} f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) L_i(\mathbf{x}, \vec{\omega}_i) V(\mathbf{x}) \cos \theta_i d\vec{\omega}_i$$

This form of the rendering equation is still not completely general, since it is based on a BRDF, so it is not possible to model subsurface scattering effects or wavelength-changing effects (like iridescence). We will extend the rendering equation in order to account for these phenomena later on in this chapter.

### 3.3.4 Fresnel equations

Until now, on the described BRDF models, we did consider only the reflected part of the radiance. When a beam of light coming from direction  $\vec{\omega}_i$  hits a surface, only part of the incoming radiance gets reflected, while another part gets refracted into the material. As we can see from the setup from figure Z, we obtain the two vectors  $\vec{\omega}_r$  and  $\vec{\omega}_t$ , the reflected and refracted vector, defined as follows:

$$\begin{aligned}\vec{\omega}_r &= \vec{\omega}_i - 2(\vec{\omega}_i \cdot \vec{n})\vec{n} \\ \vec{\omega}_t &= \eta((\vec{\omega}_i \cdot \vec{n})\vec{n} - \vec{\omega}_i) - \vec{n}\sqrt{1 - \eta^2(1 - (\vec{\omega}_i \cdot \vec{n})^2)}\end{aligned}$$

Where  $\eta = \frac{n_1}{n_2}$  is the relative index of refraction of the material. With this setup, we can use a solution to Maxwell's equations for wave propagation to describe how light spreads between reflected and refracted directions. What we obtain are called *Fresnel coefficients*. The coefficients are different according to the polarization of the incoming light, so there are two for the reflection ( $R_s$ ,  $R_p$ ) and two for transmission ( $T_s$ ,  $T_p$ ). The coefficients relate the incoming and outgoing power of the light, and by differentiation, the radiance as well.

$$\begin{aligned}R_s &= \left| \frac{n_1 \cdot \cos \theta_i - n_2 \cdot \cos \theta_t}{n_1 \cdot \cos \theta_i + n_2 \cdot \cos \theta_t} \right|^2 & R_p &= \left| \frac{n_1 \cdot \cos \theta_t - n_2 \cdot \cos \theta_i}{n_1 \cdot \cos \theta_t + n_2 \cdot \cos \theta_i} \right|^2 \\ T_s &= \frac{n_2 \cdot \cos \theta_t}{n_1 \cdot \cos \theta_i} \left| \frac{2n_1 \cdot \cos \theta_i}{n_1 \cdot \cos \theta_i + n_2 \cdot \cos \theta_t} \right|^2 & T_p &= \frac{n_2 \cdot \cos \theta_t}{n_1 \cdot \cos \theta_i} \left| \frac{2n_1 \cdot \cos \theta_i}{n_1 \cdot \cos \theta_t + n_2 \cdot \cos \theta_i} \right|^2\end{aligned}$$

In most computer graphics applications (and this is reasonable for most of the real-world lights), we assume that the two polarizations are equally mixed. So, we will use the coefficient  $R = \frac{R_s + R_p}{2}$  and  $T = \frac{T_s + T_p}{2}$  in our calculations. Note that  $R + T = 1$ , so the overall energy is conserved.

## 3.4 Light transport and subsurface scattering

When we derive our models for lighting, in general we assume that the light is traveling in vacuum. This assumption holds for light that is propagating through the air (which is assimilable to vacuum), but once we relax it, more variables

should be taken into consideration. Objects through which light travels are referred as *participating media*. In this chapter, we will derive and consider an alternative formulation of the rendering equation for light traveling into participating media, called *volume rendering equation*.

When a beam of light travels through an object, various phenomena occur. A photon on the beam can be either being absorbed (disappear), scattered (change direction) or emitted (appear). These phenomena can be uniform throughout the material (homogeneous materials), as in solid material like wax or leaves, or be not uniform (heterogeneous materials), like in smoke or clouds.

We will briefly describe all three mentioned effects, then combine them to compose the volume rendering equation. The purpose is to describe how radiance varies along a beam of light with direction  $\vec{\omega}$ . This directional derivative is indicated as:

$$(\vec{\nabla} \cdot \vec{\omega})L(\mathbf{x}, \vec{\omega}) = \frac{\partial L_x}{\partial x}\vec{\omega}_x + \frac{\partial L_y}{\partial y}\vec{\omega}_y + \frac{\partial L_z}{\partial z}\vec{\omega}_z$$

### 3.4.1 Emission

Emission is the natural property of the materials to emit light, i.e. to generate photons that add to the existing ones passing through the material. The effect is generally generated by chemical processes emitting photons (as in natural animals like fireflies), by natural black-body radiation emission in the visible spectrum (such as in a star like the sun or in incandescent bulbs), or by other radiation that changed its wavelength into the visible spectrum.

