Course Notes
SIGGRAPH 2005 Course 10
Realistic Materials in Computer Graphics
July 31, 2005

Co-Organizers:

Hendrik P. A. Lensch
Stanford University

Michael Goesele
MPI Informatik

Lecturers:

Yung-Yu Chuang
National Taiwan University

Tim Hawkins
University of Southern California

Steve Marschner
Cornell University

Wojciech Matusik
Mitsubishi Electric Research Laboratories (MERL)

Gero Mueller
Universität Bonn

Chapter 1

Course Introduction

1.1 Course Description

This course provides an introduction and overview over the field of measuring reflection properties of materials and objects targeted to the specific needs of computer graphics applications. The course will introduce and present a broad set of current acquisition methods where each approach is particularly suited for specific types of materials. The course covers the range of acquisition techniques from opaque surfaces, subsurface scattering, or fibers to complete objects.

1.2 Prerequisites

This course is self contained with regard to the principles and concepts used for each acquisition technique. However, attendees should have a working knowledge of basic concepts of realistic rendering and object representations such as triangle meshes or texture maps.

1.3 Course Web Page

Updated versions of the course notes and additional information about the course are available at the following course web page:

`http://www.mpi-sb.mpg.de/resources/siggraph05-course-realistic-materials`

1.4 Syllabus

8:30	**Welcome and Speaker Introduction**
8:45	**Introduction** (Lensch) - material properties - classification
9:30	**Acquisition Basics** (Goesele) - light sources - cameras - HDR
10:15	**Break**
10:30	**Homogeneous Isotropic BRDFs** (Matusik) - image based BRDF measurement - data driven BRDF model - validation of analytical BRDF Models
11:15	**Heterogeneous Isotropic BRDFs** (Lensch) - spatially varying BRDFs from few images
11:45	**Translucent Materials** (Goesele) - BSSRDF - simplification to 4D
12:15	**Lunch Break**
1:45	**Transparent/Specular Materials** (Chuang) - environment matting
2:30	**Fibers** (Marschner) - fiber radiometry - human hair
3:30	**Break**
3:45	**BTFs** (Mueller) - surface patches - acquisition, compression, and rendering
4:30	**Reflectance Fields** (Hawkins) - light stage 1,2,3 - opacity hulls - putting it all together
5:15	**Conclusion, Questions and Answers** (all)

1.5 Course Presenter Contact Information

Hendrik Lensch
Stanford University
Computer Graphics Laboratory
Gates Bldg. 3B-364
353 Serra Mall
Stanford, CA 94305
USA

Email: lensch@stanford.edu
WWW: http://graphics.stanford.edu/~lensch

Michael Goesele
MPI Informatik
Stuhlsatzenhausweg 85
66123 Saarbruecken
Germany

Email: goesele@mpi-sb.mpg.de
WWW: http://www.mpi-sb.mpg.de/~goesele

Yung-Yu Chuang
Department of Computer Science and Information Engineering
National Taiwan University
No.1, Sec. 4, Roosevelt Road
Taipei, 106
Taiwan

Email: cyy@csie.ntu.edu.tw
WWW: http://www.csie.ntu.edu.tw/~cyy

Tim Hawkins
University of Southern California
Institute for Creative Technologies
13274 Fiji Way, Suite 600
Marina del Rey, CA 90292
USA

Email: timh@ict.usc.edu

Steve Marschner
Cornell University
Computer Science Department
5159 Upson Hall
Ithaca, NY 14853
USA

Email: srm@cs.cornell.edu
WWW: http://www.cs.cornell.edu/~srm/

Wojciech Matusik
Mitsubishi Electric Research Laboratories (MERL)
201 Broadway
Cambridge, MA 02139
USA

Email: matusik@merl.com
WWW: http://www.merl.com/people/matusik/

Gero Mueller
University of Bonn, Institute for Computer Science II
Roemerstr. 164
53117 Bonn
Germany

Email: gero@cs.uni-bonn.de
WWW: http://cg.cs.uni-bonn.de/staff/person.asp?id=51

1.6 Acknowledgements

The authors would like to thank all of their co-authors and colleagues whose work is presented in these course notes. The authors are furthermore grateful to their funding agencies for financial support.

Contents

Part I

Course Notes

Chapter 2

Introduction

Hendrik P. A. Lensch

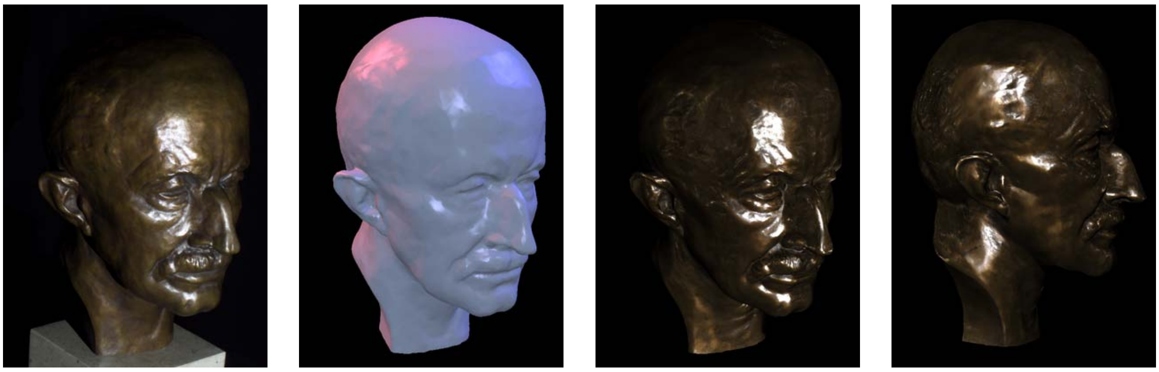
The appearance of materials play a very important role in our everyday live. Based on its appearance, that is, based on the way an object looks like we try to deduce what materials an object consists of. This has a number of consequences with regard to how we might interact with the object, what the object might be used for, or it tells us something about its mechanical properties [Adelson01]. Even the subjective value or the age of an object are often estimated based on its appearance. For example, velvety objects or surfaces with a specific shine tend to be more precious. In art, design and architecture different materials or colors are carefully composed in order to create a very specific look, to create a specific effect or atmosphere.

Because of all these reasons, materials and their appearance are of increasing interest to computer graphics. After highly advanced physics-based rendering algorithms have been developed in the last decade the limiting factor in performing photorealistic or predictive rendering is nowadays the accuracy of the scene description, specifically the material properties. Lack of accuracy or missing detail in the provided model will result in unconvincing images even with sophisticated rendering. One way to obtain highly detailed and realistic models is to resort to skilled artistry.

An alternative approach to create realistic materials is to capture the appearance of real world surfaces and objects. The field of estimating models from observations has its roots in *computer vision*. In the last decade, the requirements of computer graphics applications pushed the field towards the acquisition of high-quality appearance models.

The digitization of real word materials and objects is of importance not only to image synthesis applications, such as film production or computer games, but also to a number of other applications, such as e-commerce, education, digital libraries, cultural heritage, and so forth. The specific applications may range from presenting and preserving a digital copy of a real world original in a synthetic environment, over creating material data bases for adding a realistic look to synthetic geometry models, to advanced analysis where additional information is derived from the captured appearance data to enhance and investigate surface features or to perform material classification.

The focus of this course is on the acquisition of appearance of objects and surfaces: That is to faithfully capture the objects' reflection properties including the subtle details of the surface texture. With this data at hand the object can be rendered in arbitrary virtual environments, exactly resembling the way the original object would appear, as demonstrated in Figure 2.1. For arbitrary (non-fluorescent, non-phosphorescent) materials, the reflection properties can be described by



(a) (b) (c) (d)

Figure 2.1: *Digitized Bronze Bust. A realistic model of the original bust (a) is digitized by its 3D geometry (b). Acquiring also the reflection properties the appearance can be reproduced (c) even for novel view and lighting directions (d).*

the 8D reflectance field of the surface, also called BSSRDF. Since densely sampling an 8D function is currently not practical various acquisition methods have been proposed which reduce the number of dimensions by restricting the acquisition to specific classes of materials. In addition, the density in each sampling dimension is chosen to meet the properties of the specific materials, to achieve further speedup in the acquisition, and to reduce the amount of captured data. The acquisition is most often followed by a data modeling step which allows for interpolation of missing values, further reduces the storage cost, and depends on the envisioned class of materials as well. In this course we will give an overview over the field of measuring the reflection properties of different classes of objects targeted to the specific needs of computer graphics applications.

After a short introduction explaining some fundamental concepts regarding measuring and representing reflection properties (Chapter 3) and a taxonomy of appearance acquisition algorithms (Chapter 4), basics of data acquisition from photographs will be addressed in Chapter 5. The course presents a set of current state-of-the-art algorithms for acquiring and modeling specific types of materials, such as homogeneous, isotropic materials (Chapter 6), spatially varying materials (Chapter 7), translucent materials (Chapter 8), mirroring or transparent surfaces (Chapter 9), and fibers (Chapter 10). These chapters are followed by slightly more general approaches considering surfaces with complex meso-structure (Chapter 11), or full objects (Chapter 12). The course investigates the strengths and limitations of each technique and highlights their common features as well as their conceptual differences.

Chapter 3

Background

Hendrik P. A. Lensch

In this chapter we briefly review the physical principles related to light transport, summarize forward and inverse rendering problems, and then list and characterize the different phenomena observed when light interacts with matter. A detailed overview of light transport and light/surface interactions can be found in a review by Glassner [Glassner95].

3.1 Radiometry

In optics several models have been developed to describe the physics of light. A detailed introduction to optics can be found in Born and Wolf [Born93]. Some visual effects can be explained more easily by one model than by the others. In *particle optics*, light is modeled as a flow of *photons* each carrying a specific amount of energy. The distribution of energy can be simulated by shooting a number of photons from a light source and tracing them through the environment. In *wave optics* light is interpreted as electromagnetic waves, which allows to describe effects such as diffraction, interference, polarization. That is why wave optics are of importance when modeling the interaction of light with matter.

The energy transport that is simulated in computer graphics is often modeled using *ray optics* where light with a specific power spectrum travels along independent rays. Since there is no explicit interaction between the different light rays, effects like diffraction, interference, and polarization are hard to simulate. To some extent they are incorporated into the reflection model (see below). Otherwise interaction of light with matter is modeled purely geometrically, determining only the direction of the outgoing rays and the transported energy.

3.2 Radiometric Terms

At first we define the physical quantities that can be used to describe radiant energy transport.

Radiant Energy Q is the basic unit of radiometry. measured in Joule $[J]$.

Radiant Flux is the energy per time or *power* of radiation. It is denoted Φ and its unit is Watt $[W]$.

$$\Phi = \frac{dQ}{dt}. \tag{3.1}$$

Radiance is denoted L and is measured in $[W/m^2 sr]$. It is defined as the radiant energy traveling at some point in a given direction, per projected unit area in this direction, per unit time, per unit solid angle. Radiance can

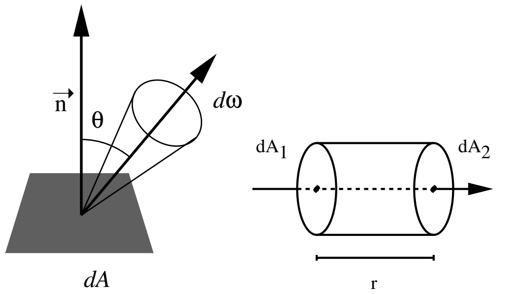


Figure 3.1: *Geometrical properties of radiance.*

be expressed by the radiant flux:

$$L(\vec{x}, \hat{\omega}) = \frac{d^2\Phi}{\cos\theta d\omega dA} \quad , \tag{3.2}$$

where θ denotes the angle between the surface normal at point $\vec{x}$ and the direction $\hat{\omega}$. The underlying geometry is depicted in Figure 3.1(left).

Radiance has one property that is very important in the context of computer graphics. It remains constant along one ray in vacuum (and approximately in air). This is demonstrated with the help of Figure 3.1(right) where two small parallel patches dA_1 and dA_2 are shown. Radiation emitted from dA_1 in direction dA_2 will be completely received by dA_2 as long as it is not absorbed in between. Because of this we get

$$L_1 d\omega_1 dA_1 = L_2 d\omega_2 dA_2. \tag{3.3}$$

The solid angles $d\omega_1$ and $d\omega_2$ for this transport are given as

$$d\omega_1 = \frac{dA_2}{r^2} \quad \text{and} \quad d\omega_2 = \frac{dA_1}{r^2}. \tag{3.4}$$

Inserting this into Equation 3.3 then yields $L_1 = L_2$. These properties of radiance describe that light will not be attenuated when traveling through unoccluded space. Thus, in order to render a scene it is sufficient to know the radiance in the direction of the viewer at each visible surface point.

Radiant flux area density is the total radiant energy falling on or leaving a surface point, per unit area. The unit of this quantity is $[W/m^2]$. Usually the incident total radiant energy (*irradiance*) is denoted E, whereas the emitted total energy is denoted B (*radiosity*) or (*radiant exitance*). They can be expressed in terms of radiance for opaque media as follows:

$$E(\vec{x}) = \frac{d\Phi_i}{dA} = \int_{\Omega^+} L_i(\vec{x}, \hat{\omega}_i) \cos\theta_i d\omega_i \tag{3.5}$$

$$B(\vec{x}) = \frac{d\Phi_o}{dA} = \int_{\Omega^+} L_o(\vec{x}, \hat{\omega}_o) \cos\theta_i d\omega_o \tag{3.6}$$

integrating the incident/outgoing radiance over Ω^+, the set of all directions in the hemisphere covering the surface at point $\vec{x}$.

The quantity, that is observed in each pixel when an image of a scene is taken, is the irradiance integrated over the pixel's field of view. This is closely related to the radiance reflected by the surface point visible in this pixel. Speaking of pixels as a point sample means observing radiance.

Intensity is the quantity describing the flux arriving or leaving with respect to a solid angle instead of an area. It is measured in $[W/sr]$:

$$I := \frac{d\Phi}{d\omega} \tag{3.7}$$

Point light sources are often described by their intensity. If a point light irradiates uniformly into all directions the intensity is $I = \Phi/4\pi sr$. The incoming radiance L_i at a point at distance r from the point light source is thus given as

$$L_i = \frac{I}{r^2}. \tag{3.8}$$

With these quantities we are able to describe the relationship between the radiant energy arriving at and leaving a surface.

3.3 The Rendering Equation

Given a scene description consisting of the geometry, the reflection properties and a specification of the light sources one can simulate the light distribution within an environment. Kajiya [Kajiya86] formulated the *rendering equation* as an energy equilibrium in a scene of arbitrary opaque surfaces:

$$L_o(\vec{x}, \hat{v}^{(g)}) = L_e(\vec{x}, \hat{v}^{(g)}) + \int_S f_r(\vec{x}, \hat{l}, \hat{v}) \cdot G(\vec{x}, \vec{y}) \cdot V(\vec{x}, \vec{y}) \cdot L_o(\vec{y}, \hat{l}^{(g)}) dA_y \quad (3.9)$$

It is an integral equation over the surfaces S, with the following definitions for the other quantities: $\hat{l}^{(g)}$ is the normalized light direction from $\vec{x}$ to $\vec{y}$ in world coordinates; $L_e(\vec{x}, \hat{v}^{(g)})$ is the radiance emitted from the surface at point $\vec{x}$ into the global viewing direction $\hat{v}^{(g)}$ and is only relevant for light sources. $f_r(\vec{x}, \hat{l}, \hat{v})$ stands for the reflection properties of the surface (further explained in the next sections) at point $\vec{x}$, that is the fraction of radiance incident from direction $\hat{l}$ that will be reflected in the outgoing direction $\hat{v}$. Both directions have to be specified in the point's local coordinate system spanned by the surface normal $\hat{n}$, the tangent and bi-normal. The formula also includes the visibility $V(\vec{x}, \vec{y})$ whose range is between 0 ($\vec{y}$ is not at all visible from $\vec{x}$) and 1 (there is no occluder). $G(\vec{x}, \vec{y})$ refers to the geometrical properties of the energy transport between $\vec{x}$ and $\vec{y}$, see Figure 3.2:

$$G(\vec{x}, \vec{y}) = \frac{\cos\theta_x \cos\theta_y}{\|\vec{x} - \vec{y}\|^2} \quad (3.10)$$

The integral in Equation 3.10 accounts for both *direct illumination* where the point $\vec{x}$ is illuminated directly by a light source and for *indirect illumination* since the incident radiance L_i incorporates light reflected by other surfaces in the scene. A slightly different formulation of the rendering equation concentrates more on the incident illumination at one point from the upper hemisphere Ω^+ and drops the direct relationship to other surfaces:

$$L_o(\vec{x}, \hat{v}^{(g)}) = L_e(\vec{x}, \hat{v}^{(g)}) + \int_{\Omega^+} f_r(\vec{x}, \hat{l}, \hat{v}) L_i(\vec{x}, \hat{l}^{(g)}) (\hat{n} \cdot \hat{l}^{(g)}) d\hat{l} \quad (3.11)$$

In this equation visibility is included implicitly since the radiance L_i impinging from one direction stems from the first surface that is visible in this direction. Dropping the emitting term $L_e(\vec{x}, \hat{v}^{(g)})$ one ends up with the so called reflectance equation.

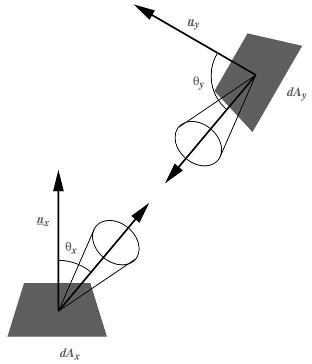


Figure 3.2: *Geometrical properties of the energy transport*

Based on Equation 3.10 a number of algorithms have been proposed to solve for *global illumination*, i.e., accounting for indirect illumination and other global effects. They can be grouped into two main categories:

Finite element methods subdivide the surfaces in the scene into patches (elements) and simulate the energy transport between sending and receiving patches. Representatives for finite element methods are for example radiosity [Heckbert92, Cohen93, Sillion94], or hierarchical radiosity [Hanrahan89] which solve the equation for diffuse surfaces only. Extensions of the radiosity method to work with glossy surfaces can be found in [Immel86, Cohen93].

Another class are so called *Monte Carlo algorithms* where basically a large number of rays is traced through the scene. The direction of the rays is derived from stochastical distributions imposed for example by the type of light source, the BRDF or other information like importance. Examples are distribution ray tracing [Cook84], bidirectional path tracing [Lafortune93], density estimation [Shirley95] and photon mapping [Jensen96].

Environment Maps

A simplistic approach to render an object in an environment is to use an *environment map* [Blinn76] representing fixed incident illumination $L_{env}(\hat{l}^{(g)})$. The environment is assumed to be at infinity and there is no light emitted or reflected by the object into the environment. Additionally, *interreflections* accounting for the energy exchange within the object itself are typically ignored:

$$L_o(\vec{x}, \hat{v}^{(g)}) = \int_{\Omega+} f_r(\vec{x}, \hat{l}, \hat{v}) V(\vec{x}, \hat{l}^{(g)}) L_{env}(\hat{l}^{(g)})(\hat{n} \cdot \hat{l}^{(g)}) d\hat{l} \quad (3.12)$$

The infinity assumption allows us to parameterize the incident radiance by the incident direction only. As a consequence parallax effects, where the incident light changes with the position on the surface, cannot be represented using a single environment map. We may however include a visibility term $V(\vec{x}, \hat{l}^{(g)})$ accounting for self-shadowing or shadowing by other nearby objects.

Environment maps without shadowing have been successfully applied in hardware-accelerated algorithms to render mirror reflections on surfaces. In the case of a perfect mirror the integral in Equation 3.12 is reduced to a lookup of the radiance incident from the mirror direction. For some glossy BRDFs it has been shown that Equation 3.12 can be precomputed for all outgoing directions resulting in prefiltered environment map [Miller84, Greene86, Cabral99, Heidrich99a, Kautz00a, Latta02, McAllister02]. Ramamoorthi and Hanrahan [Ramamoorthi01a, Ramamoorthi02a] have demonstrated how the integral can be solved efficiently for the case of a diffuse surface using spherical harmonics.

Point Light Sources

The evaluation of the rendering equation is further simplified if besides of ignoring interreflections, all light sources are restricted to be point light sources. In this case the so-called *local illumination* is evaluated as

$$L_o(\vec{x}, \hat{v}^{(g)}) = \sum_{j=0}^{n} f_r(\vec{x}, \hat{l}_j, \hat{v}) V(\vec{x}, \hat{l}_j^{(g)}) \cdot \frac{I_j}{r_j^2} (\hat{n} \cdot \hat{l}_j^{(g)}), \quad (3.13)$$

where I_j is the intensity of the j-th point light and r_j^2 is the squared distance from $\vec{x}$ to the light source.

3.4 Inverse Rendering

The rendering equation is most often used for synthesis of images of a correctly illuminated scene based on a complete description of the scene including the scene geometry, the definition of light sources, and a description of the reflection properties at the surface. The process may however be reverted: given a number of correct input images $L_{observed}$, derive the scene description, given as $\vec{x}, \hat{n}, L_i, f_r$, or parts of it based on the observations. This general process is called *inverse rendering*, and it can be roughly partitioned into three different categories. Each of the following categories can be characterized using the rendering equation since their solutions minimizes the error over all visible surface points between the observed radiance and the simulated radiance based on the scene description:

$$E(\vec{x}, \hat{v}) = L_{observed}(\vec{x}, \hat{v}) - \int_{\Omega+} f_r(\vec{x}, \hat{l}, \hat{v}) L_i(\vec{x}, \hat{l}^{(g)})(\hat{n} \cdot \hat{l}^{(g)}) d\hat{l}. \quad (3.14)$$

Photometric Stereo/Shape from Shading

The reconstruction of geometry from a number of photographs with known incident lighting is referred to as *photometric stereo*. It is a problem extensively investigated in the context of computer vision through the last decades [Woodham81, Horn86]. The geometry $\vec{x}$ is reconstructed by adjusting the normals $\hat{n}$ per pixel such that the shaded surface matches the input images. More precisely, photometric stereo yields partial spatial derivatives which are integrated to obtain a smooth surface. Frequently, the reconstructed surface is assumed to be diffuse, and lit by a point light source from several positions. Photometric stereo has also been applied to more complex reflection properties using reflectance maps [Tagare91, Nayar90, Klette98, Lin99]. The special case, where geometry is inferred from a single images is known as *shape from shading*.

Inverse Lighting

Another branch in the field of inverse rendering tries to recover the incident illumination L_i from observations. *Inverse lighting* recovers the intensity and/or position of discrete light sources [Schoeneman93, Kawai93] or tries to determine the incident light field from a distant environment with the same assumptions made in the case of environment maps [Marschner98, Marschner97, Nishino01, Ramamoorthi01b].

Inverse Reflectometry

In *inverse reflectometry* the goal is to measure the reflection properties f_r of the surfaces in the scene based on given geometry and lighting. All the problems solved in this course are centered around inverse reflectometry.

These three inverse problems are not necessarily treated separately. There are cases where more than one part of the scene description is unknown. Nishino et al. [Nishino01] and Ramamoorthi and Hanrahan [Ramamoorthi01b] reconstructed both the material properties and the incident lighting at the same time for a given geometric model.

3.5 Interaction of Light with Matter

In order to simulate light interacting with surfaces or to measure their appearance one has to represent the surfaces' reflective properties. A taxonomy of object appearance descriptions with different levels of abstraction is shown in Figure 3.3. The taxonomy is adapted from [Rusinkiewicz00].

In the most general case when light interacts with matter there is one photon striking the surface and one photon leaving the surface:

$$\begin{aligned}(x, y, \theta, \phi, t, \lambda)_{in} &\rightarrow (x, y, \theta, \phi, t, \lambda)_{out}, \quad \text{or} \\ (\vec{x}_i, \hat{\omega}_i, t_i, \lambda_i) &\rightarrow (\vec{y}_o, \hat{\omega}_o, t_o, \lambda_o) \qquad (3.15)\end{aligned}$$

Since each photon is described by six parameters (the position on the surface (x, y), the incident/outgoing direction (θ, ϕ), the time of interaction t and a specific wavelength λ), a 12D function is necessary to describe the general case.

To simplify this function one may drop the dependency on time assuming a constant appearance and ignoring the effect of *phosphorescence*. The photon is reflected instantaneously. A second simplification treats each wavelength independently assuming there is no *fluorescence*. The interaction with the material does not effect the wavelength of the photon. We obtain the generalized 9D *scattering function*. The discretization of the wavelength into bands saves another dimension, for example by representing only three color bands red, green, and blue. This eight dimensional function is also called *bidirectional scattering surface distribution function* (BSSRDF), see Nicodemus et al. [Nicodemus77].

Two different simplifications can be made to obtain a 6D function. If the effect of subsurface scattering is ignored, light is entering and leaving the surface at exactly the same position resulting in a spatially varying *bidirectional reflectance distribution function* (BRDF). The other simplification assumes a homogeneous material resulting in a six-dimensional BSSRDF.

Applying both simplifications yields a homogeneous BRDF parameterized by four dimensions. Restricting the reflections to be independent from rotations about the surface normal further eliminates one dimension. In traditional appearance measurement [Hunter87] qualitative measures of the reflection properties are sometimes represented by a one-dimensional slice through the 4D BRDF or by a small set of BRDF samples at specific incident and outgoing directions.

3.5.1 Gloss

A very restrictive approach is to represent the appearance only by the *gloss* measured for a surface. A lot of research in the field of appearance mea-

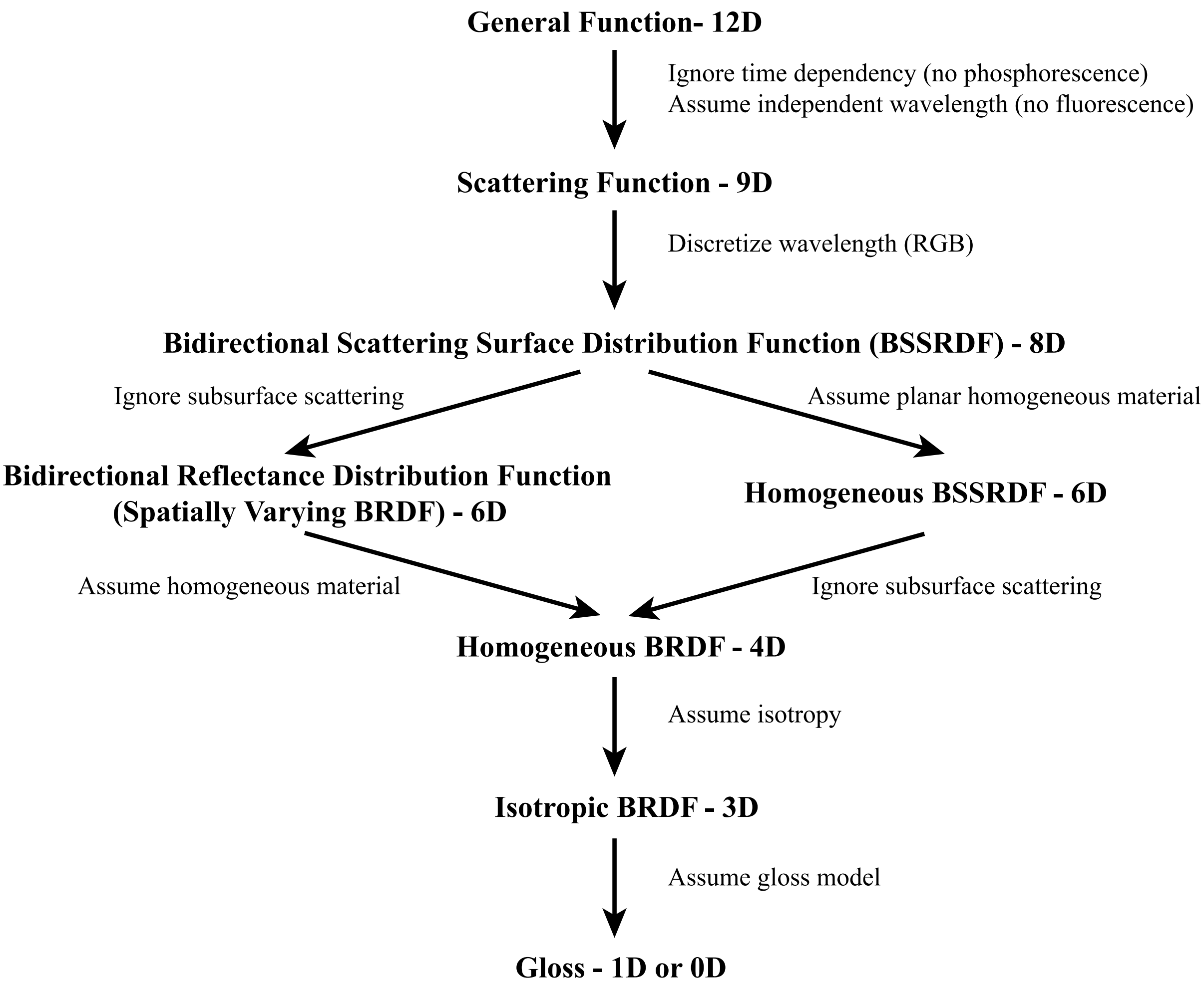


Figure 3.3: *Taxonomy of appearance representation adapted from [Rusinkiewicz00]. The dimension of the most general description of light interacting with matter is reduced by adding more and more assumptions resulting in manageable functions like the BRDF.*

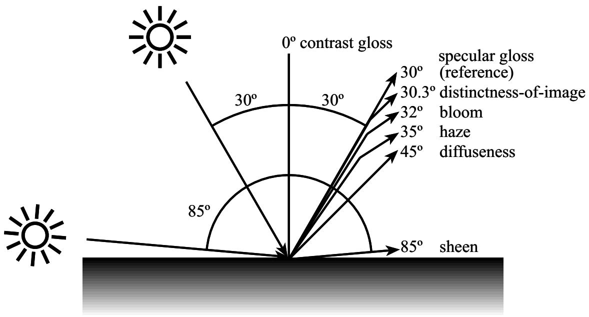


Figure 3.4: *Measurements of Gloss. Different kinds of gloss are measured as the reflected radiance at specific incident and outgoing directions.*

surement has been focused on the measurement and classification of individual materials for quality control and specification purposes – mostly in industrial applications. A lot of measures for individual aspects of surfaces have been derived and corresponding measurement tools have been developed. Due to their focus on single materials and specialized measurement devices these techniques are however not applicable to the acquisition of whole objects with varying surface properties. Hunter and Harold [Hunter87] present a detailed overview in this field.

Gloss is a qualitative measure to describe the specular appearance of a homogeneous material. Hunter and Harold distinguish several types of gloss, expressing different visual effects.

Specular gloss is measured for the exact mirror direction.

Sheen measures the shininess at grazing angles.

Contrast gloss or luster is a relative measure between specular reflecting areas and other areas.

Bloom and haze describe the milky appearance, adjacent to specular reflections.

Distinctness-of-image gloss measures the sharpness of mirror images, relevant for high-gloss surfaces.

In addition to the gloss one may express the *diffuseness* of a high-gloss surface by a measurement far of the mirror direction. Measurements of gloss are taken at specific incident and outgoing angles with respect to the surface normal, and many different measurement setups have been proposed (see [Hunter87]). One example set of specific angles is illustrated in Figure 3.4.

3.5.2 Bidirectional Reflectance Distribution Functions

A more complete model of the reflection properties of opaque surfaces is given by a *bidirectional reflection distribution function* (BRDF), which describes how incident light is reflected off the surface into all directions. The BRDF has the unit $[1/sr]$:

$$f_r(\vec{x}, \hat{\omega}_i \rightarrow \hat{\omega}_o) := \frac{dL_o(\vec{x}, \hat{\omega}_o)}{dE(\vec{x}, \hat{\omega}_i)} = \frac{dL_o(\vec{x}, \hat{\omega}_o)}{L_i(\vec{x}, \hat{\omega}_i) \cos\theta_i d\omega_i}. \quad (3.16)$$

It is defined as the quotient of the radiance L_o leaving the surface at point $\vec{x}$ in direction $\hat{\omega}_o$ and the irradiance arriving at $\vec{x}$ from direction $\hat{\omega}_i$. Since only reflections are described by this function $\hat{\omega}_i$ and $\hat{\omega}_o$ are directions within the same hemisphere above the surface.

In general, the BRDF is a 6D function: Two dimensions fix the location on the surface, representing spatially varying BRDFs. Homogeneous materials can be described by a 4D function $f_r(\hat{\omega}_i \rightarrow \hat{\omega}_o)$. Four dimensions are required to represent the incident and outgoing directions. They may be parameterized as $\hat{\omega}_i = (\theta_i, \phi_i)$ and $\hat{\omega}_o = (\theta_o, \phi_o)$, respectively. In the case of isotropic materials the BRDF actually simplifies to a 5D function $f_r(\vec{x}, \theta_i, \theta_o, \phi_0 - \phi_i)$. Quite often spatial variation is implicitly assumed and the alternative writing $f_r(\hat{\omega}_i, \hat{\omega}_o)$ is used even for isotropic materials.

In addition to the BRDF which only describes reflections, one must consider the *bidirectional* *transmission distribution function* (BTDF) f_t to simulate transparent media. The BTDF is similar to the BRDF but only considers radiance transmitted through the surface. Combining both functions forms the *bidirectional scattering distribution function* (BSDF), simply f, which of course is a function of all possible directions in the surrounding sphere; rather than just the hemisphere Ω^+.

Both the BRDF and the BSDF make the assumption that the reflected/transmitted light leaves the surface at exactly the same point where the incident light arrives. This is not true for surfaces that exhibit subsurface scattering. Depending on the scattering coefficients and on the scale the BRDF may however be a valid approximation to describe subsurface scattering materials. For highly translucent materials the eight-dimensional *bidirectional subsurface scattering distribution function* (BSSRDF) has to be used. It explicitly models the scattering based on different locations for the point of incidence and the point where the light leaves the surface.

3.5.3 Physical Properties

In order to be physically plausible a BRDF must fulfill two important constraints: the Helmholtz reciprocity and energy conservation [Beckmann63]. Some of the effects described by a BRDF or BSDF are implicitly given by the following three principles: reflection at planar surfaces, Snell's law that determines the direction of refracted light and the Fresnel formulae which determine the amount of energy that will be reflected or refracted respectively.

Helmholtz reciprocity

Helmholtz [v. Helmholtz25] stated that the role of incident and reflected energy may be reversed. If a photon incident from $\hat{\omega}_i$ is reflected or scattered into direction $\hat{\omega}_o$ with some probability, the same probability holds for a photon incident from $\hat{\omega}_o$ reflected into direction $\hat{\omega}_i$. One may swap incident and outgoing direction and still obtain the same value of the BRDF:

$$f_r(x, \hat{\omega}_i \to \hat{\omega}_o) = f_r(x, \hat{\omega}_o \to \hat{\omega}_i). \tag{3.17}$$

Energy Conservation

The reflection at a surface must always be energy conserving since real physical materials never propagate more light than they receive, but may well dissipate some energy:

$$\int_{\Omega^+} f_r(x, \hat{\omega}_i \to \hat{\omega}_o) \cos\theta_i d\omega_o \quad \leq 1 \quad \forall \quad \hat{\omega}_i \in \Omega^+. \tag{3.18}$$

Snell's Law

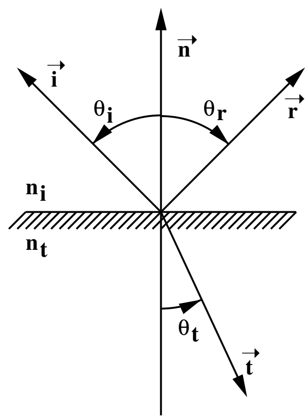


Figure 3.5: *Light ray reflection and refraction at an ideally smooth surface.*

Snell's law determines the direction of a wave or a ray that is refracted after penetrating the interface between two transparent media with different optical densities. When light travels through a medium denser than vacuum, its velocity decreases. For a specific material the index of refraction $n(\lambda)$ denotes the quotient of speed of light in vacuum c and the speed of light in this medium v_λ depending on wavelength λ:

$$n(\lambda) = \frac{c}{v_\lambda}. \tag{3.19}$$

Figure 3.5 depicts how a ray of light traveling along direction $\vec{i}$ hits a surface that separates two materials with indices of refraction n_i and n_t. The incident ray results in two new rays: a reflected one in direction $\vec{r}$ and a transmitted one in direction $\vec{t}$. Assuming a perfectly smooth surface, all three rays are in the same plane. The direction of the reflected ray is easily computed:

$$\vec{r} = 2 < \vec{i} \mid \vec{n} > -\vec{i}. \quad (3.20)$$

In order to compute the direction of the refracted ray, the angle θ_t between this ray and the surface normal can be derived from the incident angle θ_i by Snell's law:

$$n_i(\lambda) \sin\theta_i = n_t(\lambda) \sin\theta_t. \quad (3.21)$$

With this angle it is possible to calculate the direction $\vec{t}$ (see [Glassner95]):

$$\vec{t} = -\frac{n_i}{n_t}\vec{i} + \vec{n}\left(\frac{n_i}{n_t}\cos\theta_i - \cos\theta_t\right). \quad (3.22)$$

The Fresnel Formulae

Since the incoming ray results in a reflected and a transmitted component on a surface of a transparent medium it is important to know how the energy is split between them. To describe this, the *reflectance* ρ is defined as the ratio of reflected flux to incoming flux. The *transmittance* or *transmission* τ denotes the ratio of transmitted to incoming flux.

$$\rho = \frac{d\Phi_r}{d\Phi_i} \quad \text{and} \quad \tau = \frac{d\Phi_t}{d\Phi_i}. \quad (3.23)$$

A third quantity that must be considered for this interaction is the *absorption* α which is the fraction of flux that is absorbed by the surface. Because we do not consider emitting surfaces, the three quantities always sum up to 1: $\rho + \tau + \alpha = 1$.

Fresnel formulae can be used to compute ρ and τ for given angles θ_i and θ_t. These formulae specify the ratios of the reflected or transmitted amplitude to the incoming amplitude of a polarized electrical field (see [Born93, Glassner95, Heidrich99b]). Let $r^\perp$ and $r^\parallel$ be the reflected amplitude ratios of the field perpendicular and parallel to the plane spanned by $\vec{i}$ and $\vec{n}$. Accordingly, $t^\perp$ and $t^\parallel$ denote the ratios of the transmitted amplitudes. Those can be calculated as

$$r^\perp = \frac{n_1\cos\theta_i - n_2\cos\theta_t}{n_1\cos\theta_i + n_2\cos\theta_t}, \quad (3.24)$$

$$r^\parallel = \frac{n_2\cos\theta_i - n_1\cos\theta_t}{n_2\cos\theta_i + n_1\cos\theta_t}, \quad (3.25)$$

$$t^\perp = \frac{2n_1\cos\theta_i}{n_1\cos\theta_i + n_2\cos\theta_t} \quad (3.26)$$

$$t^\parallel = \frac{2n_1\cos\theta_i}{n_2\cos\theta_i + n_1\cos\theta_t}. \quad (3.27)$$

Because ρ and τ are ratios of radiant flux, they are proportional to the square of the amplitude of the electrical fields $r^\perp, r^\parallel, t^\perp$ and $t^\parallel$. In order to obtain ρ and τ for unpolarized light we have to average the perpendicular and the parallel part:

$$\rho = \frac{(r^\perp)^2 + (r^\parallel)^2}{2}, \quad \text{and} \quad (3.28)$$

$$\tau = \frac{n_2\cos\theta_i}{n_1\cos\theta_t} \cdot \frac{(t^\perp)^2 + (t^\parallel)^2}{2} = 1 - \rho - \alpha. \quad (3.29)$$

In Figure 3.6 the reflectance off a surface between air and glass and the transmittance from glass to air is plotted. Note that when light travels trough a material n_1 and hits a surface of some medium with lower index n_2 ($n_1 > n_2$), like from glass to air, the transmittance τ will be zero if the incident angle θ_i exceeds the critical angle θ_c. This critical angle is defined as the angle where the incident ray will be refracted in a direction parallel to the surface. Thus it is given by:

$$n_1 \sin\theta_c = n_2 \sin\frac{\pi}{2}, \quad \text{that is} \quad \theta_c = \arcsin\frac{n_1}{n_2}. \quad (3.30)$$

For angles greater than θ_c there will be no refracted ray and all energy will be transported by the reflected ray.

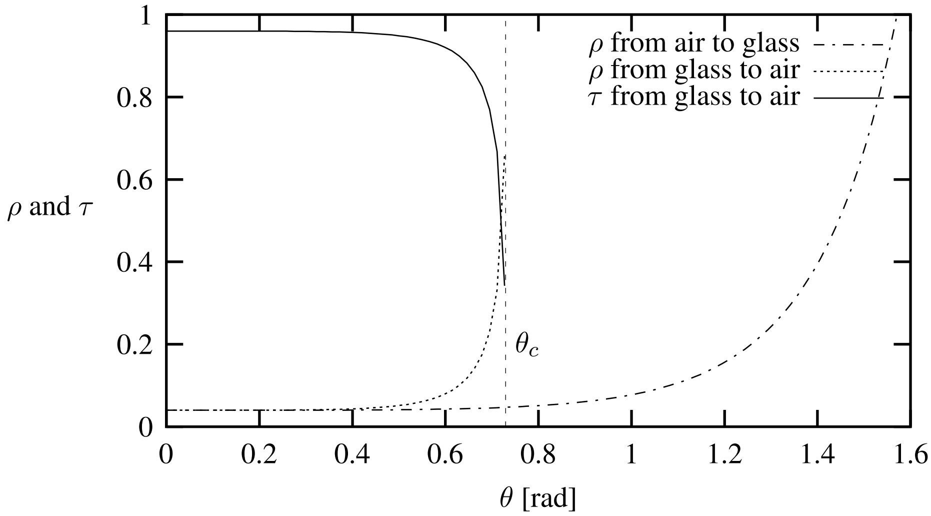


Figure 3.6: *Reflectance* ρ *and transmittance* τ*, determined by Fresnel's formulae at a surface that separates air* $(n \approx 1)$ *and glass* $(n = 1.5)$*. The critical angle is* $\theta_c \approx 0.73$.

Fresnel's formulae are able to describe why more light is reflected for larger viewing angles. At grazing angles less energy is refracted while the reflected portion increases. Since this effect can be observed at many surfaces a Fresnel factor is included into some BRDF reflectance models (see Section 3.7). For more efficient evaluation Schlick et al. [Schlick94] propose to use the simpler approximation based on the spectral distribution f_λ of the Fresnel factor at normal incidence, corresponding to the reflected color of white light. It is expressed with respect to the halfway vector $\vec{h}$ between the viewing and lighting direction:

$$F_\lambda(\hat{h} \cdot \hat{v}) = f_\lambda + (1 - f_\lambda)(1 - (\hat{h} \cdot \hat{v}))^5 \quad (3.31)$$

3.6 Reflection Properties

While the BRDF is a general way to represent the reflection properties of opaque surfaces, we classify the different properties and list phenomena commonly observed. The reflection properties depend on the materials the object is made of (e.g, metal vs. plastic), on the surface structure (e.g, rough vs. polished), and on whether they are homogeneous or heterogeneous. These attributes form three almost orthogonal classification schemes. Figure 3.8 demonstrates the appearance various different materials.

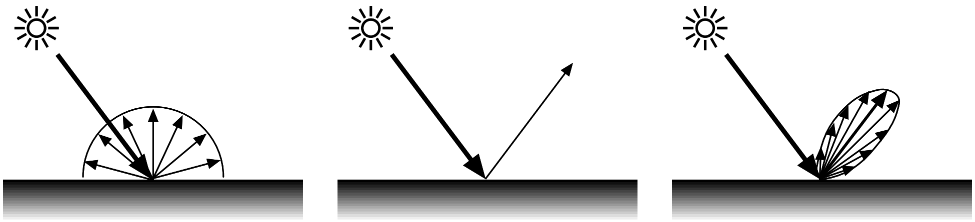
3.6.1 Materials

According to Hunter and Harold [Hunter87] objects may be classified into four groups based on the dominant way incident light is distributed, while the other ways may be present as well:

material	dominant distribution
opaque nonmetals	diffuse reflection
metallic surfaces	specular reflection
translucent objects	diffuse transmission
transparent objects	regular transmission

Surface Reflections

At *opaque* surfaces light is reflected directly at the surface. Examples for opaque materials are metals, chalk, and to some extent wood and plastic. There is no significant amount of light penetrating



(a) diffuse (b) mirror (c) glossy

Figure 3.7: *Qualitative description of reflection at opaque surfaces*

an opaque surface. A light beam illuminating an opaque surface from one direction is partially absorbed and the rest is reflected and scattered at the surface and distributed into many outgoing directions. One may think of the scattered distribution as composed by three qualitatively different components.

Diffuse reflections. Some portion of light is scattered equally into all directions (Figure 3.7a) independent on the incident direction. This diffuse or *Lambertian* [Lambert60] reflection is view-independent, i.e., the illuminated surface looks the same for any viewing direction. An almost perfectly diffuse material is chalk.

Specular reflections. Many other materials exhibit additional view-dependent reflections. Highly polished or varnished surfaces show *mirror* or *specular* reflections (Figure 3.7b).

Glossy reflections. If the surface is not perfectly smooth, light will be scattered around the mirror direction, called *glossy* reflection (Figure 3.7c). The roughness of the surface determines how broad a glossy highlight will appear. In the computer graphics literature the term specular is often used to describe the entire range of view-dependent reflections including glossy reflections. The shape of the directional distribution of glossy reflections lead to the term “specular lobe”.

One important optical effect of glossy reflections is that they increase towards grazing angles. The ratio of reflected to refracted light across a planar surface drastically changes near grazing angles as it is described by the Fresnel formulae (see Section 3.5.3). As Marschner [Marschner98] pointed out, this increase at glancing angles of incidence also causes the phenomena of off-specular peaks. Near grazing angles one can observe that more light is reflected into a direction close to the mirror direction rather than into the mirror direction itself.

While diffuse reflections typically alter the color or frequency distribution of the incoming light, this is not necessarily true for specular reflections. Only *metals* cause colored highlights. Specular reflections at normal nonmetallic materials (*dielectrics*) typically reflect the color of the light source.

Pigment Particles

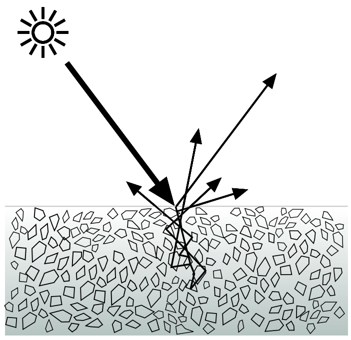


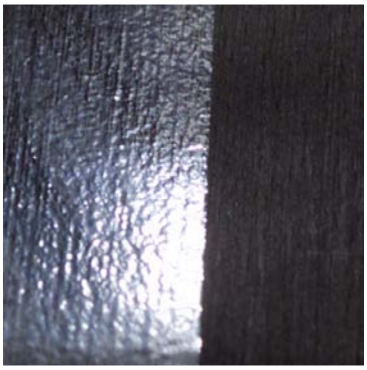
Figure 3.9: *Pigment Particles. Many surfaces obtain their diffuse color from pigment particles. Light is either reflected directly at the surface of the embedding material causing a white highlight or penetrates the surface and is diffused and filtered by the particles.*

(a)

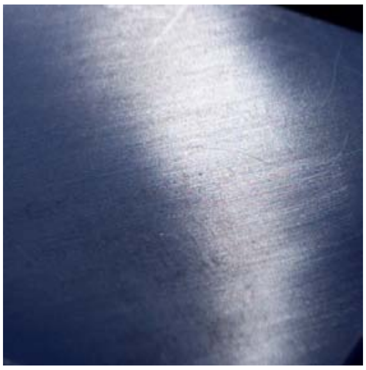
(b)

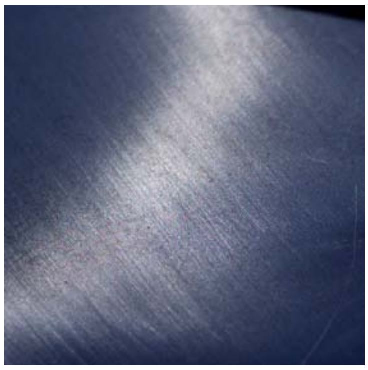
(c)

(a) Diffuse Reflection. No view-depedent highlight is visible. (b) Mirror Reflection. A clear image of the light source is reflected. (c) Glossy Reflection. An extended highlight region caused by a glossy surface.



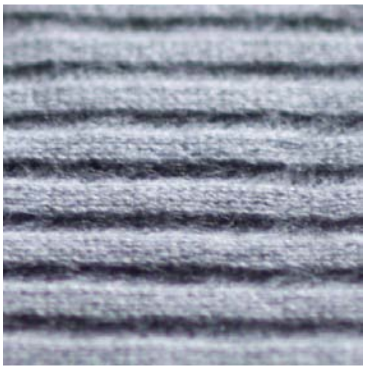
(d) Layered Materials. A layer of varnish drastically changes the reflection properties.





(e–f) Anisotropic Reflections. The shape of highlights caused by anisotropic materials change when the surface is rotated around around the surface normal.

(g) Subsurface Scattering. Light is shining through translucent alabaster.



(h–i) Volumetric Structures. Volumetric representations are required to model the appearance caused by highly complex geometry.

Figure 3.8: *The appearance of surfaces depends on the material and on the surface structure. In (a–f) camera and light source are at approximately the same position with respect to the surface.*

A set of materials is composed by a supporting substrate into which colored pigment particles are embedded. The substrate forms the actual surface causing specular reflection (with white highlights). The amount of light that is reflected directly at the surface depends on the smoothness of the surface, the refractive index of the material and the angle at which the beam strikes the surface (see Fresnel's formulae). The refracted ray changes its direction due to Snell's law and then encounters the surfaces of the pigmented particles where the light is again partially reflected and refracted. The numerous interactions with the particles produce an isotropic distribution of scattered light resulting in diffuse reflection. The pigments determine the diffuse color of the material acting as a filter on light that penetrates the pigments. The process of diffusing the light is called *subsurface scattering*.

Many colored plastics are composed this way and the pigments are typically at microscopic scale, too small to be distinguished. In car paint often larger particles are included, for example to obtain a slightly heterogeneous metallic look.

If the scattering of light within the material is restricted to a sufficiently small region with respect to the scale of the object, materials with pigment particles may still be observed as opaque surfaces and the representation using a BRDF is valid. Otherwise, they need to be described in terms of a BSSRDF.

Translucent/Transparent Materials

A class of materials, which is similar to pigmented materials are *translucent* materials. However, the light path through the material is much longer in translucent materials. This results in the point of incidence and the exit point being significantly separated (see Figure 3.10). Light is shining through a translucent object diffusing the incident light. Many natural materials are translucent, e.g., skin, flesh, milk, fruit, or vegetables. The appearance of materials with significant subsurface scattering and appropriate rendering algorithms have

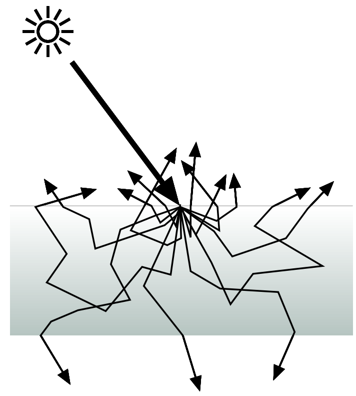


Figure 3.10: *Subsurface Scattering. For translucent materials light penetrates the surface and is scattered (multiple times) inside the material before it leaves the material probably at some other position.*

been investigated in [Hanrahan93, Koenderink96b, Koenderink01, Jensen01, Lensch02b].

If no scattering takes place or the number of scattering events is negligible we have a *transparent* material. The refracted light is directly transmitted through the object, as for example by glass.

Layered Materials

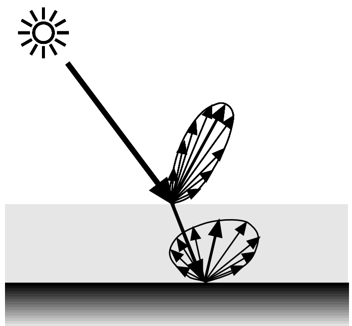


Figure 3.11: *Layered Materials. The appearance of layered materials is determined by the reflection properties at the different surfaces.*

As the name suggests, layered materials consists of layers of different materials. Materials formed by multiple layers show additional effects if at least the top level is partially translucent. Typical examples are coated surfaces, like varnished wood or a thin film of oil on skin. In a simple case the varnish seals a rough surface and produces more glossy reflections. One may think of combining the two kinds of reflections as illustrated in Figure 3.11. The overall appearance is influenced by the optical properties of the layers, their thickness and the roughness of the interfaces. Depending on the thickness of the layers interferences are noticeable amplifying or damping particular wavelengths, e.g., on an oily water surface.

3.6.2 Surface Structure

The second classification is on the surface structure. Besides the material itself, the directional distribution of the reflected light is also dependent on the structure of the surface. A smoother surface results in sharper highlights than a rough surface. The surface structure may influence the diffuse reflectance, but its main influence is on the specular reflections.

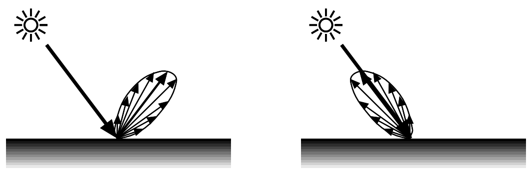
Microfacets

The structure of many rough surfaces can be described as a collection of a large number of so-called *microfacets* which are too small to be resolved (see Figure 3.12). Each microfacet acts as a mirror and may possess an arbitrary orientation. Given an incident light direction, the reflected radiance in any direction is proportional to the number of microfacets in a mirroring orientation, i.e., to the number of microfacets that are oriented such that the outgoing direction is the mirrored incident direction. They must be oriented in the direction of the halfway vector $\hat{h}$ between the incident $\hat{l}$ and outgoing direction $\hat{v}$: $\hat{h} = \frac{\hat{l}+\hat{v}}{2}$. The distribution of normals of the microfacets thus determines the direction distribution of the reflected light. For a continuous rough surface *masking* and *shadowing* are important since not all microfacets are visible or lit when looking at the surface from grazing angles.

Anisotropy

If the distribution of microfacets shows a preferred direction within the tangential plane of the surface we obtain *anisotropic* reflection properties. For anisotropic materials the reflection changes when the surface is rotated around its normal, which is in contrast to *isotropic* materials. The preferred direction of a highlight is caused by oriented structures like grooves or bumps prolongated in one direction. In this case, the reflection will be different when the light shines perpendicular or parallel to the preferred direction. The orientation of the structures may be the result of processing the surface, e.g., brushed metal, or the structures may be due to a growth process, e.g., on wood.

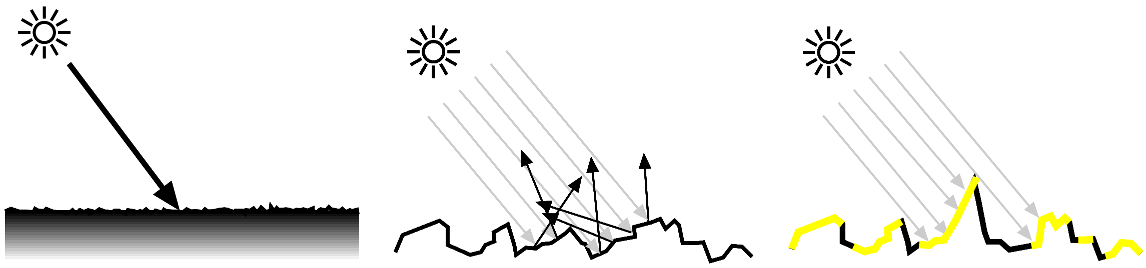
Forward/Backward Scattering



(a) forward scattering (b) backward scattering

Figure 3.13: *Some materials reflect most of the light back into the direction of incidence*

The direction of the specular lobe is for most materials oriented roughly in the mirror direction. When viewed from the top, the specular lobe is forward oriented with respect to the lighting direction (*forward scattering*). There are however materials that are *backward scattering*. Most energy is reflected back into the direction of the incoming light by backward scattering materials (Figure 3.13d). Such reflection behavior is exhibited by some porous surfaces or some crystal structures. On a macroscopic level, backward scattering can be observed in the eyes of some night active ani-



(a) rough surface (b) zoom-in (c) shadowing and masking

Figure 3.12: *Microfacet Model. (a) A rough surface may be seen as composed by small mirroring microfacets. (b) Incident illumination is scattered based on the normal distribution of the microfacets. (c) Shadowed or occluded microfacets (black) do not contribute to the reflection.*

mals. Some traffic signs and cinema screens have a backward scattering layer to make better use of the incident light.

3.6.3 Spatial Variation

In reality most surfaces are not perfectly homogeneous. Often different materials are involved, so appearance varies across the surface. This is the case for a painting or for wood, or patina sedimented on weathered surfaces [Dorsey96]. Even though an object may consist of a homogeneous material the surface structure typically adds a spatially varying reflection behavior, where subtle changes are noticeable. Realistic appearance can only be achieved if this spatial variation in the reflection properties is modeled.

Representing Surface Structure

Depending on the the resolution of the measurements performed the surface structure may be integrated in the measured BRDF or has to be modeled explicitly. Larger structures alter the surface normal and add a specific appearance due to shadowing and masking. Depending on the size, they may be seen and modeled as real surface geometry, or with *displacement maps*. If they are smaller so called *normal* or *bump maps* may be applied. With normal maps, a curved surface is approximated by a planar one and surface detail are represented by a per-pixel normal which is used to evaluate the shading at the surface (see Figure 3.14).

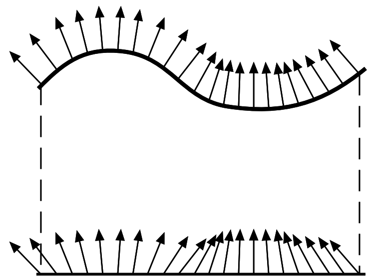


Figure 3.14: *Normal Map. The characteristic shape of a curved surface is transformed into a normal map defined on a flat surface. At each texel the original surface normal is stored.*

Volumetric Structures

So far we have assumed that there is a locally planar surface where the interaction of light with the material takes place. The reflection properties of many soft materials like textiles or fur are due to a number of fibers or hair populating a volume around the actual surface geometry. Self-occlusion and shadowing are major issues concerning the appearance of those volumetric structures. Volumetric structures are currently hard to model and require specialized algorithms for interactive rendering [Meyer98, Lengyel01, Daubert01, Lensch02a, Daubert02].

3.6.4 Hierarchy of Detail

Surface structures can be identified at various levels of detail. Fournier and Lalonde [Fournier00] introduce a hierarchy of detail: If surface features are too small to be individually resolved from the current observation they are at the *microscopic* level. Their influence on the appearance of an object is integrated into the BRDF. At the *mesoscopic* level the structures are discernible but small enough to be represented by bump maps [Koenderink96a]. The actual geometric shape of an object then forms the *macroscopic* level. Depending on the distance of the observer and the spatial sampling rate the boundaries between these three levels are floating.

Measuring the BRDF of an object from a larger distance may integrate surface features, texture and geometric structure, into the BRDF. The features may be resolvable if the measurements are performed from a closer distance, probably resulting in a different BRDF. Thus all we will measure is an *apparent* BRDF based on the observations with a specific sampling density.

3.7 Representations of Reflection Properties

Reflection properties of opaque surfaces are fully described by their BRDFs. We will now list different approaches to represent and to model BRDFs. The BRDF models proposed range from raw sampled data over representations using general functions like spherical harmonics to explicit formulae which resemble specific effects.

3.7.1 Tabulated BRDFs

The most straight forward representation of the reflection properties is to store a BRDF sample for a discrete set of directions in tabulated form [DeYoung97]. The necessary data can be obtained from simulations [Heidrich00] or real measurements [Ward Larson92, Dana99, Marschner99]. In order to evaluate the BRDF for a given incident and outgoing direction, the tabulated entries are interpolated. Problems arise in undersampled regions of the 4D space of directions. Tabulated BRDFs are very memory intensive and are often used as intermediate data to derive a more compact representation by fitting.

3.7.2 Factorization and Spherical Harmonics

A general approach to reduce the data is to transform it into a lower dimensional function space by *factorization*. Kautz and McCool [Kautz99] factorized the BRDF into a sum of one or two-dimensional textures. The numerical decomposition was further improved by McCool et al. [McCool01]. Latta and Kolb even include the incident lighting into the factorization [Latta02].

Basis transformations into *spherical harmonics* are suited to model smoothly varying data. Spherical harmonics have been applied to represent exit radiance [Sillion91] and BRDFs [Cabral87, Ramamoorthi01b]. Since convolution of a BRDF with the incident illumination can be expressed by a simple dot product in spherical harmonics they are particularly attractive for interactive rendering algorithms, simulating objects with complex BRDFs in complex environments [Ramamoorthi01a, Ramamoorthi02a, Sloan02, Kautz02].

3.7.3 BRDF Models

The most compact representation is by BRDF models where the reflectance for a pair of directions is obtained by evaluating a formula depending on a small number of parameters. The reflection properties of a homogeneous material are summarized by just one set of parameters. The parameters are either determined by fitting to measured data or they are adjusted manually in order to approximate the desired material. In addition to their compactness BRDF models fitted to measured data provide a reasonable approximation even for directions which have not been sampled during the measurements.

Empirical Models

Empirical models are typically not physically motivated, but provide a class of functions that can be used to approximate BRDFs. One of the earlier lighting models in the field of computed graphics has been proposed by Phong [Phong75] which consists of a diffuse term and one specular lobe. Unfortunately, this empirical model is neither energy conserving nor reciprocal and is only well-suited to approximate plastic-like materials. Lewis corrected these problems [Lewis93] and Blinn [Blinn77] adopted the model for more realistic reflections. Here, we present the Blinn-Phong Model [Blinn77] which computes the specular component based on the halfway vector $\hat{h}$:

$$f_r(\hat{l}, \hat{v}) = \frac{\rho_d}{\pi} + \frac{\rho_s}{(\hat{n} \cdot \hat{l})}(\hat{h} \cdot \hat{n})^N. \qquad (3.32)$$

The reflection properties are modeled by three parameters only: a diffuse coefficient ρ_d, a specular coefficient ρ_s and a specular exponent N.

A more expressive model based on the Phong model has been presented by Lafortune et al. [Lafortune97]. It can handle off-specular peaks, backscattering and even anisotropy:

$$f(\hat{l}, \hat{v}) = \frac{\rho_d}{\pi} + \sum_i [C_{x,i}(l_x v_x) + C_{y,i}(l_y v_y) + C_{z,i} l_z v_z]^{N_i}. \qquad (3.33)$$

The specular part is formed by a combination of several specular lobes. Anisotropy is achieved by different values of $C_{x,i}$ and $C_{y,i}$. The lobe is forward-reflective if $C_{x,i}$ is negative and retro-reflective if $C_{x,i}$ is positive. The off-specularity of the lobe is expressed by the ratio between $C_{x,i}$ or $C_{y,i}$ and $C_{z,i}$.

Microfacet Models

Other models derive the specular part supposing a distribution of microfacets forming the surface. In the Torrance-Sparrow model [Torrance67] the specular component is influenced by the distribution of normal directions of the microfacets. The model further accounts for shadowing and masking with respect to the normal distribution function and includes a Fresnel term for off-specularity. While the Torrance-Sparrow model assumes a Gaussian distribution of microfacets, Ashikhmin [Ashikhmin00] derived a shadowing and masking term for arbitrary distributions.

A model for anisotropic reflections has been proposed by Ward [Ward Larson92]. He assumes the normals to be distributed following an elliptical Gaussian. It is an empirical BRDF model without a shadowing term or a Fresnel term:

$$f_r(\hat{l}, \hat{v}) = \frac{\rho_d}{\pi} + \rho_s \frac{1}{\sqrt{(\hat{l} \cdot \hat{n})(\hat{v} \cdot \hat{n})}} \frac{1}{4\pi\alpha_x\alpha_y} \cdot \exp\left(-2\frac{\left(\frac{\hat{h}\cdot\hat{t}}{\alpha_x}\right)^2 + \left(\frac{\hat{h}\cdot\hat{b}}{\alpha_y}\right)^2}{1 + \hat{h}\cdot\hat{n}}\right) \qquad (3.34)$$

The anisotropy is indicated by different values for α_x and α_y defined along the tangent $\hat{t}$ and binormal $\hat{b}$ at the surface point. Ward has shown that his model approximates measured data like brushed aluminum sufficiently well.

To describe anisotropic reflections Poulin and Fournier [Poulin90] model the surface as being composed by oriented cylindrically shaped grooves or bumps. They also derived a shadowing and masking term for this structure. The model by Banks [Banks94] emulates fibers defined by a gradient direction per pixel.

Other physical effects like interference with the surface microstructure are included in the BRDF model by He et al. [He91]. BRDF Models based on single scattering in subsurface scattering materials have been presented by Hanrahan and Krueger [Hanrahan93] and later by Koenderink et al. [Koenderink96b]. Various other BRDF models have been developed in the fields of computer vision, or astronomy.

Chapter 4

Taxonomy of Acquisition Approaches

Hendrik P. A. Lensch

The goal of measuring the reflection properties of materials, surfaces and objects has lead to a number of different approaches. While a dense sampling of the general 8D reflectance field is still out of reach for current systems, a number of very successful acquisition techniques have been developed which reduce the number of dimensions and the sampling density by restricting themselves to either specific classes of materials, or to specific illumination and viewing scenarios.

A large class of acquisition techniques deal with capturing the four-dimensional BRDFs of homogeneous surfaces, often performing highly precise measurements, as explained in Chapter 6. We will treat the special case of reflection off homogeneous fibers in Chapter 10.

The approaches that concentrate on spatially varying surfaces or entire objects can roughly be sorted by the taxonomy given in Figure 4.1. The simplest representation for a spatially varying materials is a diffuse texture map. Here, view and light dependent effects cannot be represented at all, i.e., relighting of diffuse texture maps can only result in a rough approximation of the original material. Light fields cannot be re-lit as well, although they store view-dependent data and thus may exhibit moving highlights. A light field contains only information about the appearance of the object in one particular environment: the illumination that was present during capturing.

Relightable representations are best characterized starting from the 8D reflectance field or BSSRDF which is the most general form. In order to reduce the dimensions to six, spatially varying BRDFs make the assumption that the appearance can be described by a per-texel BRDF (see Chapter 7) restricting the acquisition to opaque materials. The BRDFs are represented by a per-texel set of parameters of a BRDF model. A further constraint is that interreflections are not included in this representation. They have to be added afterwards by a global illumination computation. Often the variability of the BRDF is restricted to the diffuse part only, while the specular part remains constant. 6D reflectance fields (Chapter 12) and BTFs (Chapter 11) are far less restricted. They only assume that the incident illumination originates from a far away light source, and otherwise contain all global lighting effects such as self shadowing, interreflections, etc. The effects of projected illumination patterns, such as cast shadows, or a light source that is very close are however difficult to achieve since the necessary information is not contained in the data. These effects due to spatially varying illumination can be modeled if the reflectance is captured with respect to an arbitrary incident light field, which has so far been demonstrated for a single view point

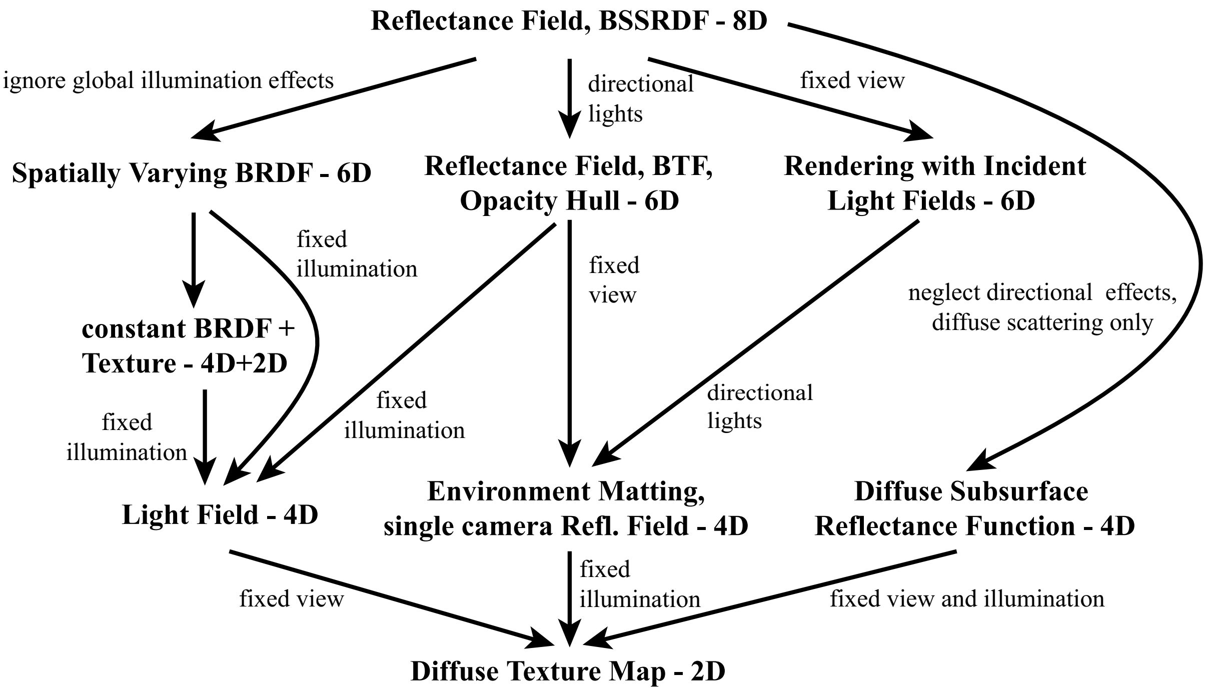


Figure 4.1: *Taxonomy of Appearance Acquisition Techniques.*

only [Masselus03, Sen05]. The reflectance field from a single camera requires a 4D data set but of course restricts the view to the originally chosen perspective. Environment mattes (Chapter 9) provide a very high sampling with regard to the incident light directions and are therefore particularly suited for highly specular and transparent materials. Opacity hulls combine the high sampling density with view dependent information. If one concentrates on translucent materials for which the subsurface light transport is more or less diffuse, and the dependence on the specific light and viewing direction can be neglected, a 4D representation of the diffuse subsurface reflectance function is sufficient to capture all effects due to spatially varying illumination (see Chapter 8).

Depending on the selected dimensions, the sampling rate for each dimension, and the envisioned dependency on spatially varying illumination, quite some different equipment must be used to capture the data (see Chapter 5). In addition, the various approaches require drastically different numbers of images or illumination patterns to capture the full data set. Table 4.1 summarizes some of the specific aspects of the mentioned acquisition approaches.

Technique	Acquisition Environment	#Images	View Res.		Light Res.		Comment
			spatial	directional	spatial	directional	
texture mapped geometry	diffuse illumination	10-20	hi	-	-	-	- hard to remove highlights
light fields	arbitrary environments	100-500	hi	med	-	-	+ geometry independent - ghosting
surface light fields	arbitrary environments	100-500	hi	med	-	-	+ better quality through explicit geometry
spatially varying BRDFs	various setups	10-40	hi	hi	hi	hi	- implicit BRDF model - restrictions on types of materials - no global illumination effects
BTFs	automated point light and surface orientation	500-10000	hi	med	-	med	+ arbitrary materials - flat samples only
reflectance fields	automated point light	2000-10000	hi	lo	-	med	+ geometry independent + arbitrary materials
relighting with incident light fields	moving projector	50000-100000	hi	-	med	lo	- very time consuming
environment mattes	monitors	1 - 1200	hi	-	-	hi	+ good on transparent and mirroring objects
opacity hulls	automated point light and object orientation monitor	20000-60000	hi	med	-	hi	+ arbitrary complex geometry through view-dependent geometry
diffuse subsurface scattering	automated laser and object rotation	500000-1000000	hi	-	hi	-	- restricted to translucent objects - no directional dependence

Table 4.1: *Comparison of Appearance Acquisition Techniques. The quality of the representation and the visual effects that can be reproduced is correlated to the number of images that have to be acquired, and thus to the acquisition time. The approaches differ with regard to the resolution of the incident and outgoing light field for which we distinguish between spatial and the directional resolution with regard to the objects surface.*

Chapter 5

Acquisition Tools and Systems

Michael Goesele

In this chapter, we describe some basic hardware and software that are often used to perform the measurements of real world materials and objects. This includes 3D scanning systems and the processing software used to acquire geometry models. We furthermore summarize basic properties of digital camera systems and discuss their geometric and photometric calibration. Finally, we briefly introduce lighting systems used for acquisition and point out the importance of a well designed measurement environment.

Although these tools are normally not the main focus of an acquisition system, their performance and accuracy is crucial for the quality of the results. We therefore include also some examples of measurements and benchmarks for parts of the described equipment.

5.1 Geometry Acquisition

A geometry model of a target object can be created using a variety of methods including hand modeling, tactile methods, active and passive stereo systems. We will here describe the basic operating principle of a laser range scanner such as the Minolta VI-910 system. This active optical range scanning system sweeps a laser line over the target object and observes the reflection of the line with a video camera. The position of the reflected laser line in the recorded image depends on the geometry of the setup and the distance of the illuminated points on the target object from the 3D scanner (see Figure 5.1). The laser line defines a plane P in space whereas each pixel in an image captured by the video camera defines a ray $\vec{r}$ in space. If the reflection of the laser line on the target object is visible in a pixel, the intersection of P and $\vec{r}$ yields a single 3D point on the surface of the target object. A single complete scan returns a depth map where each valid pixel contains a distance value from the 3D scanner to the object.

5.1.1 Performance and Accuracy

The underlying principle of active triangulation-based optical range scanners is the stereo vision principle [Klette98]. (The laser projector could be replaced by a second camera leading, however, to the – in general difficult – stereo correspondence problem.) A surface point can therefore only be scanned successfully if it is visible from both, the position of the laser projector and the position of the video camera. Concave objects may lead to occlusions which often result in holes in the depth maps and are therefore hard to scan. The larger the baseline (the distance between laser projector and video camera) of a triangulation-based 3D scanning system, the more difficult concave objects are to scan. There is however a trade-off as a larger

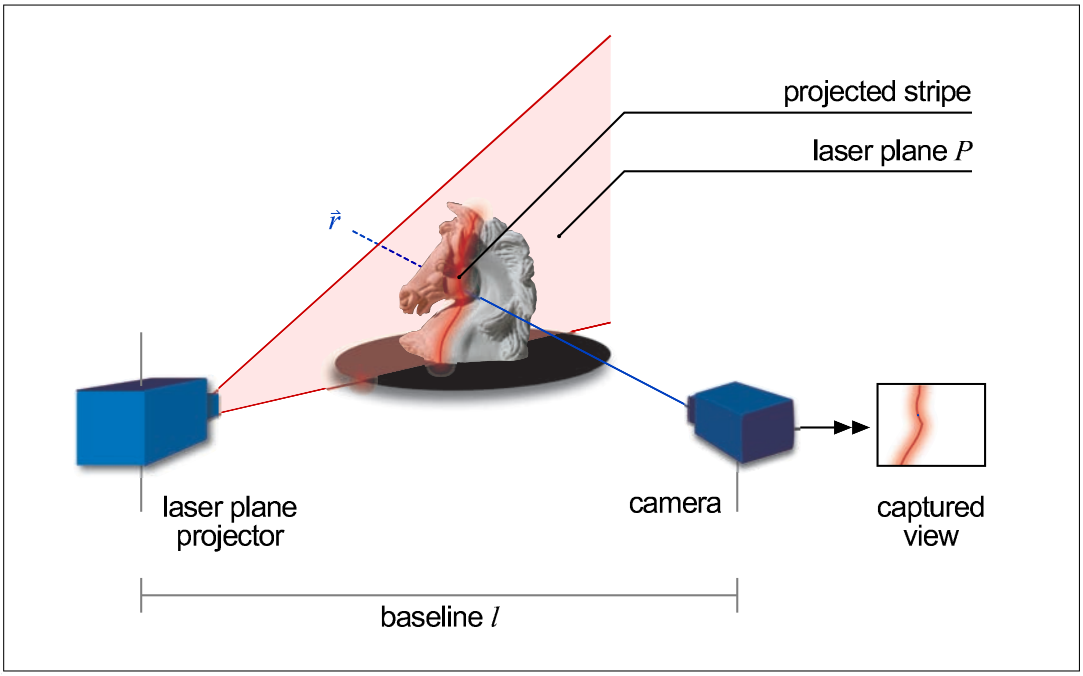


Figure 5.1: The principle of operation of the 3D laser range scanner used to acquire geometry models. Valid surface points are defined by the intersection of the laser plane P with a ray $\vec{r}$ (corresponding to positions in the captured image).

baseline leads to higher accuracy in the resulting models.

A laser range scanner relies on the constant, near-diffuse local light reflection on the surface of an object. Unfortunately, many of our test objects do not fulfill this requirement leading to noisy or incomplete depth maps: Colorful objects can absorb a large portion of the red laser light so that the reflected signal becomes too weak. Likewise, specular surfaces reflect hardly any light in off-specular directions so that the laser line may not be recorded by the video camera depending on surface orientation. Even worse, translucent or transparent objects exhibit non-local light reflection, i.e., light enters and leaves the surface at possibly distinct locations. In order to acquire good quality models even for difficult to scan objects, objects are often sprayed with white dust (e.g., Helling Met-L-Chek Developer D70) in order to obtain a surface with Lambertian reflection. Figure 5.2 shows an example of a translucent alabaster horse sculpture. The translucent behavior masks many geometric details that can be easily seen when the object is covered with dust.

5.1.2 3D Geometry Processing Pipeline

A single 3D scan of an object is usually not sufficient to cover its whole surface. To achieve complete coverage, the object is scanned several times from different directions (possibly using a turntable to rotate the object in front of the 3D scanner). The individual scans are then aligned and combined into a single data set. Based on this point cloud, a surface triangulation is constructed and remaining holes are filled manually or semi-automatically. Various postprocessing operations can be applied to the final triangle mesh, e.g., to remove noise or to reduce the complexity of the mesh. A more detailed overview

Figure 5.2: The alabaster horse. Left and middle: Photographs of the sculpture with and without subsurface scattering under identical illumination conditions. The sculpture was covered with fine dust to "turn off" subsurface scattering. Much surface detail is masked by the translucency and active optical range scanning systems have grave problems to scan the translucent object. Right: Rendering of the acquired geometry model (100 k triangles) with flat shading.

over tools and methods for geometry processing can be found in [Kobbelt00]. There are various software tools available (e.g., Minolta Polygon Editing Tool, Steinbichler COMETplus, INUS Rapidform, Innovmetric Polyworks, TowerGraphics Easy3DScan, and ViHAP3D's mesh processing tools) to perform all necessary processing steps from the initial 3D scanning until the generation of a final model.

5.2 Digital Camera Systems

A digital camera can be seen as a measurement device consisting of millions of sensors that convert luminance arriving at the pixels into digital values. The general assumption in image-based modeling applications is that after some calibration steps the acquired images are a faithful representation of reality.

We will in this section first discuss the general setup of a digital camera concentrating on the lens system and the sensor chip. We then summarize common calibration techniques for geometric and photometric camera calibration.[1] We also mention techniques to judge the accuracy of acquired images. We finally present examples of cameras used for material acquisiton at MPI Informatik and discuss their specific properties.

5.2.1 Camera Setup and Components

The two most important components of a digital camera are the lens system and the imaging sensor that ideally should be tailored to each other. Several mechanical components such as the aperture and the shutter as well as the camera electronics and host software influence the system performance, too, but will not be discussed in detail.

Imaging Sensor

An imaging sensor converts irradiance incident on a light sensitive area into an electrical signal whose exact nature depends on the sensor technology. Imaging sensors in digital cameras consist generally of a one or two dimensional array of sensor elements (sensels) or picture elements (pixels).

In the case of a *charge coupled device (CCD)* sensor, light is absorbed and creates free electrons that are stored inside the sensor element which serves as capacitor (e.g., [Janesick01]). At the end of an exposure, the collected charge is shifted towards one end of the sensor using an analog shift register and digitized using an analog-digital converter. Due to the physical principles involved, CCD sensors are highly linear, i.e., the amount of charge collected is proportional to the irradiance. The main performance limitation are leakage currents caused by chip defects and thermal noise. Cooling the sensor will greatly improve its performance as the noise doubles with every increase in

[1] There is some confusion about the meaning of the word calibration: In some areas such as color science *calibration* means to set a device into a defined state. In an additional *profiling* step, this state is characterized and the characterization is later taken into account in order to achieve a defined behavior of a device. In computer graphics, *calibration* is tuning a general model of the physical device in order to best describe the specific instance on the device. Unless otherwise noted, we will use this second semantics.

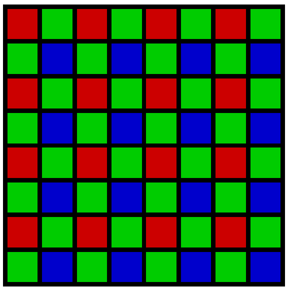


Figure 5.3: The Bayer pattern [Bayer76] for color image capture.

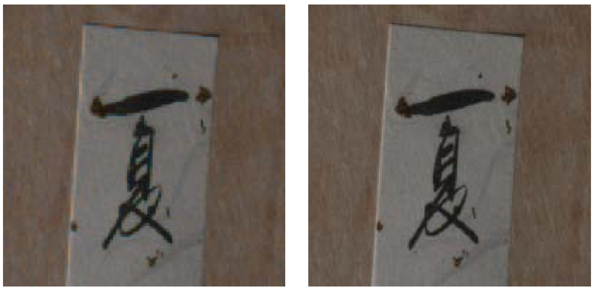


Figure 5.4: *Left:* Interpolation artifacts in an image acquired with a Bayer pattern sensor (Jenoptik ProgRes C14, single shot, fast reconstruction). *Right:* Ground truth image acquired using 4-shot mode (see Section 5.2.4 for more details). Note also the visually higher resolution of the ground truth image.

temperature of about 6 Kelvin. The Jenoptik ProgRes C14 digital cameras shown in Section 5.2.4 contains a cooled CCD imaging sensor.

The dynamic range, i.e., the ratio of brightest to darkest non-zero recordable intensity values of a CCD sensor is not only limited by noise but also by its capacity. A sensel of the Sony ICX085AK chip used in the Jenoptik ProgRes C14 camera can typically store about 25,000 electrons so that the dynamic range even of a noise free sensor is limited to about 1:25,000 requiring less than 15 bit for digitization. Very bright spots in an image can also lead to blooming artifacts if the irradiance creates a strong current and charge is overflowing from one sensel to neighboring sensels.

In contrast to CCD sensors, *complementary metal oxide semiconductor (CMOS)* sensors are based on the technology used for modern random access memory chips, processors, or analog circuitry. They are more versatile and flexible but often suffer from stronger image noise. In particular, modern CMOS cameras can achieve a dynamic range of more than 140 dB which is often required for computer graphics applications (see, e.g., Kang et al. [Kang03] for an overview of camera systems with extended dynamic range). The Silicon Vision Lars III camera is such a camera with a CMOS imaging sensor.

Color image acquisition is difficult as a single sensor element can capture only a single value (defined by the convolution of its spectral sensitivity with the spectrum of the incident light) but not a full set of RGB tristimulus values. The most common solution is to cover individual sensels with color filters arranged in a Bayer pattern [Bayer76] (see Figure 5.3) and to capture individual color components at different spatial locations. Spatial resolution is in this case traded for color information. Tristimulus color values (RGB) are reconstructed for each pixel using information from neighboring sensels. This reconstruction can introduce color artifacts so that the pixel values do not necessarily correspond to the irradiance incident at the pixel location (see Figure 5.4).

Alternatively, the image can be simultaneously captured by 3 sensors with corresponding color filters (3 chip cameras) or at different times using, e.g., a color filter wheel or tunable filters. Only recently, a new technology was introduced by Foveon that makes use of the different penetration depths of light into the chip and stacks three sensels on top of each other [Lyon02]. It allows to capture tristimulus color values without spatial or temporal aliasing artifacts.

Lens System

The lens system projects an image of a scene onto the imaging sensor (see Figure 5.5). A lens with

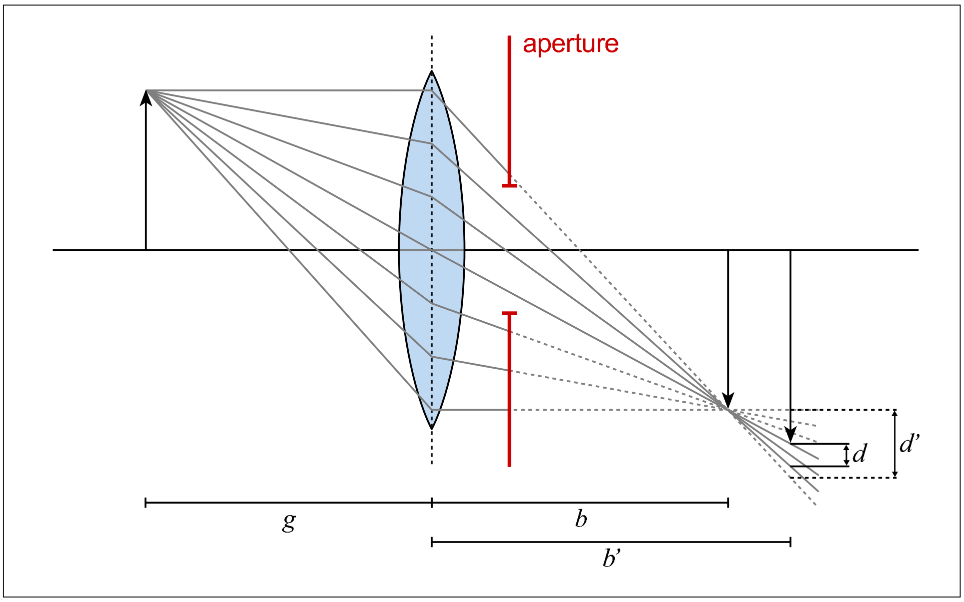


Figure 5.5: Projection by a lens system with adjustable aperture. See text for a more detailed description.

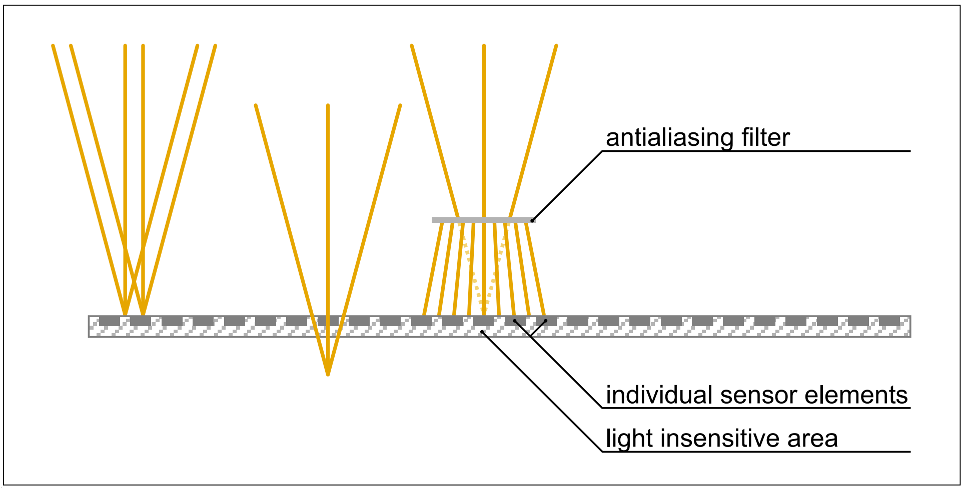


Figure 5.6: Projection onto a sampled imaging sensor with light insensitive areas (fill factor < 100 %). *Left:* A small feature is either focused on a light sensitive or insensitive area leading potentially to aliasing. *Middle:* A slightly out-of-focus projection reduces the aliasing potential but will still result in a focused digital image if the circle of confusion is smaller than the sensel spacing. *Right:* Some cameras include an antialiasing filter to prefilter in the optical domain.

focal length f focuses a point at distance g onto a point at distance b according to

$$\frac{1}{f} = \frac{1}{g} + \frac{1}{b}$$

leading to a sharp image. At a point b' farther away, the image is blurred within a circle of confusion whose diameter d is proportional to the radius of the aperture.

While achieving a sharp image is desirable for an analog, film-based camera system, it can be harmful for the sampled imaging system of a digital camera: If the light sensitive portion of a sensor element covers not its full area, a small feature could be focused on a light-insensitive part and would not be captured at all (see Figure 5.6). It has also the potential of severe aliasing if the fill factor, the percentage of light sensitive area of a pixel, is much smaller than 100 %. The fill factor can be increased by adding small lenses on top of the sensor elements as shown in Figure 5.6. This is however not sufficient for color sensors with Bayer pattern [Bayer76] where even a small feature should be registered by red, green, and blue color sensors in order to minimize interpolation artifacts. An additional antialiasing filter can be used to ensure this by slightly blurring the projected image.

To achieve good image quality, it is therefore important that the lens system matches the imaging sensor. A way to check this is to determine the modulation transfer function (MTF) [Williams99, Burns01] of an imaging system that describes its properties in frequency space.

The lens system can furthermore lead to a variety of artifacts such as chromatic aberration (the focal length f varies for different wave lengths of light leading to color artifacts at image edges), geometric distortions (the acquired image is distorted compared to an image generated using a pin-hole projection), or diffraction at the aperture blades (visible as star-shaped artifacts around bright features). It is therefore mandatory to employ high quality lenses and apply where possible correction techniques in order to acquire high quality data.

5.2.2 Geometric Calibration

The lens system of a digital camera projects an image of a scene onto the imaging sensor. The mapping between points in space and 2D image coordinates depends on the position and orientation of the camera (so called *extrinsic parameters*) and the *intrinsic parameters* such as focal length, distortion parameters, and the position of the optical axis on the image plane (principal point). Various authors such as [Tsai87, Heikkilä97, Zhang00] proposed camera models and calibration techniques to estimate extrinsic and intrinsic camera parameters from a set of images of a calibration target.

Jean-Yves Bouguet's calibration toolbox for Matlab [Bouguet] (which is based on the estimation technique by Zhang [Zhang00] and uses a camera model similar to the model of Heikkilä and Silvén [Heikkilä97]) provides a convenient way to determine the intrinsic camera parameters such as the actual focal length $fc = (fc(1), fc(2))^T$ and the distortion coefficients $kc = (kc(1), kc(2))^T$ of a lens. The simplified version of Bouguet's camera model projects a 3D point $P = (P_x, P_y, P_z)^T$ onto a 2D image coordinate $p_p = (x_p, y_p)^T$ as follows:

$$\begin{pmatrix} x_p \\ y_p \end{pmatrix} = d(r^2) \begin{pmatrix} fc(1) \cdot P_x/P_z \\ fc(2) \cdot P_y/P_z \end{pmatrix} + \begin{pmatrix} cc(1) \\ cc(2) \end{pmatrix}$$

One can often assume that the lens is centered with respect to the image sensor and set the center of projection $cc = (cc(1), cc(2))^T$ to the image center. The radial distortion $d(r^2)$ is modeled by the fourth order approximation

$$d(r^2) = 1 + kc(1)r^2 + kc(2)r^4$$

and depends on $r^2 = (P_x/P_z)^2 + (P_y/P_z)^2$.

5.2.3 Photometric Calibration

The pixel values stored in a digital image are normally not linearly related to intensity values in the captured scene. The sRGB standard [IEC99] defines for example a gamma value of 2.2 in order to avoid quantization artifacts, i.e., a 2.2-power function needs to be applied to an sRGB color value in order to get a value linearly related to intensity.[2] If linear values are needed it is therefore mandatory to recover the response curve (also called transfer function or opto-electronic conversion function (OECF) [ISO99]) that performs the mapping from digital image values to linear intensity values.

A simple method to recover the response function is to acquire an image of a uniformly illuminated test target with patches of known reflectance and to compute the response curve from the digital image values of the patches. In this course, we apply however high-dynamic range imaging methods that also recover the response curve.

High-Dynamic Range Imaging

Standard digital cameras can only capture intensity values within a quite limited dynamic range: For a given exposure setting, too dark areas of a scene are usually mapped to zero (or vanish in the devices noise floor) while too bright areas are mapped to some maximum value. For both cases, the digital image values cannot be mapped back to linear intensity values and therefore are only of limited use for image-based measurement methods.

The principle of high-dynamic range (HDR) imaging [Debevec97, Robertson99, Robertson03] is therefore to acquire multiple images of the same scene with different exposure settings ensuring that at least one meaningful digital value is available for each pixel. The cameras response curve I is then computed from this image series. Using [Robertson99], a HDR image with linear floating point intensity values of pixel x_j at image position j is generated as weighted average of the linearized input images scaled according to their exposure times t_i:

$$x_j = \frac{\sum_i w_{ij}\, t_i\, I_{y_{ij}}}{\sum_i w_{ij}\, t_i^2}.$$

$I_{y_{ij}}$ is the linearized pixel value at position j of input image i. The weighting function $w_{ij}\, t_i^2$ is a Gaussian-like function multiplied by the square of the exposure time. It emphasizes pixel values with high confidence and helps to suppress image noise.

Color Calibration

The tristimulus color value (e.g., the RGB triplet) recorded for a color patch in a scene depends not only on the spectral reflectance of the patch but also on the spectrum of the illumination and on the spectral response of the imaging sensors. The most basic method to ensure correct color acquisition is therefore to perform a white balancing step. To this end, a neutral white or gray color patch is acquired under the given illumination and multiplicative scaling factors for the linearized color values are determined that lead to equal intensity values.

5.2.4 Camera Systems

We give here examples of two different cameras used to perform some of the acquisitions described in these course notes – a high-end color video camera for scientific applications and a prototype implementation of a high-dynamic range video camera.

Jenoptik ProgRes C14

The Jenoptik ProgRes C14 video camera was designed for scientific photography and features a C-Mount lens connector. Its 1300×1030 pixel CCD sensor is cooled by a Peltier element and an additional fan to keep image noise at a minimum enabling exposure times of up to 10 min without visible noise. The Bayer pattern color sensor is mounted onto a piezo micro-scanning actuator and can be shifted by a whole pixel to sequentially

[2]Due to numerical considerations, the sRGB standard [IEC99] specifies more exactly that small values up to a threshold of 0.0031308 (on a scale of [0..1]) are kept linear while a gamma of 2.4 is applied to larger values.

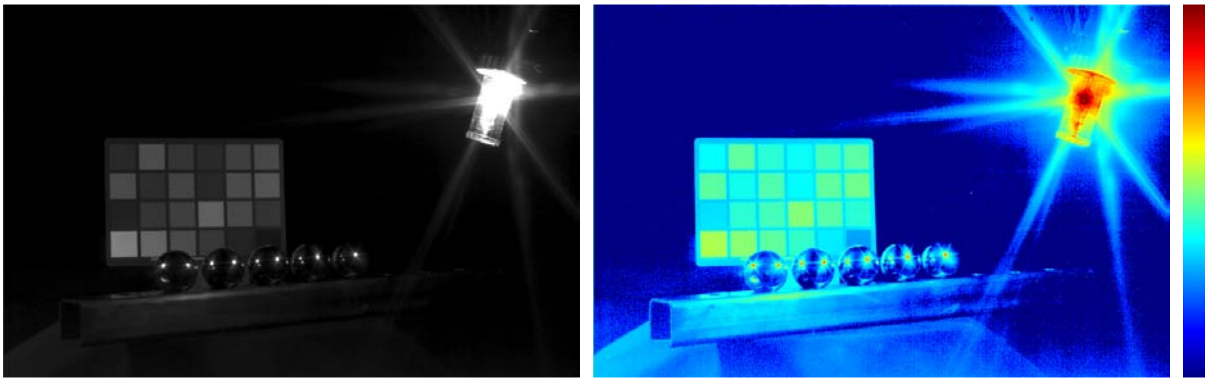


Figure 5.7: Scene with a contrast of 1:3,700,000 captured with the Lars III camera. *Left:* HDR image with medium exposure. *Right:* Color-coded version (logarithmic) encoding the full dynamic range.

capture per-pixel RGB information. Additional shifting by sub-pixel steps allows to increase image resolution up to 3900×3090 pixels. The camera records 14 bit intensity values with a typical dynamic range of 66 dB. It can be controlled remotely via IEEE 1394 FireWire and custom software using the provided SDK. The camera is usually used either with the high-resolution Jenoptik Lametar 2.8/25 mm lens or with a Schneider Cinegon 1.4/4.8 mm wide-angle lens.

The excellent noise characteristics make the C14 an ideal camera for low light-level applications. Static scenes can be captured with high resolution in multi-shot mode requiring no color interpolation. Dynamic scenes can however only be captured in single shot mode with up to 10 frames per second (fps). Scenes with a strong infrared signal might require an additional infrared cut-off filter to achieve good color fidelity.

Silicon Vision Lars III

Since a couple of years, high dynamic range video cameras are available from several manufacturers. They are able to capture scenes with much higher dynamic range than standard CCD cameras with a single exposure and are therefore able to considerably speed up complex acquisitions such as the DISCO system [Goesele04] presented in Chapter 8.

The Silicon Vision Lars III is a prototype implementation of a black-and-white high-dynamic range video camera (up to 30 fps) with C-Mount lens connector. Its 768×496 pixel CMOS sensor achieves a fill factor of 100 % by placing the light sensitive photo diode on top of the per-pixel circuitry. Each sensel decides at up to 12 time steps (usually powers of 2) whether or not the exposure should be continued. The collected charge is digitized using 12 bit A/D converters leading to a linear dynamic range of 120 dB. Figure 5.7 shows a color-coded version of a HDR image captured with the Lars III camera. At MPI Informatik, we generally use a Jenoptik Lametar 2.8/25 mm lens with fully opened aperture to minimize artifacts introduced by the interplay of lens system and HDR scene.

Lars III Photometric Calibration

Performing a photometric calibration to evaluate the linearity of a HDR camera is difficult as it requires both, a HDR calibration target as well as suitable software. Furthermore, the exposure time of the Lars III video camera cannot be changed in a controlled way.

To compute its response curve, we therefore captured 3 video streams (483 images total) of a

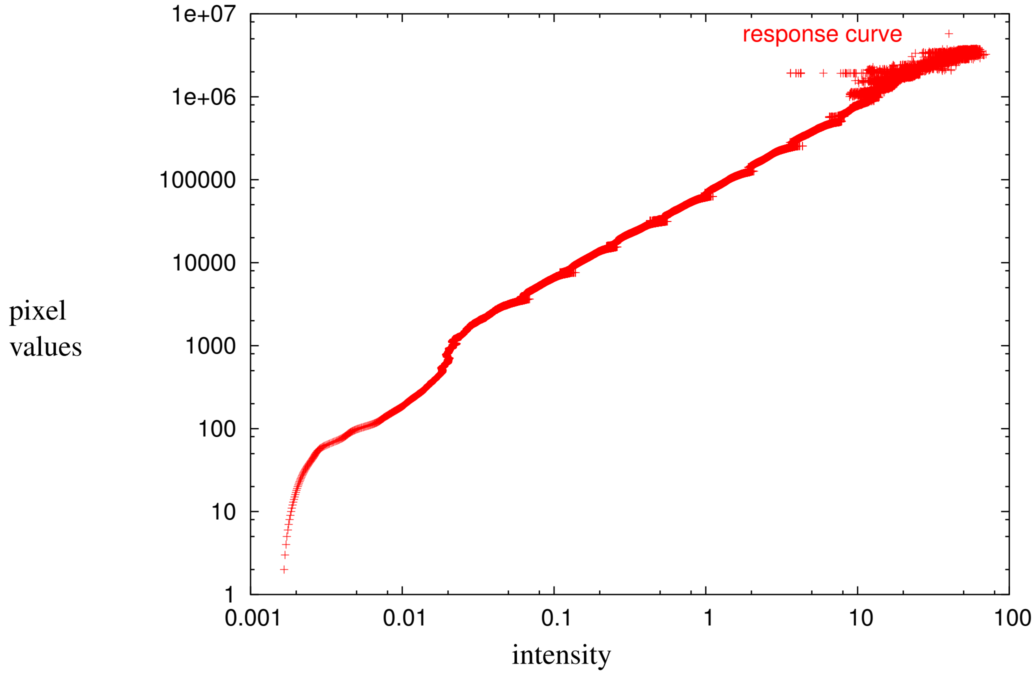


Figure 5.8: Recovered response curve for the Lars III camera (horizontal axis: intensity, vertical axis: pixel values). Three of the 483 input image for the response curve recovery are shown in Figure 5.9.