In the directional 3D derivative, the variance in emission is modeled as a constant depending only on the current position and direction.

$$(\vec{\nabla} \cdot \vec{\omega})L(\mathbf{x}, \vec{\omega}) = \epsilon(\mathbf{x}, \vec{\omega})$$

This means that emission increases linearly along the body: if the beam travels a distance  $d$  within the medium,  $d \cdot k$  photons are emitted. Emission is generally isotropic, not depending on the direction ( $\epsilon(\mathbf{x}, \vec{\omega}) = \epsilon(\mathbf{x})$ ).

### 3.4.2 Absorption

Absorption is a property of materials that models a simple physical phenomenon: a photon, traveling through the material, hits one atom of the material. The energy carried by the photon is then absorbed by the atom, augmenting its kinetic energy. This directly translates in an increase of heat in the material. Usually a certain percentage of the photons hits the atoms and gets absorbed. This means, if  $k$  is the percentage of the photons absorbed in a unit, after one unit the original radiance will become  $k \cdot L_i$ , then  $k^2 \cdot L_i$ , etc.

If we write this phenomena as a differential equation, we get after a distance  $d$  a decaying in radiance of  $k^d = e^{-\sigma_a d}$ , that leads to the following 3D directional derivative:

$$(\vec{\nabla} \cdot \vec{\omega})L(\mathbf{x}, \vec{\omega}) = -\sigma_a(\mathbf{x}, \vec{\omega})L(\mathbf{x}, \vec{\omega})$$

$\sigma_a$  is referred as the *absorption coefficient*. Also this coefficient is generically isotropic, and constant for homogenous materials.

### 3.4.3 Out-scattering

Out scattering is the radiance lost due to scattering. The scattering phenomenon happens when photons are deflected away from the current direction  $\vec{\omega}$ . As in the previous case, the phenomena is modeled as a percentage of the radiance lost per unit length. So the loss due to out-scattering is modeled as:

$$(\vec{\nabla} \cdot \vec{\omega})L(\mathbf{x}, \vec{\omega}) = -\sigma_s(\mathbf{x}, \vec{\omega})L(\mathbf{x}, \vec{\omega})$$

$\sigma_s$  is referred as the *scattering coefficient*. We note that in this case we are not interested in which direction the photons are actually going. That will be accounted in the in-scattering term of another point in the material.

### 3.4.4 In-scattering

Given some loss due to some of the photons changing direction, there will be some of them that from other scattering events will change to the  $\vec{\omega}$  direction.

We need then to discover the number of photons that comes from all the other directions. To do this, we integrate the incoming radiance from all directions in the point  $\mathbf{x}$ . This quantity, similar to irradiance, in an infinite medium is called *fluence*, and indicated as  $\phi$ :

$$\phi(\mathbf{x}) = \int_{4\pi} L(\mathbf{x}, \vec{\omega}') d\omega'$$

This quantity should be then averaged over the entire sphere, yielding  $\frac{\phi}{4\pi}$  as a result. This quantity then is then multiplied by the scattering coefficient, because only some photons on average align to the current direction  $\vec{\omega}$ . This yields:

$$(\vec{\nabla} \cdot \vec{\omega})L(\mathbf{x}, \vec{\omega}) = \sigma_s(\mathbf{x}) \frac{1}{4\pi} \int_{4\pi} L(\mathbf{x}, \vec{\omega}') d\omega' \quad (3.3)$$

However, equation 3.4 assumes that radiance scatters equally in all directions. This is not usually the case, and the  $\frac{1}{4\pi}$  term needs to be replaced by a probability distribution function that describes how the photons scatter in the medium. This function is called phase function, and indicated as  $p(\mathbf{x}, \vec{\omega}, \vec{\omega}')$ . In the actual models its integral on the hemisphere is often used as a parameter, called *mean cosine* ( $g$ ):

$$g(\mathbf{x}) = \int_{4\pi} p(\mathbf{x}, \vec{\omega}, \vec{\omega}') \vec{\omega} \cdot \vec{\omega}' d\omega'$$

This term indicates the general direction of the scattering in the material. If positive, the scattering is prevalent along the beam (forward scattering), if negative is prevalent in the opposite direction (backward scattering). If zero, the scattering is isotropic, i.e. equal in all directions.