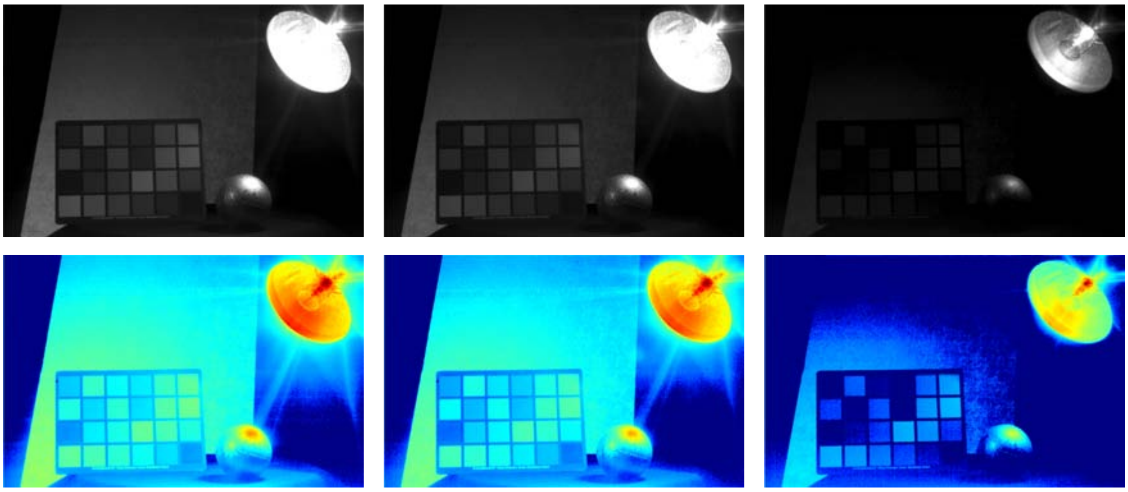


Figure 5.9: Three of the input images used to recover the response curve of the Lars III camera (Figure 5.8). All images show the same scene consisting of a bright light source as well as specular and diffuse reflecting objects in order to achieve a good luminance distribution. For the middle and right image, the lens was covered with a $1.41\times$ extension factor gray filter (middle) and a $10\times$ extension factor gray filter (right), respectively, to emulate varying exposure times.

static scene. While one of the streams was captured without modifications, we added a neutral density filter (gray filter) with 1.41× and 10× extension factor, respectively, in the optical system for the other two streams. The extension factors are not powers of 2 in order to avoid interference with the camera's switching behavior. As shown in Figure 5.9, the scene contains an HMI light with a metallic reflector and other specular and diffuse reflecting objects to achieve a good luminance distribution.

The images were encoded in a 4+12 bit exponent-mantissa format and fed into a 16 bit version of an HDR recovery algorithm [Robertson99] to compute the response curve (Figure 5.8). It confirms the linear response but also shows artifacts correlated to the sensor's switching points. The non-linearities at high intensities are probably caused by intensity noise in the input images and by the small number of samples per intensity. The bend at low intensities is at least partially caused by a problem of the measurement methodology: The neutral density filters influence only the irradiance at the image sensor but do not change the exposure time. The dark current noise (charge collected in some pixels independent of irradiance) is consequently not influenced by the filters. As the camera contains a significant number of pixels with dark current noise of some 10s or 100s, the recovered response will be unreliable in this range. More details about photometric calibration of HDR camera systems can be found in [Krawczyk05].

5.3 Illumination

Choices for camera systems are quite limited and most acquisitions of materials could be performed with a single type of high quality camera. In contrast to that, the illumination is often an integral part of the measurement system. The illumination systems presented in these course notes include e.g. standard tungsten bulbs, LEDs, Laser, and monitors. More details are given in the individual chapters.

Figure 5.10: The acquisition lab at the MPI Informatik. A test object on the table is illuminated by two HMI light sources with different reflectors.

5.4 Environment

The influence of the environment (most often the room in which the setup is located) on the measurements needs to be carefully evaluated. Large diffuse reflecting walls can introduce strong ambient illumination if they are lit by the measurement setup. Uncontrolled light sources can invalidate all measurements. Walls, floor and ceiling of general purpose measurement labs are therefore often covered with dark and diffuse reflecting material. The lab at MPI Informatik shown in Figure 5.10 has walls and ceiling covered with black felt that diffusely reflects about 1.4 % of the incoming light. The floor is covered with black needle fleece carpet. This ensures that the influence of the environment on the measurements is reduced to negligible levels. A large measurement area allows to place experiments far away from the walls further reducing the amount of reflected light.

Chapter 6

Homogeneous Isotropic BRDFs

Wojciech Matusik

This chapter deals with Bidirectional Reflectance Distribution Functions or BRDFs. First, we formally define BRDFs and describe their properties. Second, we present our image-based measurement system. Third, we show how to make measurements more efficient using appropriate representations. Next, we perform experimental validation of analytical BRDF models. Finally, we show how to construct a data-driven BRDF model.

6.1 Reflectance

Nicodemus et al. [Nicodemus77] defines reflection as the process by which electromagnetic flux incident on a stationary surface leaves the surface without a change in frequency. The reflectance is the fraction of the incident flux that is reflected. Thus, phenomena like transmission, absorption, spectral effects, polarization, and fluorescence are not considered in this chapter. We consider the reflectance under the assumption of simple ray optics – the light travels only along the straight lines. Therefore, phenomena explained by wave optics (interference, diffraction) are also not considered.

BRDF

One of the most useful models of reflectance is Bidirectional Reflectance Distribution Function (BRDF), denoted with f_r. It deals with the light that is immediately reflected when reaching the surface (the light that scatters inside the material and then leaves is not considered). This simplification of the general reflectance takes advantage of the observation that, for most materials, the radiant flux incident at a point emerges from a point very near the point of incidence. This model of reflectance assumes that the surface is homogenous. BRDF is a function of four variables: two variables specify the incoming light direction, two other variables specify the outgoing light direction. It is defined as a ratio of incoming irradiance $dE_\mathrm{i}(\theta_\mathrm{i}, \phi_\mathrm{i})$ to the outgoing radiance $dL_\mathrm{r}(\theta_\mathrm{r}, \phi_\mathrm{r})$ (see Figure 6.1):

$$f_\mathrm{r}(\theta_\mathrm{i}, \phi_\mathrm{i}, \theta_\mathrm{r}, \phi_\mathrm{r}) = \frac{dL_\mathrm{r}(\theta_\mathrm{r}, \phi_\mathrm{r})}{dE_\mathrm{i}(\theta_\mathrm{i}, \phi_\mathrm{i})} \qquad (6.1)$$

The unit of BRDF is $\frac{1}{steradian}\left[\frac{1}{sr}\right]$.

Isotropic BRDF

Isotropic BRDFs are an important subclass of BRDFs. The isotropic model is valid for materials for which rotations about the surface normal can be

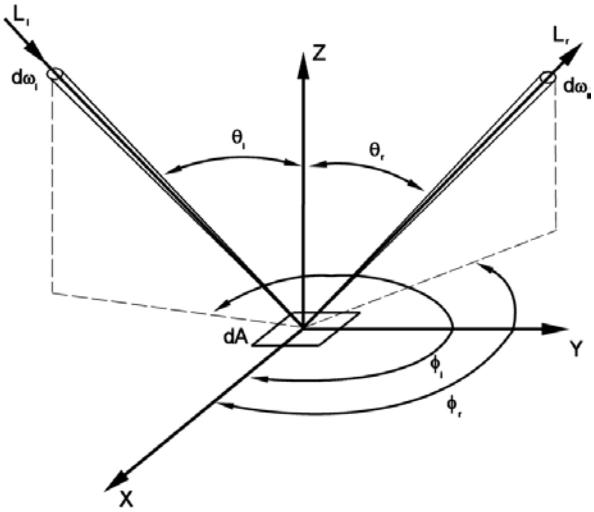


Figure 6.1: *BRDF, f_r, is defined as a ratio of incoming irradiance $dE_i(\theta_i, \phi_i)$ to the outgoing radiance $dL_r(\theta_r, \phi_r)$.*

ignored. (The reflectance is the same when the material sample is rotated about the normal while the directions of the incoming irradiance and outgoing radiance are fixed.) In this case BRDF can be written as a function of only three variables. The variables $\phi_{\rm r}$ and $\phi_{\rm i}$ can be replaced by one variable $\phi_{\rm diff}$ = ($\phi_{\rm r}$ - $\phi_{\rm i}$). Therefore, the expression for isotropic BRDF can be expressed as follows:

$$f_{\rm r}(\theta_{\rm i}, \theta_{\rm r}, \phi_{\rm diff}) = \frac{dL_{\rm r}(\theta_{\rm r}, \phi_{\rm diff})}{dE_{\rm i}(\theta_{\rm i}, \phi_{\rm diff})} \tag{6.2}$$

However, anisotropic surfaces – surfaces with preferred directions – cannot be modelled using isotropic BRDFs that are parameterized using only three angles. Anisotropic surfaces include brushed or burnished metals, hair, and fur. Kajija [Kajiya85] gives as an example cloth that is composed of a weave of threads. Light which strikes threads along their length scatters differently than when it strikes them perpendicularly. Anisotropic surfaces have microgeometry with strongly oriented elements. The orientation of these elements causes the light to reflect in preferred directions relative to the local coordinate system expressed by the tangent, normal, and binormal. A distant observer does not see the microstructure, but instead only its effect on the reflected light. However, for the vast majority of the natural and man-made surfaces the microgeometry is randomly distributed and does not have any preferred direction.

6.1.1 Properties of BRDFs

BRDFs are not arbitrary functions. In order to obey the basic principles of physics they must satisfy certain constraints. Next, we will describe the most important of these properties.

Non-negativity All values of function $f_{\rm r}$ must be non-negative. In fact, they can be any value from the interval $[0,\infty)$. This is because both radiance and irradiance must be non-negative.

Energy Conservation The energy conservation property can be stated in the following way:

$$\int_\Omega f_{\rm r}(\theta_{\rm i}, \phi_{\rm i}, \theta_{\rm r}, \phi_{\rm r})\ d\mu(\theta_{\rm r}, \phi_{\rm r}) \leq 1 \text{ for all } (\theta_{\rm i}, \phi_{\rm i}). \tag{6.3}$$

This property means that the amount of energy that is received by the surface element from some specific direction must be greater than the sum of the energy emitted by the surface element in all possible directions. This must be true for the energy received from all possible directions. The underlying assumption is that the surface element does not emit the energy by itself (e.g., the surface is not fluorescent).

Reciprocity This property is also called Helmholtz's law of reciprocity. It states that the surface reflectance should be independent of the direction of the light flow – if the flow of light is reversed the value of the BRDF should be the same. This means that swapping the incoming and outgoing directions should yield the same value of

f_r :

$$f_r(\theta_i, \phi_i, \theta_r, \phi_r) = f_r(\theta_r, \phi_r, \theta_i, \phi_i). \tag{6.4}$$

Veach [Veach97] has a detailed discussion of the reciprocity. He argues that reciprocity cannot be derived from the second law of thermodynamics or from the principle of time reversal invariance. However, reciprocity is widely observed and commonly assumed in reflection modelling.

6.2 Measurement

This section describes the construction of our BRDF measurement system. We discuss the whole acquisition process. This includes geometric calibration, radiometric measurements, and computation of actual BRDF samples. Next, we describe parameterization and representation of the acquired BRDF data – reflectance of each material is represented as a data structure composed of dense BRDF samples. In this way we measure and represent reflectance for a large collection of different materials.

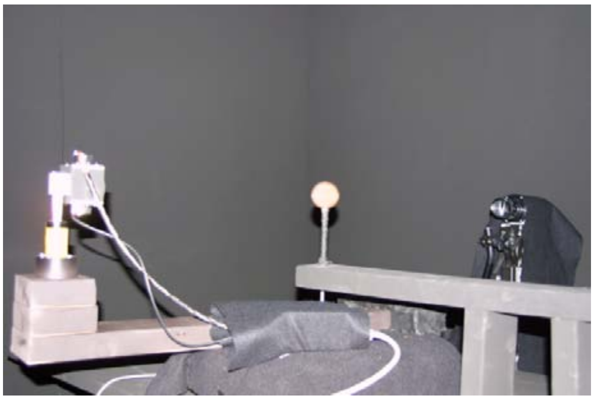


Figure 6.2: *A photograph of our high-speed BRDF measurement gantry.*

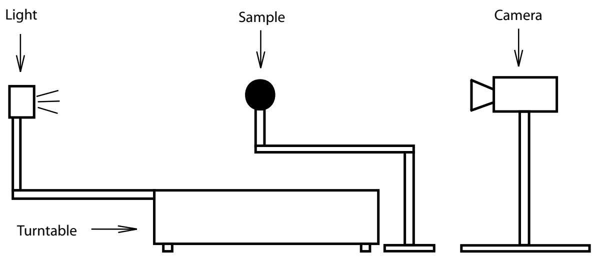


Figure 6.3: *A schematic of our high-speed BRDF measurement gantry.*

6.2.1 Measurement System

In order to acquire a sufficient number of adequately sampled BRDFs, it was necessary to build a measurement device. Our modelling approach placed two requirements on the acquired data: first, each BRDF should be sampled densely enough to be used directly as a table-based model, and second, the space of BRDFs should be sampled adequately so as to span the range of surface reflectances that we hope to generate. Accordingly, we have built a BRDF measurement device suitable for rapidly acquiring high-quality isotropic BRDFs for a wide range of different materials. Our design was inspired by Marschner's image-based BRDF measurement device [Marschner00b]. His device, like ours, requires a spherically homogenous sample of material.

A photograph of our measurement system is shown in Figure 6.2 and its schematic is shown in Figure 6.3. The system is placed in a completely isolated room, with all of the surfaces of the instrument covered in either black felt or black construction paper. In addition, all walls of the room are painted a matte black. It consists of the following components: a QImaging Retiga 1300 (a 10-bit, and a 1300 × 1030 resolution Firewire camera), a Kaidan MDT-19 (a precise computer-controlled turntable), and a Hamamatsu SQ Xenon lamp (a lamp with stable light emission output and a continuous and relatively constant radiation spectrum over the visible light range – Figure 6.4). The lamp is mounted on an arm to a turntable. The lamp orbits the measurement sample placed at the center of rotation; the camera and the sample are stationary. The light source moves in increments of approximately 0.5° from the point opposite the camera (the sample is in between the camera and the light

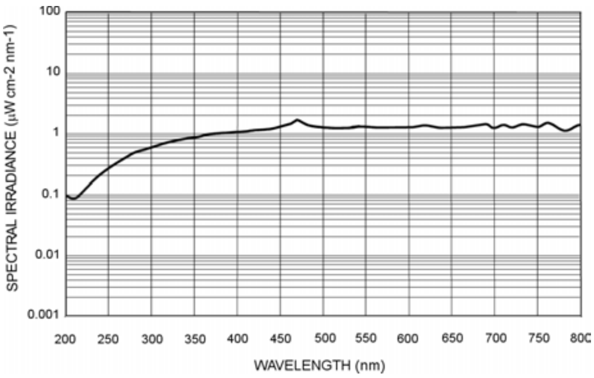


Figure 6.4: *Spectral Irradiance 50 cm from the light source.*

source) to the point in front of the camera. We take a total of 330 high dynamic range pictures to cover the required half circle. This process takes about four hours.

6.2.2 Geometric Calibration

Before proceeding with BRDF measurements, the relative positions of the camera, light source, and material sample must be found. In addition, the imaging parameters of the camera and the precise size of the material sample are determined. This process is called geometric calibration.

In order to geometrically calibrate the camera we need to determine its internal and external parameters. The internal parameters are computed using the technique developed by Zhang [Zhang98]. The external parameters are computed using about 30 correspondences between the 3D coordinates of the tip of a contact digitizer (FARO Arm) and its 2D position in the image. Thus, we obtain the relationship between image pixels and the corresponding rays in 3D.

Next we determine the locations of the light source. The light source is assumed to move on a circle in 3D. Therefore, we have to determine the circle parameters and a reference light position on the circle. First, we move the light source and measure approximately 30 of its locations using a contact digitizer. Then, we determine the plane of the circle (this requires solving an overconstrained system of linear equations). Next, we project all the 3D locations onto this plane and compute the equation of the circle in 2D using a non-linear optimization. We obtain the circle center and a vector from the circle center to the current position of the light source. Using this parameterization we can move the light source using the turntable and determine its precise location in 3D.

Computing the position of the material sample is relatively simple. First, we measure the diameter of the sample using calipers. Next, we measure the locations of about 30 points on the surface of the sample using a contact digitizer. Then, we use non-linear optimization to find the center of the sample.

6.2.3 High-Dynamic Range Radiance Measurements

The dynamic range of the scene exceeds the 10-bit range of Retiga 1300. Therefore, we use multi-exposure photography to derive high-dynamic range images of the material sample. For each high dynamic range picture we take a total of 18 10-bit photographs with increasing exposure times. The exposure time increases from 40 microseconds to 20 seconds. We take advantage of the fact that CCD cameras have a linear response curve; thus, we do not need to recover the response curve of the camera [Debevec97].

For each pixel in the image we have 18 measurements. We discard all measurement values below 10 and above 1000. We then fit a line to the exposure time vs. measured radiance values for the remaining measurements. The slope of the line is used as the radiance estimate. The correlation of this line for all pixels in the image is higher than 0.998.

Some of the CCD imager sensor elements (photosites) have high dark current values. This means they generate a significant output even in the absence of any incoming visible radiation. These dark current levels are a function of the ambient room temperature, localized impurities, and processing variations in the CCD's construction. As a result, these elements do not produce valid measurements. In order to detect these elements we capture one image with 60 second exposure time and a closed lens cover (no external light gets to the imager). The elements that have values above 16 are considered to have excessive dark current values, and are not used in the BRDF measurement process.

6.2.4 BRDF Computation

Each acquired image of the material sample represents many simultaneous BRDF samples. Each imaged pixel of the sphere is treated as a separate BRDF measurement. This measurement is either red, green, or blue since the Retiga 1300 produces a Bayer pattern image. (We do not attempt to interpolate the other color components using nearby sensor elements.) In order to compute the specific BRDF value for a given image pixel we perform the following steps. First, we intersect the ray defined by the pixel with the sphere to determine point P. Then, we compute the normal at point P on the sphere, the vector and the distance to the light source, and the vector to the camera pixel. Next, we compute the irradiance at point P due to the light source (taking into account distance to the light source and foreshortening). Finally, we compute the BRDF value as the ratio of the high dynamic range radiance to the irradiance.

6.2.5 Data Representation

The specular peaks of highly reflective materials are particularly difficult to represent using the natural isotropic BRDF parameterization (θ_r, θ_i, ϕ_diff). Even when binning a BRDF at a dense grid (every 1 degree spacing for each dimension), it is not possible to reproduce original images (the specular highlight becomes an oval shape, oriented at different directions). In order to address these sampling problems, we used a different coordinate system, introduced by Rusinkiewicz [Rusinkiewicz98]. This coordinate frame is based on the angles with respect to the half-angle (half-vector between incoming and outgoing directions). The half-angle is a relevant reflection parameter because it defines an ideal surface normal for which a mirror surface (perfect reflector) would reflect all of the incoming light in the specified outgoing direction. This coordinate frame allows us to vary the sampling density near the specular highlight. Specifically, we vary θ_h (angle between the actual surface normal and the half-vector), assigning smaller bins for values near specular reflection and larger bins for angles far away from the specular reflection.

We still subdivide θ_h, θ_d into 90 bins and ϕ_d into 360 bins. This results in a total of 90 x 90 x 360 = 2,916,000 bins for each color component. We half this number to 1,458,000 by enforcing the reciprocity constraint:

$$f(\theta_h, \theta_d, \phi_d) = f(\theta_h, \theta_d, \phi_d + \pi) \quad (6.5)$$

With this constraint we need only to subdivide ϕ_d into 180 bins.

Our measurement process gives us typically 20-80 million BRDF samples for each material. We reduce the noise in the measurements by removing the outliers in each bin (lowest and highest 25% of the values), and we average the remaining measurements. This statistical smoothing is intended to remove systematic noise as well as compensate for small variations in material properties over the sample. As a final validation we render a synthesized version of our sample sphere and compare it to the corresponding acquired high dynamic range image. We conduct this inspection for all input light configurations. The rendered images reproduce the input images very well. We have used our device to acquire BRDF measurements of more than 130 different materials, including metals, plastics, painted surfaces, and cloth. Upon analyzing the data we removed some materials that exhibited significant subsurface scattering, anisotropy, or nonhomogeneity (using the isotropic

BRDF model it is not possible to reproduce input images for these materials).

6.3 Efficient Measurement

This section describes how to improve the BRDF acquisition by optimizing sampling. We observe that the solutions for compact BRDF representations imply more efficient measurement (i.e., sampling) procedures for isotropic BRDFs. These procedures can, in principle, significantly accelerate the BRDF acquisition process. It has previously been observed that wavelets are a very compact representation for BRDFs. Therefore, we start by using a wavelet analysis to compute a single wavelet basis for all BRDFs in our dataset. The coefficients of this basis are a compact representation for most BRDFs. The wavelet analysis also results in an efficient measurement procedure that derives a non-uniform sampling of the BRDF domain required to reconstruct most BRDFs well. Furthermore, we show that each BRDF in the dataset can be represented well using linear combination of other BRDF. Therefore, one needs only a small number of coefficients to represent a BRDF – this is a compact representation. This observation results in an another efficient measurement procedure. The procedure expresses new BRDFs as a linear combination of source BRDFs. It estimates the coefficients of the linear combination using few (less than a thousand) BRDF samples.

6.3.1 Efficient BRDF Sampling Based on Wavelet Analysis

The measurement procedure we propose focuses on how to efficiently sample isotropic BRDFs. First, we observe that typical BRDFs exhibit high frequencies in only very specific regions of their parameter space (e.g., near specular peaks). Only these regions require dense sampling. Otherwise, BRDFs are smooth and slowly varying over most of their domain, and, thus, require fewer samples for accurate reconstruction in these regions. Non-uniform sampling can be used to exploit this "spatially varying" localized spectrum property, which is characteristic of BRDFs. The precise densities and patterns of this non-uniform sampling are largely a matter of guesswork, and it is likely that one should err on the side of oversampling the function, as we attempted to do in the acquisition of example BRDFs. However, once given a large set of oversampled representative BRDFs, it is possible to analyze the entire corpus in order to reveal the maximum localized signal frequencies for any point in the domain, which in turn implies the maximum necessary sampling frequency for that point. Assuming that the example BRDFs in our database are representative of the entire space of isotropic BRDFs, we can then sample the BRDF of any new material correctly at a lower non-uniform sampling rate without any a priori knowledge.

Next, we provide the details of wavelet analysis and representation. Each color component (red, green, blue) of a tabulated BRDF is stored as a function of three variables θ_h, θ_d, and ϕ_d. It is represented as a $90 \times 90 \times 180$ three-dimensional array of floating point values. Standard packages that perform discrete wavelet transform require data dimensions that are powers of two. Therefore, we insert each BRDF component into a $256 \times 256 \times 256$ array and pad the rest of the array with zeros. Next, we perform a nonstandard discrete wavelet transform on each 3D array to obtain an array of $256 \times 256 \times 256$ wavelet coefficients. For each color component of a BRDF, we keep the highest coefficients that are necessary to reconstruct the component with 3% precision. The sets of non-zero coefficients are generally different for each color component of a measured BRDF. They are also different for each BRDF in our dataset. However, there is a large degree of coherence between these sets. When we take the union of the 300 sets (there are 100 BRDFs in our dataset and three sets of coefficients for each BRDF), the size of the union set grows to approximately 69,000 wavelet coefficients – 4.7% of the original data. The union of non-zero coefficients corresponds to a set of wavelet functions. The accuracy of this representation is at least 3% but likely better. We call this set of wavelet functions the *Common Wavelet Basis (CWB)* for isotropic BRDFs.

Standard Fourier analysis could be used to determine the frequency spectrum of the BRDFs in our database. However, the maximum signal frequency would be very high since the Fourier basis functions span the whole domain of the sample space. Consequently, a Fourier analysis would suggest a dense and uniform sampling. We use wavelet analysis to obtain the maximum signal frequency for each part of the BRDF domain. The advantage of wavelet analysis is its ability to perform localized analysis of a larger signal because the underlying wavelet basis functions vary in both the spatial and frequency domain.

Wavelet analysis represents a particular BRDF function as a linear combination of basis functions of varying scale. At the same time it specifies the signal frequency over each interval of the function domain that is required to represent the function. The signal frequency translates directly to the required sampling frequency for each interval. In general, this sampling frequency is only adequate for one particular BRDF. When computing the Common Wavelet Basis, however, we perform a wavelet analysis for all of the BRDFs in the dataset. This wavelet basis defines the maximum signal frequency over each interval of the function domain needed in order to reconstruct any of the measured BRDFs. Using CWB we derive the sampling density and corresponding sampling points (θ_{h}, θ_{d}, ϕ_{d}) that should be measured in order to sample any arbitrary BRDF correctly. Our algorithm also reconstructs a dense BRDF representation from the measured BRDF values at the specified sampling points.

Next, we discuss the details of the algorithm – how to define an efficient BRDF sampling pattern, how to compute the CWB coefficients using the BRDF values at these sampling points, and how to reconstruct the full BRDF from the CWB. First, we note that each BRDF component $f_r(\theta_{\mathrm{h}}, \theta_{\mathrm{d}}, \phi_{\mathrm{d}})$ can be represented as a weighted sum of the CWB functions $H_i(\theta_{\mathrm{h}}, \theta_{\mathrm{d}}, \phi_{\mathrm{d}})$ as follows:

$$f_r(\theta_{\mathrm{h}}, \theta_{\mathrm{d}}, \phi_{\mathrm{d}}) = \sum_{i=1}^{n} H_i(\theta_{\mathrm{h}}, \theta_{\mathrm{d}}, \phi_{\mathrm{d}}) \times C_i, \quad (6.6)$$

where C is a vector of coefficients for the CWB functions that need to be computed. We also note that all of the CWB functions are known (for the sake of simplicity we use Haar wavelets) and they can be evaluated at any point $(\theta_{\mathrm{h}}, \theta_{\mathrm{d}}, \phi_{\mathrm{d}})$. Each BRDF value $f_r(\theta_{\mathrm{h}}, \theta_{\mathrm{d}}, \phi_{\mathrm{d}})$ produces one linear constraint on the values of the wavelet coefficients C. The approximately 69,000 non-zero coefficients as constraints allow us to compute all wavelet coefficients C by solving a system of linear equations.

Next, we need to select the sampling points $(\theta_{\mathrm{h}}, \theta_{\mathrm{d}}, \phi_{\mathrm{d}})$ that produce the linearly independent equations, and, therefore, allow us to compute the wavelet coefficients. We note that there is no unique set of sampling points; thus, we just select one possible set that leads to linearly independent equations. First, we compute one constraint for each original sampling point for a total of 1,458,000 equations. Many of these are linearly dependent. We determine a set of 69,000 equations that are linearly independent. Given these equations and the corresponding BRDF values, we solve the following system of equations:

$$G = H \times C. \quad (6.7)$$

Matrix H is large $(69,000 \times 69,000)$ but typically very sparse – usually there are around 40 non-zero elements in a row. This is a result of the small support of the wavelets at higher levels in the wavelet tree. We use the MATLAB sparse matrix routine to directly perform the operation H^T/C^T for each color channel (R, G, and B).

In order to reconstruct a BRDF we use the estimated coefficients of the CWB and we set coefficients for the wavelets not in the CWB to zero. Then we perform the inverse wavelet transform to compute the BRDF values at the original sampling grid locations.

6.3.2 Pull-Push Reconstruction of BRDFs

In the previous section we have shown how to reconstruct a BRDF on a uniform grid using 69,000 BRDF samples at specified locations. In this section, we present an alternative reconstruction method that in practice yields lower reconstruction errors.

Since we are given sparse BRDF samples we can treat the problem of reconstructing the full grid BRDF as a scattered data interpolation problem. One simple and fast methods is the pull-push method. This method relies on the pyramid data structure of a progressively downsampled BRDF. The algorithm consists of two steps. (1) The pull step is applied hierarchically from the highest to the lowest resolution in the BRDF pyramid. Each lower resolution of the pyramid is obtained from the higher resolution version. First, the higher resolution BRDF is convolved with a low pass filter. Then, the result of this convolution is downsampled by a factor of two to obtain the lower resolution BRDF. (2) The push step is also applied hierarchically to the BRDF pyramid. It starts at the lowest resolution and progresses to the highest, original, resolution. Low resolution data computed during the pull phase is used to fill in the gaps at the higher resolution. If the higher resolution BRDF value has high enough confidence then the lower resolution value is not used. Otherwise, the higher and lower resolution values are blended together. The implementation details of the algorithm are described by Gortler et al. [Gortler96].

6.3.3 Linear Combinations of BRDFs

In this section we show that new BRDFs can be represented equally well using the BRDFs in the original set. It follows that one needs only to estimate appropriate weighting factors for each of the original BRDFs in order to estimate any new BRDF. Since there are only 100 of these coefficients, the number of BRDF samples needed for this estimation should be relatively small.

We represent each densely sampled BRDF as a high-dimensional vector composed of all values for R, G, and B. Let P be the matrix of all BRDFs in the original set, C the vector of coefficients for the linear combinations, and B the new BRDF we measure. It follows that:

$$P \times C \approx B. \tag{6.8}$$

This system of equations is overconstrained since it has $90 \times 90 \times 180 \times 3 = 4,374,000$ equations and only 100 unknowns. However, a lot of these equations are linearly dependent. Therefore, we need to select a only small set of equations that allows us to robustly estimate the coefficients C.

We present a simple strategy for selecting equations that allow for computing the coefficients of the linear combination. Let X be a matrix composed of some subset of rows of the matrix P. (As noted before, P is a matrix composed of column BRDF vectors.) A good measure of how robustly we can estimate C is the ratio between the highest and lowest eigenvalue of the matrix $X^T X$. The system is well conditioned if this ratio is small. Since finding an optimal set exhaustively is prohibitive, we resort to a simple greedy strategy. We start with an initial set of n constraints. We select a constraint outside of the set X and one constraint in the set X. We swap them only if the ratio of the eigenvalues decreases in matrix $X^T X$ with the constraints swapped. We repeat this procedure until X converges to a stable set and we perform this procedure for different set sizes n.

This procedure guarantees that the system is numerically well conditioned, which in turn makes it robust to perturbations of the constraints. It also approximates a "most informative set" of measurements. In order to expect a good generalization of the known BRDFs, we must also ensure that the system is well overconstrained. We found that using $n = 800$ linear constraints provides a robust computation of the coefficients, while adding more constraints does not improve the solution. This implies that we can measure new BRDFs using only 800 samples. (At each point we measure either R, G, or B.) We note that each of the 800 equations corresponds to a specific value of $(\theta_{\mathrm{h}}, \theta_{\mathrm{d}}, \phi_{\mathrm{d}})$ and does not depend on the value of the BRDF at that point.

The procedure presented here is well defined, simple, and fast. In order to compute the coefficients C we need only to compute a pseudo-inverse of the 800×100 matrix and perform one vector-matrix multiplication. In contrast to fitting samples to analytic BRDF models, this procedure is not dependent on a good initial guess.

6.4 Validation of Analytic BRDF Models

In this section we evaluate the performance of several popular BRDF models in terms of their ability to fit the measured data. While previous work in BRDF modeling have validated these models with some measurements, our work is the first to quantitatively compare different models based on one sizable dataset covering a wide class of materials.

6.4.1 Single Lobe

First, we fit the dataset using a single specular lobe. The Ashikhmin and Cook-Torrance models perform the best in most cases, while Blinn-Phong and Ward yield the highest errors. The errors for the Lafortune model are systematically higher than Cook-Torrance and Ashikhmin, despite having the same number of free parameters. The differences between the BRDF fits are also perceptually significant in rendered images.

6.4.2 Multiple Lobes

Cook and Torrance suggest that two or more specular lobes are necessary to model surfaces where there are multiple scales of roughness. Lafortune's model cosine lobes serve as basis functions for representing BRDFs and thus multiple lobes are also expected in general. To study the effect of an additional lobe, we choose 26 materials from the dataset. These materials have multiple layers of finishing, e.g. metallic paints. In 14 cases for Cook-Torrance and 11 cases for Lafortune, an extra lobe reduces the error by more than 20 percent. We note that the Lafortune model performs worse than the Cook-Torrance model.

6.4.3 Shape of the specular lobe

Our experimental validation highlights the profound difference between two popular formulations of the specular lobe: 1. defined around the mirror direction symmetrically $(V \cdot R)$, and 2. defined by the normal and the half-vector $(H \cdot N)$. Most previous discussions of the behavior of reflectance have focused on the increase in intensity of the reflection when the incident angle moves toward grazing angles. We found that the shape of the lobe also has important implications. We observe that the lobe corresponding to $H \cdot N$ is more circular when L is near N, and becomes narrower and asymmetric as L moves towards grazing angle. On the contrary, the $V \cdot R$ lobe is always circular around the mirror direction R. Plots of the measured BRDFs verify that the shapes of the specular lobes are more consistent with the $H \cdot N$ lobe in almost all cases. To quantify the differences, we re-express the Blinn-Phong model using $V \cdot R$, and we find that the fitting errors for the $V \cdot R$ version are higher than the original Blinn-Phong model in almost all cases (they are 40 percent higher on average). We believe this is one reason for the inferior performance of the Lafortune model, as it uses a formulation similar to $V \cdot R$, where R is replaced by a generalized mirror direction. The superiority of the $H \cdot N$ lobe is in strong opposition to some beliefs that the half-vector representation is just another way to express the specular lobe.

6.5 Data-driven BRDF model

This section describes the construction of a data-driven reflectance model. The construction process uses machine learning techniques and, in particular, unsupervised learning methods. The classic problem in unsupervised learning is to infer a function from a set of example outputs. Each acquired isotropic BRDF can be treated as a vector in a high dimensional space – each tabulated 3D isotropic BRDF is unrolled to form a D-dimensional vector. Only a subset of all possible vectors corresponds to valid isotropic BRDFs. In order to construct a data-driven reflectance model we attempt to find a description of this subset based only on a relatively small number of training vectors. Dimensionality reduction methods are used to solve this problem. This analysis assumes that all BRDFs corresponding to real-world materials lie on a relatively low dimensional manifold embedded within the high-dimensional space. This is a common assumption is consistent with the relatively small number of parameters seen in analytic BRDF models.

First, we use Principal Component Analysis (PCA) to discover a linear subspace (a low dimensional hyperplane embedded in the high dimensional space) that best describes our dataset (in a least-squares sense). We discuss the validity of the linear subspace model and the problems associated with it. Next, we analyze the dataset using non-linear dimensionality reduction techniques that are more general than the linear analysis methods since they are capable of discovering curved low-dimensional manifolds embedded in high dimensional spaces. Once we obtain the description of the manifold (or the linear subspace) for isotropic BRDFs we can impose a parameterization of the discovered manifold and synthesize new plausible isotropic BRDFs by interpolating between points on this manifold. Thus, we obtain a generative data-driven BRDF model.

6.5.1 Analysis Using PCA

The first isotropic BRDF modelling approach that we will explore is a linear model. Finding a linear model involves computing an orthogonal basis vector set of a specified dimension that best represents the BRDFs in our dataset. In this case, the analysis tools for both manifold discovery and interpolation are well known. Principal Component Analysis effectively determines a set of basis vectors that span the desired subspace. Linear combinations of these basis vectors, sometimes called principal components or directions, can be used for approximation and interpolation within the space of given samples. Potential linear manifolds are generally suggested when there is a noticeable plateau in the magnitudes of the ordered eigenvalues associated with each principal direction. When this plateau occurs on the k^{th} eigenvalue, we can model the data as a k-dimensional linear subspace with a residual error bounded by the square root of the sum of the squares of the remaining eigenvalues. Independent of any plateau, however, the best k-dimensional approximation (under a least-squares metric) is given by a linear combination of the fist k principle directions.

We began the analysis of the BRDF samples by searching for a linear embedding manifold (a hyperplane). In this approach each of 104 BRDF samples is treated as a column vector to form a 4,374,000 by 104 measurement matrix. Initially we performed the analysis in the linear domain, but the reconstructed BRDFs exhibited large errors in the non-specular areas. Upon inspecting the data we found that the specular reflections were several orders of magnitude larger than values in areas of non-specular reflection, and errors in these regions dominate under a least-squares metric (e.g., measurement errors in the specular regions have larger values than the values in the off-specular areas). Therefore, we have decided to perform the analysis in the log space. This operation flattens the range of the BRDF values. It increases the importance of the reflectance in the off-specular regions

– these areas are also perceptually very important. The operation can be also justified by the fact that the human visual system is more sensitive to ratios of radiance values rather than absolute radiance values.

In order to compute the principal components and the coordinates of the original BRDFs in the new basis, the following steps are performed. First, the data points have to be mean centered (the mean of all log-BRDFs is computed and subtracted). Next, the covariance matrix is computed by taking dot products between all pairs of the BRDF vectors. Note that the matrix is symmetric; thus, the number of computations is halved. Moreover, it is only a 104 by 104 matrix. Then, the eigenvalue decomposition of this matrix is performed (e.g., using `eig` function in Matlab).

A plot of the eigenvalues sorted in order of decreasing magnitude the corresponding reconstruction error for the BRDF database is shown in the course slides. Furthermore, we show the reconstruction of a few different BRDFs using first 1, 5, 10, 20, 30, 45, 60, and all principal components. Good reconstruction is usually obtained using the first 30-40 components. Note that there is a considerable fall off in the sequential values seen in this plot but there is no noticeable plateau. However, the residual noise reaches a level that is comparable to the estimated measurement noise around the 45^{th} eigenvalue. This dimension of the embedding subspace is considerably higher one would assume based on the typical number of parameters used in analytic BRDF models. Nevertheless, it is easy to verify that the 45-dimensional space defined by the first principal components reconstructs all our measured BRDFs well.

6.5.2 Limitations of the Linear Model

The model obtained using Principal Component Analysis has some drawbacks for practical use. The most serious is the large number of principal vectors – users need to specify at least 45 different parameters in order to span the whole BRDF space. Furthermore, the space spanned by the 45 principal vectors includes vectors that correspond to unlikely or physically implausible BRDFs, as we will demonstrate. We start by considering the combinations of BRDFs outside the convex hull of the original BRDFs. Since the subspace extends infinitely, some of the vectors outside the convex hull will produce physically implausible BRDFs (e.g., the non-negativity or energy conservation property is not met).

Determining the validity of BRDFs within the convex hull of measured data samples is a much more difficult undertaking. First, we will show that conceptually all convex combinations of the source BRDFs would produce physically valid BRDFs (non-negativity, energy conservation, and reciprocity properties are preserved). To demonstrate this point we describe a thought experiment that explains how to "manufacture" a material with a BRDF equivalent to any convex combination of the source BRDFs. First, consider that a BRDF is defined with respect to an infinitesimally small surface element, but in practice sensors integrate over finite surface patches when measuring reflectance. Let's imagine a specific surface patch that is viewed at exactly one pixel of the sensor. If we assume that this surface patch is composed of tiles, and each of these tiles can be made of a different base material, then one could design material with a reflectance equivalent to any convex combination of the source BRDFs by assigning appropriate (possibly dithered) combinations of materials to each tile. This construction process is illustrated in Figure 6.5. While it would be possible to "manufacture" these materials, not all of them appear "natural." This is illustrated in course slides in which we show BRDFs that are convex combinations of the source BRDFs – these BRDFs correspond to unlikely materials. In general, we focus on modelling reflectance of real-world materials. Therefore, these limitations lead us to analyze the dataset using non-linear dimensionality reduction techniques.

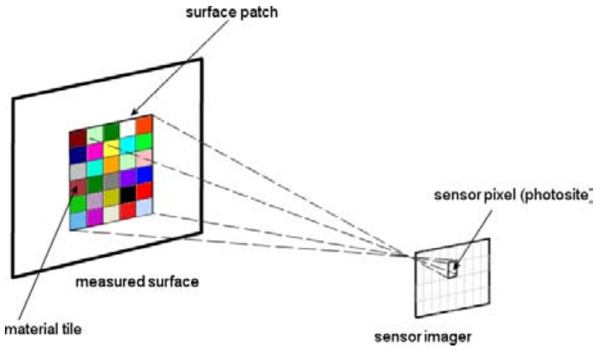


Figure 6.5: *"Manufacturing" a material with a BRDF equivalent to any convex combination of the source BRDFs.*

6.5.3 Non-linear Dimensionality Reduction

The discussion in the previous section, combined with the relative effectiveness of many low-parameter count analytic BRDF models, suggests that the space of BRDFs corresponding to materials found in nature lies on a lower-dimensional non-linear manifold that is embedded in the 45D linear subspace. We have applied recently developed non-linear dimensionality reduction techniques to discover this lower dimensional manifold. In order to construct an effective data-driven model one has to compute a continuous mapping between the embedding and the original (ambient) space.

We chose to use charting [Brand03] because it is explicitly designed to work well with a small numbers of samples and to suppress measurement noise, two conditions that tend to break methods for dimensionality reduction from local relationships. For charting, one must specify a set of chart centers, a width parameter σ for the Gaussians, and a target dimensionality d. We used the default settings: one chart centered on each data point and σ = half the average distance between each point and its closest neighbor. Note that locating a chart on a point does not cause the manifold to pass through that point – only near it.

As with PCA, the data-reconstruction error of a charted dataset gives an indication of the true dimensionality of the manifold. A plot of reconstruction error vs. target dimensionality (in course slides) shows a sharp drop in the residual error at 10D embedding space. However, when using 10D embedding space some of the reconstructed BRDFs have visual artifacts. These artifacts are not present at 15D when the residual error drops below 1% (comparable to reconstructions with 45 principal components). Observe that the reconstruction error does not decline monotonically because each dimensionality may merit a different flattening. For example, if the data were sampled from a truncated cone, the best 1D chart would simply be height along the cone, while the best 2D chart would flatten the cone into an annulus. Each flattening would suppress the noise in different directions, some more fortuitous than others.

A charted manifold of BRDF data makes it possible to treat the space of BRDFs as if it were linear, and to identify meaningful axes of variation in this embedding space. An interpolating or extrapolating line in this space is a non-linear curve in the original BRDF space that passes closer to the data density that the equivalent straight line would (on average), simply because it stays on the manifold where a straight line does not.

Chapter 7

Heterogeneous Isotropic BRDFs

Hendrik P. A. Lensch

In this chapter we will concentrate on spatially varying BRDFs, provide a categorization of existing techniques and will detail one specific acquisition technique based on clustered BRDFs and linear combinations of these.

Real-world objects are usually composed of a number of different materials that often show subtle changes even within a single material, e.g., caused by imperfections, dust, scratches, and so on. Photorealistic rendering of such objects requires accurate measurements of the reflection properties of each material, as well as the spatially varying effects.

As seen in the previous chapter, the process of measuring the BRDF of a homogeneous material is already quite demanding. In order to obtain spatially varying BRDFs of heterogeneous materials one can either try to capture a fully sampled BRDF for each individual texel (see Chapter 11 and 12) or one can combine the BRDF samples of multiple surface points in order to obtain more reliable measurements and a significant speed-up.

Spatially varying BRDFs are most often represented by a per-texel set of parameters of a BRDF model. The texture is specified over a given 3D geometry, which is either scanned, in the case of a complex object, or assumed to be flat. Fitting a BRDF model to the measured data allows for reconstruction from arbitrary view points with arbitrary illumination. Given the prior information that a BRDF model can be applied, one can extrapolate to viewing and lighting directions never seen before. One could even illuminate texels individually enabling spatially varying illumination.

Of course, the use of a BRDF model restricts the class of materials that can be represented in such a way that all reflection effects are reproduced correctly. Due to a variety of reasons the measured samples might deviate from what can be expressed by a physically plausible BRDF model. The observed 4D sampling is often called *apparent BRDF*. The apparent BRDF might be polluted by global effects such as interreflecions, subsurface scattering, or because the geometry model does represent only a coarse approximation of the real surface resulting in combining samples from different surface points, or in a wrong parameterization of the BRDF if the normal is not estimated correctly. Optimized BRDF parameters therefore can only approximate the full reflection behavior.

Existing approaches for measuring spatially varying BRDFs can be categorized with regard to the illumination setup during capture and the way spatial variation is represented. Table 7.1 lists and classifies the most common publications in the area of spatially varying BRDF estimation. The illumination is either highly controlled by using just a single point light source, the illumination is assumed to be far away, represented by an environment map, or the light sources are within the scene demanding a global inverse illumination approach in order to estimate the reflection properties.

One property of a acquisition algorithm that largely influences the reconstruction quality is the way spatial variation is represented. The best quality is achieved if a BRDF is estimated completely independent for each texel. This however, requires a sufficient number of BRDF samples per texel. If the number of input images and thus the number of samples is limited, quite often the specular part is restricted to be constant over a surface patch or the entire object. Since most objects consist of a variety of materials, each with its own diffuse and specular components, these approaches show limited approximation capabilities if the patch boundaries do not coincide with the material boundaries. A better approximation can be obtained if the material boundaries are found during a clustering process and if the per-texel BRDF is represented as a weighted sum of cluster BRDFs. We will detail this approach, which has been introduced by Lensch et al. [Lensch01, Lensch03], in the remainder of this chapter.

7.1 BRDF Measurement by Clustering

The overall pipeline for image-based measurement of spatially varying materials from a sparse set of images consists of the following steps:

- acquisition of 3D geometry
- capturing high dynamic range images with varying illumination
- resampling
- clustering of surface points into clusters of similar materials
- fitting a BRDF model to the material clusters
- representing the BRDF at each point as a weighted sum of cluster BRDFs.

The pipeline can be reorganized and individual components can be exchanged. But many of the principal considerations will remain valid so that the insight gained form studying a single approach can be applied to a whole range of other techniques.

7.1.1 Image-based BRDF Measurement for Homogeneous Materials

Given a flat sample surface, a gonioreflectometer samples the BRDF by varying the position of a light source and a sensor over the hemisphere. At each position the reflectivity is measured for exactly one pair of incident and exitant directions. A dense sampling of the BRDF thus requires moving the light source and the sensor to a large number of different positions which is very time consuming. In Figure 7.1.1a the principle setup of a gonioreflectometer is depicted. The same effect can be obtained by keeping either the light source or the sensor fix while tilting the surface and changing its orientation instead (see Figure 7.1.1b).

A more efficient technique to acquire BRDFs of homogeneous materials has been proposed by Marschner et al. [Marschner99]. Here, the sensor is replaced by a digital camera which acts as a sensor array. Instead of a planar sample this technique uses a curved surface. Since the surface normal for a curved object is different at each point a single image of the object lit by a point light source captures a large number of different incident and exitant light directions with respect to the surface. Only a small number of images is needed for a dense sampling of the BRDF.

7.1.2 Image-Based Measurement of Material Clusters

Based on the image-based BRDF measurement technique, we have developed a method that captures the spatial variation of reflection properties across the surface just from a small number of images (around 30).

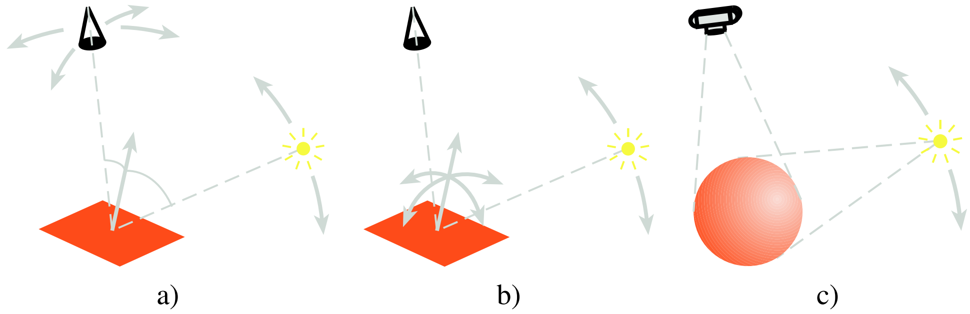


Figure 7.1: *a) Principle setup of a gonioreflectometer: The BRDF of a flat, homogeneous surface is sampled by varying the position of the light source and the detecting sensor. b) Instead of moving the sensor, the surface sample is tilted, varying the surface normal. c) In image-based BRDF measurement the sensor is replaced by a camera which captures many samples of the BRDF at one shot since the surface is curved, resulting in a different normal at each pixel.*

The goal is to determine the BRDF of each single surface point or texel. However, instead of trying to measure the BRDF for each texel separately, we first measure the BRDF of whole groups or clusters of surface points.

The approach is motivated by the fact that many man-made objects and even quite a lot of naturally grown objects are composed of only a few visually distinguishable materials or mixtures of them. The partitioning of surface points into clusters of similar material is the first step of the algorithm. In a Lloyd iteration the clusters are separated based on their actually measured BRDF samples. The measurements of each texel are compared to the current estimate of the cluster BRDF in order to decide to which cluster it might belong. While in theory a clustering based on the diffuse color is possible, taking into account the full BRDF is much more reliable and can resolve the clustering problems even for materials where only the specular component is different as demonstrated in Figure 7.2. A complete description of how the clustering is performed can be found [Lensch03].

Given a partitioning of the surface points into a finite number of clusters, a single image of a curved object already yields a more or less dense sampling of the clusters' BRDFs, like in the image-based measurement approach for homogeneous materials. Multiple images have to be taken to increase the sampling rate concerning the incident and outgoing directions, and at the same time to ensure that each point on the surface is visible in at least one image. The number of images necessary for the sampling rate in many cases matches the number of images required to fulfill the visibility constraint.

After clustering, the parameters of a BRDF model are determined for each cluster by non-linear optimization using all samples of texels within one cluster.

7.1.3 Adding Spatial Variation

Unfortunately, we are not done by just measuring and representing the BRDFs of material clusters. Assuming that all texels within one cluster show the same reflection properties does not result in realistic images since real surfaces exhibit subtle changes like dust or impurities even within the same material. Additionally, smooth transitions between materials should be modeled.

Again we have to determine the BRDF of each surface point separately. However, the small number of images yields only a very small number of samples of the BRDF of single texels (3-10 samples on average). These are clearly too sparse to

Figure 7.2: *Measured bathroom tile. The entire tile has the same diffuse color but distinct specular components for the background and the design. Left: No pattern is visible if the diffuse reflection is dominant. Middle: The differences come out only in highlight regions. Right: Close to the mirror direction the narrow specular lobe of the pattern is brighter than the background. The patterns is reverted.*

represent the BRDF directly using scattered data interpolation. Even fitting the parameters of a BRDF model will not work reliably for so few samples.

In order to model the spatial detail, we represent the BRDF at a single texel as a mixture of the BRDFs that were measured for the clusters. As the cluster BRDFs are blended by a weighted sum, we now have to determine the optimal weights for each texel. Most often a very small set of samples is enough to compute these weights reliably.

Even if there are too few samples for a texel, a plausible BRDF can be obtained since the sum is formed by reasonable BRDFs determined for whole groups of texels with a large number of samples. Further details on the representation can be found in the attached paper.

Figure 7.3: *Picture of the acquisition setup in a photo studio covered with dark felt (from left to right): point light source, metal spheres for light source tracking, object to be measured, digital still camera.*

7.2 Acquisition

Before fitting any parameters to samples, the samples have to be acquired. In this section we give a brief overview over the necessary steps, consisting of 3D scanning, calibration, high-dynamic range imaging, and so on.

We obtain the 3D models with a structured light

3D scanner and a computer tomography scanner both generating dense triangle meshes. The triangle meshes are smoothed [Garland97, Kobbelt96], manually cleaned, and decimated.

All images are acquired in a measurement lab (see Figure 7.3) using a professional digital camera. An HMI metal halide bulb serves as point light source for the BRDF measurements. The interior of the photo studio is covered with dark and diffusely reflecting felt to minimize the influence of the environment on the measurements.

Several views of each object are captured with different camera and light source positions. Light source and camera are positioned manually, which is however easily possible since only a few different views are required. For each view we acquire three sets of images: one image of the object's silhouette to register the 3D model with the images, and two images to recover the light source position (see Section 7.2.1). We then acquire a high dynamic range image of the object lit by the point light source by taking a series of photographs with varying exposure times [Debevec97]. One high dynamic range image of a gray card with known camera and light position is taken in order to allow for an absolute calibration of the reflectance.

In addition, a series of calibration images of a checkerboard pattern is taken whenever the lens settings are changed. The calibration method proposed by Zhang [Zhang99] is used to recover the intrinsic camera parameters.

To register the images with the 3D model we use a silhouette-based method [Lensch00] that yields the camera position relative to the object.

7.2.1 Recovering the Light Source Position

In order to recover the position of the point light source, a geometric approach was used which requires no user interaction. Six steel spheres of known, equal diameter are used (see Figure 7.3). A metal fixture manufactured using a CNC milling machine ensures that the spheres are lying exactly

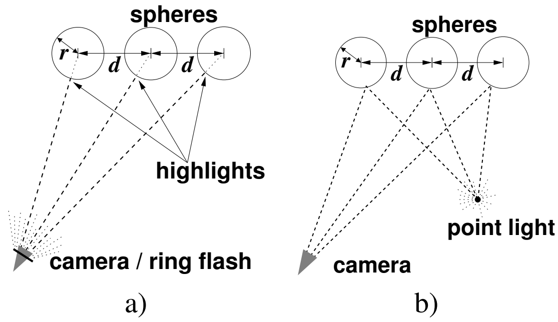


Figure 7.4: *a) the ring flash mounted on the camera yields a highlight in the center of the spheres. b) rays from the camera to the light source highlights will be reflected to the point light.*

on a straight line, and are separated by the same known distance.

For each view two images of the spheres are acquired. One view shows only the reflection of the point light source in the spheres. For the second view a ring flash mounted on the camera lens produces a highlight on the center of each sphere. The exact centers of these reflections in the images are determined automatically by fitting ellipses to the highlights.

Given the intrinsic parameters of the camera, the pixel coordinates of the reflections of the ring flash define rays in space on which the centers of the spheres are located (see Figure 7.4a). Knowing the distance between the spheres we can triangulate their positions relative to the camera.

Now, rays are sent from the camera to the positions of the light source highlights and reflected off the spheres. The light source position is located at the intersection of the rays (see Figure 7.4b). To increase the stability of this method more than the three necessary spheres are used and a least squares approximation is computed.

After measuring the light source position the metal spheres are covered with black cloth in order to avoid disturbing the acquisition of the HDR images.

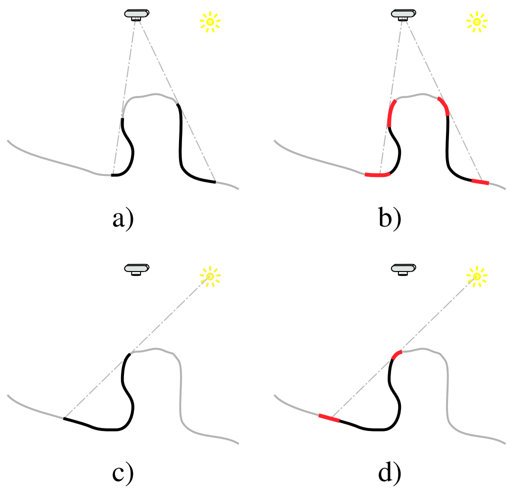


Figure 7.5: *Due to registration errors and imprecise 3D geometry depth discontinuities may cause problems. Data near depth discontinuities have to be discarded because it is not clear whether the front or the back part of the surface is visible at a point close to depth discontinuities (a and b). The same also applies to shadow boundaries where it is not clear where the shadowed area ends.*

7.3 Resampling

After acquisition of the geometric model, high dynamic range image recovery, and registration, it is necessary to merge the acquired data for further processing. For each point on the model's surface we collect the available information from all the images where the point is visible and lit.

How the points for which the data is collected are determined in order to best match the resolution of the input images and what data actually is stored per texel is described in the attached paper (Section 4).

Here, we want to point out some important issues that have to be considered when resampling the data of the input images and combining it with the geometry.

7.3.1 Discarding Data at Depth Discontinuities

In order to increase the quality of the resampled data it is sometimes necessary to discard some of the input data. Especially near depth discontinuities the resampled data is prone to registration errors and imprecise 3D geometry. If the 3D model is not perfectly aligned with the 2D image, the part of the surface that is visible in one pixel may not correspond to the surface part predicted by the 3D model. In the case of depth discontinuities the visible part and the predicted part will not even be adjacent (see Figure 7.5a). Radiance samples would then be assigned to completely wrong surface points or texels.

Furthermore, since a sensor element of the camera always integrates over a finite area, the radiance values reported at depth discontinuities are never reliable even if the registration with the 3D model were perfect. Thus, it is necessary to discard the image data at depth discontinuities as depicted in Figure 7.5b. The depth discontinuities are easily detected since the 3D model of the object is given.

The same approach also applies to the shadowing problem. Here, depth discontinuities result in shadow boundaries whose position can only be determined up to some uncertainty. Hence, also pixels near shadow boundaries have to be discarded (Figure 7.5c and d). They can be determined by a filtered shadow map.

The results of removing samples at depth discontinuities are displayed in Figure 7.6. Note that the dark stripes on the dress and across the hand have been removed by this step.

7.4 Conclusion

Putting it all together, the complete pipeline for the BRDF measurement of spatially varying materials consists of the following steps: After the acquisition of geometry and images, and the resampling, the surface points are clustered into groups showing similar reflection properties. The parameters

Figure 7.6: *Left: Dark stripes on the dress and on the hand are due to depth discontinuities. Right: By discarding samples at depth discontinuities and shadow boundaries these effects have been removed.*

of a BRDF model are fit to the acquired samples of each cluster. Finally, spatial variation is obtained by representing the BRDF of each single point as a weighted sum of BRDFs of the cluster. Please refer to the original paper for more presentations of results.

Technique	sampling	environment	interreflections	inverse lighting	specular variation	isotropic	anisotropic	normal estimation	staticdynamic
[Sato97]	sparse	PL	-	-	patch	+	-	-	s
[Lu98]	sparse	PL	-	-	object	+	+	-	s
[Yu98]	sparse	E	+	-	object	+	-	-	s
[Yu99]	sparse	E	+	-	patch	+	+	-	s
[Marschner00a]	sparse	PL	-	-	patch	+	-	-	s
[Gibson01]	sparse	E	+	+	patch	+	-	-	s
[Boivin01]	single view	E	+	-	patch	+	+	-	s
[Ramamoorthi01b]	sparse	E	-	+	object	+	-	-	s
[Nishino01]	sparse	E	-	+	object	+	-	-	s
[Li02]	single view	PL	-	-	pixel/clustered	+	-	-	s
[Lensch01, Lensch03]	sparse	PL	-	-	pixel/clustered	+	-	+	s
[Georghiades03]	sparse	PL	-	-	object	+	-	+	s
[Carceroni01]	sparse	PL	-	-	object	+	-	+	d
[Debevec00]	dense	PL	-	-	pixel	+	+	+	s
[Müller03]	dense	PL	-	-	pixel	+	+	-	s
[McAllister02]	dense	PL	-	-	pixel	+	+	-	s
[Gardner03]	dense	LL	-	-	pixel	+	-	-	s
[Goldman04]	sparse	PL	-	-	pixel/clustered	+	-	+	s
[Debevec04]	sparse	E	+	-	pixel/clustered	+	-	-	s

Table 7.1: *Comparison of algorithms acquiring spatially varying BRDFs. BRDF measurements are performed using point lights (PL), a linear light source (LL) or allowing arbitrary environments (E). Interreflections are most often ignored when a point light is applied. Only a few techniques recover BRDFs and incident illumination at the same time. Major differences are with respect to the variation of the specular part which is frequently restricted to be constant over some surface area.*

Chapter 8

Translucent Materials and BSSRDFs

Michael Goesele

8.1 Introduction

Light interacts with ideal opaque objects purely locally – it is scattered at the point of incidence according to the bidirectional reflectance distribution function (BRDF) [Nicodemus77]. In contrast, light passes straight through transparent objects (e.g., glass) and is only reflected or refracted at material boundaries. Many daily life objects (e.g., milk, skin or marble) are translucent and belong to neither of these categories. They are characterized by multiple light scattering inside the object. This *subsurface scattering* behavior leads to a very distinct appearance which is visually important: light shines through objects, they appear smooth and surface details are hidden (see Figure 8.1 for an example of the same object with and without subsurface scattering).

Figure 8.1: Photographs of an alabaster horse model with and without subsurface scattering under identical illumination conditions. The object was covered with fine dust to “turn off” subsurface scattering. Much surface detail is hidden by the translucency.

8.2 Models for Translucent Objects

Translucent objects can be modeled by defining their basic physical properties (e.g., the absorption and scattering cross sections σ_a and σ_s [Ishimaru78]) for each point inside their volume. A variety of rendering techniques such as finite element methods [Rushmeier90, Sillion95, Blasi93], bidirectional path tracing [Hanrahan93, Lafortune96], photon mapping [Jensen98, Dorsey99], Monte Carlo simulations [Pharr00, Jensen99], or diffusion [Stam95, Stam01] are able to simulate the effects of subsurface scattering based on physical principles. Subsurface scattering can also be integrated in the framework of precomputed radiance transfer [Sloan03a].

Alternatively, it is sufficient to record the visible effects of subsurface scattering. Translucent objects can be modeled using a bidirectional scattering-surface reflectance distribution func-

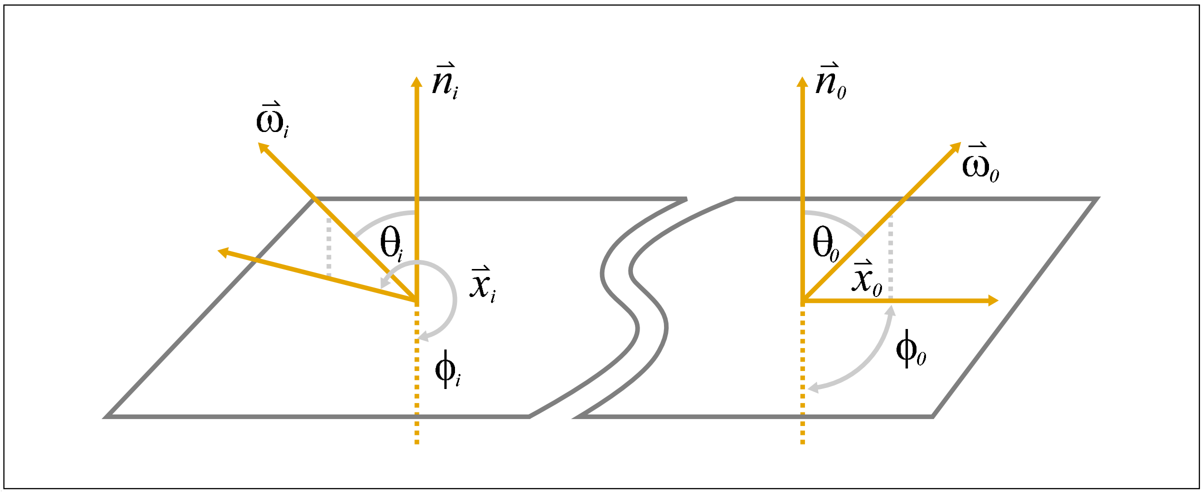


Figure 8.2: Geometric configuration of a BSSRDF.

tion $S(\vec{x}_i, \hat{\omega}_i; \vec{x}_o, \hat{\omega}_o)$ (BSSRDF, see Section 8.3) [Nicodemus77] that relates irradiance at a surface position $\vec{x}_i$ to the reflected radiance at $\vec{x}_o$ for arbitrary incoming and outgoing directions $\hat{\omega}_i$ and $\hat{\omega}_o$. The complexity of this 8D function makes handling it quite cumbersome. Many image-based acquisition and rendering techniques are able to record and reproduce some aspects of translucent objects – see e.g. Chapters 9, 11 and 12 for an overview.

8.3 The BSSRDF

The *bidirectional scattering-surface reflectance distribution function* (BSSRDF) [Nicodemus77] provides a general model of light transport inside an object and is defined as follows:

$$S(\vec{x}_i, \hat{\omega}_i; \vec{x}_o, \hat{\omega}_o) := \frac{dL^{\rightarrow}(\vec{x}_o, \hat{\omega}_o)}{d\Phi^{\leftarrow}(\vec{x}_i, \hat{\omega}_i)} \tag{8.1}$$

The BSSRDF S is the ratio of reflected radiance $L^{\rightarrow}(\vec{x}_o, \hat{\omega}_o)$ leaving the surface at a point $\vec{x}_o$ in direction $\hat{\omega}_o$ to the incident flux $\Phi^{\leftarrow}(\vec{x}_i, \hat{\omega}_i)$ arriving at a point $\vec{x}_i$ from a direction $\hat{\omega}_i$. The unit of S is $[m^{-2} \cdot sr^{-1}]$. Note that the BSSRDF is closely related to the reflectance field [Debevec00] (see Chapter 12). The only difference between them is that a reflectance field does not need to be attached to a real surface A.

Given the incoming radiance $L^{\leftarrow}(\vec{x}_i, \hat{\omega}_i)$ and the BSSRDF S for an object, the outgoing radiance $L^{\rightarrow}(\vec{x}_o, \hat{\omega}_o)$ at a point $\vec{x}_o$ in direction $\hat{\omega}_o$ can be computed by integrating over the whole surface A and all incoming directions Ω

$$L^{\rightarrow}(\vec{x}_o, \hat{\omega}_o) = \int_A \int_\Omega L^{\leftarrow}(\vec{x}_i, \hat{\omega}_i) \cdot S(\vec{x}_i, \hat{\omega}_i; \vec{x}_o, \hat{\omega}_o) \langle \hat{n}_i \cdot \hat{\omega}_i \rangle d\hat{\omega}_i d\vec{x}_i . \tag{8.2}$$

The BSSRDF varies also with the wavelength λ of the incoming light. Apart from multi-spectral applications, this dependence is in computer graphics often approximated by defining and evaluating the BSSRDFs separately per color channel. To include *fluorescence* and *phosphorescence* where light within a range of wavelengths is absorbed and re-emitted at some later time in a different range of wavelengths would require additional parameters.[1]

[1]Fluorescent materials re-emit the light no later than 10 ns after absorption whereas phosphorescent re-emit the light after more than 10 ns – sometimes even after hours.

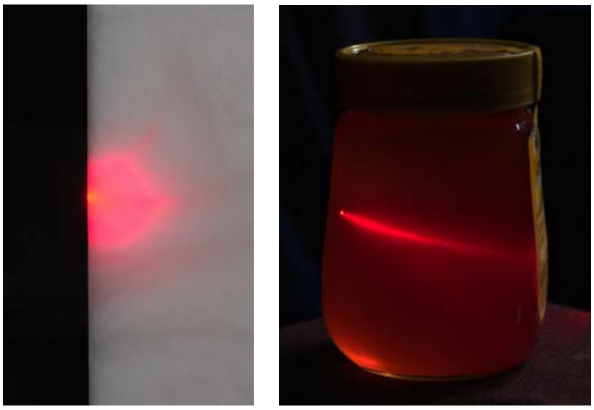


Figure 8.3: Single vs. multiple scattering. *Left:* Material dominated by multiple scattering (alabaster block). *Right:* Glass of honey whose optical properties are dominated by single scattering. Both objects are illuminated from the left by a laser beam. Note that the direction of the incoming beam is clearly visible in the right images.

8.3.1 Single vs. Multiple Scattering

A BSSRDF is often used to model the light transport in optically dense scattering materials such as skin or marble. The photons interact with the object's material so that after multiple scattering events, their propagation direction does not depend any more on the incoming illumination direction $\hat{\omega}_i$ [Ishimaru78]. It is therefore sometimes convenient to split S into a sum of a single scattering contribution $S^{(1)}$ and a multiple scattering component S_d

$$S(\vec{x}_i, \hat{\omega}_i; \vec{x}_o, \hat{\omega}_o) = S^{(1)}(\vec{x}_i, \hat{\omega}_i; \vec{x}_o, \hat{\omega}_o) + S_d(\vec{x}_i, \hat{\omega}_i; \vec{x}_o, \hat{\omega}_o).$$

They can then be treated separately, e.g., using a ray tracing approach to evaluate $S^{(1)}$ and a multiple scattering model such as the dipole approximation described in the following section for S_d [Jensen01]. The different behavior of single and multiple scattering is also illustrated in Figure 8.3.

8.4 Dipole Approximation

Unlike for BRDFs (see Chapters 6 and 7), general BSSRDF models that can be used to describe the reflection properties of subsurface-scattering material are rare. Jensen et al. [Jensen01] introduced a model for the diffuse light propagation S_d in translucent materials to computer graphics that is now widely used. The model assumes an infinite half-space of homogeneous, optically thick material with given material properties (reduced scattering coefficient σ'_s, absorption coefficient σ_a, relative index of refraction η – see Ishimaru [Ishimaru78] and Jensen et al. [Jensen01] for a description of these parameters and a method to recover them for real materials). Light propagation is modeled by a dipole approximation as follows:

$$S(\vec{x}_i, \hat{\omega}_i; \vec{x}_o, \hat{\omega}_o) = \frac{1}{\pi} F_t(\eta, \hat{\omega}_i) R_d(\vec{x}_i, \vec{x}_o) F_t(\eta, \hat{\omega}_o) \tag{8.3}$$

$$R_d(\vec{x}_i, \vec{x}_o) = \frac{\alpha'}{4\pi} \left[z_r(1 + \sigma_{tr} d_r) \frac{e^{-\sigma_{tr} d_r}}{d_r^3} + z_v(1 + \sigma_{tr} d_v) \frac{e^{-\sigma_{tr} d_v}}{d_v^3} \right] \tag{8.4}$$

Figure 8.4 shows the geometrical configuration of the setup and Figure 8.5 defines the remaining parameters. An example evaluation of the model using measured parameters for marble taken from [Jensen01] is given in Figure 8.6.

8.4.1 Determining Physical Material Parameters

The dipole model is now widely used in rendering systems to approximate multiple scattering in optically dense translucent materials. Like other physically based rendering approaches, it requires however knowledge of the relevant material properties. Jensen et al. [Jensen01] built therefore a measurement setup to determine the reduced scattering coefficient σ'_s and the absorption coefficient

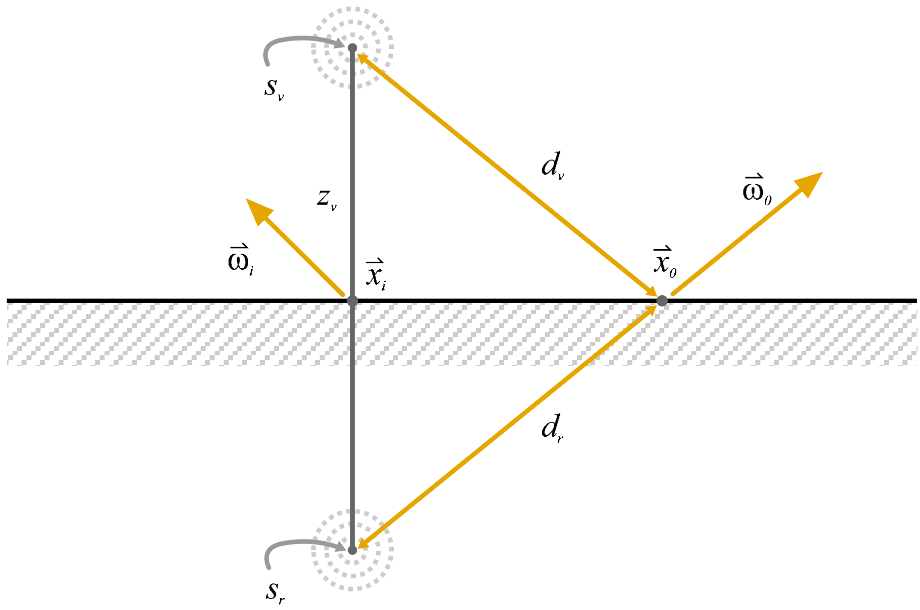


Figure 8.4: The geometry of the setup for the BSSRDF model of Jensen et al. [Jensen01].

$$
\begin{aligned}
z_r &= 1/\sigma_t' \\
z_v &= z_r + 4AD \\
d_r &= \|\vec{x}_r - \vec{x}_o\|, \text{ with } \vec{x}_r = \vec{x}_i - z_r\hat{n}_i \\
d_v &= \|\vec{x}_v - \vec{x}_o\|, \text{ with } \vec{x}_v = \vec{x}_i + z_v\hat{n}_i \\
A &= \frac{1 + F_{dr}}{1 - F_{dr}} \\
F_{dr} &= -\frac{1.440}{\eta^2} + \frac{0.710}{\eta} + 0.668 + 0.0636\eta \\
D &= 1/3\sigma_t' \\
\sigma_{tr} &= \sqrt{3\sigma_a\sigma_t'} \\
\sigma_t' &= \sigma_a + \sigma_s' \\
\alpha' &= \sigma_s'/\sigma_t'
\end{aligned}
$$

Figure 8.5: Additional definitions used in the BSSRDF model of Jensen et al. [Jensen01]. See text for more details.

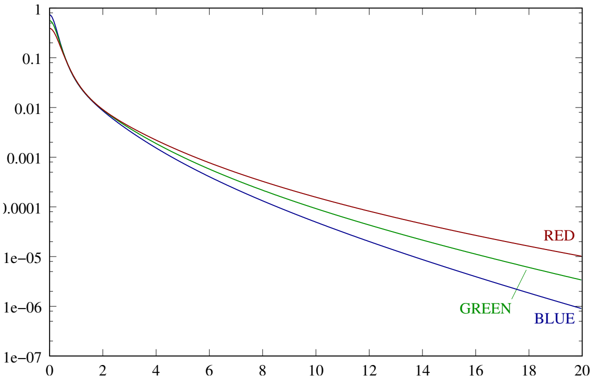


Figure 8.6: The BSSRDF model of Jensen et al. [Jensen01] evaluated for marble. The graphs show the diffuse reflectance due to subsurface scattering R_d for a measured sample of marble with σ'_s=(2.19,2.62,3.00), σ_a=(0.0021,0.0041,0.0071) and η=1.5 (parameters extracted from [Jensen01]). $R_d(r)$ indicates the radiosity at a distance r in a plane, due to unit incident power at the origin.

σ_a for various materials. Their setup illuminates a small surface area of a flat material sample with a tightly focused beam of white light along the surface normal. The resulting light distribution on the surface is observed by digital camera using a high-dynamic range imaging technique [Debevec97]. Physical parameters are determined from the acquired samples using a diffusion model of the multiple scattering process [Farell92]. [Jensen01] contains also a table with reduced scattering and absorption coefficients for 12 materials.

8.5 Hierarchical Model for Multiple Scattering

Under the assumption of diffuse multiple scattering inside the object, light transport can be modeled using the 4D diffuse subsurface reflectance function $R_d(\vec{x}_i, \vec{x}_o)$ as described in Equation 8.3. In the following section, we will introduce a method to measure $R_d(\vec{x}_i, \vec{x}_o)$ per color channel for all incoming and outgoing surface locations $\vec{x}_i$ and $\vec{x}_o$. The target object is illuminated at a single surface point $\vec{x}'_i$ with known incident radiance $L^{\leftarrow}(\vec{x}'_i, \hat{\omega}_i)$ and we can record $L^{\rightarrow}(\vec{x}_o, \hat{\omega}_o)$ on the object's surface.

Storing and rendering the full 4D function R_d with high resolution is impractical. The nature of optically dense translucent objects (see Figure 8.7) however makes R_d well suitable for a hierarchical modeling approach: R_d will typically vary strongly in the proximity of $\vec{x}_i$ due to the exponential fall-off of radiosity inside the scattering volume. In distant areas, R_d varies quite smoothly. Sharp features are usually caused by inhomogeneous material properties within a small volume dV on and right below the object's surface at $\vec{x}_i$ and $\vec{x}_o$. All photons pass through these volumes and their influence is significant (see Feng et al. [Feng93] for a study of photon path distributions within scattering material).

8.5.1 Discretization of Light Transport

The hierarchical BSSRDF model is based on a discrete version of the integral expression in Equation 8.2 with the diffuse scattering assumption from Equation 8.3. For each hierarchy level, we introduce a set of basis functions for the discretization. Multiple discrete representations of the same data are of course redundant but allow us to adapt the sampling density to the smoothness of R_d. Appropriate blending will be necessary for correct results when combining them into a single model.

On the finest level of the hierarchy we generally employ a set of piecewise constant basis functions assuming a sufficiently high sampling density to model the objects response in the vicinity of an incoming light impulse. At coarser levels, which will be used to represent the smoother global response, we use more elaborate basis functions such as hat functions.

For the discretization, we first reformulate Equation 8.2 in order to be able to address the irradiance $E(\vec{x}_i)$ collected at a surface point $\vec{x}_i$ and the radiosity $B(\vec{x}_o)$ at the outgoing surface position $\vec{x}_o$

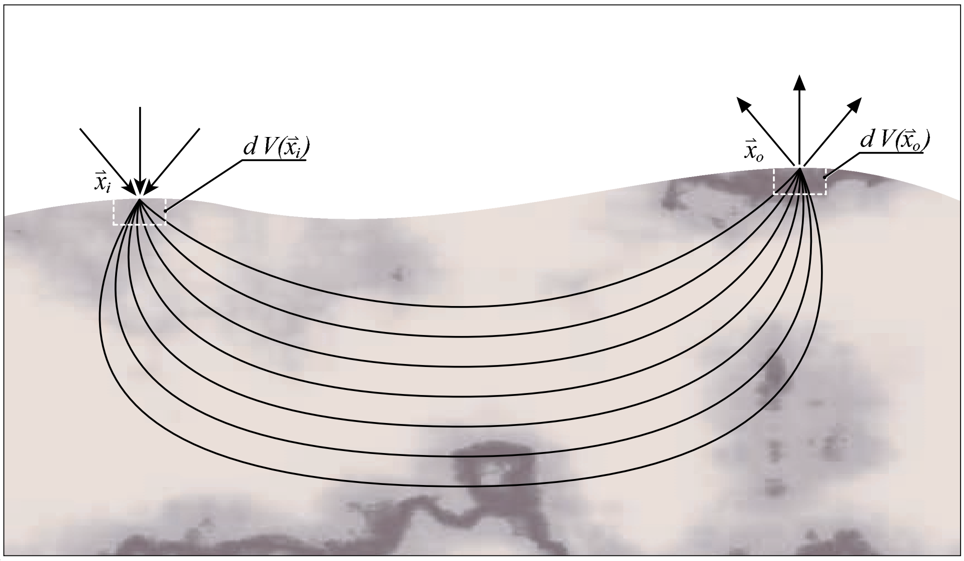


Figure 8.7: Light propagation in translucent objects. All photons incident at $\vec{x}_i$ and exiting at $\vec{x}_o$ must pass through two small volumes dV around $\vec{x}_i$ and $\vec{x}_o$ while their paths are spread out inbetween.

individually:

$$L^{\rightarrow}(\vec{x}_o, \hat{\omega}_o) = \frac{1}{\pi} F_{t,o}(\eta, \hat{\omega}_o) B(\vec{x}_o) \quad (8.5)$$

$$B(\vec{x}_o) = \int_A R_d(\vec{x}_i, \vec{x}_o) E(\vec{x}_i) d\vec{x}_i \quad (8.6)$$

$$E(\vec{x}_i) = \int_{\Omega_+(\vec{x}_i)} L^{\leftarrow}(\vec{x}_i, \hat{\omega}_i) F_{t,i}(\eta, \omega_i) \langle \hat{n}_i \cdot \hat{\omega}_i \rangle d\hat{\omega}_i. \quad (8.7)$$

For each of the discretizations we first choose a set of spatial basis functions $\psi_u(\vec{x})$ and project the irradiance $E(\vec{x}_i)$ onto the chosen basis:

$$\tilde{E}_u(\vec{x}_i) = \sum_u E_u \psi_u(\vec{x}_i) \approx E(\vec{x}_i)$$

The coefficients E_u are found by calculating scalar products of $E(\vec{x}_i)$ with dual basis functions $\tilde{\psi}_u(\vec{x})$:[2]

$$E_u = \int_A E(\vec{x}_i) \tilde{\psi}_u(\vec{x}_i) d\vec{x}_i \quad (8.8)$$

[2] The dual basis functions are the unique set of linear combinations of the primary basis functions $\psi(\vec{x})$ that fulfill the orthonormality relations $\int_A \psi_u(\vec{x}) \tilde{\psi}_v(\vec{x}) d\vec{x} = \delta_{uv}$. The Kronecker delta function δ_{uv} is defined as $\delta_{uv} = 1$ if $u = v$ and $\delta_{uv} = 0$ otherwise.

Equation 8.6 is then transformed into a matrix-vector multiplication:

$$B_v = \sum_u E_u \mathbf{F}_{u,v} \quad (8.9)$$

with throughput factors

$$\mathbf{F}_{u,v} = \int_A \int_A \psi_u(\vec{x}_i) R_d(\vec{x}_i, \vec{x}_o) \tilde{\psi}_v(\vec{x}_o) d\vec{x}_o d\vec{x}_i. \quad (8.10)$$

The radiosity $B(\vec{x}_o)$ at surface position $\vec{x}_o$ is reconstructed as

$$B(\vec{x}_o) \approx \sum_v B_v \psi_v(\vec{x}_o). \quad (8.11)$$

The conversion of radiosity into outgoing radiance $L^{\rightarrow}(\vec{x}_o, \hat{\omega}_o)$ for a particular direction $\hat{\omega}_o$ can then be performed according to Equation 8.5.

8.5.2 Modulation of Smooth Hierarchy Levels

Using a coarse hierarchy level to model the scattering behavior of an object will lead to a smooth radiosity distribution on its surface (depending on

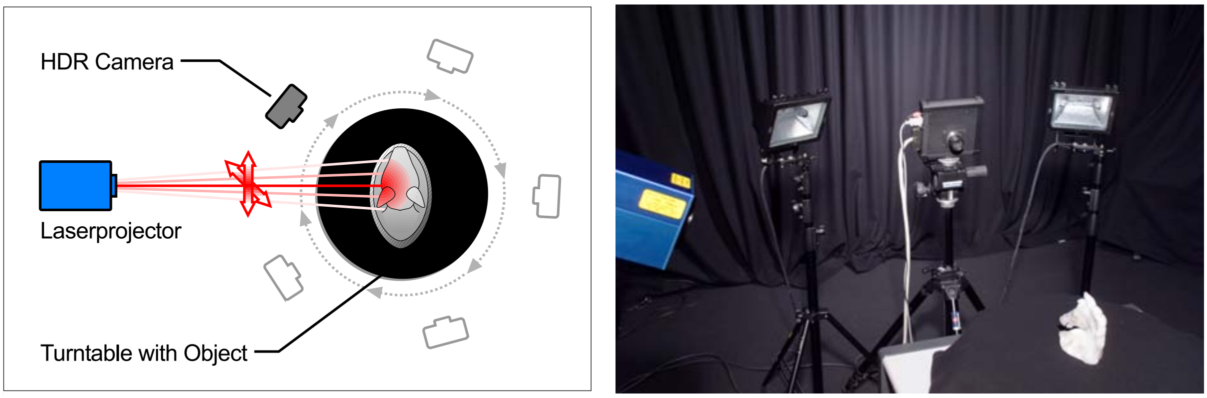


Figure 8.8: Acquisition setup. The laser projector illuminates the target object (resting on the turntable). The scene is observed by the HDR video camera from several positions. The two spotlights visible on both sides of the camera in the right image are used to illuminate the object during silhouette detection.

the sample spacing and the chosen set of basis functions $\psi_u(\vec{x})$). For an object with inhomogeneous scattering properties, that exhibits high frequency variations on the surface as shown in Figure 8.7, this will lead to a too smooth reconstruction hiding important details.

If we assume that the high frequency variations depend on the properties of a small volume $dV(\vec{x}_o)$ surrounding an exitant surface location $\vec{x}_o$, the effect of the details is independent of $\vec{x}_i$ and applies to all discretized radiosity contributions B_v. We can therefore introduce a modulation factor $T_\rho(\vec{x}_o)$ that modulates the computed radiosity for each surface location. Using Equation 8.6, $T_\rho(\vec{x}_o)$ should be chosen such that the objective function

$$O = \int_A R_d(\vec{x}_i, \vec{x}_o) E(\vec{x}_i) d\vec{x}_i - T_\rho(\vec{x}_o) \cdot \sum_v \left(\psi_v(\vec{x}_o) \cdot \sum_u E_u \mathbf{F}_{u,v} \right) \tag{8.12}$$

is minimal for all incoming surface positions $\vec{x}_i$.

Overall, the outgoing radiance $L^{\rightarrow}(\vec{x}_o, \hat{\omega}_o)$ of one hierarchy level is now approximated by

$$L^{\rightarrow}(\vec{x}_o, \hat{\omega}_o) \approx \frac{1}{\pi} F_{t,o}(\eta, \hat{\omega}_o) \cdot T_\rho(\vec{x}_o) \cdot \sum_v \left(\psi_v(\vec{x}_o) \cdot \sum_u E_u \mathbf{F}_{u,v} \right). \tag{8.13}$$

8.6 DISCO Acquisition System

In this next sections we will describe DISCO – a method to acquire a model for real world translucent objects introduced by Goesele et al. [Goesele04]. The system employs a laser projection system to sequentially illuminate individual surface locations $\vec{x}_i'$ with a narrow beam of light (see Figure8.8 for an overview over the acquisition setup). Under the assumption of diffuse light propagation inside the translucent object, the objects impulse response corresponds to the diffuse subsurface scattering reflectance function R_d. The outgoing radiance can be captured by a high-dynamic range video camera. Given the observations of the HDR video camera, we can now determine $R_d(\vec{x}_i', \vec{x}_o)$ for all visible surface locations and store them in the hierarchical model.

Figure 8.9: *Left:* Rendering of the horse model acquired by the DISCO method. Differences in material are clearly visible when the object is lit from behind. *Right:* Photograph of the horse sculpture taken under similar lighting conditions.

8.6.1 Acquisition Results

Figures 8.9, 8.10, and 8.11 show renderings of two acquired models – an alabaster horse sculpture and a starfruit (carambola). The alabaster horse has strong variation in its subsurface scattering properties and complex geometry. It consists of regions with varied translucency and contains cracks inside the material. The starfruit is an example of a translucent biological object with concave geometry and a relatively specular surface. It is evident that the acquisition system and the proposed hierarchical model reproduce the multiple scattering behavior quite well. Artifacts are however visible for surface areas with non-diffuse BRDF and strong single scattering contribution as these effects are not included in the scattering model.

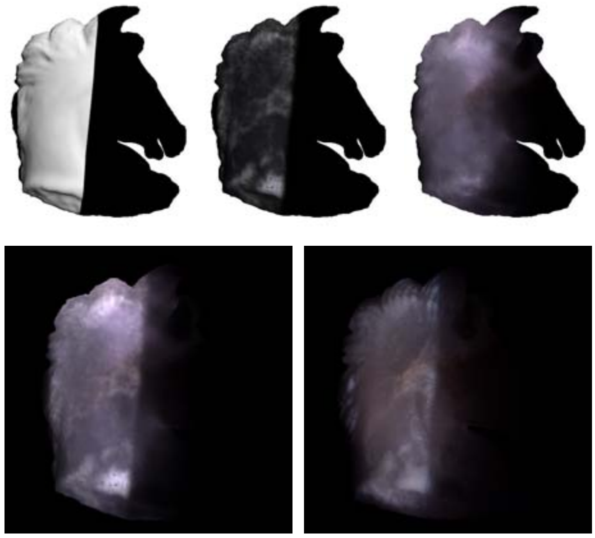


Figure 8.10: The horse model with two hierarchy levels. *Top:* Irradiance, local light transport by filter kernels, and global light transport due to a global throughput factor matrix. *Bottom left:* Combined rendering of local and global term. The local light transport adds surface detail and gives an impression of the material properties at the object's surface whose basic structure is also visible in the global term. *Bottom right:* Photograph of the real object under similar conditions. A slide projector was used to produce the sharp shadow boundary.

Figure 8.11: Starfruit model. *Left:* The hierarchical model of the starfruit is illuminated from the front right. *Right:* The same model illuminated from the back.

Chapter 9

Transparent and Specular Materials, Environment Matting

Yung-Yu Chuang

In this chapter, we describe methods for capturing object's light transport due to refraction and reflection. One naïve way is to acquire an accurate 3D geometric model and accurate refractive indices or reflective properties for the object. However, obtaining these quantities in itself may be very difficult. Instead of capturing the 3D shape and material properties, in this chapter, we describe techniques for *environment matting*, a method to parameterize surface reflectance and refractive properties of the object into a 2D matte for a fixed viewpoint.

Environment matting is a generalization of traditional bluescreen matting. In contrast to traditional matting, it captures not just a foreground object and its traditional opacity matte from a real-world scene, but also a description of how object refracts and reflects light, which we call an *environment matte*. Figure 9.1 shows a water goblet captured by traditional matting and environment matting.

(a) (b)

Figure 9.1: Comparison of traditional matting and environment matting. Traditional mattes (a) only model the color blending effects for transparency while environment mattes (b) also record how light from the surrounding environment interacts with the object.

9.1 Introduction

Matting and compositing are fundamental operations in graphics. In the matting process, a foreground element of arbitrary shape is extracted from a background image. The matte extracted by this process describes the opacity of the foreground element at every point. In the compositing process, the foreground element is placed over a new background image, using the matte to hold out those parts of the new background that the foreground element obscures.

Although matting and compositing have proven tremendously useful in film, video, and computer graphics production, they nevertheless fail to simulate two key effects that are essential for realism:

- refraction, exhibited by transparent objects; and
- reflection, exhibited by shiny objects, objects seen at a grazing angle.

Moreover, in the case of translucent or glossy materials, refraction and reflection are further coupled with various light scattering effects. For colored materials, refraction and reflection may also exhibit selective attenuation and scattering of light, according to wavelength.

In 1999, Zongker *et al.* [Zongker99] introduced *environment matting and compositing*, a process which generalizes the traditional matting and compositing processes to incorporate all of these effects. The resulting environment matte record how a foreground object refracts and reflects light. With the environment matte, the foreground object can then be placed in a new environment, where it will refract and reflect light from that scene.

9.2 The Environment matting equation

We begin by developing a general expression for the environment compositing equation and showing how it reduces to the traditional compositing equation.

An imaging system, such as a CCD camera, records a discrete set of samples over an image plane. Let's assume for the moment that we have a camera that measures the irradiance at each wavelength separately. Then, for a given pixel p, the camera records a value C for each wavelength.[1] Following the environment mapping work of Blinn and Newell [Blinn76], we can express this color in terms of an infinitely distant environment illumination $\boldsymbol{E}(\boldsymbol{\omega})$,

$$C \;=\; \int \boldsymbol{W}(\boldsymbol{\omega})\, \boldsymbol{E}(\boldsymbol{\omega})\, d\boldsymbol{\omega}\,. \tag{9.1}$$

The weighting function $\boldsymbol{W}$ comprises all means of transport of environment lighting from all directions $\boldsymbol{\omega}$ through a foreground object to the camera, including any blurring due to the camera optics and area integration at a sensor cell. This equation holds under the assumption that none of the materials that are scattering light from the environment exhibit any wavelength coupling (e.g., fluorescence).

Next, we rewrite this equation as a spatial integral over a bounding surface (e.g., an environment map). Further, we augment the equation to include an additive foreground color F. This foreground color is typically due to some additional lighting that is separate from the environment map, though it could encompass object emissivity as well. Under these assumptions, our equation becomes

$$C \;=\; F \;+\; \int \boldsymbol{W}(\boldsymbol{x})\, \boldsymbol{B}(\boldsymbol{x})\, d\boldsymbol{x}\,, \tag{9.2}$$

where $\boldsymbol{B}(\boldsymbol{x})$ is the environment map. In this chapter, for simplicity, we restrict ourselves with the situation that the environment map consists of only the background face, but it is not difficult to extend to a full environment. We call this equation *environment matting equation*. From this equation, we can develop a series of approximations that allow us to embed a foreground object in a new environment with varying degrees of quality.

To arrive at the traditional compositing equation [Brinkman99], we assume that the straight-through background pixel is the only environment sample that affects the camera pixel. Let $\sqcap_p$ be the rectangular-area support of the pixel p on the background. We describe the (in this case, monochromatic) weighting function as

$$\boldsymbol{W}(\boldsymbol{x}) \;=\; (1-\alpha)\, \boldsymbol{\Pi}(\boldsymbol{x}; \sqcap_p)\,, \tag{9.3}$$

where α represents the foreground's transparency or partial pixel coverage, and $\boldsymbol{\Pi}(\boldsymbol{x}; A)$ is the box

[1] Throughout this chapter, the derivations of equations are for a pixel p of the foreground plate. For example, C is for $C(p)$ and $\boldsymbol{W}(\boldsymbol{x})$ is for $\boldsymbol{W}(\boldsymbol{x}; p)$. Exceptions are $\boldsymbol{E}(\boldsymbol{\omega})$ and $\boldsymbol{B}(\boldsymbol{x})$, which represent the environment and are hence independent of p.

function of unit volume supported over an arbitrary axis-aligned area A. Next we define $\mathcal{M}(\boldsymbol{B}, A)$ as the "texture-mapping operator" that performs the area integral and returns the average value of the texture $\boldsymbol{B}$ over region A,

$$\mathcal{M}(\boldsymbol{B}, A) \equiv \int \boldsymbol{\Pi}(\boldsymbol{x}; A)\, \boldsymbol{B}(\boldsymbol{x})\, d\boldsymbol{x}\,. \quad (9.4)$$

Finally, defining the filtered background B to be the integral over the pixel's support,

$$B \equiv \mathcal{M}(\boldsymbol{B}, \sqcap_p)\,, \quad (9.5)$$

and substituting the previous three equations into equation (9.2), gives the traditional compositing equation,

$$C = F + (1-\alpha)\, B\,. \quad (9.6)$$

Note that α typically does not have any wavelength dependence and thus cannot model color-filtered transparency. In addition, F is a measured quantity that is added directly to the attenuated background—in effect, it is pre-multiplied by α.

Environment matting approaches often involves making assumptions on the weighting function, simplifying the environment matting equation, photographing the object in front of a series of different backgrounds and estimating the environment matte by observing how the backgrounds are distorted. The backgrounds can be either controlled (active approaches) or uncontrolled (passive approaches). In the next two sections, we will discuss these approaches.

9.3 Active approaches

Active approaches estimate the weighting function by photographing the object in front of a series of controlled stimulus backdrop patterns displayed by a light emitter (usually a monitor) as shown in Figure 9.2. In what follows, we call the displayed stimulus patterns the *backdrop image*, and the image of the foreground object in front of the backdrop the *object image*. These methods mainly differ in what approximation is made on the weighting function and how the stimulus backdrop function is designed accordingly to reconstruct the weighting function.

Figure 9.2: A photograph of the experimental setup used to capture environment mattes. The camera in the foreground photographs objects surrounded by stimulus backdrop patterns displayed on the computer monitors. The image is processed to extract only the area covered by the backdrop pattern, resulting in an image as shown in the inset.

9.3.1 Hierarchical environment matting (HEM)

In their pioneer paper, Zongker *et al.* [Zongker99] introduce the first environment matting method by approximating the weighting function as

$$\boldsymbol{W}(\boldsymbol{x}) = (1-\alpha)\, \boldsymbol{\Pi}(\boldsymbol{x}; \sqcap_p) + R\, \boldsymbol{\Pi}(\boldsymbol{x}; A)\,, \quad (9.7)$$

where A represents an axis-aligned region on the background and R is reflectance coefficient describing the amount of light from the designated area of background that is reflected or transmitted by the object at a given wavelength. In this formulation, R captures color-filtered transparency, and α represents only partial pixel coverage of the object. Substituting this weighting function into equation (9.2) gives the environment compositing equation used by Zongker *et al.* [Zongker99],

$$C = F + (1-\alpha) B + R \mathcal{M}(\boldsymbol{B}, A). \quad (9.8)$$

Hence, the environment matte for each pixel is represented by the foreground color F, the partial coverage α, a single reflection coefficient R and a normalized box filter on a rectangular support area A on the backdrop.

To recover the foreground color F and the reflection coefficient R, we can photograph the object against two solid backgrounds . Replacing $\boldsymbol{B}$ in equation (9.8) with constant colors B_0 and B_1, we obtain

$$\begin{aligned} C_0 &= F + (1-\alpha) B_0 + R B_0 , \\ C_1 &= F + (1-\alpha) B_0 + R B_1 . \end{aligned}$$

We now have two equations in two unknowns, which are easily solved for R and F, expressed here as functions of α:

$$\begin{aligned} R(\alpha) &= (C_0 - C_1)/(B_0 - B_1) - (1-\alpha) \\ F(\alpha) &= C_0 - (1 - \alpha + R(\alpha)) B_0 \end{aligned}$$

To determine α for pixels on the boundary and the axis-aligned rectangle A that best approximates the reflection and refraction in the scene, an objective function over a series of n photographed images for each pixel in the scene is minimized:

$$E = \sum_{i=1}^{n} \| \quad C_i - F(\alpha) - (1-\alpha)B_i - R(\alpha)\mathcal{M}(\boldsymbol{B}_i, A)\|^2, \quad (9.9)$$

where C_i and B_i are the colors of the pixel in question in the object image and the backdrop image, respectively, when the i-th patten $\boldsymbol{B}_i$ is displayed as a backdrop.

The goal will be to find the rectangular area A and α that minimize the objective function E. The rectangle A can be parameterized by its left, right, top and bottom (l, r, t, b). The dimensionality can be further reduced by using horizontal and vertical stripes for backgrounds as shown in Figure 9.3. Hence, (l, r) can be optimized using only vertical stripes and (t, b) can be optimized using only horizontal stripes. To solve the optimization problem, α is set to nine values, $\alpha \in \{\frac{1}{8}(0, 1, 2, \cdots, 8)\}$, and then (l, r) and (t, b) are optimized for each α. The combination of (α, l, r, t, b) leading to the smallest E is used as the environment matte. We call this approach hierarchical environment matting because it uses hierarchical stripes as the stimulus function.

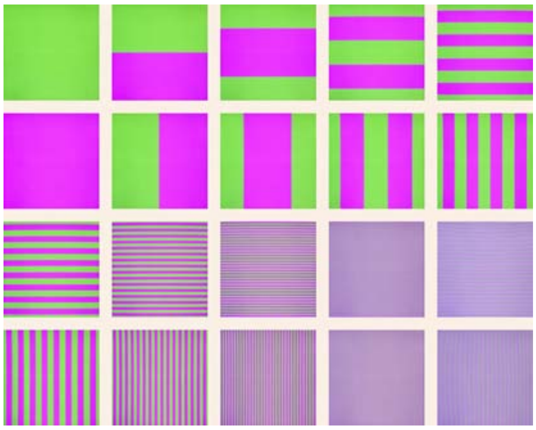


Figure 9.3: Hierarchical horizontal and vertical stripe patterns used in hierarchical environment matting [Zongker99].

This formulation not only permits colored filtering of light, but also enables effects such as reflection and refraction since the light contributing to a pixel can be scattered from parts of the environment other than just the pixel directly behind the object. Hence, it is effective for refraction, translucency, highly specular, gloss and color transparency.

This approach, however, does have several distinct limitations. First, the components of the weighting function are assumed to be separable products of wavelength functions R_i and spatial functions $\boldsymbol{\Pi}(\boldsymbol{x}; A_i)$. Thus, phenomena such as dispersion are not handled (Figure 9.7(a)), since these require the weighting functions to shift spatially with wavelength. Second, the axis-aligned rectangle weighting functions do not simulate the effects of, for example, smooth BRDF's, which when mapped onto a background have a smooth, oriented footprint (Figure 9.7(b)). Finally, other than the straight-through α-component, the approach models only a single mapping from the background to the camera. In reality, multiple mappings to the same face can and do happen and must be modeled, for example, when reflection and refraction at an interface cause view rays to split into distinct groups that collect light from the same background (Figure 9.7(c)). In addition, this approach requires $O(\log k)$ (typically about 20) images for extracting an environment matte of resolution $k \times k$ and thus does not lend itself to real-time acquisition.

In the following sections, we describe the follow-on methods attempting to either improve accuracy of the resulting matte or allow real-time acquisition.

9.3.2 Real-time environment matting (RTEM)

To address the problem of requiring too many images, Chuang *et al.* [Chuang00] introduce real-time environment matting which requires only one backdrop. It is interesting for two reasons. First, it represents a definite, extreme end of the image-count-versus-accuracy spectrum—a sort of lower bound on the quality of the whole environment matting technique. Secondly, unlike a solution that requires even two or three images of the object, a single-image solution makes it straightforward to capture video environment mattes of *moving* objects in front of a still, structured background. Hence, real-time EM could be used to capture breaking glass, sloshing liquids, and other kinds of non-repeatable, uninterruptible motions. Although the data capture itself is real-time, the matte extraction process is performed off-line at a slower speed. The matte extraction process analyzes the frames of a captured video and constructs a matte for each frame.

In the setup for real-time environment matting, we only have three samples per pixel: red, green, and blue. The first objective, then, is to simplify the problem just enough to be solvable—that is, so that there will be only three unknowns remaining. Simplifications are chosen to maximize the visual impact of the final matte.

Starting from Equation (9.8), Chuang *et al.* first assume there is no foreground color (the backdrop is the only light source in the scene) and ignore the partial coverage around the object boundary α. Furthermore, they assume that the object is both colorless and specularly reflective and refractive (i.e., has no roughness or translucency). If the object is colorless, then R becomes a scalar ρ, with no wavelength dependence. In addition, pure specularity implies that neighboring pixels do not have overlapping support in their weighting functions. If we break the A term into $\{\boldsymbol{c}, \boldsymbol{w}\}$, where $\boldsymbol{c} = (c_x, c_y)$ is the center of the area and $\boldsymbol{w} = (w_x, w_y)$ is the width in x and y. Thus, $\boldsymbol{c}$ is an image warping function, and $\boldsymbol{w}$, derived from the warping function, indicates the size of the filter support for proper antialiasing [Wolberg90]. The following approximation are used for $\boldsymbol{w}$:

$$\begin{aligned} w_x &\equiv \frac{\partial}{\partial x} c_x \approx \frac{1}{2}\left[c_x(x+1, y) - c_x(x-1, y)\right] \\ w_y &\equiv \frac{\partial}{\partial y} c_y \approx \frac{1}{2}\left[c_y(x, y+1) - c_y(x, y-1)\right]. \end{aligned}$$

Thus, the environment matting equation becomes

$$C = \rho \mathcal{M}(B, A), \tag{9.10}$$

with only three variables ρ, c_x and c_y.