So, the final 3D equation for in-scattering, accounting for the phase function, is as follows:

$$(\vec{\nabla} \cdot \vec{\omega})L(\mathbf{x}, \vec{\omega}) = \sigma_s(\mathbf{x}) \int_{4\pi} p(\mathbf{x}, \vec{\omega}, \vec{\omega}') L(\mathbf{x}, \vec{\omega}') d\omega' \quad (3.4)$$



### 3.4.5 Final formulation of the volume rendering equation

Combining emission, absorption, scattering described in the previous sections, we reach the final formulation of the volume rendering equation:

$$(\vec{\nabla} \cdot \vec{\omega})L(\mathbf{x}, \vec{\omega}) = -\sigma_t(\mathbf{x})L(\mathbf{x}, \vec{\omega}) + \epsilon(\mathbf{x}) + \sigma_s(\mathbf{x}) \int_{4\pi} p(\mathbf{x}, \vec{\omega}, \vec{\omega}')L(\mathbf{x}, \vec{\omega}')d\omega'$$

Where the two reducing term, scattering and absorption, have been combined together in  $\sigma_t = \sigma_a + \sigma_s$ , called the *extinction coefficient*.

### 3.4.6 BSSRDF functions and generalized rendering equation

In order to be able to model subsurface scattering

### 3.4.7 Standard dipole model

### 3.4.8 Directional dipole model



## CHAPTER 4

# Results

---



# Bibliography

---

Jesper Børlum, Brian Bunch Christensen, Thomas Kim Kjeldsen, Peter Trier Mikkelsen, Karsten Østergaard Noe, Jens Rimestad, and Jesper Mosegaard. Sslpv: Subsurface light propagation volumes. In *Proceedings of the ACM SIGGRAPH Symposium on High Performance Graphics*, HPG '11, pages 7–14, New York, NY, USA, 2011. ACM. URL <http://doi.acm.org/10.1145/2018323.2018325>.

Carsten Dachsbacher and Marc Stamminger. Translucent shadow maps. In *Proceedings of the 14th Eurographics Workshop on Rendering*, EGRW '03, pages 197–201, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association. URL <http://dl.acm.org/citation.cfm?id=882404.882433>.

Eugene D'Eon. A better dipole (a publicly available manuscript). Technical report, -, 2012. URL <http://www.eugenedeon.com/papers/betterdipole.pdf>.

Eugene D'Eon and Geoffrey Irving. A quantized-diffusion model for rendering translucent materials. In *ACM SIGGRAPH 2011 Papers*, SIGGRAPH '11, pages 56:1–56:14, New York, NY, USA, 2011. ACM. URL <http://doi.acm.org/10.1145/1964921.1964951>.

Eugene d'Eon, David Luebke, and Eric Enderton. Efficient rendering of human skin. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques*, EGSR'07, pages 147–157, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association. URL <http://dx.doi.org/10.2312/EGWR/EGSR07/147-157>.

Craig Donner and Henrik Wann Jensen. Light diffusion in multi-layered translucent materials. In *ACM SIGGRAPH 2005 Papers*, SIGGRAPH '05, pages