What kind of stimulus function could we use to recover all the parameters in this problem? A logical choice would be a smooth function, where we define smoothness as

$$\mathcal{M}(B, A) \approx B(\boldsymbol{c}) \tag{9.11}$$

for any area A. Such a function would have the property under our simplified model that

$$C \approx \rho B(\boldsymbol{c}). \tag{9.12}$$

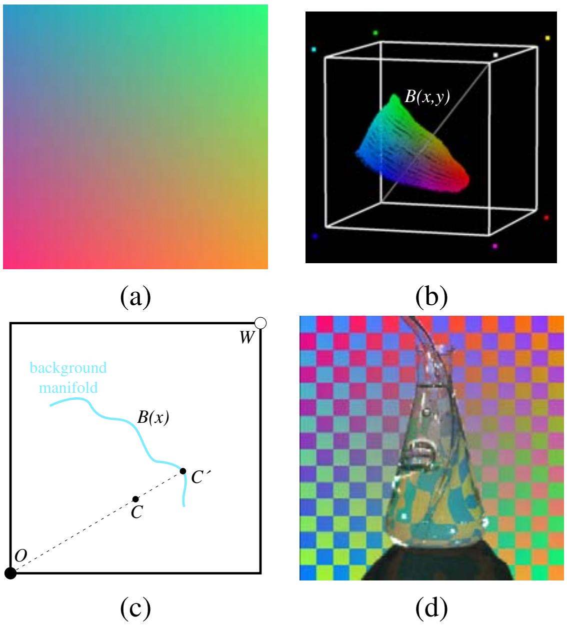


Figure 9.4: (a) The smooth color ramp used for real-time environment matting. b) In reality, the background colors lie on a 2D manifold within the *rgb* cube (c) 2D version of the RTEM algorithm for constructing environment mattes from a single image. The observed color C is projected from black onto the background manifold. The position of point C' on the manifold gives the position x, and $\rho = \overline{OC}/\overline{OC'}$. (d) Sample composite using the real-time environment matting technique.

Treating this equation as three equations in r, g, and b, we can easily solve for the three unknowns ρ, c_x, c_y. Backgrounds that are smooth according to the definition in Equation (9.11) include constant color functions and linear color ramps. However, this function must also be invertible, so that we can identify which pixel we are seeing from its color. Obviously, a constant color function does not fill this requirement. Hence, a color ramp background is used (Figure 9.4(a)). Due to nonlinearities in the system, including crosstalk between the spectra of the monitor phosphors and the CCD elements, the gamma of the backdrop display, and processing in the camera's electronics, this plane in color space will be distorted, becoming a curved 2D manifold lying within the *rgb* cube, as in Figure 9.4(b).

To extract matte parameters at each pixel, we consider the line joining the observed color and the black point in *rgb* space. The point where this line intersects the background-color manifold gives us the point $\boldsymbol{c}$, and the fractional distance of the observed color to the manifold gives us ρ, as illustrated in Figure 9.4(c). The manifold is difficult to characterize algebraically, so rather than projecting the observed color onto it, a multiresolution search is used to find the point on the manifold closest to the construction line.

Because a single frame of video captured with a CCD camera will have considerable noise, this leads to grainy mattes. To combat this effect, the extracted warping function, $\boldsymbol{c}$ is smoothed by an edge-preserving smoothing operator of Perona and Malik [Perona90]. Finally, heuristics are used to estimate specular highlights as the foreground color to add realism. Figure 9.4(d) shows an example composite of real-time environment matting.

9.3.3 Gaussian environment matting (GEM)

In the previous section, we considered ways to restrict the input to just a single image of the object at the expense of realism. For the following sections, we will attempt to go the other way— to increase the accuracy of mattes produced with the environment matting technique at the expense of increasing the number of input images required.

Inspired by the BRDF fitting work of Ward [Ward Larson92], Chuang *et al.* [Chuang00] approximate the weighting function as a sum of Gaussians,

$$\boldsymbol{W}(\boldsymbol{x}) = \sum_{i=1}^{m} R_i \, \boldsymbol{G}_i(\boldsymbol{x}) \, . \tag{9.13}$$

This allows any number of contributions from the background. Here, R_i is an attenuation factor, and each $\boldsymbol{G}_i$ is a normalized, elliptical, oriented $2D$ Gaussian,

$$\boldsymbol{G}_i(\boldsymbol{x}) \;\equiv\; \boldsymbol{G}_{\text{2D}}(\boldsymbol{x};\; \boldsymbol{c}_i,\, \boldsymbol{\sigma}_i,\, \theta_i)\,, \qquad (9.14)$$

where $\boldsymbol{G}_{\text{2D}}$ is defined as

$$\boldsymbol{G}_{\text{2D}}(\boldsymbol{x};\boldsymbol{c},\boldsymbol{\sigma},\theta) \equiv \frac{1}{2\pi\sigma_u\sigma_v} e^{\left[-\frac{u^2}{2\sigma_u{}^2}-\frac{v^2}{2\sigma_v{}^2}\right]} \quad (9.15)$$

with

$$u \;=\; (x-c_x)\cos\theta - (y-c_y)\sin\theta\,, \quad (9.16)$$
$$v \;=\; (x-c_x)\sin\theta + (y-c_y)\cos\theta\,. \quad (9.17)$$

Here, $\boldsymbol{x} = (x, y)$ are the pixel coordinates, $\boldsymbol{c} = (c_x, c_y)$ is the center of each Gaussian, $\boldsymbol{\sigma} = (\sigma_u, \sigma_v)$ are the "unrotated" widths (a.k.a. standard deviations) in a local uv-coordinate system, and θ is the orientation. Figure 9.5 illustrates these parameters. Thus, the weighting function is some m-modal Gaussian with each term contributing a reflective or refractive effect from the object. Substituting into equation (9.2), we arrive at a new form of the compositing equation,

$$C = F + \sum_{i=1}^{m} R_i \int \boldsymbol{G}_{\text{2D}}(\boldsymbol{x},\boldsymbol{c}_i,\boldsymbol{\sigma}_i,\theta_i)\boldsymbol{B}(\boldsymbol{x})d\boldsymbol{x}. \quad (9.18)$$

The key advantages of this weighting function over the one used by Zongker *et al.* are that: (1) the spatial variation can be coupled with wavelength to permit modeling of dispersion; (2) it supports multiple mappings to a single texture; and (3) it approximates the behavior of BRDF's more closely (using oriented Gaussian weighting functions rather than axis-aligned box functions).

In practice, each of the values C, F, R_i, $\boldsymbol{c}_i$, $\boldsymbol{\sigma}_i$, θ_i and $\boldsymbol{B}$ in equation (9.18) is implemented as an *rgb* vector. So, this equation actually represents three independent equations, one for each of the color components. Our unknowns are $F, R_i, \boldsymbol{c}_i, \boldsymbol{\sigma}_i, \theta_i$, which means that each pixel encodes $3 + 18m$ parameters.

To recover these parameters, Chuang *et al.* use a set of swept Gaussian stripes as the stimulus function. Let's see how sweeping stripes is used to recover the parameters of the weighting functions.

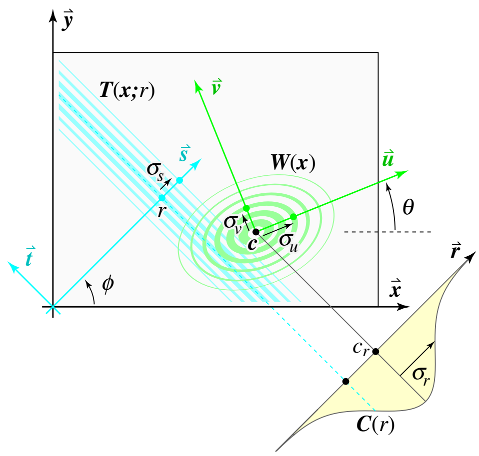


Figure 9.5: Illustration of the variables used in recovering an unknown elliptical, oriented Gaussian by sweeping out convolutions with known Gaussian stripes. As a tilted stripe $\boldsymbol{B}(\boldsymbol{x}; r)$ of width σ_s and position r sweeps across the background in direction s, it passes "under" the elliptical Gaussian weighting function $\boldsymbol{W}(\boldsymbol{x})$ associated with a single camera pixel. The camera records the integral of the product of the stripe and the weighting function, which describes a new, observed function $C(r)$ as the stripe sweeps. The center c_r and width σ_r of this observed function are related to the center $\boldsymbol{c}$ and width $\boldsymbol{\sigma} = (\sigma_u, \sigma_v)$ of the weighting function and the width of the stripe through equation (9.25).

To begin, let us assume that the weighting function is unimodal and axis-aligned ($m = 1$ and $\theta = 0$). Under these assumptions, we can decompose the 2D Gaussian weighting function into two 1D components,

$$\boldsymbol{W}(\boldsymbol{x}) = R\, \boldsymbol{G}_{\text{1D}}(x; c_x, \sigma_u)\, \boldsymbol{G}_{\text{1D}}(y; c_y, \sigma_v)\,, \quad (9.19)$$

where

$$\boldsymbol{G}_{\text{1D}}(x; c, \sigma) \;=\; \frac{1}{\sqrt{2\pi}\sigma} e^{\left[-\frac{(x-c)^2}{2\sigma^2}\right]}. \qquad (9.20)$$

The first stimulus function will be a vertical stripe that is constant in y and has a 1D Gaussian intensity profile in x with width σ_s,

$$\boldsymbol{B}(\boldsymbol{x}) = \boldsymbol{G}_{\text{1D}}(x; 0, \sigma_s)\,. \tag{9.21}$$

Now consider sweeping the stripe horizontally, displacing it at each step by some offset r,

$$\boldsymbol{B}(\boldsymbol{x}; r) = \boldsymbol{G}_{\text{1D}}(x-r; 0, \sigma_s)\,. \tag{9.22}$$

The camera observation at a pixel is then given by

$$\begin{aligned} C(r) &= \int \boldsymbol{W}(\boldsymbol{x})\, \boldsymbol{B}(\boldsymbol{x}; r)\, d\boldsymbol{x} \\ &= R\, \boldsymbol{G}_{\text{1D}}(r; c_x, \sqrt{{\sigma_u}^2 + \sigma_s^2})\,. \end{aligned} \tag{9.23}$$

Thus, at each pixel, we expect to record a Gaussian evolving over time. Given an illumination stripe of known width, we can now estimate the *rgb* parameters c_x and σ_u using a nonlinear optimization method such as Levenburg-Marquardt method [Press92]. By symmetry, we can recover the vertical center coordinate and width by sweeping a horizontal Gaussian stripe in the vertical direction behind the foreground object. Thus, for the case of a single, unoriented Gaussian weighting function, a horizontal and a vertical swept Gaussian stripe are enough to estimate all the remaining parameters of the environment matte.

Figure 9.5 illustrates the more general case of a sweeping stripe that is constant in the t-direction and has Gaussian profile in the s-direction. This stripe is oriented at an angle ϕ with respect to the xy-coordinate system and travels in the s-direction. Under these circumstances, it is straightforward to show that the observation at a pixel will be

$$C(r) = R\, \boldsymbol{G}_{\text{1D}}(r; c_r, \sigma_r)\,, \tag{9.24}$$

where

$$\begin{aligned} c_r &= c_x \cos\phi + c_y \sin\phi\,, \\ \sigma_r &= \sqrt{{\sigma_u}^2 \cos^2(\phi-\theta) + {\sigma_v}^2 \sin^2(\phi-\theta) + \sigma_s^2}\,. \end{aligned} \tag{9.25}$$

Here, c_r is the center of the weighting function projected onto the r-axis, and σ_r is the projected, convolved standard deviation of the observed Gaussian.

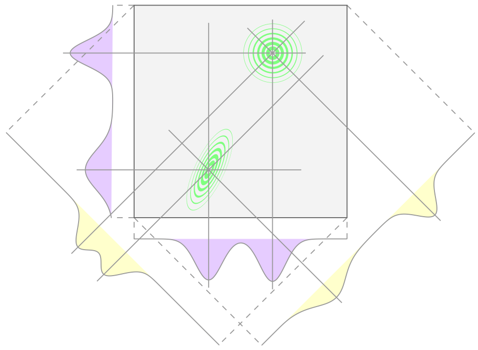


Figure 9.6: The horizontal and vertical sweeps alone (purple) are insufficient to determine the mode positions, but by adding additional diagonal sweeps (yellow) the correct modes can be determined.

Horizontally and vertically swept stripes alone ($\phi = 0°$ and $90°$, respectively) are not enough to determine the weighting function, two diagonal passes at $\phi = 45°$ and $-45°$ are added. The additional oriented stripes serve another purpose: disambiguating multiple mappings to the backdrop. As shown in Figure 9.6, a bimodal weighting function results in two Gaussian images over time at a pixel as the stripe sweeps across. If we use just the horizontal and vertical stripes, the two modes recorded in each sweep yield multiple indistinguishable interpretations of the bimodal weighting function. The oriented stripes can also be used to disambiguate these choices. These parameters are again estimated with Levenburg-Marquardt method. Figure 9.7 shows results of this approach and comparisons with HEM.

9.3.4 Frequency-based environment matting (FBEM)

Although GEM is capable of acquiring more accurate environment mattes than HEM, it assumes the weighting function is a sum of Gaussians and thus places restrictions on the materials that can be acquired. To address this problem, Zhu *et al.* [Zhu04]

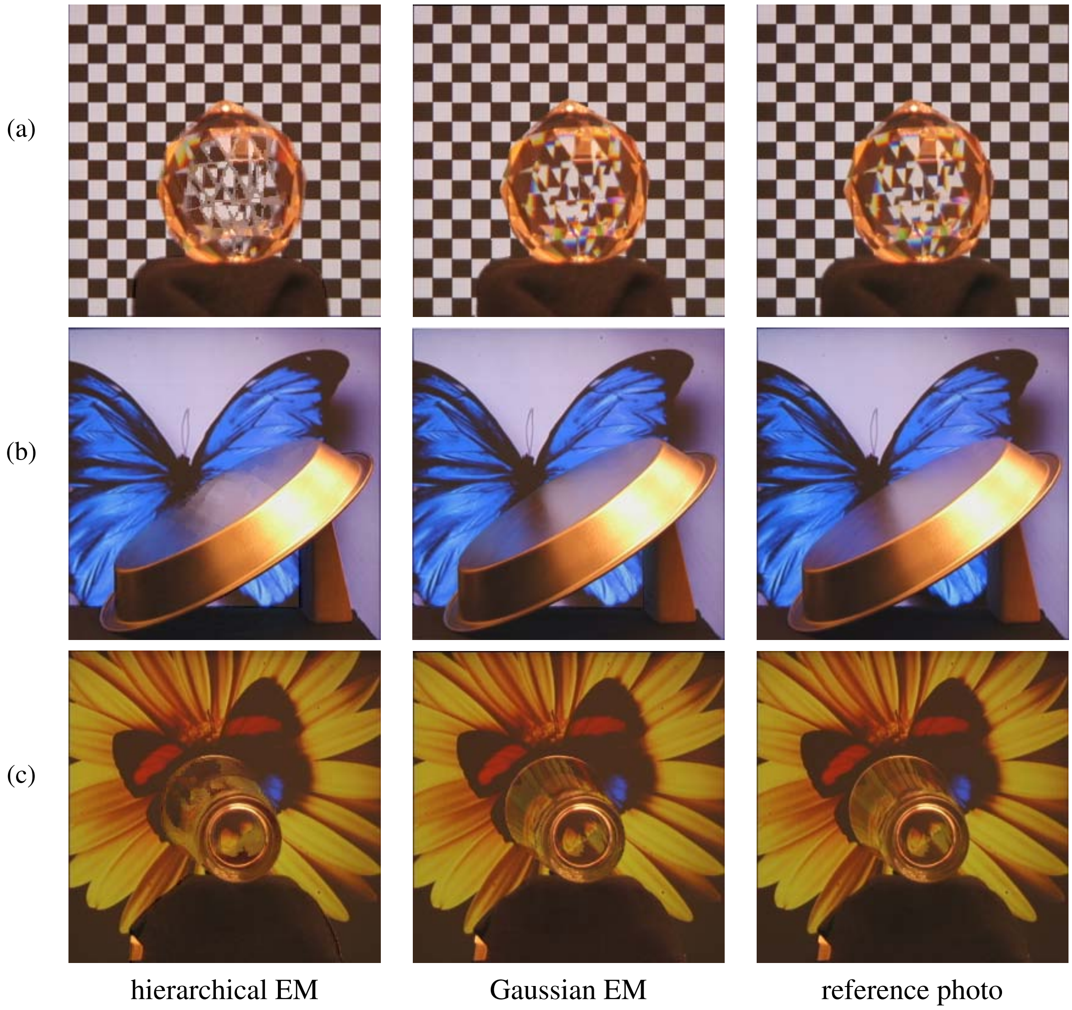


Figure 9.7: Comparisons between the composite results of the hierarchical environment matting, the Gaussian environment matting technique, and reference photographs of the matted objects in front of background images. Lighting in the room contributed a yellowish foreground color F that appears, e.g., around the rim of the pie tin in the bottom row. (a) A faceted crystal ball causes rainbowing due to prismatic dispersion, an effect successfully captured by GEM since shifted Gaussian weighting functions are determined for each color channel. (b) The weighting functions due to reflections from a roughly-textured pie tin are smooth and fairly broad. GEM handles such smooth mappings successfully, while HEM based on square-wave illumination patterns and rectangular weighting functions yields blocky artifacts. (c) Light both reflects off and refracts through the sides of a glass. This bimodal contribution from the background causes catastrophic failure with the previous unimodal method, but is faithfully captured by GEM.

propose frequency-base environment matting. The principle of FBEM is pretty simple and elegant. As shown in Figure 9.8, a unique frequency $f_{\boldsymbol{x}}$ is assigned to each position $\boldsymbol{x}$ of the backdrop and a time-evolving function $\boldsymbol{B}(\boldsymbol{x}, t)$ with this particular frequency, say $\boldsymbol{B}(\boldsymbol{x}, t) = \cos(2\pi f_{\boldsymbol{x}} t)$, is displayed at $\boldsymbol{x}$ on the backdrop. Assuming F is zero and substituting $\boldsymbol{B}(\boldsymbol{x}, t)$ into Equation 9.2, we have the time-evolving function $C(t)$ for a pixel of the object image as:

$$C(t) = \int \boldsymbol{W}(\boldsymbol{x}) \, \cos(2\pi f_{\boldsymbol{x}} t), d\boldsymbol{x} \,. \qquad (9.26)$$

Thus, $C(t)$ is a sum of several functions with different frequencies. The contribution of each function $\boldsymbol{B}(\boldsymbol{x}, t)$ to $C(t)$ is proportional to the weighting function $W(\boldsymbol{x})$. Hence, if we take the Fourier transform of $C(t)$ and look at its spectrum, we can directly reconstruct the weighting function.

The approach of assigning a unique frequency to each pixel, however, would require $O(k^2)$ images because of the Nyquist limit. To reduce the number of images, Zhu *et al.* assume that the weighting function is separable. Hence, the acquisition can be divided into two passes. In the first pass, the frequencies $f_{\boldsymbol{x}}$ only depends on the horizontal component of $\boldsymbol{x}$. In the second pass, $f_{\boldsymbol{x}}$ only depends on the vertical component of $\boldsymbol{x}$. Thus, the number of images is reduced to $O(k)$. However, although FBEM makes no assumption on the form of the weighting function as GEM and thus allows more flexibility, this separability assumption places restrictions on the material properties. For example, FBEM is not suitable for acquiring materials with multiple mappings since the corresponding weighting function is not separable.

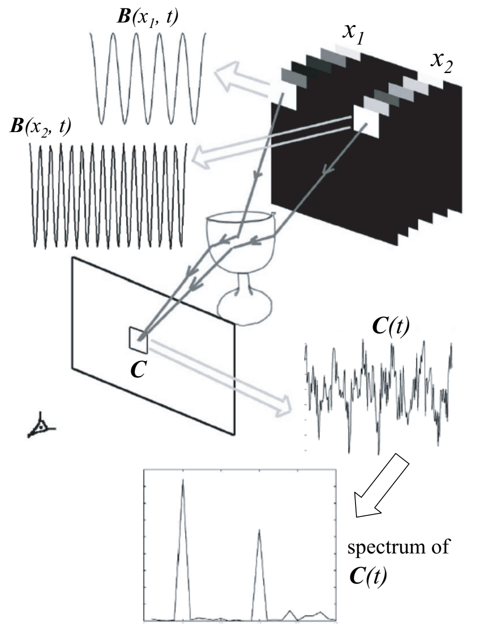


Figure 9.8: Illustration of frequency-based environment matting.

9.3.5 Wavelet environment matting (WEM)

All approaches we have discussed so far make certain assumptions on the weighting function. Hence, they impose limitations on the material properties that can be acquired. For example, diffuse surfaces are problematic to these techniques because diffuse materials have a large area of support which can be irregularly shaped because of occlusion and self-shadowing. These irregularly shaped support areas are difficult to approximate accurately with the approaches we have discussed. Peers *et al.* [Peers03] propose wavelet environment matting that makes no assumption on the form of the weighting function at the expense of using even more backdrops. Theoretically, their method would require $O(k^2)$ backdrops.

Assuming that F is known in the environment matting equation (9.2) and leaving it out in the following derivation[2], we have

$$C \;=\; \int \boldsymbol{W}(\boldsymbol{x}) \, \boldsymbol{B}(\boldsymbol{x}) \, d\boldsymbol{x} \,. \qquad (9.27)$$

When photographing the object in front of a series of n stimulus function $\boldsymbol{B}_i(\boldsymbol{x})$, we obtain images

[2] F can be estimated by photographing the object when the backdrop display is turned off.

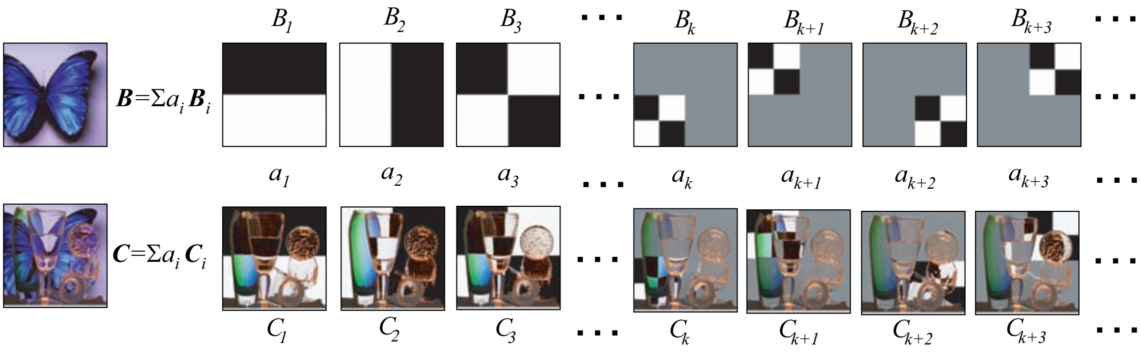


Figure 9.9: Illustration of wavelet environment matting.

C_i with

$$C_i \;=\; \int \boldsymbol{W}(\boldsymbol{x})\, \boldsymbol{B}_i(\boldsymbol{x})\, d\boldsymbol{x}\,. \tag{9.28}$$

We observe that if the background image $\boldsymbol{B}$ for compositing can be written as a linear combination of $\boldsymbol{B}_i$:

$$\boldsymbol{B} \;=\; \sum_{i=1}^{n} a_i \boldsymbol{B}_i\,, \tag{9.29}$$

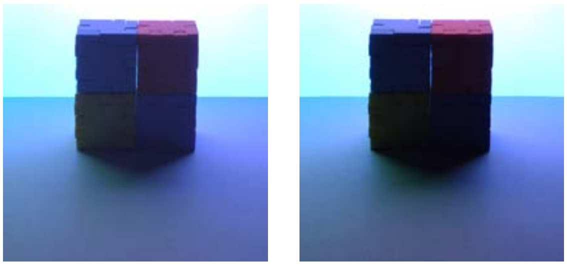
where a_i are the coefficients associated with each $\boldsymbol{B}_i$. We can write the composite C as:

$$\begin{aligned} C &= \int \boldsymbol{W}(\boldsymbol{x})\, \boldsymbol{B}(\boldsymbol{x})\, d\boldsymbol{x} \\ &= \int \boldsymbol{W}(\boldsymbol{x}) \sum_{i=1}^{n} a_i \boldsymbol{B}_i(\boldsymbol{x})\, d\boldsymbol{x} \\ &= \sum_{i=1}^{n} a_i \int \boldsymbol{W}(\boldsymbol{x})\, \boldsymbol{B}_i(\boldsymbol{x})\, d\boldsymbol{x} \\ &= \sum_{i=1}^{n} a_i C_i. \end{aligned} \tag{9.30}$$

Hence, instead of representing the weighting function by a set of parameters, WEM directly represents the weighting function as a set of n photographs C_i acquired in front of the basis backdrops B_i. During compositing, given a novel background $\boldsymbol{B}$, $\boldsymbol{B}$ is projected onto the space spanned by $\boldsymbol{B}_i$ to obtain coefficients a_i. Then, the composite can be computed as $\sum_{i=1}^{n} a_i C_i$. Figure 9.9 illustrates this process.

To efficiently representing images, WEM uses a set of wavelet illumination patterns B_i to exploit the hierarchical nature for reducing the number of backdrops. The backdrop patterns shown in Figure 9.9 are 2D Haar wavelet basis images. In addition, during acquisition, important patterns are displayed first and then an error-tree and a feedback loop are employed to decide the next wavelet patterns to display based on previously recorded photographs. Hence, the displayed patterns are determined on fly, adapting to the observed properties of the environment matte. This enables WEM to stop the acquisition process prematurely when the contribution of the patterns to the illumination is below some threshold or when the acquisition time has exceeded a time-limit. In practice, WEM sets a time limit of 12 hours as the stopping criterion and found it sufficient for capturing most types of objects. Typically, 400 to 2400 wavelet patterns are required.

WEM avoids the use of an optimization procedure for finding parameters of the weighting functions and can represent arbitrary weighting functions. For example, it can model diffuse materials reasonably well as shown in Figure 9.10. However, it requires a large amount of storage for the images C_i. As reported by Peers *et al.* [Peers03], an av-



(a) (b)

Figure 9.10: A scene containing diffuse surfaces. The composite (b) of wavelet environment matting reasonably approximates the photograph (a).

erage of 2.5GB is required for a single object. In addition, the acquisition time is considerably much longer than other environment matting methods.

9.4 Passive approaches

The environment matting methods we have discussed so far are all active approaches, and hence require carefully controlled backgrounds and a calibrated laboratory setting. In this section, we describe passive approaches which extract environment mattes from a sequence of the object in front of natural backgrounds and don't require a controllable background.

9.4.1 Image-based environment matting (IBEM)

Wexler *et al.* [Wexler02] propose an approach, called image-based environment matting, to pull an environment matte from a sequence of images C_i in which a static foreground object is filmed in front of a moving planar background or by a panning camera in front of a planar background. For IBEM to work, the clean plate B_i for each frame C_i is required. These clean plates could be obtained by mosaicking the moving-background sequence with planar homographies. Figure 9.11 illustrates an example of moving the camera with a planar background and warping the image (b) to obtain clean plate (c) for the base frame (a).

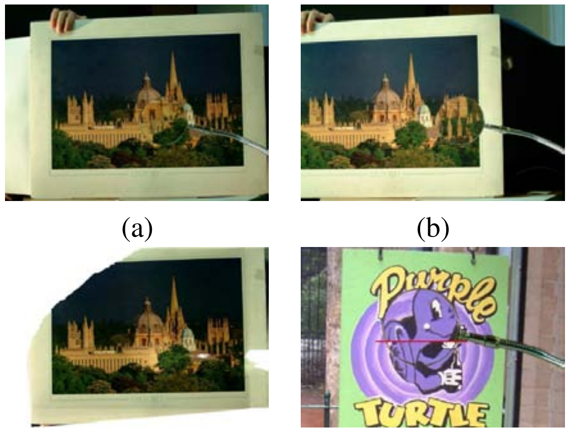


Figure 9.11: Clean plate estimation and a composite of image-based environment matting. For the base frame (a), all other frames (b) are warped using planar homographies to assemble the clean plate (c) for the base frame. Then, the weighting function can be estimated and the composite (d) mimics how the magnifying glass distorts lights.

Given a series of images C_i (Figure 9.12(a)) and their corresponding clean plates B_i (Figure 9.12(b)), for each pixel p of the environment matte, a probability map P_i is built using

$$P_i(q) \propto e^{-\|C_i(p) - B_i(q)\|^2}, \qquad (9.31)$$

where q is a pixel on the background. The resulting probability map (Figure 9.12(c)) potentially has many peaks where the color $B_i(q)$ on the background is similar to $C_i(p)$. However, since we have multiple backgrounds, we can integrate these probability maps P_i's by multiplying them together (Figure 9.12(f)) to get rid of the false peaks, essentially identifying the locations of the pixel contributing to p on the background. Figure 9.12 (d) and (e) show the cross sections of the probability maps and accumulated probability maps for a specific scanline.

Although, in principle, this formulation allows arbitrary weighting functions (essentially the integrated probability map), the way of construct-

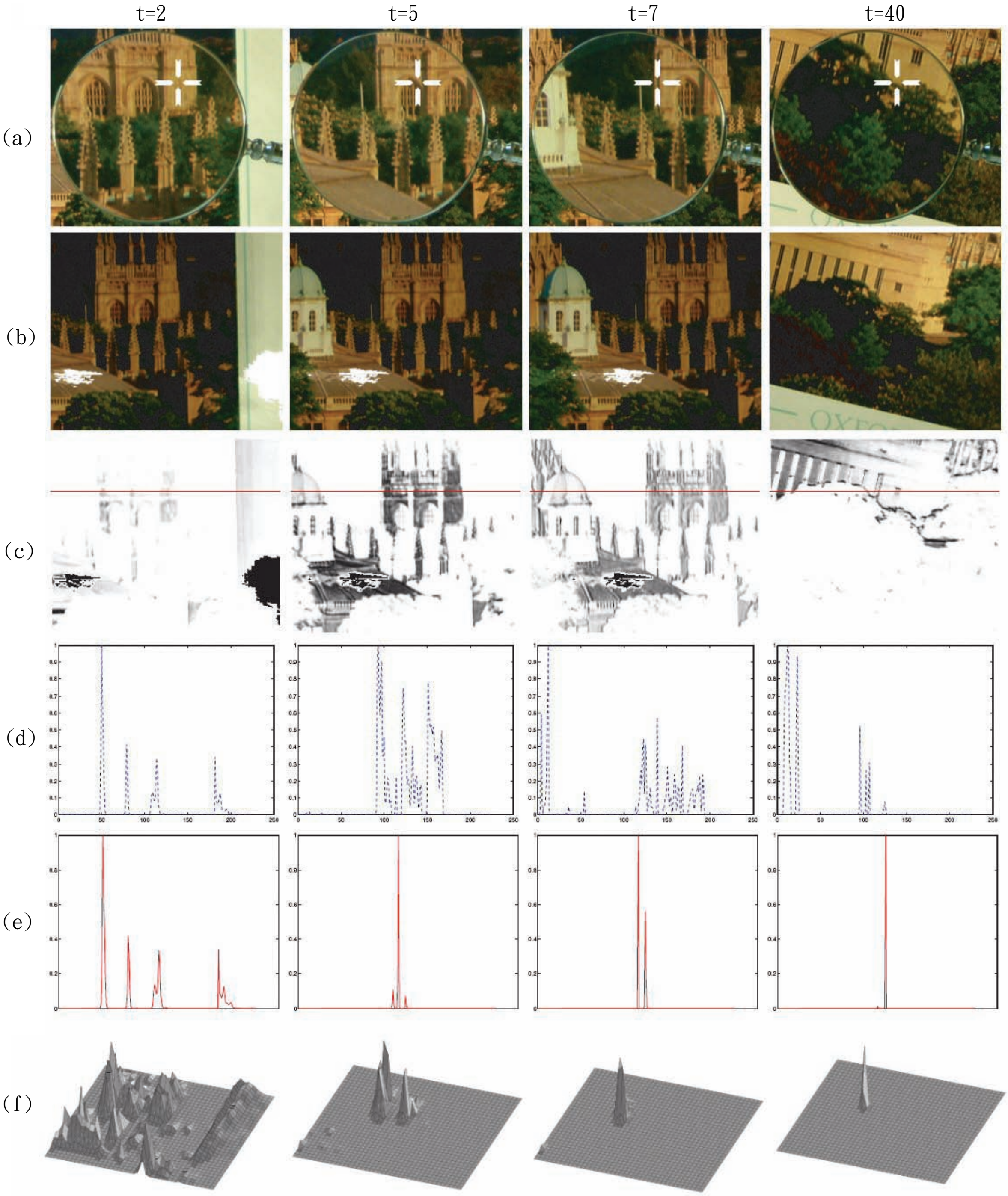


Figure 9.12: Illustration for image-based environment matting.

category	method	asymptotic # of images	typical # of images	weighting function	materials	comment
active	RTEM	1	1	warping function	colorless, specularly refractive	the only real-time method to date
	HEM	$O(\log k)$	20	box filter	refraction, translucency, highly specular, color transparency	the first EM method
	GEM	$O(k)$	600	sum of Gaussians	+color dispersion, multiple mappings and glossy reflection	good accuracy, reasonable time
	FBEM	$O(k)$	1,200	product of two 1D functions	-multiple mappings	temporal-domain stimulus function
	WEM	$O(k^2)$	1,200	object images	+diffuse reflection	the most accurate method to date, but it requires long capturing time and large storage
passive	IBEM	N/A	40	probability map	colorless, specularly refractive	the first passive method
	ROEM	N/A	200	warping function	colorless, specularly refractive	works with unknown backgrounds

Table 9.1: Summary of available environment matting methods.

ing the probability map restricts the object to be specularly refractive or reflective. Hence, IBEM is only effective for capturing colorless specular objects. Furthermore, though it is passive, IBEM is a multiple-frame approach, requiring multiple frames for extracting a single environment matte. Hence, the object must be static during the acquisition, and the success of this method relies on having enough background samples or sufficiently rich backdrop images.

9.4.2 Refractive optical flow environment matting (ROEM)

Agarwal *et al.* [Agarwal04] approximate the weighting function with an image warping function similar to RTEM:

$$\boldsymbol{W}(\boldsymbol{x}) = \rho(x)\delta(\boldsymbol{u} - T(\boldsymbol{x})), \tag{9.32}$$

where ρ is the attenuation factor, $\delta(\cdot)$ is Dirac's delta function and T is the warping function. The environment matting equation (9.2) can now be rewritten as

$$C(\boldsymbol{x}, t) = \rho(x)\boldsymbol{B}(T(\boldsymbol{x}), t), \tag{9.33}$$

and it leads to the refractive optical flow equation:

$$C_t(\boldsymbol{x}, t) + \boldsymbol{v}(T(\boldsymbol{x}), t))^T \boldsymbol{J}^{-T}(T(\boldsymbol{x}))) \left[\nabla_{\boldsymbol{x}} C(\boldsymbol{x}, t) - C(\boldsymbol{x}, t)\frac{\nabla_{\boldsymbol{x}}\rho(\boldsymbol{x})}{\rho(\boldsymbol{x})}\right] = 0, \tag{9.34}$$

where $\boldsymbol{v}(\boldsymbol{u}, t)$ is the velocity of the point $\boldsymbol{u}$ at time t in the background plane and $\boldsymbol{J}(T(\boldsymbol{x}))$ is the Jacobian of the warping function $T(\boldsymbol{x})$. The refractive optical flow equation (9.34) is independent of $\boldsymbol{B}(\boldsymbol{x}, t)$. This means that we can solve the refractive flow through an object by observing the distorted versions of the background image without knowing it. Assuming that $\boldsymbol{v}(\boldsymbol{u}, t)$ is known, we can set up a linear system and solve for the estimations for the partial derivatives of $T(\boldsymbol{x})$ and $\rho(\boldsymbol{x})$. The warping function and attenuation factors can then be recovered by integrating the respective gradient fields to construct the matte. Similar to IBEM, this approach is only effective for specularly refractive objects.

9.5 Conclusion

Environment matting involves an inherent tradeoff: the amount of input data required versus the quality of the resulting mattes versus the requirement of a controllable background. In this chapter, we describe several environment matting methods representing several points in the design space, each with its own strength and weakness. Table 9.1 summarizes these methods.

Chapter 10

Fibers

Steve Marschner

When we think of light reflection in computer graphics, we usually think of reflection from surfaces. But when we are rendering materials like hair, we generally can't afford to compute the reflection from the surface of each fiber; this would imply modeling each individual hair as a little tube and using several samples across the width of the tube for antialiasing. Instead, hairs are modeled as 1D curves with a small thickness, and light reflection should be computed for the whole width of the hair at once. That is, we always assume the hair is not resolved—that it's not substantially thicker than one pixel. What we need from the light reflection model is the *average* radiance across the width of the fiber, based on the *average* irradiance across the width of the fiber.

This amounts to treating fibers as one-dimensional entities. Of course, they have to have some width in order to cast shadows and occlude things. And if we are trying to be physically correct and account for where all the light energy goes, we have to think about the fiber's width when we figure out how much light is falling on it. But the actual model for reflection should abstract away the width: it will be a function only of the incident and exitant directions. This is like a BRDF, but since these directions can vary over the whole sphere, things look a bit different. For the surface, only one direction is normal and a whole circle of directions is "grazing," but for the fiber there's a whole circle of normal directions but only two "grazing" directions.

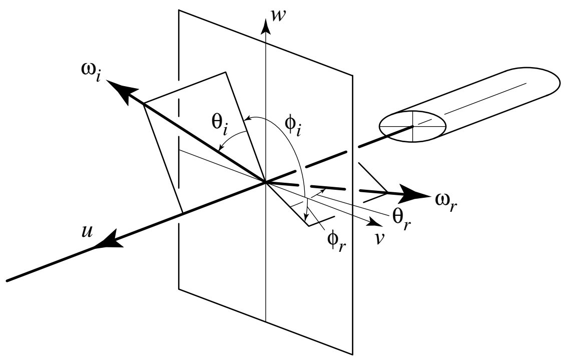


Figure 10.1: Notation for fiber scattering geometry

Our notation for scattering geometry is summarized in Figure 10.1. We refer to the plane perpendicular to the fiber as the *normal plane*. The direction of illumination is ω_i, and the direction in which scattered light is being computed or measured is ω_r; both direction vectors point away from the center. We express ω_i and ω_r in spherical coordinates. The inclinations with respect to the normal plane are denoted θ_i and θ_r (measured so that $0°$ is perpendicular to the hair). The azimuths around the hair are denoted ϕ_i and ϕ_r, and the relative azimuth $\phi_r - \phi_i$, which is sufficient for circular fibers, is denoted $\Delta\phi$.

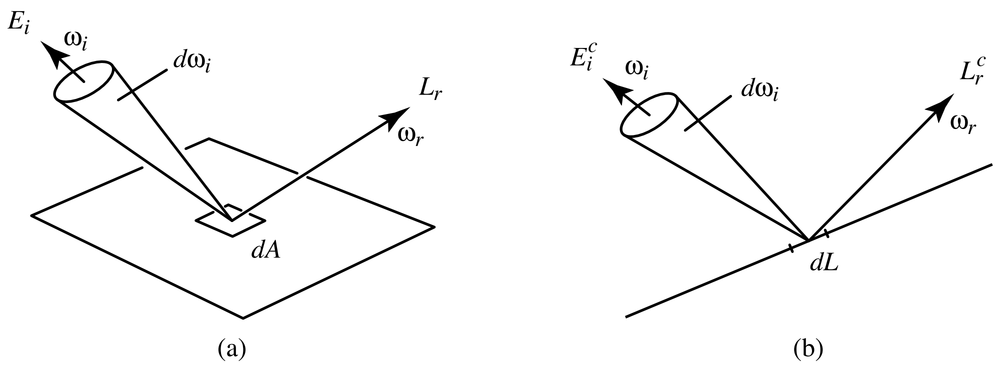


Figure 10.2: Geometric setup for reflection from (a) surfaces, in which the reflected light is resolved per unit area and (b) curves, in which the reflected light is resolved only per unit length.

10.1 Radiometry

Because fibers are usually treated as one-dimensional entities, light reflection from fibers needs to be described somewhat differently from the more familiar surface reflection.

As we've seen already earlier in this course, light scattering at a surface is conventionally described using the bidirectional reflectance distribution function (BRDF), $f_r(\omega_i, \omega_r)$. The BRDF gives the density with respect to projected solid angle of the scattered flux that results from directional illumination from the direction ω_i. It is defined as the ratio of surface radiance (intensity per unit projected area) exiting the surface in direction ω_r to surface irradiance (flux per unit area) falling on the surface from a differential solid angle in the direction ω_i (Figure 10.2a):

$$f_r(\omega_i, \omega_r) = \frac{dL_r(\omega_r)}{dE_i(\omega_i)}.$$

Under this definition, the radiance due to an incoming radiance distribution $L_i(\omega_i)$ is

$$L_r(\omega_r) = \int_{H^2} f_r(\omega_i, \omega_r) L_i(\omega_i) \cos\theta_i d\omega_i$$

where H^2 is the hemisphere of directions above the surface.

Light scattering from fibers is described similarly, but the units for measuring the incident and reflected light are different because the light is being reflected from a one-dimensional curve [Marschner03]. If we replace "surface" with "curve" and "area" with "length" in the definition above we obtain a definition of the scattering function f_s for a fiber: "the ratio of *curve radiance* (intensity per unit projected length) exiting the curve in direction ω_r to *curve irradiance* (flux per unit length) falling on the curve from a differential solid angle in the direction ω_i" (Figure 10.2b). The curve radiance due to illumination from an incoming radiance distribution L_i is

$$L_r^c(\omega_r) = D \int_{H^2} f_s(\omega_i, \omega_r) L_i(\omega_i) \cos\theta_i d\omega_i$$

where D is the diameter of the hair as seen from the illumination direction.

The physical units of radiance and curve radiance are also parallel. Irradiance measures the total radiant power falling on the relevant bit of geometry: surface irradiance is power per unit area, while curve irradiance is power per unit length.

$$E = \frac{dP}{dA} \left(\frac{\mathrm{W}}{\mathrm{m}^2}\right) \quad E^c = \frac{dP}{dL} \left(\frac{\mathrm{W}}{\mathrm{m}}\right)$$

Radiance is different in two ways: it is a solid angle density as well as an area/length density, and it is measured with respect to area/length projected in the direction of propagation. Surface radiance has units of power per unit solid angle per unit projected area, while curve radiance has units of power per unit solid angle per unit projected length.

$$L = \frac{d^2P}{dA^{\perp}d\omega} \left(\frac{\mathrm{W}}{\mathrm{m}^2\,\mathrm{sr}}\right) \quad L^c = \frac{d^2P}{dL^{\perp}d\omega} \left(\frac{\mathrm{W}}{\mathrm{m}\,\mathrm{sr}}\right)$$

Curve radiance is in some sense halfway between the concepts of radiance and intensity, and it describes the contribution of a thin fiber to an image independent of its width. Curve irradiance measures the radiant power intercepted per unit length of fiber and therefore increases with the fiber's width. Thus, given two fibers with identical properties but different widths, the two will have the same scattering function but the wider fiber will produce a brighter curve in a rendered image, because the fiber intercepts more incident light. This is consistent with the behavior of real fibers: very fine hairs do appear fainter when viewed in isolation.

Chapter 11

Bidirectional Texture Functions

Gero Mueller

Figure 11.1: Images of a corduroy cloth under different lighting and viewing directions projected onto a common plane, i.e., samples of the material's BTF.

11.1 Introduction

Most standard graphics applications model materials with spatially varying appearance using a single texture only, i.e., a single image of the material taken under one given illumination condition. Assuming that the color values of the texture represent the diffuse albedo of the material this approach is only able to capture the reflectance field of a flat Lambertian surface. In contrast real materials are neither flat nor ideal diffuse but exhibit complex surface structures and reflectance characteristics that are responsible for their unique look-and-feel (Figure 11.1 shows an example).

Many different techniques have been developed over the years to cope with the deficiencies of simple diffuse texture mapping and to increase the realism of material rendering. E.g., bump mapping simulates variation of outgoing radiance due to changing surface normals, displacement mapping models occlusion effects, and volume based techniques can be used to render arbitrary complex non-heightfield geometry. Nevertheless, all these improvements share the same problem of how to generate the input data. Capturing the small-scale geometry of materials is currently possible for a limited class of materials only using, e.g., shape-from-shading and related techniques. Modeling the geometry explicitly is practical for special types of surfaces like fur or hair only, where procedural modeling tools are available. But even if the material's geometry is at hand or the material is flat enough to neglect it, the correct simulation of light transport on this surface requires additional information about the scattering characteristics at every point on the surface which again has to be modeled or captured somehow.

The image-based solution to these modeling problems is taking images of the material of interest under controlled viewing and lighting conditions in order to sample its reflectance field more densely. For example Malzbender et al. [Malzbender01] captured the lighting variability of a texture by taking images lit from different directions and compressed the data to a compact representation called Polynomial Texture Maps. If also the viewing direction is varied we arrive at the Bidirectional Texture Function (BTF), which was introduced by Dana et al. [Dana97] as a general texture representation for computer vision tasks and rendering.

Mathematically the BTF can be expressed as a measured 6D-slice of the general 8D-reflectance field $\mathbf{RF}_{rgb}(\mathbf{x}_i \rightarrow \mathbf{x}_o, \omega_i \rightarrow \omega_o)$ parameterized over a planar base surface S (if S coincides with the physical surface, the reflectance field $\mathbf{RF}_{rgb}$ is equivalent to the BSSRDF, cf. Chapter 4):

$$\mathbf{BTF}_{rgb}(\mathbf{x}, \omega_i \rightarrow \omega_o) := \int_S \mathbf{RF}_{rgb}(\mathbf{x}_i \rightarrow \mathbf{x}, \omega_i \rightarrow \omega_o) d\mathbf{x}_i$$

It this sense the BTF integrates subsurface scattering from neighboring surface locations and occlusion effects (shadowing and masking) resulting from deviations of the physical surface from the planar surface S. As illustrated in Figure 11.2 this leads to the interpretation of the BTF as spatially varying *apparent* BRDF (ABRDF; the term was introduced in a paper of Wong et al. [Wong97]). For convenience in the following we will denote the incoming *light* direction by **l** and the outgoing *view* direction by **v**.

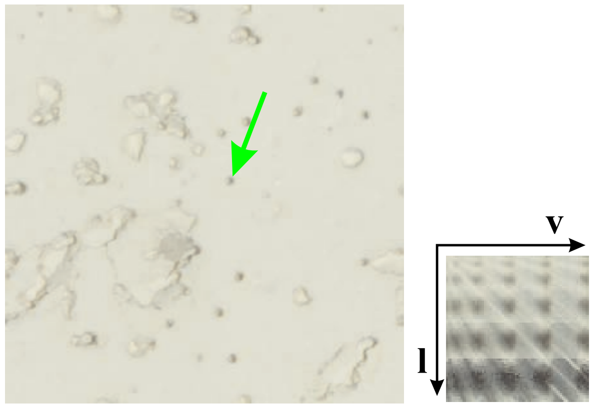


Figure 11.2: An apparent BRDF (right) from the *plasterstone* BTF (left). While the reflectance characteristic of the white plaster alone is quite regular, the holes introduce strong effects from meso-scale shadowing and masking.

11.2 Measurement

The acquisition of the BTF of real world materials requires a complex and controlled measurement environment. As BTF acquisition is physical measurement of real-world reflection, special attention has to be paid to the device calibration and image registration. Otherwise the measurements will contain inaccuracies which may generate visible rendering artifacts. In this section current techniques for accurate BTF measurement will be presented. The techniques aim at capturing the appearance of an opaque material independently from geometry and therefore capture the BTF of a planar material sample. The acquired data can be used instead of simple 2D-textures and mapped onto arbitrary geometry given a parameterization of this geometry. In this section we will introduce shortly the main existing BTF-measurement approaches and then describe our setup at Bonn in greater detail.

11.2.1 Gonioreflectometer-like Setups

The most common approaches use a gonioreflectometer-like setup with a CCD-chip instead of a spectrometer in order to capture the spatial variation of reflection. This approach has proved to be reliable and several variations of it have been published. However its drawback are the long measurement times.

The first measurement system that used such a gonioreflectometer-like setup as depicted in Fig-

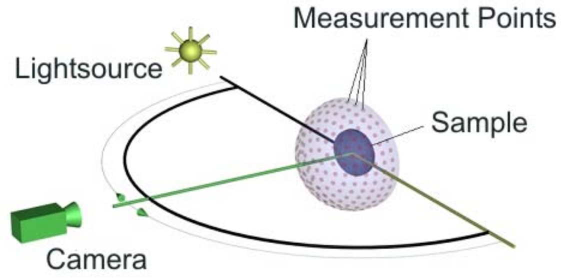


Figure 11.3: Capturing the BTF of a planar sample using a gonioreflectometer-like setup with a fixed light source, a rotating sampleholder and a moving camera.

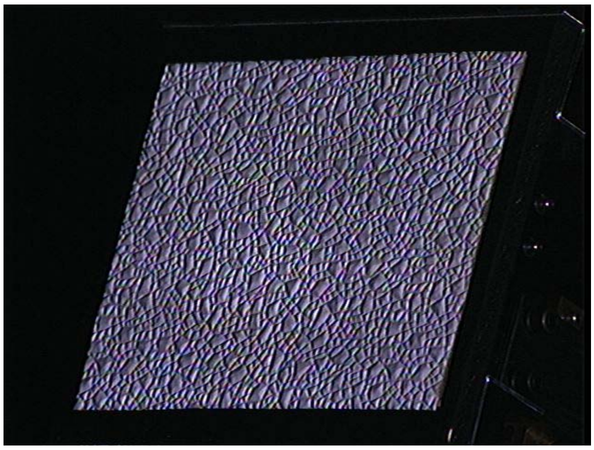


Figure 11.4: One image of the sample *rough plastic* from the CUReT database.

ure 11.3 was presented in the pioneering work of Dana et al. [Dana97]. Their system takes 205 images of isotropic materials which is a too sparse sampling for high-quality rendering, in particular for rough surfaces and materials with strong specular pikes. Even though they mentioned the possibility of using the data for rendering, the original intent of the work was building up a material database for computer vision related tasks such as texture recognition, texture segmentation and shape-from-texture. They measured 61 real-world surfaces and made them available through the CUReT database [Reflectance]. Figure 11.4 shows a sample from the database. Similar, but improved versions of the measuring system were described in [McAllister02] and [Hauth02]. Some measurements of the latter system are now also publicly available through the BTF database Bonn [BTF Database Bonn]. We will describe this system in greater detail later in this section.

11.2.2 Using Video Cameras

Koudelka et al. [Koudelka03] presented a system resembling the before mentioned ones, but it fixes the image sensor (a Canon XL-1 digital video camera) and moves the light source (a white LED mounted on a robot arm). The employed hemisphere sampling results in a dataset of about 10.000 images. Due to the use of a video camera with relatively low resolution compared to a high-end still camera, a measurement takes about 10 hours. Samples from this system are publicly available for research purposes and include interesting natural materials like lichen or moss and man-made materials like carpet or even LEGO™bricks.

11.2.3 Using Mirrors

Inspired by BRDF measurement techniques, it has also been proposed to use mirrors for BTF measurement in order to avoid hemispherical movements or to make several measurements in parallel.

An approach using a concave parabolic mirror has been published by Dana and Wang [Dana04]. In this device the parabolic mirror is focused on the sample and thus an image of the mirror captures the appearance of the surface point in focus as seen from different viewing directions. Spatial variation of the sample is captured by planar movement of the mirror (or the sample). Illumination from different directions is achieved by focusing a light beam on the appropriate spot in the mirror. With this setup no hemispherical movement is required and the resulting data is of high quality. But a high spatial resolution (comparable to the gonioreflectometer-like devices which achieve about 300DPI) requires an enormous amount of images. Please note also that interreflections and subsurface scattering from neighboring parts of the surface are not integrated. Furthermore for a substantial range of azimuthal angles ϕ the covered range of the polar angle θ is restricted by the size of the mirror. E.g. for $\phi = \pi$ the presented prototype can capture polar angles only up to $22.6°$.

Han et al. [Han03] presented a measurement system based on a kaleidoscope which allows to capture several images of the whole sample at once. The advantages in measurement time and registration precision (no moving parts) are accompanied by a number of disadvantages. Multiple reflections on mirrors (not perfect reflectors) cause low image quality and lead to a difficult color calibration. Slight asymmetries in the configuration of the mirrors result in registration errors. Angular sampling and spatial resolution are often coupled, i.e. a higher angular resolution leads to a lower spatial resolution. Radloff [Radloff04] has analyzed different kaleidoscope configurations by simulation and built several prototypes. But due to the mentioned difficulties in building a perfect kaleidoscope the quality of the results turned out to be rather low.

11.2.4 Camera arrays

For a fast high quality acquisition of BTFs in our recent approach we propose an array of 151 digital still cameras mounted on a hemispherical gantry (see Figure 11.5). A similar gantry with mounted light sources was used by Malzbender et al. [Malzbender01] to capture PTMs. In our setup the built-in flash lights of the cameras are used as light sources.

Although the setup is costly to build, a camera array is capable of measuring many samples in a short time. Due to the parallel structure of the acquisition, the example setup is capable of capturing a BTF dataset of $151^2 = 22801$ images in less than half an hour. No moving parts are needed. Therefore, the region of interest (ROI) is known for every camera and can be extracted at subpixel precision. Hence, there is no need for a time-consuming detection of the ROI, the post-processing (reprojection, geometric correction, color correction) is fast enough to be done in parallel to the measurement. The angular resolution depends on the number of cameras and the spatial resolution on the imaging chips. This will result in a high angular resolution; every measured direction represents an average solid angle of only 0.04161 steradians. For the 3 MPixel cameras we use the spatial resolution is up to 240DPI.

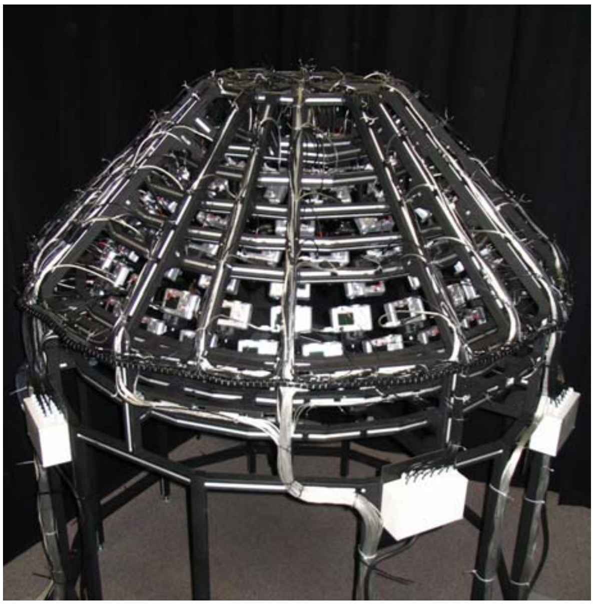


Figure 11.5: 151 digital cameras with built-in flash lights are mounted on a gantry, focusing on the sample, which is placed in the center of the hemisphere.

11.2.5 Details of setup at University of Bonn

Measurement Setup The measurement setup is designed to conduct a fully automatic measurement of a BTF including the automatic alignment and postprocessing of the captured data. A high-end digital still camera is used as image sensor.

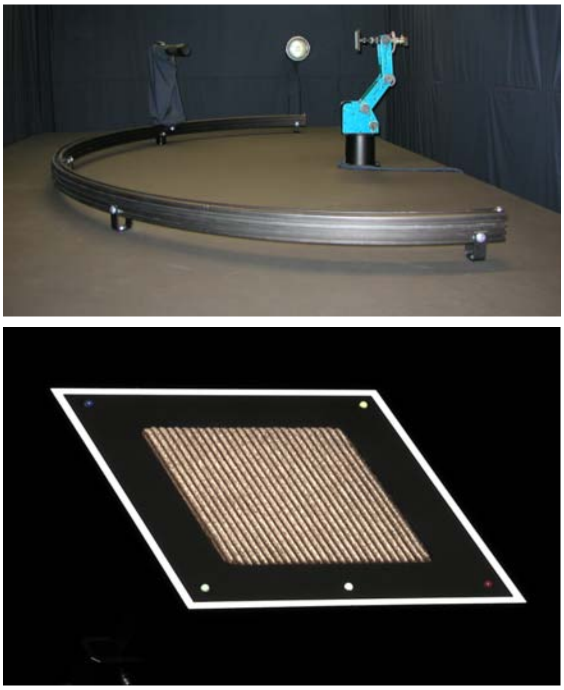


Figure 11.6: Top: Measurement setup of the Bonn-System consisting out of an HMI lamp, a CCD camera and a robot arm with a sample holder. Bottom: Raw image of *corduroy* BTF measurement.

θ [°]	$\Delta\phi$ [°]	No. of images
0	$-^*$	1
15	60	6
30	30	12
45	20	18
60	18	20
75	15	24

Table 11.1: Sampling of viewing and illumination angles of the BTF database Bonn. *= only one image taken at $\phi = 0°$

The complete setup, especially all metallic parts of the robot, are covered with black cloth or matte paint, with strong diffuse scattering characteristics.

The system uses planar samples with a maximum size of $10 \times 10\ cm^2$. In spite of these restrictions, measurement of a lot of different material types, e.g., fabrics, wallpapers, tiles, wood etc. is possible. As shown in Figure 11.6, the laboratory consists of a HMI (Hydrargyrum Medium Arc Length Iodide) bulb, a robot holding the sample and a rail-mounted CCD camera (Kodak DCS Pro 14N). Table 11.1 shows the sampling density of the upper hemisphere for light and view direction which results in $n = 81$ unique directions for camera and light position. Hence, 6561 pictures of a sample are taken. The bottom of Figure 11.6 shows a raw image as taken from an oblique view during the measurement process. Each raw image is 12 megabytes in size (lossless compression) with a resolution of 4500×3000 pixels (Kodak DCR 12-bit RGB format). With this setup, the measuring time is about 14 hours, where most of the time is needed for the data transfer from the camera to the host computer.

Calibration As described in Chapter 5 calibration of the equipment is absolutely necessary for high quality measurements. In particular we perform the following calibration steps:

- To compensate the positioning error due to the robot and the rail system, one has to track the sample holder mounted on the robot arm using the camera. Experiments determined that these errors are small in the described setup. Therefore, marker points, which are placed on the sample holder, are detected only during the post-processing phase, allowing a software jitter correction.
- A geometric calibration has to be applied to the camera to reduce geometric distortion, caused by the optical system of the camera.
- For each sample to be measured, the aperture of the camera is adjusted in such a way that the number of saturated or dark pixels in the pictures is minimized given a fixed aperture during the measurement process.
- To achieve the best possible color reproduction, the combination of the camera and the

light source has to be color calibrated. For the measurement of the camera color profile a special CCD-Camera standard card (Gretag Macbeth - Color Checker DC) is used.

Data Postprocessing After the measurement the raw image data is converted into a set of rectified, registered images capturing the appearance of the material for varying light and view directions. Now a complete set of discrete reflectance values for all measured light and viewing directions can be assigned to each texel of a 2D texture.

Registration is done by projecting all sample images onto the plane which is defined by the frontal view ($\theta = 0, \phi = 0$). To be able to conduct an automatic registration, borderline markers were attached to the sample holder plate, see Figure 11.6. After converting a copy of the raw data to a binary image, standard image processing tools are used to detect the markers. In following steps the mapping (which maps these markers to the position of the markers in the frontal view) is computed and utilized to fill the registered image with appropriate colors.

To convert the 12-bit RGB images stored in the proprietary format of the camera manufacturer to standard 8 bit RGB file formats, the standard color profiles provided with the Camera SDK (look and output profile) and camera (tone curve profile) are applied to the image. The most appropriate 8 bit color range is extracted after applying an exposure gain to the converted data.

After this postprocessing step, the final textures are cut out of the raw reprojected images and resized appropriately (256×256 pixels in size for probes in the database, up to about 800×800 in principle). A final dataset with 256×256 pixels spatial resolution has a data amount of 1.2GB.

11.3 Compression

Since uncompressed BTF data demands from hundred of megabytes up to dozens of gigabytes compression is absolutely necessary for BTFs in order to be used in practical applications. A compression technique for BTFs should exploit the redundancy in the data in an efficient way and provide a fast, preferably real-time decompression stage. For integration into current real-time rendering systems an implementation of the decompression stage on modern GPUs would also be of great value.

Existing BTF compression techniques can be roughly separated into two categories. The methods of the first category are based on the interpretation of the BTF as spatially varying ABRDFs and apply BRDF modeling techniques. The second category consists of methods which interpret the measured BTF as a high-dimensional signal and apply more general statistical modeling techniques.

11.3.1 Analytical BRDF-Models

A natural approach for compressing BTFs is via a pixel-wise representation using BRDF models which are fitted to the synthetic or measured BTF data. Candidate BRDF models need to be efficiently computable to achieve real-time capabilities. In the following we will describe techniques based on this approach.

Lafortune Lobes

The simplest BTF model based on analytic function fitting was published by McAllister et al. [McAllister02] and is directly based on the Lafortune [Lafortune97] model. Lafortune et al. propose to approximate the BRDF by a sum of lobes

$$s(\mathbf{v}, \mathbf{l}) = \left(\mathbf{v}^t \cdot \mathbf{M} \cdot \mathbf{l}\right)^n \qquad (11.1)$$

with $\mathbf{v}$ and $\mathbf{l}$ denoting local view and light direction respectively, while the general 3×3 matrix $\mathbf{M}$ and the exponent n define the lobe.

To fit these parameters to the reflectance properties of a synthetic or measured BRDF, non-linear fitting methods like the Levenberg-Marquardt algorithm [Press92] are employed. Fitting the complete matrix allows for very general BRDFs but is very time consuming. Therefore, McAllister et al. decided to employ a more restricted, diagonal matrix $\mathbf{D}$, since fitting and rendering efforts are significantly reduced without major loss in rendering quality. Thus, they use the following, spatially varying lobes:

$$s_{\mathbf{x}}(\mathbf{v},\mathbf{l}) = \left(\mathbf{v}^t \cdot \mathbf{D}_{\mathbf{x}} \cdot \mathbf{l}\right)^{n_{\mathbf{x}}}. \quad (11.2)$$

This results in the following BTF approximation:

$$\mathbf{BTF}(\mathbf{x},\mathbf{v},\mathbf{l}) \approx \rho_{d,\mathbf{x}} + \sum_{j=1}^{k} \rho_{s,\mathbf{x},j} \cdot s_{\mathbf{x},j}(\mathbf{v},\mathbf{l}) \quad (11.3)$$

where ρ_d and ρ_s denote diffuse and specular albedo (specified as RGB values) and k is the number of lobes.

The model requires only a few parameters to be stored per pixel resulting in a very compact material representation (about 2 MB per material depending on the spatial resolution and number of lobes). Due to the expensive non-linear minimization, the number of Lafortune lobes is practically limited to about 4 lobes.

Scaled Lafortune Lobes

BRDF models were not designed for the spatially varying ABRDFs which can contain strong effects from meso-scale shadowing, masking (Figure 11.2). Therefore, more specialized models for ABRDFs were developed which also try to model some of these effects.

Daubert et al. [Daubert01] proposed a material representation, which is also based on the Lafortune model but includes an additional, multiplicative term $T_{\mathbf{x}}(\mathbf{v})$ modelling occlusion. Following their proposal, the BTF is evaluated as follows:

$$\mathbf{BTF}(\mathbf{x},\mathbf{v},\mathbf{l}) \approx T_{\mathbf{x}}(\mathbf{v}) \cdot \left(\rho_{d,\mathbf{x}} + \sum_{j=1}^{k} s_{\mathbf{x},i}(\mathbf{v},\mathbf{l})\right). \quad (11.4)$$

The view-dependent lookup-table T is defined per pixel and therefore the model requires significantly more parameters to be stored. It is thus necessary to combine this method with quantization approaches when handling materials that require significant spatial resolution. The model, as presented originally, was intended to independently represent the three channels of the RGB model by fitting individual Lafortune lobes and lookup-tables for each color channel.

Reflectance Functions

As a qualitative improvement over the previous method, Meseth et al. [Meseth04] published an approach to BTF rendering based on fitting a set of functions to the spatially varying reflectance functions of the BTF only and performing a simple linear interpolation for view directions not contained in the measured set. Following this proposal, the BTF is evaluated as follows:

$$\mathbf{BTF}(\mathbf{x},\mathbf{v},\mathbf{l}) \approx \sum_{v \in N(\mathbf{v})} w_{\mathbf{x},v} \cdot rf_{\mathbf{x},v}(\mathbf{l}) \quad (11.5)$$

Here, $N(\mathbf{v})$ denotes the set of closest view directions (a subset of the measured view directions), $w_{\mathbf{x},v}$ denotes the spatially varying interpolation weight, and $rf_{\mathbf{x},v}$ is the spatially varying reflectance function for view direction v which is approximated either by a biquadratic polynomial following the Polynomial Texture Map approach of Malzbender et al. [Malzbender01] or adopting Lafortune lobes as follows:

$$rf_{\mathbf{x},v}(\mathbf{l}) \approx \rho_{d,\mathbf{x}} + \rho_{s,\mathbf{x},v}(\mathbf{l}) \cdot \sum_{i=1}^{k} s_{\mathbf{x},v}(\mathbf{l}) \quad (11.6)$$

with $s_{\mathbf{x},v}$ similar to a spatially varying Lafortune lobe but for fixed view direction and k being the number of lobes.

Since the reflectance functions are fitted per pixel and measured view direction, the amount of parameters necessary to evaluate the model is higher than for the scaled Lafortune lobes model. Like the model of McAllister et al. [McAllister02], the approach is designed for efficient rendering and therefore the lobes are intended to compute luminance values that scale the RGB color albedo instead of fitting individual lobes for each color channel. Unlike previous methods based on function fitting, the approach requires interpolation between view directions, since the reflectance functions are defined for fixed view directions.

11.3.2 Linear Basis Decomposition

The other large group of BTF compression techniques interpret the measured data as a high-dimensional signal. Then general signal-processing techniques such as Principal Component Analysis (PCA) [Press92] can be applied. PCA minimizes the variance in the residual signal and provides the in a least-squares sense optimal affine-linear approximation of the input signal. We use the terms PCA and Singular Value Decomposition (SVD) synonymously during the rest of this section, since the principal components of the centered data matrix X are the columns of the matrix V with $X = U\Lambda V^T$ being the SVD of X.

PCA has been widely used in the field of image-based rendering to compress the image data. For example Nishino et al. [Nishino99] applied PCA to the reparameterized images of an object viewed from different poses and obtained so-called *eigen-textures*. Matusik et al. [Matusik02a] compressed the pixels of the captured reflectance field applying PCA to 8 by 8 image blocks.

The several BTF-compression methods that use PCA differ mainly in two points: (i) the slices of the data to which PCA is applied independently and (ii) how these slices are parameterized.

Per-Texel Matrix Factorization

One approach especially suited for real-time rendering applies PCA to the per-texel ABRDFs. Such methods were developed in the context of real-time rendering of arbitrary BRDFs at the time when the Phong-model was the state of the art in real-time rendering. The original idea as introduced by Kautz and McCool [Kautz99] can be stated as follows: Given the 4-dimensional BRDF $B_{\mathbf{x}}$, find a factorization into a set of 2-dimensional functions:

$$B_{\mathbf{x}}(\mathbf{v},\mathbf{l}) \approx \sum_{j}^{c} g_{\mathbf{x},j}(\pi_1(\mathbf{v},\mathbf{l}))h_{\mathbf{x},j}(\pi_2(\mathbf{v},\mathbf{l})) \tag{11.7}$$

The functions π_1 and π_2 are projection functions which map the 4D-dimensional BRDF parameters $(\mathbf{v},\mathbf{l})$ to a 2D-space. These projection functions have to be chosen carefully, because the parameterization significantly affects the quality of low-term factorizations. Given such a factorization real-time reconstruction of the BRDF using graphics hardware becomes easy, since the functions $g_{\mathbf{x},j}$ and $h_{\mathbf{x},j}$ can be stored in texture maps and combined during rendering. A trade-off between quality and speed is possible by controlling the number of terms c.

Several methods to find such factorizations have been proposed. Recently Suykens et al. [Suykens03] presented a method developed for BTFs called Chained Matrix Factorization (CMF) which has the following general factorization form:

$$B_{\mathbf{x}}(\mathbf{v},\mathbf{l}) \approx \prod_{j}^{d}\sum_{k}^{c_j} P_{\mathbf{x},j,k}(\pi_{j,1}(\mathbf{v},\mathbf{l}))Q_{\mathbf{x},j,k}(\pi_{j,2}(\mathbf{v},\mathbf{l})). \tag{11.8}$$

Such a chained factorization is computed using a sequence of simple factorizations using for example SVD, but each time in a different parameterization.

The compression ratio of per-texel matrix factorization depends on the size of the matrix X, i.e. the sampling of the angular space, and the number of factors. Please note, that these techniques where originally designed for BRDF rendering and for scenes containing a few (maybe some hundred) BRDFs. A single BTF with 256^2 texels contains 64k ABRDFs! Hence to reduce the memory requirements for real-time rendering a clustering of the factors may be necessary.

Full BTF-Matrix Factorization

The per-texel factorization methods from the previous section have the disadvantage, that they do not exploit inter-texel coherence. This can be accomplished by applying a PCA to the complete BTF-data arranged in a $|\mathcal{M}| \cdot |I|$ matrix

$$X = \left(B_{\mathbf{x}_0}, B_{\mathbf{x}_1}, \ldots, B_{\mathbf{x}_{|I|}}\right).$$

Keeping only the first c eigenvalues results in the following BTF-approximation:

$$\mathbf{BTF}(\mathbf{x}, \mathbf{v}, \mathbf{l}) \approx \sum_{j}^{c} g_j(\mathbf{x}) h_j(\mathbf{v}, \mathbf{l}) \tag{11.9}$$

This approach was used in the works of Liu et al. [Liu04] and Koudelka et al. [Koudelka03].

The remaining issue is, how to choose c. Ravi Ramamoorthi [Ramamoorthi02b] showed by an analytic PCA construction, that using about five components is sufficient to reconstruct lighting variability in images of a convex object with Lambertian reflectance. Therefore, for nearly diffuse and relatively flat samples a good reconstruction quality can be expected for low c. However, as illustrated in Figure 11.7, this will not be sufficient for complex BTFs containing considerable self-shadowing, masking and obviously for non-diffuse reflectance. Therefore, Koudelka et al. chose $c = 150$ to represent all significant effects with enough fidelity. To reduce the size of the resulting dataset even further they stored the basis-vectors as JPG-images resulting in very high compression rates. But of course real-time reconstruction from this representation is not possible.

An alternative approach for full BTF-matrix factorization was presented by Vasilescu and Terzopoulos [Vasilescu04]. They arranged the BTF-data in a 3-mode tensor and applied multi-linear analysis (3-mode SVD), which corresponds to the application of standard SVD to different arrangements of the data. It would be beyond the scope of this course notes to introduce the necessary concepts of tensor algebra, but it is worth noting that the resulting reconstruction formula provides a more flexible dimensionality reduction by allowing to reduce the represented variation in view and lighting directions independently. Compared to standard PCA with the same number of components the methods leads to a higher RMS-error, but the authors claim that keeping more variation in the viewing direction gives perceptually more pleasing results.

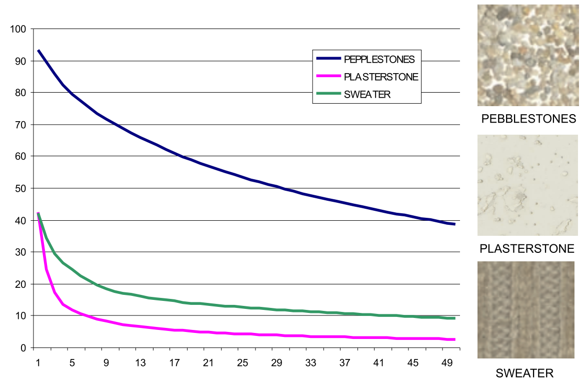


Figure 11.7: RMS-error of the full BTF-matrix factorization depending on the the number of terms c. There is a direct correspondence between the magnitude and decrease of error and the BTF-complexity.

A serious practical problem of the full BTF-matrix factorization methods is the size of the matrix X which easily could reach several gigabytes in float-precision. In this case, even the computation of the covariance matrix XX^T would be a very lengthy operation and special out-of-core routines have to be implemented. As an alternative the factorization can be computed for a subset of the data only.

Per-View Factorization

The reconstruction from the full BTF-matrix factorization is only fast *and* visually pleasing for relatively simple materials. Therefore, Sattler et al. [Sattler03] published a method that deals with these problems. Because their original intention was to visualize cloth BTFs that exhibit a significant amount of depth variation and hence highly non-linear view-dependence, they use an approach similar to the work of Meseth et al. [Meseth04] in the sense that they applied PCA to slices of the BTF with fixed view-direction. This leads to the following set of data-matrices:

$$X_{\mathbf{v}_i} := \left(T_{(\mathbf{v}_i,\mathbf{l}_0)}, T_{(\mathbf{v}_i,\mathbf{l}_1)}, \ldots, T_{(\mathbf{v}_i,\mathbf{l}_{M_{\mathbf{v}_i}})}\right)$$

with $M_{\mathbf{v}_i}$ denoting the number of sampled light directions for the given view direction $\mathbf{v}_i$. The PCA is applied to all matrices $X_{\mathbf{v}_i}$ independently. Then keeping only the first c eigenvalues gives the following BTF approximation:

$$\mathbf{BTF}(\mathbf{x}, \mathbf{v}, \mathbf{l}) \approx \sum_{j=1}^{c} g_{\mathbf{v},j}(\mathbf{l}) h_{\mathbf{v},j}(\mathbf{x}) \qquad (11.10)$$

Compared to Equation 11.9, the value of c can be set much lower (the authors chose c between 4 and 16) which enables interactive or even real-time rendering frame rates with good visual quality. However, the memory requirements are much higher. For example, $c = 16$ and $M_{\mathbf{v}_i} = 81$ lead to more than 1200 terms that have to be stored.

Per-Cluster Factorization

As already mentioned, a complex BTF contains highly non-linear effects like self-shadowing, self-occlusion and non-diffuse reflectance. Nevertheless, many high-dimensional data sets exhibit a local linear behavior. Applying per-texel or per-view factorization implicitly exploits this observation by selecting fixed subsets of the data and approximating these subsets with an affine-linear subspace. A more general approach would choose these subsets depending on the data. This is the idea behind the local PCA method, which was introduced by Kambhatla and Leen [Kambhatla97] to the machine-learning community in competition to classical non-linear/neural-network learning algorithms. It combines clustering and PCA using the reconstruction error as metric for choosing the best cluster.

Recently, Müller et al. [Müller03] applied this method to BTF-compression and proposed the following approximation:

$$\mathbf{BTF}(\mathbf{x}, \mathbf{v}, \mathbf{l}) \approx \sum_{j}^{c} g_{k(\mathbf{x}),j}(\mathbf{x}) h_{k(\mathbf{x}),j}(\mathbf{v}, \mathbf{l}) \qquad (11.11)$$

The operator $k(\mathbf{x})$ is the cluster index look-up given a position $\mathbf{x}$. In this case clustering is performed in the space of ABRDFs which was found being better suited for real-time rendering than clustering in the space of images. Now the number of clusters can be chosen according to computational resources and quality requirements. Good results were obtained for cluster counts between 16 and 32, which is much smaller than the fixed cluster number (e.g. $M_{\mathbf{v}_i} = 81$) used in per-view factorization.

11.3.3 Discussion

The following discussion is based on Figure 11.8.

It can be noted that the reconstruction quality from linear basis decompositions is better than from parametric reflectance models even if additional parameters like scaling values per view or even a full fit per view are used. Furthermore, the increased quality achieved by using this additional complexity does not legitimate the increased memory requirements. The qualitatively best result is achieved using per-view factorization but unfortunately the memory requirements are very high, since for every measured view direction a set of textures and weights has to be stored. Using per-texel matrix factorization does not exploit spatial coherence and thus the quality is not as good as

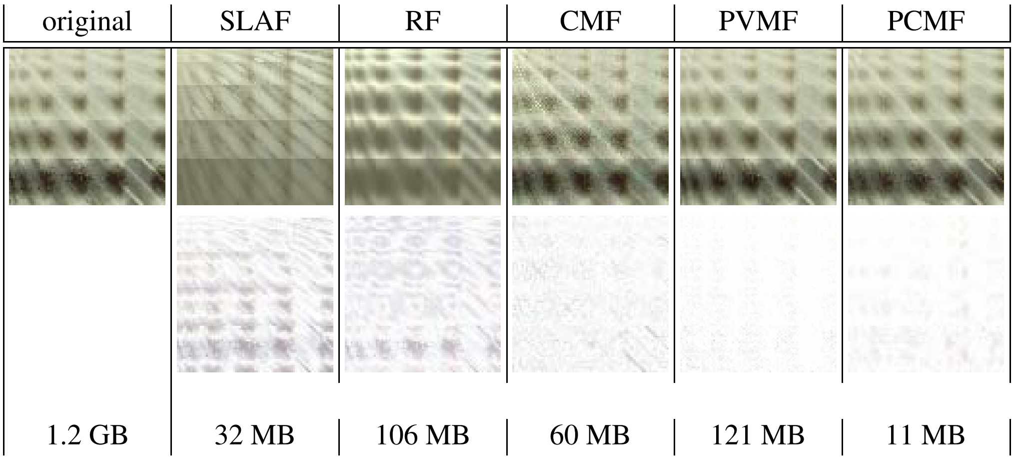


Figure 11.8: Comparison of selected BTF-compression methods. Top row: original and reconstructed ABRDFs. Second row: inverted difference images. Bottom row: storage requirements for the compressed representations using parameters suitable for interactive rendering. From left to right: Original ABRDF of PLASTERSTONE BTF, Scaled Lafortune Lobes with 2 lobes (SLAF), Reflectance Functions with per-view polynomial (RF), Per-Texel Chained Matrix Factorization with 4 factors (CMF), Per-View Matrix Factorization with 8 terms (PVMF), Per-Cluster Matrix Factorization with 32 clusters and 8 terms (PCMF). The original dataset consists of 6561 8-bit RGB-images with 256^2 pixels in size. Numerical precision is 16 bit for PCMF and 8 bit for all other methods.

per-view or per-cluster factorization even for considerable memory requirements. Furthermore, the chained resampling steps can introduce additional resampling error. Suykens et al. [Suykens03] propose k-means clustering of the factors across the spatial dimension to reduce memory requirements. This obviously could be applied to every per-texel BTF compression method, but for complex materials that do not contain uniform areas this will introduce cluster artifacts. These cluster artifacts are reduced using PCA in each cluster as done in the per-cluster factorization method. Hence, this method seems to offer a good compromise between reconstruction cost, visual quality and memory requirements.

11.4 Rendering

Generally accurate and physically plausible rendering algorithms have to compute a solution of the rendering equation (cf. Chapter 3) at every surface point $\mathbf{x}$ (neglecting emissive effects):

$$L_r(\mathbf{x},\mathbf{v}) = \int_{\Omega_i} \rho_{\mathbf{x}}(\mathbf{v},\mathbf{l}) L_i(\mathbf{x},\mathbf{l})(n_{\mathbf{x}} \cdot \mathbf{l})\, d\mathbf{l} \quad (11.12)$$

Here, $\rho_{\mathbf{x}}$ denotes the BRDF, L_i denotes incoming radiance, $n_{\mathbf{x}}$ is the surface normal and Ω_i is the hemisphere over $\mathbf{x}$.

Including measured BTFs into the rendering equation 11.12 is very simple:

$$L_r(\mathbf{x},\mathbf{v}) = \int_{\Omega_i} BTF(\mathbf{x},\mathbf{v},\mathbf{l}) L_i(\mathbf{x},\mathbf{l})(n_{\mathbf{x}} \cdot \mathbf{l})\, d\mathbf{l} \quad (11.13)$$

Now the BRDF at point $\mathbf{x}$ is simply looked up from the BTF. Here we assume that a mapping from the 3D-surface to the 2D spatial texture domain exists already. Please note that the BTF now also models small-scale geometry projected into planar domain of the BTF. Furthermore it can contain the foreshortening factor $(n_{\mathbf{x}} \cdot \mathbf{l})$ if it is not removed during the post-processing steps.

Equation 11.13 can be solved using monte-carlo techniques [Dutre04]. In this case a particular BTF compression scheme from Section 11.3 has to be supported which in fact corresponds to the implementation of the decompression stage on the CPU. Figure 11.9 shows an example.

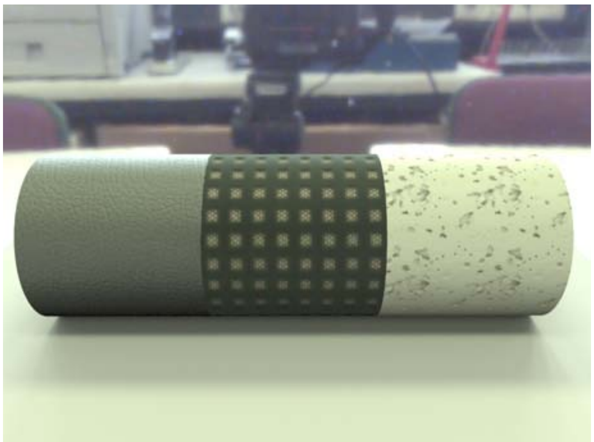


Figure 11.9: Cylinders covered with different BTFs (*leather, rustika, plasterstone*) and rendered using monte-carlo techniques. The BTFs have been compressed using per-cluster factorization.

The integration of BTFs in current real-time rendering systems is much more challenging because the evaluation of the BTF should be done on the graphics processing unit (GPU), i.e., integrated into the fragment-shading process. To achieve this, the compressed representation of the BTF must be stored in textures and the reconstruction is performed using the programmable units of the graphics board. The parameters for BTF evaluation are the standard 2D-texture coordinates $\mathbf{x}$, the local view direction $\mathbf{v}$ and in the case of point light sources the local light direction $\mathbf{l}$.

A crucial point in all BTF-rendering methods is interpolation. Since a measured BTF is discrete in all 6 dimensions, smooth renderings require interpolation in each dimension. To achieve high frame-rates it is indispensable, that at least some of these dimensions are interpolated using built-in hardware interpolation. For the other dimensions either the nearest neighbor must be chosen or the interpolation of the nearest neighbors has to be executed explicitly in the fragment shader. In both cases there has to be an operator $N(\cdot)$ that supplies the nearest samplings. Such a look up can be performed on the GPU using dependent texture look-ups.

In the following sections we will present and discuss the existing BTF-rendering methods that achieve interactive or even real-time frame-rates exploiting the capabilities of current graphics hardware.

11.4.1 Interactive Rendering of BTFs with Point Lights

The most popular simplification of the rendering equation allows only point light sources and ignores interreflections (cf. Equation 3.13). In this case the rendering equation reduces to a finite sum and can be efficiently evaluated on graphics hardware.

Figure 11.10 shows an example of BTF rendering with point light sources.

Parametric Reflectance Models

Efficient implementations of the parametric reflectance models from Section 11.3.1 have been presented by various publications. McAllister et al. [McAllister02] describe a real-time evaluation scheme for Equation 11.3. Coefficients of the spatially varying Lafortune lobes are stored in 2D textures. Evaluation can efficiently be done using vertex and fragment shaders. Since Lafortune lobes are continuous in the angular domain, no interpolation is required in the angular domain. Spatial interpolation has to be done explicitly in the fragment shader (magnification) or is left to the multi-sampling and built-in MIP-mapping capabilities of

Figure 11.10: Visualization of car interior (Mercedes C-class) with point light illumination. The leather and wood materials are modeled by BTFs. All other materials are standard phong.

graphics hardware, although MIP-maps have to be built manually (e.g. by fitting new sets of Lafortune lobes for each BTF resolution).

As an extension to the model of McAllister et al., the BTF model of Daubert et al. [Daubert01] only requires an additional evaluation of the view-dependent visibility term. Although significant numbers of parameters are required to store this term, it can easily be encoded in a texture. Interpolation of the view-direction can be achieved using graphics hardware. Spatial interpolation is done like in the previous approach.

The even more complex model of Meseth et al. [Meseth04] is evaluated very similarly to the previous two ones with the significant difference that the discretization of the view direction requires an additional manual interpolation (as denoted in Equation 11.5). Therefore, two cube maps are utilized which store for each texel in the cube map (representing a certain view direction) the three closest view directions and respective interpolation weights. Interpolation is achieved by evaluating the reflectance functions for the three closest view directions and interpolating the results based on the interpolation factors stored in the cube map. Spatial interpolation can be done exactly like in the previous approaches.

Per-Texel Matrix Factorization

Generally, the rendering algorithms for BRDF-factorizations can be used [Kautz99] with the difference, that the factors now change in every fragment. Suykens et al. [Suykens03] detailed how this can be accomplished:

Every factor is reparameterized and stored into a parabolic map [Heidrich98]. Then all these factors are stacked into 3D-textures and normalized to a range between 0 and 1. The resulting scale factors are stored in a scale map. A particular factor is selected in its 3D-texture using transformed 2D texture coordinates or, if clustered factors are employed, by the values from an index map. To avoid mixing of neighboring factors in the 3D-texture the z-coordinate has to be chosen carefully. A value within a particular factor is indexed using the appropriately reparameterized local view and light directions. While interpolation inside a factor is supported by the hardware, spatial interpolation between neighboring texels is not implemented. Equation 11.8 now can be executed in a fragment shader.

Per-View Matrix Factorization

Sattler et al. [Sattler03] demonstrated how Equation 11.10 can be implemented using a combination of CPU and GPU computations.

Figure 11.11 shows the basic approach: Combination of the eigen-textures $h_{\mathbf{v},j}(\mathbf{x})$ with the appropriate weights $g_{\mathbf{v},j}(\mathbf{l})$ using the multi-texturing capabilities of modern graphics boards. This combination is done for every triangle-vertex. A smooth transition is ensured by blending the resulting three textures over the triangle using a fragment program [Chen02].

While the interpolation in the spatial domain is done by the graphics hardware, light and view direction are interpolated explicitly. To interpolate

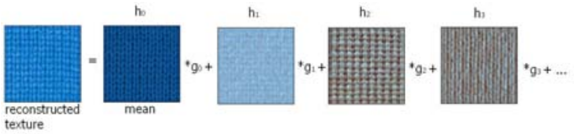


Figure 11.11: Schematic rendering using the per-view matrix factorization.

between the nearest measured light directions the term

$$\mathbf{BTF}(\mathbf{x},\mathbf{v},\mathbf{l}) \approx \sum_{\tilde{l}\in N(\mathbf{l})} \tau_{\tilde{l}} \sum_{j=1}^{c} g_{\mathbf{v},j}(\tilde{l}) h_{\mathbf{v},j}(\mathbf{x}) \tag{11.14}$$

has to be evaluated. In order to speed up this computation the weights $\lambda_{\mathbf{v},j}(\mathbf{l}) = \sum_{\tilde{l}\in N(\mathbf{l})} \tau_{\tilde{l}} \cdot g_{\mathbf{v},j}(\tilde{l})$ are computed on the CPU per frame and sent to the GPU resulting in the term

$$\mathbf{BTF}(\mathbf{x},\mathbf{v},\mathbf{l}) \approx \sum_{j=1}^{c} \lambda_{\mathbf{v},j}(\mathbf{l}) h_{\mathbf{v},j}(\mathbf{x}). \tag{11.15}$$

To perform view-interpolation different sets of eigen-textures have to be combined resulting in an expensive combination:

$$\mathbf{BTF}(\mathbf{x},\mathbf{v},\mathbf{l}) \approx \sum_{\tilde{v}\in N(\mathbf{v})} \tau_{\tilde{v}} \sum_{j=1}^{c} \lambda_{\tilde{v},j}(\mathbf{l}) h_{\tilde{v},j}(\mathbf{x}). \tag{11.16}$$

Full BTF-Matrix Factorization

To evaluate Equation 11.9 the factors $g_j(\mathbf{x})$ and $h_j(\mathbf{v},\mathbf{l})$ have to be evaluated, whereas the factors $g_j(\mathbf{x})$ can be stored as simple 2D-textures and the factors $h_j(\mathbf{v},\mathbf{l})$ as 4D-textures. But unfortunately neither 4D-textures nor their full interpolation is currently supported by graphics hardware.

Therefore, Liu et al. [Liu04] store the factors $h_j(\mathbf{v},\mathbf{l})$ into stacks of 3D-textures. The tri-linear filtering capabilities of graphics hardware now can be exploited to interpolate the view direction and the polar angle of the light direction. The final step for 4D filtering is performed manually by blending two, tri-linearly filtered values with closest azimuth angles in light direction. As usually, the values of $h_j(\mathbf{v},\mathbf{l})$ parameterized over the hemispheres of view and light directions have to be resampled and reparameterized in order to be stored in textures. In order to avoid the fragment shader's online effort of calculating the reparameterized local light and view directions, which are necessary for accessing the 3D-textures and interpolation weights, the mapping is precomputed and stored in a cube map.

Per-Cluster Matrix Factorization

Apart from the additional cluster look-up, evaluating Equation 11.11 is essentially the same as evaluating 11.9. Hence the real-time rendering algorithm for Equation 11.11 presented by Schneider [Schneider04] is similar in style to the approach presented in the previous section. He also stores the factors $h_{k(\mathbf{x}),j}(\mathbf{v},\mathbf{l})$ in stacks of 3D-textures and accesses them through reparameterized local light and view directions. The cluster index introduces an additional dependent texture look-up. Since existing graphics boards only support hardware supported interpolation of fixed point values, every factor is quantized to 8-bit separately yielding scaling factors $s_{j,k}$. As compensation the factors $g_j(\mathbf{x})$, which can be stored as floating-point values since they require no interpolation, have to be divided by the corresponding scaling factor $s_{j,k}$.

Mipmapping the BTF can simply be implemented by executing the shader instructions twice (once for the currently best mipmap level and once for the next best mipmap level) and interpolating the resulting colors. Bilinear, spatial interpolation is currently not supported, since the additional overhead is prohibitive. Fortunately, hardware supported full-screen antialiasing can reduce potential artifacts significantly.

11.4.2 Combining BTFs and Precomputed Radiance Transfer

We consider relighting of BTFs with an infinitely distant illumination environment L_i. Typically this distant illumination is represented by an environment map. Such an environment map can either be computed by the graphics hardware or captured from a natural environment by taking pictures (e.g. of a metallic sphere) [Debevec97]. Now the dependency of the incident lighting on surface position $\mathbf{x}$ can be removed and equation 11.12 reads as follows:

$$L_r(\mathbf{x}, \mathbf{v}) \approx \int_{\Omega_i} \rho_{\mathbf{x}}(\mathbf{v}, \mathbf{l}) L_i(\mathbf{l})(n_{\mathbf{x}} \cdot \mathbf{l})\, d\mathbf{l} \quad (11.17)$$

For special types of BRDFs this integral can be precomputed (e.g. [Kautz00b]) and the results can be displayed in real-time. Corresponding approaches for BTFs exist ([McAllister02, Meseth04] which we will not discuss here (for space reasons). Instead we will concentrate on recent approaches in the field of Precomputed Radiance Transfer (PRT) which reduce the integral to a finite sum by projecting light and BRDF onto a basis like Spherical Harmonics (SH) [Sloan02] or wavelets [Ng03] and keeping only a finite number of terms. These approaches not only allow for dynamic change of lighting but can also include precomputed transport effects like shadows and interreflections.

Bi-Scale Radiance Transfer

PRT, as originally introduced by Sloan et al. [Sloan02], is evaluated per vertex only and interpolated across the triangle. In a follow up work, Sloan et al. [Sloan03b] extended PRT to support also spatially varying reflectance across the triangle. They projected the BTF per pixel and fixed view direction onto the SH basis generating a Radiance Transfer Texture (RTT) which now represents the per-view response of the material to the spherical harmonics basis. To cover large geometry with the memory-intensive texture they used the synthesis algorithm of Tong et al. [Tong02] to generate an ID-map over the mesh which references into the RTT.

Generation of the RTT and the ID-map The generation of the RTT $B(\mathbf{x}, \mathbf{v})$ is accomplished by projecting the BTF onto the first c elements of the SH basis $\{Y_j\}_{j \in \mathbf{N}}$:

$$B(\mathbf{x}, \mathbf{v})_j = \int_{\Omega_i} BTF(\mathbf{x}, \mathbf{v}, \mathbf{l}) Y_j(\mathbf{l})\, d\mathbf{l} \quad (11.18)$$

Thus, the RTT is a 4D-array of c Spherical Harmonics coefficients (typically $c = 25$).

The ID-map is generated using BTF synthesis over densely resampled geometry called a *meso-mesh*. This step assigns every vertex an ID into the RTT. Then a texture atlas for the original coarse mesh is generated and for every texel in this atlas the nearest vertex in the meso-mesh is found and the corresponding ID is assigned to the texel.

Real-Time Rendering Standard PRT-rendering is performed per-vertex. That means computation of a matrix-vector multiplication between the transfer matrix and the incident lighting SH-coefficients. The resulting transferred lighting vector is interpolated across the triangle. To compute exitant radiance per-fragment the interpolated transferred lighting vector is dot multiplicated with the corresponding vector in the RTT which is accessed by the ID map and the local view direction. This is done in a fragment program. Each texel of the RTT is stored in an 8x8 texture block encoding the view dependence. This allows smooth bilinear interpolation across views using built-in hardware interpolation. Interpolation between spatial samples is not performed.

Since the RTTs are not compressed, the method supports only sparse samplings. They used 8x8 view-samples and 64x64 spatial samples which results in $64^3 * 25$ SH-coefficients that have to be stored per color band.

Per-View Matrix Factorization

Sattler et al. [Sattler03] also proposed a method to relight the per-view factorized BTFs by environment maps that also supports large scale shadows. The main idea is to discretize the integral in Equation 11.17 using a hemicube [Cohen85], which leads to

$$L_r(\mathbf{x}, \mathbf{v}) \approx \sum_{\hat{l} \in H_i} \rho_{\mathbf{x}}(\mathbf{v}, \hat{l}) H_{i,\mathbf{x}}(\hat{l})(n_{\mathbf{x}} \cdot \hat{l}) \quad (11.19)$$

where $H_{i,\mathbf{x}}$ denotes the discretized hemicube. $H_{i,\mathbf{x}}(\hat{l})$ returns the color in the hemicube over $\mathbf{x}$ at direction $\hat{l}$ or zero if the direction is occluded. The hemicube is precomputed and stored in a visibility map as follows:

Hemicube Precomputation The hemicube $H_{i,\mathbf{x}}$ stores a discretization of the hemisphere at the vertex $\mathbf{x}$. Figure 11.12 (left) shows an unfolded hemicube. Using a color-coded environment map (Figure 11.12 middle) a look-up table into a high dynamic range map (Figure 11.12 right) is created. This allows easy exchange of the environment map. By rendering the geometry in white color into the hemicube the visibility function is computed and self-shadowing can be supported. Since the directions in the visibility map not necessarily correspond to the measured light directions in the BTF, the map furthermore stores the four nearest measured light directions and the corresponding interpolation weights.

A hemicube pixel now stores the following information:

- visibility of a pixel of the environment map and if it is visible, the position of this pixel in the map
- four nearest measured directions with respect to the direction represented by this pixel
- corresponding interpolation weights

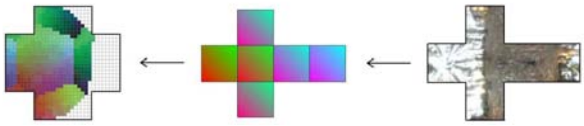


Figure 11.12: Hemicube computation. Visibility map (left) with rendered color-coded lookup environment map (middle). White color in the visibility map stands for occlusion caused by the mesh. On the right side a HDR environment is shown, which is mapped onto the color-coded one.

Rendering algorithm Given the nearest measured view direction $\mathbf{v}$ at vertex $\mathbf{x}$ and substituting the BRDF in Equation 11.19 by the per-view factored BTF representation including the foreshortening term yields the following sum:

$$L_r(\mathbf{x}, \mathbf{v}) \approx \sum_{\mathbf{l} \in H_i} \left(\sum_{j=1}^{c} \lambda_{\mathbf{v},j}(\mathbf{l}) h_{\mathbf{v},j}(\mathbf{x}) \right) H_{i,\mathbf{x}}(\mathbf{l}) \quad (11.20)$$

The factors $\lambda_{\mathbf{v},j}(\mathbf{l})$ are from Equation 11.15. As in Equation 11.14 the factors $\gamma_{\mathbf{v},j} = \sum_{\mathbf{l} \in H_i} \lambda_{\mathbf{v},j}(\mathbf{l}) H_{i,\mathbf{x}}(\mathbf{l})$ are precomputed per vertex and sent to the GPU where the final expression is evaluated:

$$L_r(\mathbf{x}, \mathbf{v}) \approx \sum_{j=1}^{c} \gamma_{\mathbf{v},j} h_{\mathbf{v},j}(\mathbf{x}) \quad (11.21)$$

which again is only a linear combination of basis textures. For view interpolation the same calculations as in Equation 11.16 have to be applied.

Since $\gamma_{\mathbf{v},j}$ is computed for all vertices $\mathbf{x}$ of the geometry a vector U holding all $\gamma_{\mathbf{x},\mathbf{v},\mathbf{j}}$ can be introduced:

$$U = (\gamma_{\mathbf{x}_0,\mathbf{v}_0,1}, \ldots, \gamma_{\mathbf{x}_0,\mathbf{v}_0,c}, \ldots) \quad (11.22)$$

This vector has do be calculated once per environment map and allows real-time changes of the viewing position. A drawback of this method is, that changing the environment map and even a simple rotation implies a complete recalculation

Figure 11.13: Combination of BTF rendering and precomputed radiance transfer for photorealistic rendering of textiles.

of U. Depending on the visibility map resolution and the number of vertices, this computation can take too long for interactive change of lighting. Reducing the visibility map resolution adaptively to achieve interactive changing rates introduces under-sampling artifacts of the environment map during motion, which can be compensated if the change stops, by using an adaptively higher resolution for the visibility map.

Per-Cluster Matrix Factorization

To reduce the memory consumption of the Bi-Scale Radiance Transfer method Müller et al. [Müller04] presented a combination of per-cluster factorized BTFs with PRT.

Data Representation Similar to Sloan et al. [Sloan03b] they compute Radiance Transfer Textures but encode them using the local PCA method [Kambhatla97]. In addition, similar to Sloan et al. [Sloan03a], they apply local PCA to the transfer matrices of the mesh vertices. An approximate solution to Equation 11.13 can now be computed by the weighted sum of dot products between the PCA-factors of the RTT and the transfer matrices.

Real-Time Rendering Since the dot products remain constant as long as only the camera is moving in the scene (i.e. neither the mesh nor the environment nor the BTF is changing), they can be precomputed on the CPU and afterwards be stored in a texture. Since precomputation and upload times of the dot products do not allow interactive rendering for very high quality settings, the number of RTT and transfer matrix components is reduced in dynamic situations. These reductions in quality do not significantly influence the perceived quality of the rendered results as long as high quality solutions are presented in static cases.

Like in Sloan et al. [Sloan03a], rendering requires clusters of triangles to be rendered independently, where a triangle belongs to a certain cluster if at least one of its vertices belongs to the respective clusters.

Most rendering power is spent in the fragment program which computes the weighted sum. Interpolation of the angular domain of the BTF can be achieved by standard filtering features, spatial interpolation and filtering can be achieved using standard multisampling and MIP-mapping of the BTF.

11.4.3 Discussion

To our experience almost all real-time BTF rendering methods pose a great challenge to the current graphics boards and the performance differences vary greatly depending on the board and driver versions. Therefore, we will discuss some general "hardware-independent" properties of the algorithms which might help judging the strengths and weaknesses of the algorithms.

The method of McAllister et al. [McAllister02] is mainly suited for nearly flat and specular materials with spatially varying reflection properties. Since it only uses slightly more memory than ordinary diffuse textures and can be rendered fastly, it fits into current rendering systems. This is not true for the methods which use additional data to capture also depth-varying BTFs such as

[Daubert01, Meseth04]. Their memory consumption and increased rendering cost restricts them to special domains and as pointed out in Section 11.3 the visual quality of the used analytical models still remains questionable.

Far better visual quality is offered by the methods based on matrix factorization. But using PCA alone as done by Liu et al. [Liu04] is only real-time for very few terms, and thus only applicable for simple materials. Therefore, a segmentation of the data into subsets may be necessary. Using a segmentation per fixed view direction as done by Sattler et al. [Sattler03] provides excellent visual quality but requires too much texture memory to be used in complex scenes with many materials. Spatial clustering as in the method of Müller et al. [Müller03] or [Suykens03] reduces the memory requirements drastically while retaining high-quality but these methods face the not sufficiently solved problem of spatial interpolation and Mip-Mapping. This can result in decreased quality for texture magnification and minification. A problem of all matrix factorization based methods is the required random access to many textures, which can form a bottleneck on current graphics architectures.