- 1032–1039, New York, NY, USA, 2005. ACM. URL <http://doi.acm.org/10.1145/1186822.1073308>.
- Craig Donner, Jason Lawrence, Ravi Ramamoorthi, Toshiya Hachisuka, Henrik Wann Jensen, and Shree Nayar. An empirical bssrdf model. *ACM Trans. Graph.*, 28(3):30:1–30:10, July 2009. URL <http://doi.acm.org/10.1145/1531326.1531336>.
- Julie Dorsey, Alan Edelman, Henrik Wann Jensen, Justin Legakis, and Hans K hling Pedersen. Modeling and rendering of weathered stone. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH ’99, pages 225–234, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co. URL <http://dx.doi.org/10.1145/311535.311560>.
- Raanan Fattal. Participating media illumination using light propagation maps. *ACM Trans. Graph.*, 28(1):7:1–7:11, February 2009. URL <http://doi.acm.org/10.1145/1477926.1477933>.
- J. R. Frisvad, T. Hachisuka, and T. K. Kjeldsen. Directional dipole for sub-surface scattering in translucent materials. Technical report, DTU Compute, 2013. URL <http://www2.imm.dtu.dk/pubdb/p.php?6646>. Unpublished manuscript.
- Henrik Wann Jensen and Juan Buhler. A rapid hierarchical rendering technique for translucent materials. *ACM Trans. Graph.*, 21(3):576–581, July 2002. URL <http://doi.acm.org/10.1145/566654.566619>.
- Henrik Wann Jensen and Per H. Christensen. Efficient simulation of light transport in scenes with participating media using photon maps. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH ’98, pages 311–320, New York, NY, USA, 1998. ACM. URL <http://doi.acm.org/10.1145/280814.280925>.
- Henrik Wann Jensen, Stephen R. Marschner, Marc Levoy, and Pat Hanrahan. A practical model for subsurface light transport. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH ’01, pages 511–518, New York, NY, USA, 2001. ACM. URL <http://doi.acm.org/10.1145/383259.383319>.
- Jorge Jimenez, Veronica Sundstedt, and Diego Gutierrez. Screen-space perceptual rendering of human skin. *ACM Trans. Appl. Percept.*, 6(4):23:1–23:15, October 2009. URL <http://doi.acm.org/10.1145/1609967.1609970>.
- Anton Kaplanyan and Carsten Dachsbacher. Cascaded light propagation volumes for real-time indirect illumination. In *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, I3D ’10,

- pages 99–107, New York, NY, USA, 2010. ACM. URL <http://doi.acm.org/10.1145/1730804.1730821>.
- Hendrik P. A. Lensch, Michael Goesele, Philippe Bekaert, Jan Kautz, Marcus A. Magnor, Jochen Lang, and Hans-Peter Seidel. Interactive rendering of translucent objects. In *Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, PG '02, pages 214–, Washington, DC, USA, 2002. IEEE Computer Society. URL <http://dl.acm.org/citation.cfm?id=826030.826632>.
- Tom Mertens, J. Kautz, P. Bekaert, F. Van Reeth, and H.-P. Seidel. Efficient rendering of local subsurface scattering. In *Computer Graphics and Applications, 2003. Proceedings. 11th Pacific Conference on*, pages 51–58, Oct 2003a.
- Tom Mertens, Jan Kautz, Philippe Bekaert, Hans-Peter Seidelz, and Frank Van Reeth. Interactive rendering of translucent deformable objects. In *Proceedings of the 14th Eurographics Workshop on Rendering*, EGRW '03, pages 130–140, Aire-la-Ville, Switzerland, Switzerland, 2003b. Eurographics Association. URL <http://dl.acm.org/citation.cfm?id=882404.882423>.
- Matt Pharr and Pat Hanrahan. Monte carlo evaluation of non-linear scattering equations for subsurface reflection. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, pages 75–84, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co. URL <http://dx.doi.org/10.1145/344779.344824>.
- M.A. Shah, J. Konttinen, and S. Pattanaik. Image-space subsurface scattering for interactive rendering of deformable translucent objects. *Computer Graphics and Applications, IEEE*, 29(1):66–78, Jan 2009.
- Jos Stam. Multiple scattering as a diffusion process. In PatrickM. Hanrahan and Werner Purgathofer, editors, *Rendering Techniques 1995*, Eurographics, pages 41–50. Springer Vienna, 1995. URL [http://dx.doi.org/10.1007/978-3-7091-9430-0\\_5](http://dx.doi.org/10.1007/978-3-7091-9430-0_5).
- Anna Tomaszewska and Krzysztof Stefanowski. Real-time spherical harmonics based subsurface scattering. In Aurélio Campilho and Mohamed Kamel, editors, *Image Analysis and Recognition*, volume 7324 of *Lecture Notes in Computer Science*, pages 402–409. Springer Berlin Heidelberg, 2012. URL [http://dx.doi.org/10.1007/978-3-642-31295-3\\_47](http://dx.doi.org/10.1007/978-3-642-31295-3_47).
- Yajun Wang, Jiaping Wang, Nicolas Holzschuch, Kartic Subr, Jun-Hai Yong, and Baining Guo. Real-time rendering of heterogeneous translucent objects with arbitrary shapes. *Comput. Graph. Forum*, 29(2):497–506, 2010. URL <http://dblp.uni-trier.de/db/journals/cgf/cgf29.html#WangWHSYG10>.