Chapter 12

Reflectance Fields

Tim Hawkins

12.1 Introduction

Image based modelling and rendering methods have become commonplace over the last decade. These methods typically provide a very high level of photorealism, but it is often difficult to edit the geometry or modify the illumination conditions. Recently, the data-driven approach embodied in image based modelling and rendering has been extended to capture models than can be reilluminated with any desired lighting. The reflectance field, as introduced in [Debevec00], provides a unifying framework for many of these new techniques.

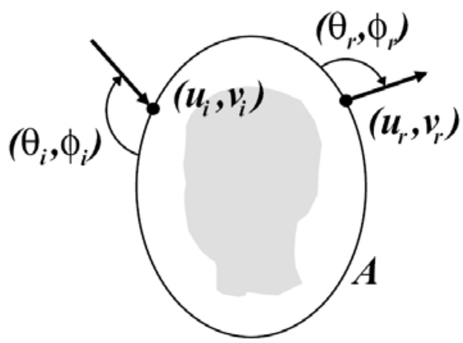


Figure 12.1: Geomtery for definition of the reflectance field.

The reflectance field describes the transfer function between an incident lighting condition represented as a light field, and the resulting exitant light field. As shown in Figure 12.1, the reflectance field is defined on a surface A that is assumed to contain some object of interest. Defined by analogy with the BRDF, the reflectance field is an 8-dimensional function expressing the differential exitant radiance at a given point (u_r, v_r) in a given direction (θ_r, ϕ_r) due to a differential incident radiant flux at the point (u_i, v_i) in the direction (θ_i, ϕ_i).

$$R = R(u_i, v_i, \theta_i, \phi_i; u_r, v_r, \theta_r, \phi_r) \qquad (12.1)$$

Whether A contains complicated geometry, materials with unusual reflectance properties, volumetric participating media, or any combination of these, the reflectance field effectively captures the appearance of the interior of A under any lighting conditions.

The reflectance field is very closely related to the Bidirectional Scattering-Surface Reflectance Distribution Function (BSSRDF) [Nicodemus77] and the Global Reflectance Distribution Function (GRDF) [Lafortune94]. Unlike the BSSRDF, the reflectance field is parameterized on a surface which is not assumed to be coincident with a physical surface. The GRDF actually describes the transfer of light power between any two rays. Because ray space is 5-dimensional, the GRDF is in general a 10-dimensional quantity. However, Lafortune deals primarily with the restriction to 8 dimensions resulting from assuming that the value of the function need only be considered at the defined surfaces of a scene.

Although the reflectance field is most easily imagined as being defined on a sphere containing the object of interest, in some cases it may be preferred to define it instead on some other surface. In the case that it is defined on a surface coinciding with the object's surface, it is called a surface reflectance field. In this case, the definition of the reflectance field is slightly complicted by the possibility that the surface A may be nonconvex, allowing for a given ray to be both exitant and incident. In this case the formulation must be made more precise. For a given pair of rays r_i and r_o, where $r_i = r_o$ is permitted, we are interested in the transfer function between an assumed source or self-emissive radiance along ray r_i, and the resulting total radiance travelling along ray r_o. (This final total radiance can be undefined in certain physically implausible circumstances. For example between two perfectly reflective mirrors or inside of a sphere with a perfectly white surface, all sourced light power can be trapped in a finite space, resulting in radiance values that increase forever.)

For an object surrounded by a surface A, the function reflectance field R on A allows the object to be rendered under arbitrary illumination and from any viewpoint, provided that the viewpoint is outside of A and that the illumination originates outside of A. If R is known, geometry and material properties of the object are no longer needed. Alternatively, for the case where A is a sphere it could be said that the reflectance field is really just an extreme version of a geometry plus reflectance model of the contained object, where the geometry has been greatly simplified and all interesting information has instead been encoded as a high-dimensional nonlocal material reflectance.

(a)

(b)

Figure 12.2: (a) The Light Stage 1 consists of a single light mounted on a system of two nested plastic arms, where the outer arm spins around a vertical axis while the inner arm can be raised and lowered. (b) A long exposure (about 60 seconds) showing the movement of the light source over the course of a data capture.

Reflectance field rendering is analogous to convolution, except that instead of a fixed convolution kernel, the lighting response kernel is generally different for every exitant ray. The rendering process requires no visibility calculations or recursive ray casting. Rather, rendering an image pixel corresponding to a ray r_o becomes a simple dot

Figure 12.3: The top row shows several captured basis images of a subject under different directions of illumination. The bottom row shows image based relightings of a subject under different environmental illumination.

product between the 4D subspace of the reflectance field describing the response of r_o and the desired lighting condition specified as a 4D light field.

Because of the high dimensionality of the reflectance field, no measurements have been made and no computations have been performed on a full 8 dimensional reflectance field. However, a number of researchers have performed measurements and computations with various simplified versions, generally 4-dimensional or 6-dimensional. We give a brief review of these efforts.

12.2 Light Stage 1 and Faces

[Debevec00] used a lighting apparatus dubbed Light Stage 1 to acquire images of a human subject under all directional illumination conditions. This apparatus, shown in Figure 12.2, allows a single light source to be moved to all directions surrounding a seated subject. A video camera then captures a sequence of images of the subject with all directional illumination conditions. Figure 12.3 shows a few examples of such captured images. The resulting dataset can be considered to be a 4-dimensional slice of the full reflectance field, capturing for a fixed viewpoint the appearance of the subject under any distant illumination. For an exitant ray r_o corresponding to a virtual camera pixel, the radiance under any combination of directional lights is proportional to

$$L(r_o) = \sum R(r_i)L(r_i)\Delta\omega(r_i) \qquad (12.2)$$

where $R(r_i)$ was the measured radiance at the pixel corresponding to r_o when illuminated from direction r_i, $L(r_i)$ is the desired illumination from direction r_i, and $\Delta\omega(r_i)$ is the solid angle determined by the spacing of incident illumination samples. This equation allows images to be produced simulating the appearance of the subject under any desired nonlocal lighting condition. A few examples of relit versions of a subject are shown in Figure 12.3.

This basic image based relighting process can reproduce the full range of reflectance phenom-

(a)

(b)

Figure 12.4: (a) The Light Stage 2 consists of 30 strobe lights mounted on a semi-circular arm. The arm spins around a vertical axis, allowing all directions of illumination to be provided. (b) A long exposure (about 10 seconds) showing the accumulation of all lights flashed during a single data capture.

ena an object can exhibit under distant illumination, including diffuse and specular reflection, self-shadowing, translucency, and caustics. Care must be taken to capture a sufficient number of incident lighting directions for the object of interest. If the lighting samples are too far apart, problems will be seen for highly specular objects and for self-shadowing objects. In particular, a highly specular surface reflecting a diffuse lighting environment will appear to reflect many small light sources rather than the smooth environment, and self-shadowing objects when illuminated by a moving sharply directional light will exhibit shadows that progress in a series of steps rather than continuously.

12.3 Light Stage 2 and Facial Animation

[Hawkins04] demonstrates a system for rapidly capturing reflectance fields from several cameras. This allows for many different facial expressions to be captured to build a morphable face model that can be lit and viewed arbitrarily. The capture system is shown in Figure 12.4. The system can capture illuminations conditions at the rate of 60 per second, allowing 480 lighting conditions to be captured in 8 seconds. This capture is fast enough that the subject is able to maintain even extreme facial expressions without excess motion.

To register the many facial expressions, roughly 300 fiducial dots are placed on the subject's face. Triangulating these dots from the six cameras provides a rough geometric model. Because six cameras do not provide an adequate sampling of the radiant light field, this rough geometry is required to allow for high quality rendering from arbitrary viewpoints. Examples of renderings are seen in Figure 12.5.

12.4 Image-Based 3D Photography using Opacity Hulls

[Matusik02b] demonstrated a system for capturing a subject under many directional illumination conditions and from many viewpoints, yielding a 6D version of the reflectance field. The capture setup for this system can be seen in Figure 12.6. In their system, a one dimensional array of cameras captures images of a central object as illuminatied by a one dimensional array of lights. By rotating the object and the array of lights independently, images can be captured of the object from every viewpoint and illuminated from every direction. Like [Debevec00] the effects of spatially inhomogeneous lighting are not captured with this technique.

Although the number of captured viewpoints was relatively large (about 200), this number is

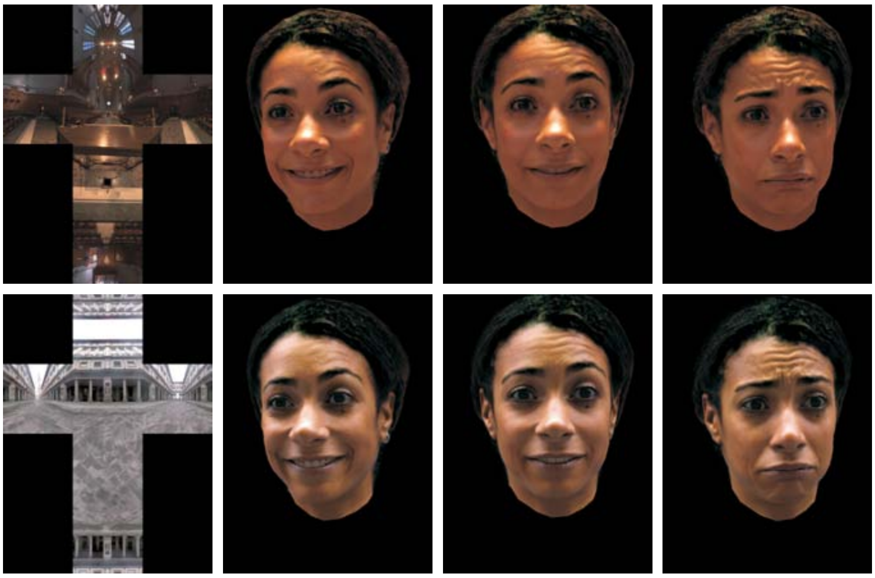


Figure 12.5: An animated face rendered from novel viewpoints and under novel illumination.

(a) (b)

Figure 12.6: (a) Apparatus for capturing a 6D reflectance field and opacity hull geometry model of an object. (b) Renderings of different objects under specified illumination conditions. The opacity hull model allows for objects with very complicated geometry such as fur or leaves to correctly occlude the background.

still too small to allow pure light field rendering with no geometry. For this reason, additional images at each viewpoint are captured of the object backlit by a monitor displaying several patterns. These images allow the object silhouette to be easily identified. Extracting geometry from visual hull (silhouette) information then allows for high quality view interpolation renderings, analogous to depth correction for light field rendering.

12.5 Relighting with 4D Incident Light Fields

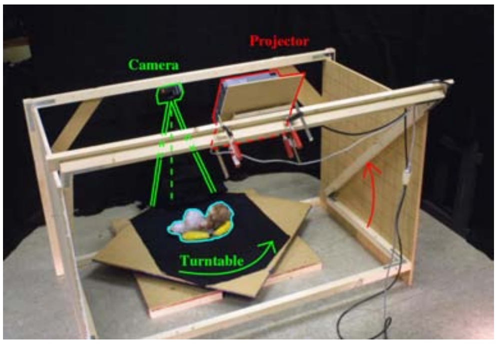
[Masselus03] used a movable projector to capture the response of an object to arbitrary incident light fields. Their capture setup is shown in Figure 12.7. The 4 dimensions of the incident light field along with the 2 dimensions of image space make this also a 6D slice of the reflectance field.

Although the images are captured only for a single viewpoint, this method allows for renderings under completely arbitrary incident illumination, including inhomogeneous lighting effects such as those due to spotlights or local occluders.

To speed up the capture process, this work takes advantage of the fact that for many objects the projection of a small spot of light primarily illuminates only a subset of the captured image. By projecting several separate spots of lights simultaneously that are relatively unlikely to illuminate the same image pixels, the reflectance capture can be parallelized.

12.6 Summary

A number of techniques in recent years have demonstrated photoreal relightable image based models. In particular, a number of efforts have focussed on capturing images under many illumination conditons, allowing for this relighting to be performed directly from captured data without making simplifying assumptions about the material properties or light transport within the scene. These approaches can often be considered attempts at characterizing or measuring the reflectance field of the scene, a prospect which remains elusive given the high dimensionality of the reflectance field.



(a)

(b)

(c)

Figure 12.7: (a) Apparatus for capturing appearance of an object under any incident light field. (b), (c) Renderings of a scene relit with different incident light fields. The effects of spatially-varying illumination are evident.

Bibliography

[Adelson01] E. H. ADELSON. On Seeing Stuff: the Perception of Materials by Humans and Machines. In Bernice E. Rogowitz and Thrasyvoulos N. Pappas, editors, *Proceedings of SPIE - Human Vision and Electronic Imaging IV*, volume 4299, pages 1–12, 2001.

[Agarwal04] SAMEER AGARWAL, SATYA P. MALLICK, DAVID J. KRIEGMAN, AND SERJE J. BELONGIE. On Refractive Optical Flow. In *Proceedings of 8th European Conference on Computer Vision*, pages 483–494, 2004.

[Ashikhmin00] M. ASHIKHMIN, S. PREMOZE, AND P. SHIRLEY. A Microfacet-based BRDF Generator. In *Proc. SIGGRAPH*, pages 65–74, July 2000.

[Banks94] D. BANKS. Illumination in Diverse Codimensions. In *Proc. SIGGRAPH*, pages 327–334, July 1994.

[Bayer76] BRYCE E. BAYER. Color imaging array. US Patent 3,971,065, 1976.

[Beckmann63] P. BECKMANN AND A. SPIZZICHINO. *The Scattering of Electromagnetic Waves from Rough Surfaces*. McMillan, 1963.

[Blasi93] PHILIPPE BLASI, BERTRAND LE SAËC, AND CHRISTOPHE SCHLICK. A Rendering Algorithm for Discrete Volume Density Objects. *Computer Graphics Forum*, 13(3):201–210, 1993.

[Blinn76] JAMES F. BLINN AND MARTIN E. NEWELL. Texture and Reflection in Computer Generated Images. *Communications of the ACM*, 19(10):542–546, 1976.

[Blinn77] J. BLINN. Models of Light Reflection For Computer Synthesized Pictures. In *Proc. SIGGRAPH*, pages 192–198, July 1977.

[Boivin01] SAMUEL BOIVIN AND ANDRÉ GAGALOWICZ. Image-Based Rendering of Diffuse, Specular and Glossy Surfaces From a Single Image. In Eugene Fiume, editor, *Proceedings of SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 107–116. ACM Press / ACM SIGGRAPH, August 2001. ISBN 1-58113-292-1.

[Born93] MAX BORN AND EMIL WOLF. *Principles of Optics*. Pergamon Press, Oxford, 6 edition, 1993.

[Bouguet] JEAN-YVES BOUGUET. Camera Calibration Toolbox for Matlab. See `http://www.vision.caltech.edu/bouguetj`.

[Brand03] M. BRAND. Charting a manifold. In *Advances in Neural Information Processing Systems*, volume 15, 2003.

[Brinkman99] RON BRINKMAN. *The Art and Science of Digital Compositing*. Morgan Kaufman, 1999.

[BTF Database Bonn] BTF DATABASE BONN. http://btf.cs.uni-bonn.de.

[Burns01] PETER D. BURNS AND DON WILLIAMS. Image Resolution and MTF for Digital Cameras and Scanners. In *Tutorial Notes, IS&T PICS Conference 2001*, 2001.

[Cabral87] B. CABRAL, N. MAX, AND R. SPRINGMEYER. Bidirectional Reflection Functions From Surface Bump Maps. In *Proceedings SIGGRAPH*, pages 273–281, Anaheim, California, July 1987.

[Cabral99] B. CABRAL, M. OLANO, AND P. NEMEC. Reflection Space Image Based Rendering. In *Proceedings SIGGRAPH*, pages 165–170, aug 1999.

[Carceroni01] RODRIGO L. CARCERONI AND KIRIAKOS N. KUTULAKOS. Multi-View Scene Capture by Surfel Sampling: From Video Streams to Non-Rigid 3D Motion Shape & Reflectance. In *ICCV*, pages 60–67, 2001.

[Chen02] YANYUN CHEN, YINGQING XU, BAINING GUO, AND HEUNG-YEUNG SHUM. Modeling and rendering of realistic feathers. In *Proceedings of SIGGRAPH*, pages 630–636. ACM Press, 2002.

[Chuang00] YUNG-YU CHUANG, DOUGLAS E. ZONGKER, JOEL HINDORFF, BRIAN CURLESS, DAVID H. SALESIN, AND RICHARD SZELISKI. Environment Matting Extensions: Towards Higher Accuracy and Real-Time Capture. In *Proceedings of ACM SIGGRAPH 2000*, pages 121–130, 2000.

[Cohen85] M. F. COHEN AND D. P. GREENBERG. The hemicube: A radiosity solution for complex environments. *Symposium on Computational Geometry*, 19(3):31–40, July 1985.

[Cohen93] MICHAEL F. COHEN AND JOHN R. WALLACE. *Radiosity and Realistic Image Synthesis*. Academic Press, Boston, 1993.

[Cook84] R. L. COOK, T. PORTER, AND L. CARPENTER. Distributed Ray Tracing. In *Proceedings SIGGRAPH*, pages 137–145, July 1984.

[Dana97] KRISTIN J. DANA, BRAM VAN GINNEKEN, SHREE K. NAYAR, AND JAN J. KOENDERINK. Reflectance and texture of real-world surfaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 151–157, 1997.

[Dana99] K. DANA, B. VAN GINNEKEN, S. NAYAR, AND J. KOENDERINK. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics*, 18(1):1–34, January 1999.

[Dana04] KRISTIN J. DANA AND JING WANG. Device for convenient measurement of spatially varying bidirectional reflectance. *Journal of the Optical Society of America*, A:1–12, January 2004.

[Daubert01] KATJA DAUBERT, HENDRIK LENSCH, WOLFGANG HEIDRICH, AND HANS-PETER SEIDEL. Efficient Cloth Modeling and Rendering. In *Proceedings of EGRW '01*, pages 63–70, 2001.

[Daubert02] K. DAUBERT AND H. SEIDEL. Hardware-Based Volumetric Knit-Wear. *Computer Graphics Forum*, 21(3):575–784, 2002.

[Debevec97] P. DEBEVEC AND J. MALIK. Recovering High Dynamic Range Radiance Maps from Photographs. In *Proc. SIGGRAPH*, pages 369–378, August 1997.

[Debevec00] P. DEBEVEC, T. HAWKINS, C. TCHOU, H.-P. DUIKER, W. SAROKIN, AND M. SAGAR. Acquiring the Reflectance Field of a Human Face. In *Proc. SIGGRAPH*, pages 145–156, July 2000. ISBN 1-58113-208-5.

[Debevec04] PAUL DEBEVEC, CHRIS TCHOU, ANDREW GARDNER, TIM HAWKINS, CHARIS POULLIS, JESSI STUMPFEL, ANDREW JONES, NATHANIEL YUN, PER EINARSSON, THERESE LUNDGREN, MARCOS FAJARDO, AND PHILIPPE MARTINEZ. Estimating Surface Reflectance Properties of a Complex Scene under Captured Natural Illumination. Technical Report ICT-TR-06.2004, USC ICT, December 2004.

[DeYoung97] J. DEYOUNG AND A. FOURNIER. Properties of Tabulated Bidirectional Reflectance Distribution Functions. In *Graphics Interface*, pages 47–55, May 1997.

[Dorsey96] JULIE DORSEY AND PATRICK M. HANRAHAN. Modeling and Rendering of Metallic Patinas. In *Proceedings of SIGGRAPH 96*, Computer Graphics Proceedings, Annual Conference Series, pages 387–396, August 1996.

[Dorsey99] JULIE DORSEY, ALAN EDELMAN, HENRIK WANN JENSEN, JUSTIN LEGAKIS, AND HANS KOHLING PEDERSEN. Modeling and Rendering of Weathered Stone. In *SIGGRAPH 1999*, pages 225–234, 1999.

[Dutre04] PHILIP DUTRE, HENRIK WANN JENSEN, JIM ARVO, KAVITA BALA, PHILIPPE BEKAERT, STEVE MARSCHNER, AND MATT PHARR. State of the Art in Monte Carlo Global Illumination. In *Siggraph Course 4*, Los Angeles, August 2004. ACM SIGGRAPH.

[Farell92] T. J. FARELL, M. S. PATTERSON, AND B. WILSON. A diffusion theory model of spatially resolved, steady-state diffuse reflectance for the non-invasive determination of tissue optical properties in vivo. *Med. Phys.*, 19:879–888, 1992.

[Feng93] SHESHAO FENG, FANAN ZENG, AND BRITTON CHANCE. Monte Carlo Simulations of Photon Migration Path Distributions in Multiple Scattering Media. In *Photon Migration and Imaging in Random Media and Tissues, Proc. of SPIE Vol. 1888*, pages 78–89, 1993.

[Fournier00] A. FOURNIER AND P. LALONDE. From Structure to Reflectance. In *Cloth Modeling and Animation*, pages 241–267, Natick, MA, 2000. A K Peters, Ltd.

[Gardner03] A. GARDNER, C. TCHOU, T. HAWKINS, AND P. DEBEVEC. Linear light source reflectometry. *ACM Trans. Graphics.*, 22(3):749–758, 2003.

[Garland97] M. GARLAND AND P. HECKBERT. Surface Simplification Using Quadric Error Metrics. In *Proc. SIGGRAPH*, pages 209–216, August 1997.

[Georghiades03] ATHINODOROS S. GEORGHIADES. Recovering 3-D Shape and Reflectance From a Small Number of Photographs. In *Eurographics Symposium on Rendering: 14th Eurographics Workshop on Rendering*, pages 230–240, June 2003.

[Gibson01] SIMON GIBSON, TOBY HOWARD, AND ROGER HUBBOLD. Flexible Image-Based Photometric Reconstruction using Virtual Light Sources. *Computer Graphics Forum*, 20(3), 2001. ISSN 1067-7055.

[Glassner95] ANDREW GLASSNER. *Principles of Digital Image Synthesis*. Morgan Kaufmann, 1995.

[Goesele04] MICHAEL GOESELE, HENDRIK P. A. LENSCH, JOCHEN LANG, CHRISTIAN FUCHS, AND HANS-PETER SEIDEL. DISCO – Acquisition of Translucent Objects. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2004)*, 23(3):835–844, 2004.

[Goldman04] DAN B GOLDMAN, BRIAN CURLESS, AARON HERTZMANN, AND STEVE SEITZ. Shape and Spatially-Varying BRDFs From Photometric Stereo. Technical report, University of Washington, 2004.

[Gortler96] S. GORTLER, R. GRZESZCZUK, R. SZELISKI, AND M. COHEN. The Lumigraph. *Computer Graphics*, 30(Annual Conference Series):43–54, 1996.

[Greene86] N. GREENE. Environment Mapping and Other Applications of World Projections. *IEEE Computer Graphics & Applications*, 6(11):21–29, nov 1986.

[Han03] JEFFERSON Y. HAN AND KEN PERLIN. Measuring bidirectional texture reflectance with a kaleidoscope. *ACM Trans. Graph.*, 22(3):741–748, 2003.

[Hanrahan89] P. HANRAHAN AND D. SALZMANN. A rapid hierarchical radiosity algorithm for unoccluded environments. In *Eurographics Workshop on Photosimulation, Realism and Physics in Computer Graphics*, 1989.

[Hanrahan93] P. Hanrahan and W. Krueger. Reflection from Layered Surfaces Due to Subsurface Scattering. In *Proceedings of SIGGRAPH 93*, Computer Graphics Proceedings, Annual Conference Series, pages 165–174, August 1993.

[Hauth02] Michael Hauth, Olaf Etzmuss, Bernd Eberhardt, Reinhard Klein, Ralf Sarlette, Mirko Sattler, Katja Daubert, and Jan Kautz. Cloth Animation and Rendering. In *Eurographics 2002 Tutorials*, 2002.

[Hawkins04] Tim Hawkins, Andreas Wenger, Chris Tchou, Andrew Gardner, Fredrik Göransson, and Paul E. Debevec. Animatable Facial Reflectance Fields. In *Proceedings of the 15th Eurographics Workshop on Rendering*, pages 309–321, 2004.

[He91] X. He, K. Torrance, F. Sillion, and D. Greenberg. A Comprehensive Physical Model for Light Reflection. In *Proc. SIGGRAPH*, pages 175–186, July 1991.

[Heckbert92] Paul Heckbert. Discontinuity mesching for radiosity. In *Proceedings of the Third Eurographics Workshop on Rendering* (Bristol, UK), pages 203–226, May 1992.

[Heidrich98] Wolfgang Heidrich. View-Independent Environment Maps. In *Proceedings of Eurographics/SIGGRAPH Workshop on Graphics Hardware '98*, 1998.

[Heidrich99a] W. Heidrich and H.-P. Seidel. Realistic, Hardware-accelerated Shading and Lighting. In *Proceedings SIGGRAPH*, pages 171–178, 1999.

[Heidrich99b] Wolfgang Heidrich. *High-quality Shading and Lighting for Hardware-accelerated Rendering*. PhD thesis, University of Erlangen-Nuremberg, April 1999.

[Heidrich00] W. Heidrich, K. Daubert, J. Kautz, and H.-P. Seidel. Illuminating Micro Geometry Based on Precomputed Visibility. In *Proc. SIGGRAPH*, pages 455–464, July 2000.

[Heikkilä97] Janne Heikkilä and Olli Silvén. A Four-Step Camera Calibration Procedure With Implicit Image Correction. In *CVPR97*, 1997.

[Horn86] B. Horn. *Computer Vision*. MIT Press, Cambridge, Mass., 1986.

[Hunter87] Richard S. Hunter and Richard W. Harold. *The measurement of appearance*. Wiley, 2. ed., 5. print. edition, 1987.

[IEC99] Multimedia Systems and Equipment – Colour Measurement and Management – Part 2-1: Colour Management – Default RGB Colour Space – sRGB. Publication IEC 61966-2-1, 1999.

[Immel86] DAVID S. IMMEL, MICHAEL F. COHEN, AND DONALD P. GREENBERG. A Radiosity Method for Non-Diffuse Environments. In *Computer Graphics (Proceedings of SIGGRAPH 86)*, pages 133–142, August 1986.

[Ishimaru78] AKIRA ISHIMARU. *Wave Propagation and Scattering in Random Media*. Academic Press, 1978.

[ISO99] ISO 14524:1999: Photography – Electronic Still-Picture Cameras – Methods for Measuring Opto-Electronic Conversion Functions (OECFs), 1999.

[Janesick01] JAMES R. JANESICK. *Scientific Charge-Coupled Devices*. SPIE, 2001.

[Jensen96] H.W. JENSEN. Global Illumination using Photon Maps. In *7th Eurographics Workshop on Rendering*, pages 21–30, jun 1996.

[Jensen98] HENRIK WANN JENSEN AND PER H. CHRISTENSEN. Efficient Simulation of Light Transport in Scences with Participating Media using Photon Maps. In *SIGGRAPH 1998*, pages 311–320, 1998.

[Jensen99] HENRIK WANN JENSEN, JUSTIN LEGAKIS, AND JULIE DORSEY. Rendering of Wet Materials. In *Rendering Workshop 1999*, pages 281–290, 1999.

[Jensen01] HENRIK WANN JENSEN, STEPHEN R. MARSCHNER, MARC LEVOY, AND PAT HANRAHAN. A Practical Model for Subsurface Light Transport. In *SIGGRAPH 2001*, pages 511–518, 2001.

[Kajiya85] J. T. KAJIYA. Anisotropic reflection models. *Computer Graphics*, 19(Annual Conference Series):15–21, 1985.

[Kajiya86] JAMES T. KAJIYA. The rendering equation. In *Computer Graphics (SIGGRAPH '86 Proceedings)*, 20(4):143-150, August 1986.

[Kambhatla97] NANDAKISHORE KAMBHATLA AND TODD K. LEEN. Dimension reduction by local principal component analysis. *Neural Comput.*, 9(7):1493–1516, 1997.

[Kang03] SING BING KANG, MATTHEW UYTTENDAELE, SIMON WINDER, AND RICHARD SZELISKI. High Dynamic Range Video. *ACM Transactions on Graphics*, 22(3):319–325, 2003.

[Kautz99] J. KAUTZ AND M. MCCOOL. Interactive Rendering with Arbitrary BRDFs using Separable Approximations. In *Tenth Eurographics Workshop on Rendering*, pages 281–292, June 1999.

[Kautz00a] J. KAUTZ AND M. MCCOOL. Approximation of Glossy Reflection with Prefiltered Environment Maps. In *Proceedings Graphics Interface*, pages 119–126, may 2000.

[Kautz00b] JAN KAUTZ, PERE-PAU VÁZQUEZ, WOLFGANG HEIDRICH, AND HANS-PETER SEIDEL. A Unified Approach to Prefiltered Environment Maps. In *11th Eurographics Workshop on Rendering*, pages 185–196, 2000.

[Kautz02] J. KAUTZ, P.-P. SLOAN, AND J. SNYDER. Arbitrary BRDF Shading for Low-Frequency Lighting Using Spherical Harmonics. In *13th Eurographics Workshop on Rendering*, pages 301–308, June 2002.

[Kawai93] JOHN K. KAWAI, JAMES S. PAINTER, AND MICHAEL F. COHEN. Radioptimization - Goal Based Rendering. In *Proceedings of SIGGRAPH 93*, Computer Graphics Proceedings, Annual Conference Series, pages 147–154, August 1993.

[Klette98] REINHARD KLETTE, KARSTEN SCHLÜNS, AND ANDREAS KOSCHAN. *Computer Vision: Three-Dimensional Data from Images*. Springer, 1998.

[Kobbelt96] L. KOBBELT. Discrete fairing. In *Proc. of the 7th IMA Conf. on the Mathematics of Surfaces*, pages 101–131, 1996.

[Kobbelt00] LEIF P. KOBBELT, STEPHAN BISCHOFF, MARIO BOTSCH, KOLJA KÄHLER, CHRISTIAN RÖSSL, ROBERT SCHNEIDER, AND JENS VORSATZ. Geometric Modeling Based on Polygonal Meshes. Technical Report MPI-I-2000-4-002, Max-Planck-Institut für Informatik, July 2000.

[Koenderink96a] J. KOENDERINK AND A. VAN DOORN. Illuminance texture due to surface mesostructure. *Journal of the Optical Society of America*, 13(3):452–463, 1996.

[Koenderink96b] J. KOENDERINK, A. VAN DOORN, AND M. STAVRIDI. Bidirectional Reflection Distribution Function expressed in terms of surface scattering modes. In *Proc. 4th Europ. Conf. on Computer Vision*, pages 28–39, 1996.

[Koenderink01] J. KOENDERINK AND A. VAN DOORN. Shading in the Case of Translucent Objects. In *Proceedings of SPIE*, volume 4299, pages 312–320, 2001.

[Koudelka03] MELLISA L. KOUDELKA, SEBASTIAN MAGDA, PETER N. BELHUMEUR, AND DAVID J. KRIEGMAN. Acquisition, Compression and Synthesis of Bidirectional Texture Functions. In *Texture 2003*, 2003.

[Krawczyk05] GRZEGORZ KRAWCZYK, MICHAEL GOESELE, AND HANS-PETER SEIDEL. Photometric Calibration of High Dynamic Range Cameras. MPI Informatik Technical Report MPI-I-2005-4-005, 2005.

[Lafortune93] E. LAFORTUNE AND Y. WILLEMS. Bidirectional Path Tracing. In *Computergraphics*, pages 95–104, 1993.

[Lafortune94] ERIC P. LAFORTUNE AND YVES D. WILLEMS. A Theoretical Framework for Physically Based Rendering. *Computer Graphics Forum*, 13(2):97–107, 1994.

[Lafortune96] ERIC P. F. LAFORTUNE AND YVES D. WILLEMS. Rendering Participating Media with Bidirectional Path Tracing. In *Rendering Workshop 1996*, pages 91–100, 1996.

[Lafortune97] E. LAFORTUNE, S. FOO, K. TORRANCE, AND D. GREENBERG. Non-Linear Approximation of Reflectance Functions. In *Proc. SIGGRAPH*, pages 117–126, August 1997.

[Lambert60] J. LAMBERT. *Photometria Sive de Mensura et Gradibus Luminus, Colorum et Umbrae*. Eberhard Klett, 1760.

[Latta02] L. LATTA AND A. KOLB. Homomorphic Factorization of BRDF-based Lighting Computation. In *Proceedings SIGGRAPH*, pages 509–516, July 2002.

[Lengyel01] J. LENGYEL, E. PRAUN, A. FINKELSTEIN, AND H. HOPPE. Real-Time Fur over Arbitrary Surfaces. In *Symposium on Interactive 3D Graphics*, pages 227–232, 2001.

[Lensch00] H. LENSCH, W. HEIDRICH, AND H.-P. SEIDEL. Automated Texture Registration and Stitching for Real World Models. In *Pacific Graphics '00*, pages 317–326, October 2000.

[Lensch01] HENDRIK P. A. LENSCH, JAN KAUTZ, MICHAEL GOESELE, WOLFGANG HEIDRICH, AND HANS-PETER SEIDEL. Image-Based Reconstruction of Spatially Varying Materials. In *Rendering Techniques 2001: 12th Eurographics Workshop on Rendering*, pages 103–114. Eurographics, June 2001. ISBN 3-211-83709-4.

[Lensch02a] HENDRIK P. A. LENSCH, KATJA DAUBERT, AND HANS-PETER SEIDEL. Interactive Semi-Transparent Volumetric Textures. In Guenther Greiner, Heindrich Niemann, Thomas Ertl, Bernd Girod, and Hans-Peter Seidel, editors, *Proceedings of Vision, Modeling and Visualization*, pages 505–512, Erlangen, Germany, 2002. unknown.

[Lensch02b] 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 Sabine Coquillart, Heung-Yeung Shum, and Shi-Min Hu, editors, *Proceedings of Pacific Graphics 2002*, pages 214–224, Beijing, China, 2002. IEEE Computer Society.

[Lensch03] HENDRIK P. A. LENSCH, JAN KAUTZ, MICHAEL GOESELE, WOLFGANG HEIDRICH, AND HANS-PETER SEIDEL. Image-Based Reconstruction of Spatial Appearance and Geometric Detail. *ACM Transactions on Graphics*, 22(2):234–257, April 2003.

[Lewis93] R. LEWIS. Making Shaders More Physically Plausible. In *4th Eurographics Workshop on Rendering*, pages 47–62, June 1993.

[Li02] YUAZHEN LI, STHEFEN LIN, SING BING KANG, HANQUING LU, AND HEUN-YEUNG SHUM. Single-Image Reflectance Estimation for Relighting by Iterative Soft Grouping. In *Pacific Graphics '02*, pages 483–485, October 2002.

[Lin99] S. LIN AND S. LEE. Estimation of Diffuse and Specular Appearance. In *International Conference on Computer Vision*, pages 855–860, 1999.

[Liu04] XINGUO LIU, YAOHUA HU, JINGDAN ZHANG, XIN TONG, BAINING GUO, AND HEUNG-YEUNG SHUM. Synthesis and Rendering of Bidirectional Texture Functions on Arbitrary Surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 10(3):278–289, 2004.

[Lu98] R. LU, J. KOENDERINK, AND A. KAPPERS. Optical Properties (bidirectional reflectance distribution functions) of velvet. *Applied Optics*, 37(25):5974–5984, September 1998.

[Lyon02] RICHARD F. LYON AND PAUL M. HUBEL. Eyeing the Camera: Into the Next Century. In *Proceedings of the 10th Color Imaging Conference on Color Science and Engeneering Systems, Technologies and Applications (CIC-02)*, pages 349–355. IS&T, 2002.

[Malzbender01] TOM MALZBENDER, DAN GELB, AND HANS WOLTERS. Polynomial texture maps. In *Proceedings of SIGGRAPH*, pages 519–528. ACM Press, 2001.

[Marschner97] S. R. MARSCHNER AND D. P. GREENBERG. Inverse Lighting for Photography. In *Proceedings of IS&T/SID Fifth Color Imaging Conference,*, pages 262–265, 1997.

[Marschner98] S. MARSCHNER. *Inverse rendering for computer graphics*. PhD thesis, Cornell University, 1998.

[Marschner99] S. MARSCHNER, S. WESTIN, E. LAFORTUNE, K. TORRANCE, AND D. GREENBERG. Image-based BRDF Measurement Including Human Skin. In *10th Eurographics Workshop on Rendering*, pages 131–144, June 1999.

[Marschner00a] S. MARSCHNER, B. GUENTER, AND S. RAGHUPATHY. Modeling and Rendering for Realistic Facial Animation. *11th Eurographics Workshop on Rendering*, pages 231–242, June 2000. ISBN 3-211-83535-0.

[Marschner00b] S. MARSCHNER, S. WESTIN, E. LAFORTUNE, AND K. TORRANCE. Image-based Measurement of the Bidirectional Reflection Distribution Function. *Applied Optics*, 39(16):2592–2600, 2000.

[Marschner03] STEPHEN R. MARSCHNER, HENRIK WANN JENSEN, MIKE CAMMARANO, STEVE WORLEY, AND PAT HANRAHAN. Light Scattering From Human Hair Fibers. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 22(3):780–791, 2003.

[Masselus03] VINCENT MASSELUS, PIETER PEERS, PHILIP DUTRÉ, AND YVES D. WILLEMS. Relighting With 4D Incident Light Fields. *ACM Transactions on Graphics*, 22(3):613–620, July 2003.

[Matusik02a] WOJCIECH MATUSIK, HANSPETER PFISTER, ADDY NGAN, PAUL BEARDSLEY, REMO ZIEGLER, AND LEONARD MCMILLAN. Image-based 3D photography using opacity hulls. In *Proceedings of SIGGRAPH*, pages 427–437. ACM Press, 2002.

[Matusik02b] WOJCIECH MATUSIK, HANSPETER PFISTER, ADDY NGAN, PAUL BEARDSLEY, REMO ZIEGLER, AND LEONARD MCMILLAN. Image-based 3D photography using opacity hulls. In *Proceedings of ACM SIGGRAPH 2002*, Computer Graphics Proceedings, Annual Conference Series, pages 427–437. ACM Press / ACM SIGGRAPH, July 2002.

[McAllister02] D. MCALLISTER, A. LASTRA, AND W. HEIDRICH. Efficient Rendering of Spatial Bi-directional Reflectance Distribution Functions. *Graphics Hardware 2002*, 2002.

[McCool01] M. MCCOOL, J. ANG, AND A. AHMAD. A Homomorphic Factorization of BRDFs for High-Performance Rendering. In *Proceedings SIGGRAPH*, pages 171–178, August 2001.

[Meseth04] JAN MESETH, GERO MÜLLER, AND REINHARD KLEIN. Reflectance Field based real-time, high-quality Rendering of Bidirectional Texture Functions. *Computers and Graphics*, 28(1):103–112, February 2004.

[Meyer98] A. MEYER AND F. NEYRET. Interactive Volumetric Textures. In *Proc. of Eurographics Workshop on Rendering*, pages 157–168, 1998.

[Miller84] G. MILLER AND R. HOFFMAN. Illumination and Reflection Maps: simulated Objects in simulated and Real Environments. In *SIGGRAPH '84 Course Notes – Advanced Computer Graphics Animation*, July 1984.

[Müller03] GERO MÜLLER, JAN MESETH, AND REINHARD KLEIN. Compression and real-time Rendering of Measured BTFs using local PCA. In *Vision, Modeling and Visualisation 2003*, pages 271–280, November 2003.

[Müller04] GERO MÜLLER, JAN MESETH, AND REINHARD KLEIN. Fast Environmental Lighting for Local-PCA Encoded BTFs. In *Computer Graphics International 2004 (CGI2004)*, June 2004.

[Nayar90] S. NAYAR, K. IKEUCHI, AND T. KANADE. Determining Shape and Reflectance of Hybrid Surfaces by Photometric Sampling. *RA*, 6(4):418–431, 1990.

[Ng03] REN NG, RAVI RAMAMOORTHI, AND PAT HANRAHAN. All-Frequency Shadows using Non-Linear Wavelet Lighting Approximation. *ACM Transactions on Graphics*, 22(3):376–381, 2003.

[Nicodemus77] FRED E. NICODEMUS, JOSEPH C. RICHMOND, JACK J. HSIA, I. W. GINSBERG, AND T. LIMPERIS. Geometrical Considerations and Nomenclature for Reflectance. National Bureau of Standards, 1977.

[Nishino99] K. NISHINO, YOICHI SATO, AND KATSUSHI IKEUCHI. Eigen-texture method: appearance compression based on 3D model. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'99)*, volume 1, pages 618 – 624, June 1999.

[Nishino01] K. NISHINO, Z. ZHANG, AND K. IKEUCHI. "Determining Reflectance Parameters and Illumination Distribution from a Sparse Set of Images for View-dependent Image Synthesis". In *in Proc. of Eighth IEEE International Conference on Computer Vision ICCV '01*, pages 599–606, july 2001.

[Peers03] PIETER PEERS AND PHILIP DUTRÉ. Wavelet Environment Matting. In *Proceedings of the 14th Eurographics Symposium on Rendering*, pages 157–166, 2003.

[Perona90] P. PERONA AND J. MALIK. Scale Space and Edge Detection using Anisotropic Diffusion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(7):629–639, July 1990.

[Pharr00] MATT PHARR AND PAT HANRAHAN. Monte Carlo Evaluation of Non-linear Scattering Equations for Subsurface Reflection. In *SIGGRAPH 2000*, pages 75–84, 2000.

[Phong75] B.-T. PHONG. Illumination for Computer Generated Pictures. *Communications of the ACM*, 18(6):311–317, June 1975.

[Poulin90] P. POULIN AND A. FOURNIER. A Model for Anisotropic Reflection. In *Proceedings SIGGRAPH*, pages 273–282, July 1990.

[Press92] W. PRESS, S. TEUKOLSKY, W. VETTERLING, AND B. FLANNERY. *Numerical recipes in C - The art of scientific computation*. ISBN 0-521-43108-5. Cambridge University Press, 2nd edition, 1992.

[Radloff04] JUDITH RADLOFF. Obtaining the Bidirectional Texture Reflectance of Real-World Surfaces by means of a Kaleidoscope. Technical report, Department of Computer Science, Rhodes University, 2004.

[Ramamoorthi01a] R. RAMAMOORTHI AND P. HANRAHAN. An Efficient Representation for Irradiance Environment Maps. In *Proceedings SIGGRAPH*, pages 497–500, August 2001.

[Ramamoorthi01b] RAVI RAMAMOORTHI AND PAT HANRAHAN. A Signal-Processing Framework for Inverse Rendering. In Eugene Fiume, editor, *Proceedings of SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 117–128. ACM Press / ACM SIGGRAPH, August 2001. ISBN 1-58113-292-1.

[Ramamoorthi02a] R. RAMAMOORTHI AND P. HANRAHAN. Frequency Space environment Map Rendering. In *Proceedings SIGGRAPH*, pages 517–526, July 2002.

[Ramamoorthi02b] RAVI RAMAMOORTHI. Analytic PCA construction for theoretical analysis of lighting variability in images of a Lambertian object. *PAMI Oct 2002*, 2002.

[Reflectance] COLUMBIA-UTRECHT REFLECTANCE AND TEXTURE DATABASE. http://www1.cs.columbia.edu/CAVE/curet/.

[Robertson99] MARK A. ROBERTSON, SEAN BORMAN, AND ROBERT L. STEVENSON. Dynamic Range Improvement Through Multiple Exposures. In *Proceedings of ICIP 1999*, 1999.

[Robertson03] MARK A. ROBERTSON, SEAN BORMAN, AND ROBERT L. STEVENSON. Estimation-Theoretic Approach to Dynamic Range Enhancement using Multiple Exposures. *Journal of Electronic Imaging*, 12(2):219–285, April 2003.

[Rushmeier90] HOLLY E. RUSHMEIER AND KENNETH E. TORRANCE. Extending the Radiosity Method to Include Specularly Reflecting and Translucent Materials. *ACM Trans. Graph.*, 9(1):1–27, 1990.

[Rusinkiewicz98] S. RUSINKIEWICZ. A New Change of Variables for Efficient BRDF Representation. In *Rendering Techniques '98*, pages 11–22, 1998.

[Rusinkiewicz00] S. RUSINKIEWICZ AND S. MARSCHNER. Measurement I - BRDFs. Script of Course CS448C: Topics in Computer Graphics, held at Stanford University, October 2000.

[Sato97] Y. SATO, M. WHEELER, AND K. IKEUCHI. Object Shape and Reflectance Modeling from Observation. In *Proc. SIGGRAPH*, pages 379–388, August 1997.

[Sattler03] M. SATTLER, R. SARLETTE, AND R. KLEIN. Efficient and Realistic Visualization of Cloth. *Proceedings of the Eurographics Symposium on Rendering 2003*, 2003.

[Schlick94] CHRISTOPHE SCHLICK. An Inexpensive BRDF Model for Physically-based Rendering. *Computer Graphics Forum*, 13(3):233–246, 1994.

[Schneider04] MARTIN SCHNEIDER. Real-Time BTF Rendering. In *Proceedings of CESCG 2004*, 2004.

[Schoeneman93] CHRIS SCHOENEMAN, JULIE DORSEY, BRIAN SMITS, JAMES ARVO, AND DONALD GREENBERG. Painting With Light. In *Proceedings of SIGGRAPH 93*, Computer Graphics Proceedings, Annual Conference Series, pages 143–146, August 1993.

[Sen05] PRADEEP SEN, BILLY CHEN, GAURAV GARG, STEVE MARSCHNER, MARK HOROWITZ, MARC LEVOY, AND HENDRIK P. A. LENSCH. Dual Photography. *ACM Transactions on Graphics*, July 2005. (accepted for publication).

[Shirley95] P. SHIRLEY, B. WADE, P. HUBBARD, D. ZARESKI, B. WALTER, AND D. GREENBERG. Global Illumination via Density Estimation. In *6th Eurographics Workshop on Rendering*, pages 219–231, 1995.

[Sillion91] F. SILLION, J. ARVO, S. WESTIN, AND D. GREENBERG. A Global Illumination Solution for General Reflectance Distributions. In *Proceedings SIGGRAPH*, pages 187–196, July 1991.

[Sillion94] FRANCOIS X. SILLION AND CLAUDE PUECH. *Radiosity and Global Illumination*. Morgan Kaufmann Publishers, San Francisco, 1994.

[Sillion95] FRANCOIS X. SILLION. A Unified Hierarchical Algorithm for Global Illumination with Scattering Volumes and Object Clusters. *IEEE Trans. Visualization and Computer Graphics*, 1(3):240–254, 1995.

[Sloan02] PETER-PIKE SLOAN, JAN KAUTZ, AND JOHN SNYDER. Precomputed Radiance Transfer for Real-Time Rendering in Dynamic, Low-Frequency Lighting Environments. In *Proceedings of SIGGRAPH*, pages 527–536. ACM Press, 2002.

[Sloan03a] PETER-PIKE SLOAN, JESSE HALL, JOHN HART, AND JOHN SNYDER. Clustered Principal Components for Precomputed Radiance Transfer. *ACM Transactions on Graphics*, 22(3):382–391, 2003.

[Sloan03b] PETER-PIKE SLOAN, XINGUO LIU, HEUNG-YEUNG SHUM, AND JOHN SNYDER. Bi-Scale Radiance Transfer. *ACM Transactions on Graphics*, 22(3):370–375, 2003.

[Stam95] JOS STAM. Multiple Scattering as a Diffusion Process. In *Rendering Workshop 1995*, pages 51–58, 1995.

[Stam01] JOS STAM. An Illumination Model for a Skin Layer Bounded by Rough Surfaces. In *Rendering Workshop 2001*, pages 39–52, 2001.

[Suykens03] FRANK SUYKENS, KARL VOM BERGE, ARES LAGAE, AND PHILIP DUTRÉ. Interactive Rendering of Bidirectional Texture Functions. In *Eurographics 2003*, pages 463–472, September 2003.

[Tagare91] H. TAGARE AND R. DE FIGUEIREDO. A Theory of photometric stereo for a class of diffuse non-lambertion surfaces. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 13(2):133–152, 1991.

[Tong02] XIN TONG, JINGDAN ZHANG, LIGANG LIU, XI WANG, BAINING GUO, AND HEUNG-YEUNG SHUM. Synthesis of bidirectional texture functions on arbitrary surfaces. In *Proceedings of SIGGRAPH*, pages 665–672. ACM Press, 2002.

[Torrance67] K. TORRANCE AND E. SPARROW. Theory for Off-Specular Reflection from Roughened Surfaces. *Journal of Optical Society of America*, 57(9), 1967.

[Tsai87] ROGER Y. TSAI. A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses. *RA*, 3(4):323–344, 1987.

[v. Helmholtz25] H. V. HELMHOLTZ. *Treatise on Physiological Optics*. Dover, New York, 1925.

[Vasilescu04] M. A. O. VASILESCU AND DEMETRI TERZOPOULOS. TensorTextures: Multilinear Image-Based Rendering. In *Proceedings of SIGGRAPH*, August 2004.

[Veach97] E. VEACH. *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Stanford University, 1997.

[Ward Larson92] G. WARD LARSON. Measuring and Modeling Anisotropic Reflection. In *Proc. SIGGRAPH*, pages 265–272, July 1992.

[Wexler02] YONATAN WEXLER, ANDREW W. FITZGIBBON, AND ANDREW ZISSERMAN. Image-Based Environment Matting. In *Proceedings of the 13th Eurographics workshop on Rendering*, pages 279–290, 2002.

[Williams99] TOM L. WILLIAMS. *The Optical Transfer Function of Imaging Systems*. Institute of Physics Publishing, London, 1999.

[Wolberg90] G. WOLBERG. *Digital Image Warping*. IEEE Computer Society Press, 1990.

[Wong97] TIEN-TSIN WONG, PHENG-ANN HENG, SIU-HANG OR, AND WAI-YIN NG. Image-based Rendering with Controllable Illumination. In *Proceedings of EGRW '97*, pages 13–22. Springer-Verlag, 1997.

[Woodham81] R. WOODHAM. Analysing Smages of Curved Surfaces. *Artificial Intelligence*, 17:117–140, 1981.

[Yu98] Y. YU AND J. MALIK. Recovering Photometric Properties of Architectural Scenes from Photographs. In *Proc. SIGGRAPH*, pages 207–218, July 1998.

[Yu99] Y. YU, P. DEBEVEC, J. MALIK, AND T. HAWKINS. Inverse Global Illumination: Recovering Reflectance Models of Real Scenes From Photographs. In *Proc. SIGGRAPH*, pages 215–224, August 1999.

[Zhang98] Z. ZHANG. A Flexible New Technique for Camera Calibration. Tr98-71, MSR Redmond, 1998.

[Zhang99] Z. ZHANG. Flexible Camera Calibration By Viewing a Plane From Unknown Orientations. In *Int. Conf. on Computer Vision*, pages 666–673, September 1999.

[Zhang00] ZHENGYOU ZHANG. A Flexible New Technique for Camera Calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[Zhu04] JIAYUAN ZHU AND YEE-HONG YANG. Frequency-Based Environment Matting. In *Proceedings of 12th Pacific conference on Computer Graphics and Applications*, pages 402–410, 2004.

[Zongker99] DOUGLAS E. ZONGKER, DAWN M. WERNER, BRIAN CURLESS, AND DAVID H. SALESIN. Environment Matting and Compositing. In *Proceedings of ACM SIGGRAPH 1999*, pages 205–214, 1999.

Part II

Annotated Bibliography

Chapter 13

Introduction of the Annotated Bibliography

The goal of this annotated bibliography is to provide an overview over the most important publications in the areas covered by the course. Our goal was especially to help newcomers to the field to quickly become familiar with the main papers and serve as a starting point for further literature study. This is naturally always a subjective choice and we claim therefore by no means that the list of selected papers is complete and apologize for any important papers we missed.

General References

[1] Richard S. Hunter and Richard W. Harold. *The Measurement of Appearance*. Wiley, 2. ed., 5. print. edition, 1987.

In this book, the various effects of reflections off surfaces are carefully described and analized. The authors provide valuable and intuitive insights on how to distinguish the appearance of two different materials. The book furthermore illustrates how the appearance of real world surfaces can be measured giving examples of techniques commonly applied in print industry. The main focus is on measuring the appearance of planar surfaces.

[2] Fred E. Nicodemus, Joseph C. Richmond, Jack J. Hsia, I. W. Ginsberg, and T. Limperis. Geometrical Considerations and Nomenclature for Reflectance. National Bureau of Standards, 1977.

This report introduces the basic concepts of BSSRDFs, BRDFs, and related functions to describe reflectance. It also defines the nomenclature for all of them and describes their relationships such as the derivation of the BRDF from the BSSRDF.

BRDFs

[1] James F. Blinn. Models of Light Reflection for Computer Synthesized Pictures. In *SIGGRAPH '77: Proceedings of the 4th annual conference on Computer graphics and interactive techniques*, pages 192–198. ACM Press, 1977.

This paper introduces the empirical Blinn-Phong model (based on the earlier Phong model [6]. It can model more realistic reflections using three parameters (diffuse and specular coefficient, specular exponent). The specular lobe is computed based on the halfway vector.

[2] R. Cook and K. Torrance. A reflection model for computer graphics. *ACM Transactions On Graphics*, 1(1):7–24, 1982.

The Cook-Torrance model is a modification of earlier reflectance models. The main assumption is that the surface is composed of tiny, perfectly reflective, smooth microfacets oriented at different directions. The facets are assumed to be V-shaped and their distribution is isotropic. The model takes into account the fact that the light might be blocked by other microfacets (shadowing). Similarly, it also considers the fact that the viewer does not see some of the microfacets since they are blocked by the other microfacets (masking effect). The model takes into account an average Fresnel term (polarization is not considered) when modelling the reflectance of individual microfacets. However, it does not allow for multiple light bounces between the microfacets. The orientation of the facets is assumed to have some distribution - Cook and Torrance use the Beckman distribution function.

[3] X. He, K. Torrance, F. Sillon, and D. Greenberg. A comprehensive physical model for light reflection. *Computer Graphics*, 25(Annual Conference Series):175–186, 1991.

This paper presents a reflectance model that accounts for the phenomena that can be explained using both geometrical optics and wave optics (diffraction, interference). The model supports arbitrary polarization of incident light, but the simplifications for unpolarized light are also presented. In general, the reflectance is modelled as a sum of three components: specular, directional diffuse, and uniform diffuse. The specular component accounts for mirror-like reflection. It depends on the Fresnel reflectivity, roughness, and shadowing factors. The directional diffuse contribution of the reflectance function is the most complex term. It accounts for diffraction and interference effects. It depends on surface statistics (the effective roughness and the autocorrelation length). The uniform-diffuse contribution is a result of multiple microfacet reflections and subsurface reflections. It is expressed as a simple function of wavelength. The resulting isotropic reflectance model for unpolarized light is a function of four parameters. Each of the parameters has some physical meaning and (at least theoretically) can be measured separately.

[4] Eric P. F. Lafortune, Sing-Choong Foo, Kenneth E. Torrance, and Donald P. Greenberg. Non-linear Approximation of Reflectance Functions. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 117–126. ACM Press/Addison-Wesley Publishing Co., 1997.

The Lafortune model presented in this paper is an extension of the Phong model [6] with a diffuse term and multiple lobes. Each lobe consists of a weighted dot product between viewing and lighting direction raised to some power. This empirical model can handle off-specular peaks, backscattering and anisotropy and is frequently used to model the reflection properties of real, measured materials.

[5] S. Marschner, S. Westin, E. Lafortune, and K. Torrance. Image-based measurement of the Bidirectional Reflection Distribution Function. *Applied Optics*, 39(16):2592–2600, 2000.

Marschner et al. describe an image-based BRDF measurement system. They use a material sample with different surface normals. Each point with a different surface normal gives a different BRDF measurement. Their system uses a spherical sample of homogenous material. A fixed camera takes images of the sample under illumination from an orbiting light source. The system, although limited to only isotropic BRDF measurements, is both fast and robust. Furthermore, they extend their method to surface geometry acquired with a laser range scanner to acquire reflectance of a human face.

[6] Bui Tuong Phong. Illumination for Computer Generated Pictures. *Commun. ACM*, 18(6):311–317, 1975.

This paper introduces the Phong model – one of the earliest empirical lighting models for computer graphics. The model consists of a diffuse term and one specular lobe. It is neither energy conserving nor reciprocal and is only well-suited to approximate plastic-like materials. Improvements and extensions of the model include [1, 4].

[7] G. Ward. Measuring and modeling anisotropic reflection. *Computer Graphics*, 26(Annual Conference Series):265–273, 1992.

This paper presents one of the first methods to speed up the BRDF measurement process. Ward's measurement device (imaging gonio-reflectometer) consists of a hemispherical mirror and a CCD camera with a fisheye lens. The main advantage of his system is that the CCD camera can take multiple, simultaneous BRDF measurements. Each photosite of the imaging sensor contains a separate BRDF value. Moving the light source and material over all incident angles enables the measurement of arbitrary BRDFs. Ward also presents a BRDF model that is based on the elliptical Gaussian distribution. The model is carefully designed to be physically plausible - it supports energy conservation and

reciprocity. It is also relatively simple and can be evaluated efficiently. The parameters of the model have physical meaning and theoretically can be measured independently.

Spatially Varying BRDFs

[1] Samuel Boivin and André Gagalowicz. Image-based rendering of diffuse, specular and glossy surfaces from a single image. In Eugene Fiume, editor, *Proceedings of SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 107–116. ACM Press / ACM SIGGRAPH, August 2001. ISBN 1-58113-292-1.

This paper tries to solve the difficult problem of measuring BRDF in indoor scenes from a single observation. The hope is that the global illumination and grouping of measurements of multiple surface points provide sufficient constraints to estimate a per-patch BRDF. At first a simple diffuse BRDF model is assumed. If the observed error is still insufficient a specular lobe is added. In case of failure, further tests involve anisotropic or mirroring BRDFs.

[2] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar. Acquiring the Reflectance Field of a Human Face. In *Proc. SIGGRAPH*, pages 145–156, July 2000. ISBN 1-58113-208-5.

While this paper actually introduced the concept of reflectance fields it also contains a section where a BRDF model is fit to the measured data of each texel. The spatially varying BRDF yields some compression compared to the full reflectance field data set.

[3] Paul Debevec, Chris Tchou, Andrew Gardner, Tim Hawkins, Charis Poullis, Jessi Stumpfel, Andrew Jones, Nathaniel Yun, Per Einarsson, Therese Lundgren, Marcos Fajardo, and Philippe Martinez. Estimating Surface Reflectance Properties of a Complex Scene under Captured Natural Illumination. Technical Report ICT-TR-06.2004, USC ICT, December 2004.

This reports combines the idea of clustered BRDFs with global inverse illumination. For a number of representative spots/materials the BRDF is captured using standard image-based BRDF measurement techniques under controlled illumination conditions. In order to capture the spatially varying BRDF of a building the incident light onto ths building is captured by an environment map which serves as a illumination source in a global illumination framework. Based on the differences between the synthesized images and the captured HDR images the weight for combining the cluster BRDFs are updated for each texel individually.

[4] A. Gardner, C. Tchou, T. Hawkins, and P. Debevec. Linear light source reflectometry. *ACM Trans. Graphics.*, 22(3):749–758, 2003.

In this paper the fully spatially varying BRDF and a transmission term is estimated for rather flat documents. The illumination is provided by a linear light source which has to be considered during the BRDF estimation. The same data is also used to scan the 3D geometry of the surface.

[5] Athinodoros S. Georghiades. Recovering 3-d shape and reflectance from a small number of photographs. In *Eurographics Symposium on Rendering: 14th Eurographics Workshop on Rendering*, pages 230–240, June 2003.

Georghiades addresses the problem of estimating shape and reflection properties at the same time. Given a set of images of the scene illuminated by a point light source of unknown position the approach sets up an optimization problem that solves for the diffuse component of the BRDF and the actual surface normal per pixel as well as a global specular component and the light source positions in the individual images. As in other shape-from-shading approaches assuming a continuous surface introduces a regularization term that allows for solving the large optimization problem.

[6] Hendrik P. A. Lensch, Jan Kautz, Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. Image-based reconstruction of spatially varying materials. In *Rendering Techniques 2001: 12th Eurographics Workshop on Rendering*, pages 103–114. Eurographics, June 2001. ISBN 3-211-83709-4.

This paper introduces the concept of capturing cluster BRDFs and expressing the spatially variation by per-texel weighted sums of cluster BRDFs. Making use if the idea of image-based BRDF measurments samples from multiple surface points are combined when determining the cluster BRDFs. This results in more reliable, that is, more plausible BRDF parameters and at the same time reduces the number of required input images. Drastically different materials distributed in the same patch can be reproduced faithfully.

[7] Hendrik P. A. Lensch, Jan Kautz, Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. Image-Based Reconstruction of Spatial Appearance and Geometric Detail. *ACM Transactions on Graphics*, 22(2):234–257, April 2003.

This paper extends the previous work towards estimating per-texel normals. Starting from a scanned and smoothed 3D geometry model the per-texel BRDF is estimated. In a photometric stereo approach the current estimate of the BRDF is used to update the surface normal.

[8] S. Marschner, S. Westin, E. Lafortune, K. Torrance, and D. Greenberg. Image-based BRDF Measurement Including Human Skin. In *10th Eurographics Workshop on Rendering*, pages 131–144, June 1999.

This paper applied the idea of image-based BRDF measurement to objects of arbitrary geometry. A 3D scan of the object provides the geometric information. Multiple images illuminated by a flash light are combined in order to estimate a single BRDF for the object.

[9] D. McAllister, A. Lastra, and W. Heidrich. Efficient rendering of spatial bi-directional reflectance distribution functions. *Graphics Hardware 2002*, 2002.

The authors present the first real-time rendering framework for BTFs. They used the Lafortune model to approximate the spatially varying BRDFs which leads to an extreme compact representation amendable to hardware implementation.

[10] K. Nishino, Z. Zhang, and K. Ikeuchi. "determining reflectance parameters and illumination distribution from a sparse set of images for view-dependent image synthesis". In *in Proc. of Eighth IEEE International Conference on Computer Vision ICCV '01*, pages 599–606, july 2001.

Nishino et al. address the complicated problem of reconstructing BRDF and incident illumination at the same time. Specular highlights observed in the individual images are projected into a global environment map to estimate incident illumination. In the next step the BRDF is estimated. Spatial variation is restricted to the diffuse component.

[11] Ravi Ramamoorthi and Pat Hanrahan. A signal-processing framework for inverse rendering. In Eugene Fiume, editor, *Proceedings of SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 117–128. ACM Press / ACM SIGGRAPH, August 2001. ISBN 1-58113-292-1.

This paper as well presents a solution to the problem of estimating BRDF and illumination from the same set of images for which an iterative algorithm has been developed. The paper is mostly well-known for the use of spherical harmonics to represent environment maps as well as BRDFs. This representation allows for computing the convolution of the BRDF with the environment map by a simple dot product.

[12] Y. Sato, M. Wheeler, and K. Ikeuchi. Object Shape and Reflectance Modeling from Observation. In *Proc. SIGGRAPH*, pages 379–388, August 1997.

In this paper shape and reflectance properties are captured using the same sensor but different illumination. There is no explicit registration step necessary to match 3D geometry and 2D images. The diffuse component of the BRDF is estimated per pixel while the specular component is constant per patch.

[13] Y. Yu, P. Debevec, J. Malik, and T. Hawkins. Inverse Global Illumination: Recovering Reflectance Models of Real Scenes From Photographs. In *Proc. SIGGRAPH*, pages 215–224, August 1999.

This paper considers indoor scenes. A few input images are aligned with a geometry model of the rooom and the furniture. Given the positions of the light sources, the light transport in the room can be simulated incorporating global illumination effects. The estimated BRDF minimizes the error between the measured values and a global illumination forward solution.

[14] Y. Yu and J. Malik. Recovering Photometric Properties of Architectural Scenes from Photographs. In *Proc. SIGGRAPH*, pages 207–218, July 1998.

The BRDFs of buildings in outdoor scenes are estimated considering the incident illumination from the sun and the sky. Only the diffuse component is allowed to vary freely across the surface.

Translucent Materials, BSSRDFs

[1] Yanyun Chen, Xin Tong, Jiaping Wang, Stephen Lin, Baining Guo, and Heung-Yeung Shum. Shell texture functions. *ACM Transactions on Graphics*, 23(3):343–353, August 2004.

This paper presents an appearance representation approach that is particularly suited for heterogeneous translucent objects. The translucent object is divided into a homogeneous diffusely scattering core surrounded by volume of heterogeneous translucent material. The shell texture function (STF) provides an intermediate data structure representing the light transport and the mesostructure of the outer shell. For each voxel in the shell volume the irradiance due to light impinging from arbitrary directions is precomputed and stored in a 5D data structure.

[2] Michael Goesele, Hendrik P. A. Lensch, Jochen Lang, Christian Fuchs, and Hans-Peter Seidel. DISCO – Acquisition of Translucent Objects. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2004)*, 23(3), 2004.

This is the first paper that captured the diffuse reflectance R_d of a real translucent object with inhomogeneous material properties. The object is pointwise illuminated and its impulse response function is captured with a HDR camera. A hierarchical model of transfer functions is computed from a large number of input images. Rendering can be performed in real time using an earlier approach.

[3] Akira Ishimaru. *Wave Propagation and Scattering in Random Media*. Academic Press, 1978.

This book describes the physical principles of single and multiple scattering in various types of media and derives the mathematical formulations.

[4] Henrik Wann Jensen and Juan Buhler. A Rapid Hierarchical Rendering Technique for Translucent Materials. In *SIGGRAPH 2002*, pages 576–581, 2002.

The authors propose a hierarchical evaluation technique to speed up the rendering of translucent objects using the dipole model [5]. This is the first of a whole series of papers proposing various rendering techniques to speed up evaluation of the dipole model – see e.g. [2] for a list of such publications.

[5] Henrik Wann Jensen, Stephen R. Marschner, Marc Levoy, and Pat Hanrahan. A Practical Model for Subsurface Light Transport. In *SIGGRAPH 2001*, pages 511–518, 2001.

This paper introduces the dipole model as an approximation for translucent objects in computer graphics. It describes the derivation of the model, it's use for rendering, and compares the results to Monte-Carlo simulations. The authors describe also a measurement setup to determine the required parameters for real materials and provide a table of measured values. The dipole model is used in many publications including [4] as a fast method to evaluate the effects of subsurface scattering.

Transparent and Specular Materials, Environment Matting

[1] Sameer Agarwal, Satya P. Mallick, David J. Kriegman, and Serje J. Belongie. On refractive optical flow. In *Proceedings of 8th European Conference on Computer Vision*, pages 483–494, 2004.

This paper presents a generalization of the optical flow equation to the case of refraction. The resulting refractive optical flow equation can be used to estimate environment mattes without knowing the background. The objects that can be captured are limited to be specularly refractive.

[2] Yung-Yu Chuang, Douglas E. Zongker, Joel Hindorff, Brian Curless, David H. Salesin, and Richard Szeliski. Environment matting extensions: Towards higher accuracy and real-time capture. In *Proceedings of ACM SIGGRAPH 2000*, pages 121–130, 2000.

This paper presents two extensions to the hierarchical approach used by Zongker *et al.* The first one allows for the capture of more complex and subtle interactions of light with objects assuming Gaussian weighting functions, while the second allows for real-time capture of colorless objects in motion.

[3] Pieter Peers and Philip Dutré. Wavelet environment matting. In *Proceedings of the 14th Eurographics Symposium on Rendering*, pages 157–166, 2003.

This paper introduces wavelet environment matting which can extract the most accurate environment mattes to date. A set of wavelet patterns is used as the stimulus function and a feedback loop is used to select the most important patterns on the fly. It however requires a long capturing time (12 hours) and a large amount of storage (2.5GB on average).

[4] Yonatan Wexler, Andrew W. Fitzgibbon, and Andrew Zisserman. Image-based environment matting. In *Proceedings of the 13th Eurographics workshop on Rendering*, pages 279–290, 2002.

This paper introduces the first passive approach for environment matting which can extract environment mattes from natural backgrounds. It however requires that it is possible to reconstruct the clean plates and is only effective for specularly refractive objects.

[5] Jiayuan Zhu and Yee-Hong Yang. Frequency-based environment matting. In *Proceedings of 12th Pacific conference on Computer Graphics and Applications*, pages 402–410, 2004.

This paper introduces an environment matting method with a stimulus function in the temporal domain rather than the spatial domain. Environment mattes can then be extracted using frequency analysis. It is, in principle, capable of representing arbitrary weighting functions. However, in this paper, for reducing the number of images, only the separable weighting functions are modeled.

[6] Douglas E. Zongker, Dawn M. Werner, Brian Curless, and David H. Salesin. Environment matting and compositing. In *Proceedings of ACM SIGGRAPH 1999*, pages 205–214, 1999.

This is the first paper on environment matting and compositing. It assumes a rectangular weighting function and uses hierarchical horizontal/vertical stripes as the stimulus backdrop function. It is effective for refraction, translucency, highly specular and color transparency.

Fibers

[1] David C. Banks. Illumination in diverse codimensions. In *Proceedings of SIGGRAPH 94*, July 1994.

> This paper contains a nice mathematical framing of the geometry of reflection from a k-dimensional subspace of n-dimensional space. Although Banks doesn't explicitly discuss radiometry, this paper is valuable background reading for understanding the radiometric issues of reflection from fibers in 3-space.

[2] James T. Kajiya and Timothy L. Kay. Rendering fur with three dimensional textures. In *Proceedings of SIGGRAPH 1989*, pages 271–280, 1989.

> Although the main focus of this paper was to introduce a method for simulating materials like fur efficiently using volume rendering, Kajiya and Kay also introduce a simple phenomenological model for scattering from a fiber. This model captures the basic phenomenon of reflection into the specular cone, and it continues to serve well after over 15 years.

[3] Stephen R. Marschner, Henrik Wann Jensen, Mike Cammarano, Steve Worley, and Pat Hanrahan. Light scattering from human hair fibers. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 22(3):780–791, 2003.

> This paper makes the first serious effort at a realistic scattering model for hair, based on the characteristics of real hair. Drawing from studies of reflection from other cylindrical structures, the authors propose an analytical model for reflection from hair that includes surface reflection, transmission, and internal reflection, and accounts for the asymmetric microgeometry of the surface. The key predictions are a double specular highlight—one colored, and one white—and a strong azimuthal variation in the internally reflected light that contributes to the visual texture of hair.

BTFs

[1] Kristin J. Dana, Bram van Ginneken, Shree K. Nayar, and Jan J. Koenderink. Reflectance and texture of real-world surfaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 151–157, 1997.

This paper introduced Bidirectional Texture Functions to the computer graphics community. The authors present the CUReT reflectance and texture database which made BTF and BRDF measurements publicly available for the first time. The sampling density of the BTFs (205 images per material) was not yet sufficient for high-quality rendering.

[2] D. McAllister, A. Lastra, and W. Heidrich. Efficient rendering of spatial bi-directional reflectance distribution functions. *Graphics Hardware 2002*, 2002.

The authors present the first real-time rendering framework for BTFs. They used the Lafortune model to approximate the spatially varying BRDFs which leads to an extreme compact representation amendable to hardware implementation.

[3] M. Sattler, R. Sarlette, and R. Klein. Efficient and realistic visualization of cloth. *Proceedings of the Eurographics Symposium on Rendering 2003*, 2003.

In this paper the first BTF real-time rendering framework based on statistical data analysis is presented. It describes the whole pipeline from measurement using fully automatic setup over compression to rendering including image-based illumination and large scale shadows.

[4] Xin Tong, Jingdan Zhang, Ligang Liu, Xi Wang, Baining Guo, and Heung-Yeung Shum. Synthesis of bidirectional texture functions on arbitrary surfaces. In *Proceedings of SIGGRAPH*, pages 665–672. ACM Press, 2002.

The authors present a fast synthesis algorithm for BTFs on arbitrary surfaces. It is based on a quantization of the BTF data leading to so called *textons* and a novel search strategy called *k-coherance search*.

Reflectance Fields

[1] Jefferson Y. Han and Ken Perlin. Measuring bidirectional texture reflectance with a kaleidoscope. *ACM Trans. Graph.*, 22(3):741–748, 2003.

This paper demonstrated the acquisition of images from many viewpoints simultaneously using a kaleidoscope. By using a video projector and a beamsplitter, the reversibility of light transport allows arbitrary directions of illumination to also be provided.

[2] Tim Hawkins, Andreas Wenger, Chris Tchou, Andrew Gardner, Fredrik Göransson, and Paul E. Debevec. Animatable facial reflectance fields. In *Proceedings of the 15th Eurographics Workshop on Rendering*, pages 309–321, 2004.

This paper demonstrated a system for rapidly capturing directional illumination on a human subject. The system was used to capture a large number of facial expressions, producing a relightable morphable model. Multiple cameras and a coarse geometric model provide for the viewpoint to also be changed.

[3] Eric P. Lafortune and Yves D. Willems. A theoretical framework for physically based rendering. *Computer Graphics Forum*, 13(2):97–107, June 1994.

This paper introduced the Global Reflectance Distribution Functions, a 10-dimensional function which captures the overall light transport in a scene.

[4] Vincent Masselus, Pieter Peers, Philip Dutré, and Yves D. Willems. Relighting with 4d incident light fields. *ACM Transactions on Graphics*, 22(3):613–620, July 2003.

This paper captured images of an object illuminated by a video projector. By moving the video projector to many different positions, and by capturing many different projected patterns, they construct a 6-dimensional subset of the object's reflectance field which allows the object to be relit by any desired incident light field.

[5] Wojciech Matusik, Hanspeter Pfister, Addy Ngan, Paul Beardsley, Remo Ziegler, and Leonard McMillan. Image-based 3d photography using opacity hulls. In *Proceedings of ACM SIGGRAPH 2002*, Computer Graphics Proceedings, Annual Conference Series, pages 427–437. ACM Press / ACM SIGGRAPH, July 2002.

This paper captures images of an object from many directions of view and under many directional illumination conditions. They map this 6-dimensional reflectance field data

onto a refined version of a visual hull known as an opacity hull. They can then produce realistic renderings from any desired viewpoint and under any directional illumination condition of objects with very intricate geometric features.

Part III

Presenters' Slides

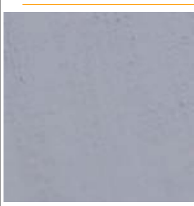
SIGGRAPH2005

Course 10
Realistic Materials in Computer Graphics

Introduction

Hendrik P.A. Lensch
Stanford University

Material Samples

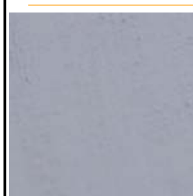


diffuse

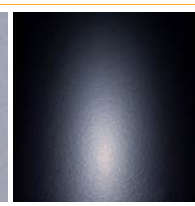
glossy

mirror

Material Samples

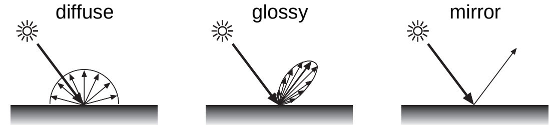


diffuse



glossy

mirror



Material Samples

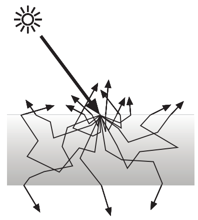
anisotropic

Material Samples

translucent

Material Samples

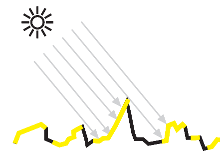
translucent



Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

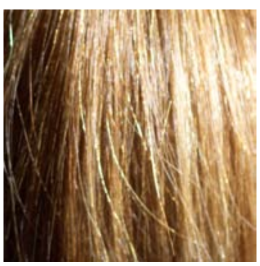
Material Samples

complex surface structure



Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

Material Samples



fibers

Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

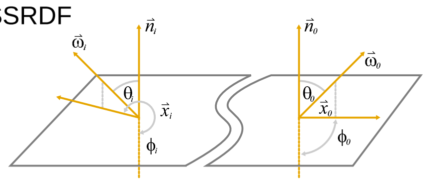
How to describe materials?

- mechanical, chemical, electric properties
- reflection properties
- surface roughness
- geometry/meso-structure
- **relightable** representation of appearance

Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

Representing Appearance

- BSSRDF

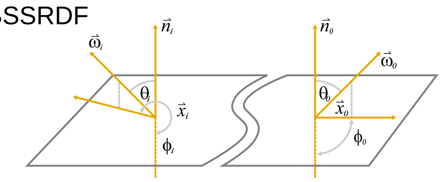


$$(\vec{x}_i, \vec{\omega}_i, t_i, \lambda_i) \rightarrow (\vec{x}_o, \vec{\omega}_o, t_o, \lambda_o)$$

Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

Representing Appearance

- BSSRDF

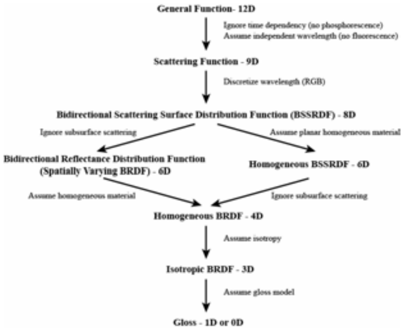


$$(x, y, \theta, \phi, t, \lambda)_{in} \rightarrow (x, y, \theta, \phi, t, \lambda)_{out}$$

Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

Taxonomy of Appearance Representations

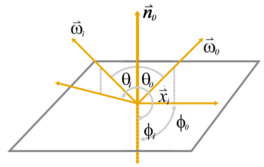
SIGGRAPH2005



Representing Appearance

SIGGRAPH2005

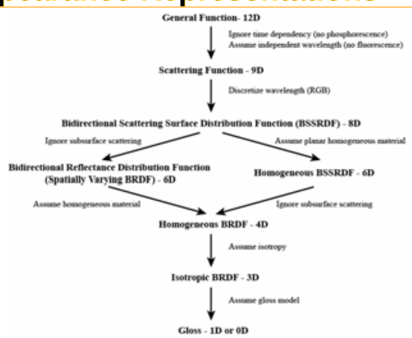
- BRDF



$$(\theta,\phi)_{in} \rightarrow (\theta,\phi)_{out}$$

Taxonomy of Appearance Representations

SIGGRAPH2005



Material Acquisition

SIGGRAPH2005

- single picture
  - no interaction

Material Acquisition

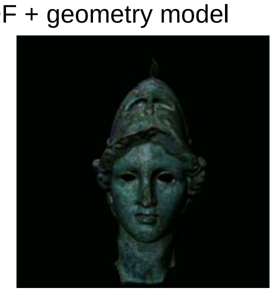
SIGGRAPH2005

- diffuse color + geometry model
  - no relighting

Material Acquisition

SIGGRAPH2005

- spatially-varying BRDF + geometry model
  - moving highlights



Acquisition Approaches

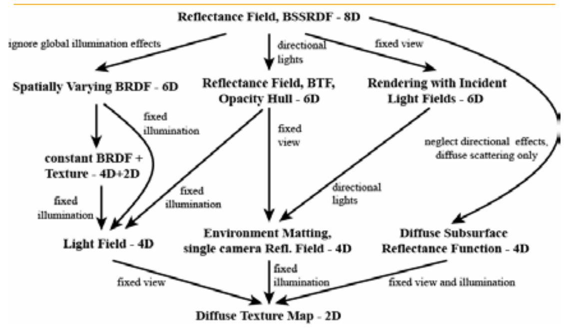
SIGGRAPH2005

- hard to sample an 8D function
- dimensionality reduction
- sampling density

- restricted viewing and relighting capabilities
- restriction to a specific class of materials/objects

Acquisition Taxonomy

SIGGRAPH2005



Comparison

SIGGRAPH2005

			View Res.		Light Res.	
Technique	Acquisition	#Images	spatial	dir.	spatial	dir.
texture maps	diffuse illum.	10-20	hi	-	-	-
light fields	fixed illum.	100-500	hi	med	-	-
image-based BRDF	point light	20-60	hi	hi	hi	hi
SBRDFs	point light	10-40	hi	hi	hi	hi
reflectance fields/BTFs	point light	500-10000	hi	med	-	med
relighting with 4D incident LFs	projector	50000-100000	hi	-	med	lo
environment mattes	monitor	1-1200	hi	-	-	hi
opacity hulls	point light + monitor	20000-60000	hi	med	-	hi
translucent	laser beam	500000-1000000	hi	-	hi	-

SIGGRAPH2005

max planck institut informatik

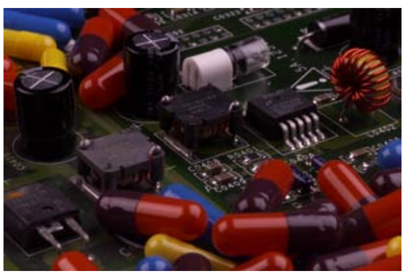
Course 10
Realistic Materials in Computer Graphics

Acquisition Basics

Michael Goesele
MPI Informatik

How can we measure material properties?

- color
- texture
- reflection properties
- normals
- ...

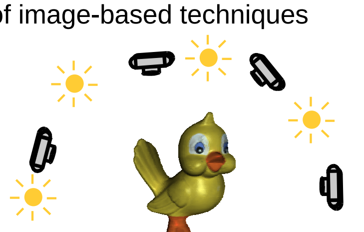


Special Purpose Tools

- spectrophotometer
  - spectral reflectance of a surface
- gloss meter, haze meter, …
  - various appearance characteristics
- often used in industry where single parameters of one material are important

General Purpose Tools

- setup with digital camera(s), controlled lighting, …
- foundation of image-based techniques



General Purpose Tools

- digital camera as
  - massively parallel sensor
  - mostly tristimulus color
  - often high quality optical system
  - tuned to make good and/or correct pictures

Overview Acquisition Basics

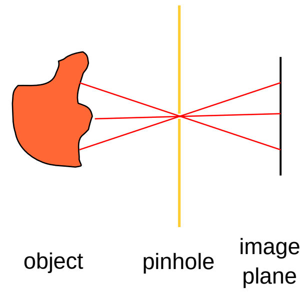
SIGGRAPH2005

- **digital cameras**
  - geometric and photometric calibration
  - high dynamic range imaging
- light sources
- lab setup
- geometry acquisition

Pinhole Camera Model

SIGGRAPH2005

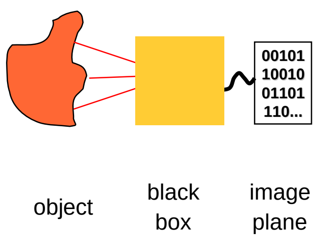
- “each pixel corresponds to one ray through the pinhole onto the object”
- not valid for most digital cameras!!!



(Pessimistic) Digital Camera Model

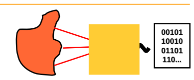
SIGGRAPH2005

- digital camera as a black box
- take only for granted what you measured (or what is given in the manual)



(Pessimistic) Digital Camera Model

SIGGRAPH2005

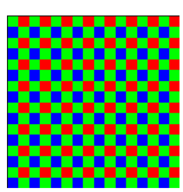


- optical lens system instead of pinhole aperture (aberration, vignetting)
- CCD/CMOS chip and A/D conversion
- normally only one color per pixel (e.g. Bayer pattern) requires demosaicing
- camera image processing
- …

Bayer Pattern

SIGGRAPH2005

- sensor records only one color per pixel
  - higher sampling rate in green channel (luminance channel)
- remaining two color values per pixel must be reconstructed
  - artifacts possible

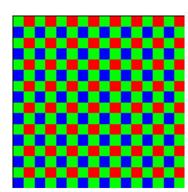


Bayer pattern

Demosaicing

SIGGRAPH2005

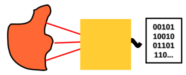
- common approach
  - combining an interpolation and a pattern matching scheme
  - groups pixels into regions and makes some continuity assumption within the regions
  - “nice pictures”, but no guarantee that two of the R,G,B values per pixel are correct



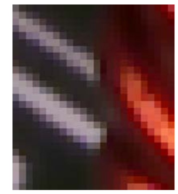
Bayer pattern

(Pessimistic) Digital Camera Model

- often globally correct image
- no guarantee that each pixel contains reliable color values
- some issues can be solved using camera calibration



Overview Acquisition Basics



- digital cameras
  - **geometric and photometric calibration**
  - **high dynamic range imaging**
- light sources
- lab setup
- common problems and solutions

Geometric Camera Calibration

- get transformation between points in space and image coordinates
- intrinsic camera parameters
  - focal length, distortion coefficients, …
- extrinsic parameters
  - position, orientation

Geometric Camera Calibration

- several methods commonly used, e.g., [Tsai ’87, Heikkila ’97, Zhang ’99]
- Matlab calibration toolbox by Jean-Yves Bouguet
  - http://www.vision.caltech.edu/bouguetj/calib_doc/
  - also included in the OpenCV Open Source Computer Vision library distributed by Intel

Camera Model (simplified from Bouguet)

- point in camera reference frame is mapped to normalized pinhole coordinates

$$x_n = \begin{bmatrix} x_n(1) \\ x_n(2) \end{bmatrix} = \begin{bmatrix} X_c / Z_c \\ Y_c / Z_c \end{bmatrix}$$

Camera Model (simplified from Bouguet)

- normalized point coordinates are computed using distortion model
  - only parameterized by distance from center

$$x_d = \begin{bmatrix} x_d(1) \\ x_d(2) \end{bmatrix} = (1 + kc(1)r^2 + kc(2)r^4 + kc(3)r^6)x_n$$

$$r^2 = x_n(1)^2 + x_n(2)^2$$

Camera Model (simplified from Bouguet)

SIGGRAPH2005

- final pixel coordinates are computed using focal length and principal point

$$\begin{bmatrix} x_p(1) \\ x_p(2) \end{bmatrix} = \begin{bmatrix} fc(1) \cdot x_d(1) \\ fc(2) \cdot x_d(2) \end{bmatrix} + \begin{bmatrix} cc(1) \\ cc(2) \end{bmatrix}$$

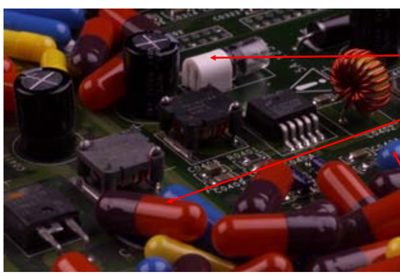
Calibration Approach

SIGGRAPH2005

- capture images of test target with known geometry
  - cover space and angles with planar target
- solve for intrinsic and extrinsic parameters
- quality can be checked by reprojection

Photometric Calibration

SIGGRAPH2005



What do these RGB values (digital counts) mean?

Camera Response Curve (OECF)

SIGGRAPH2005

- relationship between digital counts and luminance is unknown (and often non-linear)
  - gamma correction
  - image optimizations
  - ...
- can be described by response curve or OECF (Opto-Electronic Conversion Function)

Camera Response Curve (OECF)

SIGGRAPH2005

- direct measurement via test chart
  - patches with known gray levels
  - uniform illumination
- patches arranged in a circle to suppress lens effects (e.g. vignetting)
- inversion using OECF leads to pixel values linearly related to luminance values

Definition of Dynamic Range

SIGGRAPH2005

- dynamic range is the ratio of brightest to darkest (non-zero) intensity values in an image
  - assuming linear intensity
- often given as
  - ratio: 1:100.000
  - orders of magnitude: 5 orders of magnitude
  - in decibel: 100 dB

Sources of Dynamic Range

SIGGRAPH2005

- diffuse materials reflect 0.5% – >90% of incoming light
  - specular highlights much brighter
- lit regions vs. in shadow regions
- moonless night vs. sunny day

→ high dynamic range mainly caused by illumination effects

Sources of Dynamic Range

SIGGRAPH2005

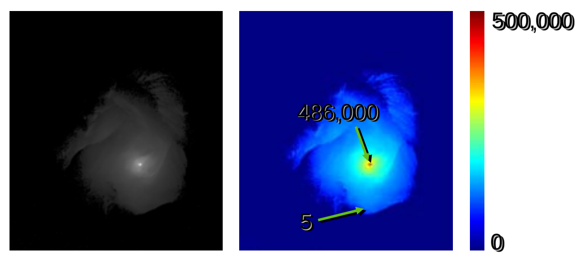
Dynamic Range of Cameras

SIGGRAPH2005

- example: photographic camera with standard CCD sensor
  - dynamic range of sensor 1:1000
  - exposure variation 1/60th s – 1/6000th s (handheld camera/non- static scene) 1:100
  - varying aperture f/2.0 – f/22.0 ~1:100
  - exposure bias/varying "sensitivity" 1:10
  - total (sequential) 1:100,000,000
  - simultaneous dynamic range still only 1:1000

High Dynamic Range (HDR) Imaging

SIGGRAPH2005



High Dynamic Range (HDR) Imaging

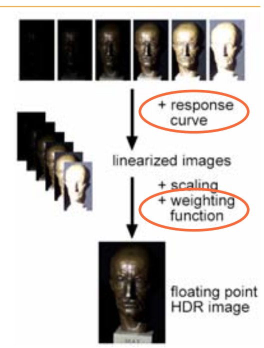
SIGGRAPH2005

- analog film with several emulsions of different sensitivity levels by Wyckoff in the 1960s
  - dynamic range of about 10^8
- modern CMOS sensors can achieve a dynamic range of 10^6 – 10^8
  - logarithm in analog domain
  - multiple exposure techniques

High Dynamic Range Imaging

SIGGRAPH2005

- extending dynamic range of ordinary camera
- combining multiple images with different exposure



Determining the Response Curve

- [Madden 1993] assumes linear response
  - correct for raw CCD data
- [Debevec and Malik 1997]
  - selects a small number of pixels from the images
  - performs an optimization of the response curve with a smoothness constraint
- [Robertson et al. 1999, 2003]
  - optimization over all pixels in all images

Algorithm of Robertson et al.

- Principle of this approach:
  - calculate a HDR image using the response curve
  - find a better response curve using the HDR image
- (to be iterated until convergence)
- assume initially linear response

Algorithm of Robertson et al.

- input:
  - series of i images with exposure times t_i
  - pixel value at image position j is $y_{ij} = f(t_i x_j)$
- find irradiance x_j and response curve $I(y_{ij})$
  - $t_i x_j$ is proportional to collected charge/radiant energy
  - f maps collected charge to intensity values

$$f^{-1}(y_{ij}) = t_i x_j =: I(y_{ij})$$

Algorithm of Robertson et al.

- additional input:
  - a weighting function $w(y_{ij})$ (bell shaped curve)
  - an initial camera response curve $I(y_{ij})$ – usually linear
- calculate HDR values x_j from images using

$$x_j = \frac{\sum_i w(y_{ij}) t_i^2 \cdot \frac{I(y_{ij})}{t_i}}{\sum_i w(y_{ij}) t_i^2} \qquad x_j = \qquad y_{ij}$$

Algorithm of Robertson et al.

- optimizing the response curve I:
  - start again with definition $f^{-1}(y_{ij}) = t_i x_j =: I(y_{ij})$
- minimization of objective function O

$$O = \sum_{i,j} w(y_{ij})(I(y_{ij}) - t_i x_j)^2$$

- using Gauss-Seidel relaxation yields

$$E_m = \{(i,j) : y_{ij} = m\}$$

$$I(m) = \frac{1}{\mathrm{Card}(E_m)} \sum_{i,j \in E_m} t_i x_j$$

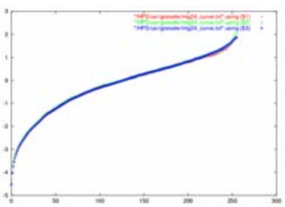
- $\mathrm{Card}(E_m)$ = number of elements in E_m

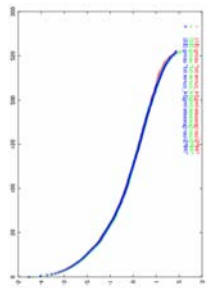
Algorithm of Robertson et al.

- both steps are iterated
  - calculation of a HDR image using I
  - optimization of I using the HDR image
    - → I needs to be normalized, e.g., $I(128)=1.0$
- stop iteration after convergence
  - criterion: decrease of O below some threshold
  - usually only a couple of iterations

HDR Imaging: Algorithm of Robertson et al.

SIGGRAPH2005



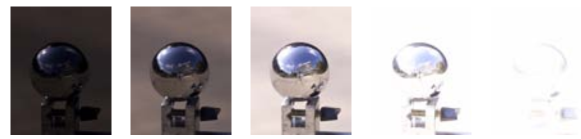


$\log(I(y_{ij}))$

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

HDR Example: Capturing Environment Maps

SIGGRAPH2005



1/2000s 1/500s 1/125s 1/30s 1/8s

series of input images

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

HDR Example: Capturing Environment Maps

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Algorithm of Robertson et al.

SIGGRAPH2005

- choice of weighting function *w(y)* for response recovery

$$w_{ij} = \exp\left(-4\frac{(y_{ij}-127.5)^2}{127.5^2}\right)$$

  - for 8 bit images
  - possible correction at both ends (over/underexposure)
  - motivated by general noise model

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Algorithm of Robertson et al.

SIGGRAPH2005

- choice of weighting function $w(y_{ij})$ for HDR reconstruction
  - introduce certainty function *c* as derivative of the response curve with logarithmic exposure axis
  - approximation of response function by cubic spline to compute derivative

$$w_{ij} = w(y_{ij}) = c(I_{y_{ij}})$$

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Input Images for Response Recovery

SIGGRAPH2005

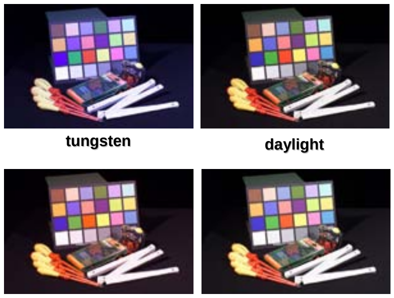
- my favorite:
  - grey card, out of focus, smooth illumination gradient
- advantages
  - uniform histogram of values
  - no color processing or sharpening interfering with the result

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

White Balance

SIGGRAPH2005

capture the spectral characteristics of the light source to assure correct color reproduction

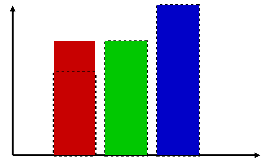


images taken with different camera settings

White Balance

SIGGRAPH2005

- capture white surface under target illumination
- scale color channels to achieve uniform intensity values
- often built-in function



Color Calibration

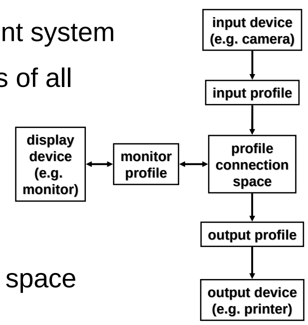
SIGGRAPH2005

- BRDF model of real object
- long processing pipeline
- which image is (more) correct?

Color Calibration

SIGGRAPH2005

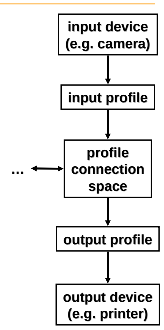
- ICC color management system
- capture the properties of all devices
  - camera and lighting
  - monitor settings
  - output properties
- common interchange space



Color Calibration

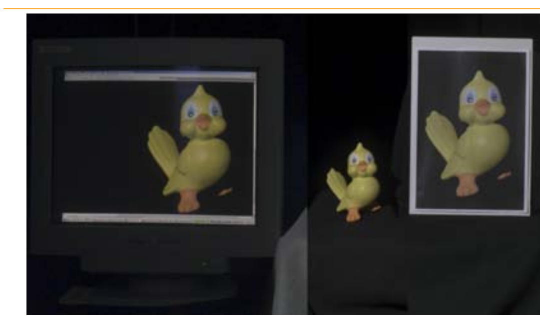
SIGGRAPH2005

- profile connection spaces
  - CIELAB (perceptual linear)
  - linear CIEXYZ color space
- can be used to create an high dynamic range image in the profile connection space
- allows for a color calibrated workflow



Color Calibration

SIGGRAPH2005



[Goesele et al. 2004]

Limits of White Balance and Color Calibration

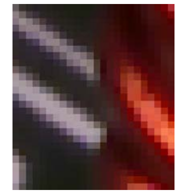
- fluorescence effects
  - signal colors
  - optical brighteners
  - test targets
- color calibration impossible
- cannot be solved using white balance

daylight (HMI)

green LED

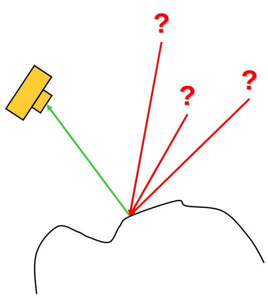
Overview Acquisition Basics

- digital cameras
  - geometric and photometric calibration
  - high dynamic range imaging
- **light sources**
- lab setup
- geometry acquisition



General Measurement Approach

- find relation between incoming and outgoing light at a surface point
- derive information from this data
- knowledge and control over light sources needed



Lighting Requirements

- photometric properties
  - uniform spatial distribution
  - color constant over time
  - even spectral distribution
  - very bright and efficient

Lighting Requirements

- emission pattern
- requirements depend on application, e.g.,
  - well defined light source
  - incident angle as small as possible
    - ➔ *parallel light source (e.g. laser beam)*
    - ➔ *point light source*
  - lens or reflector based systems are not ideal

Point Light Source Example

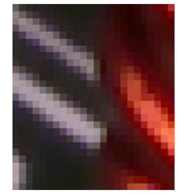
- 800 W HMI light source
- very efficient (equals 2500 W tungsten light)
- (almost) daylight spectrum
- constant colors
- point light source

Point Light Source Example

- more information about lighting in the individual sections of the course …

Overview Acquisition Basics

- digital cameras
  - geometric and photometric calibration
  - high dynamic range imaging
- light sources
- **lab setup**
- geometry acquisition



Lab Setup

- part of the lighting considerations
- often low and diffuse reflection required to minimize the influence of the environment
- MPI photo studio
  - walls and ceiling covered with black felt
  - black needle fleece carpet

Lab Setup

Lab Setup

- tuned for efficiency and flexibility
  - enough space
  - enough stands, supporting materials, …
- have some lighting available in dark areas
  - e.g., radio controlled light switch
- safety concerns

Overview Acquisition Basics

- digital cameras
  - geometric and photometric calibration
  - high dynamic range imaging
- light sources
- lab setup
- **geometry acquisition**

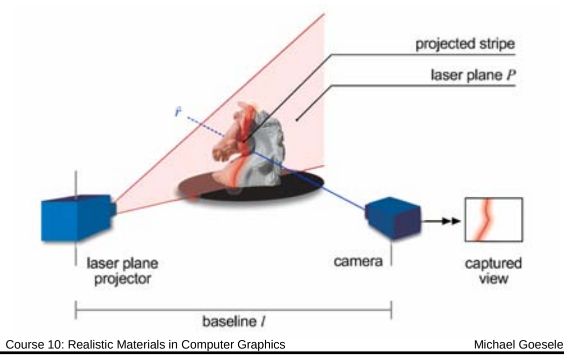
Geometry Acquisition

SIGGRAPH2005

- geometry of test targets often required
- could teach a separate course about the topic
- but some comments …

Geometry Acquisition

SIGGRAPH2005



Geometry Acquisition

SIGGRAPH2005

- usage of standard tools for acquisition pipeline
  - aligning several scans
  - merging into single dataset
  - surface triangulation
  - hole filling
  - noise removal
  - …

Uncooperative Materials

SIGGRAPH2005

- realistic materials are nice to look at but often difficult to scan
- requires some creativity, …
- angel model: based on CT scan

Uncooperative Materials

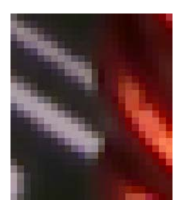
SIGGRAPH2005

- alabaster horse: covered with white dust

Overview Acquisition Basics

SIGGRAPH2005

- digital cameras
  - geometric and photometric calibration
  - high dynamic range imaging
- light sources
- lab setup
- geometry acquisition



Course 10
Realistic Materials in Computer Graphics

Homogeneous Isotropic BRDFs

Wojciech Matusik

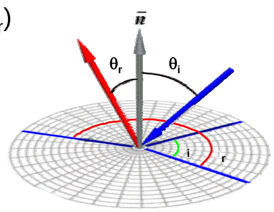
MERL

Outline

- Image-based BRDF measurement
- Efficient measurement
- Validation of analytic BRDF models
- Data-driven BRDF model

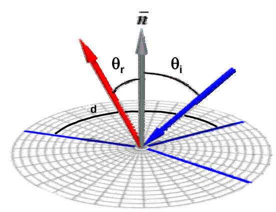
Reflectance

- Bidirectional Reflectance Distribution Function (BRDF)
- $f_r(\theta_i,\phi_i,\theta_r,\phi_r)$



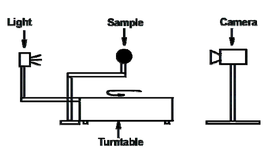
Reflectance

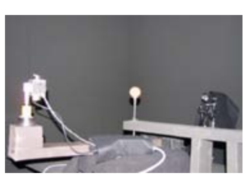
- Isotropic BRDF
- $f_r(\theta_i,\theta_r,\phi_d)$



Measurement

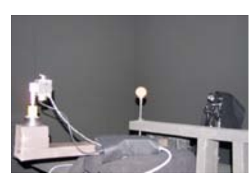
- Inspired by Marshner [1998]





Measurement

- Camera
  - QImaging Retiga
    - Firewire Connection
    - High-Res (1300x1030)
    - 10 bits, Bayer pattern
    - Exposure 40µs - 16 min
  - HDR imaging
    - 17 pictures: 40 µs - 30 s
    - Linear Response Curve



Measurement

SIGGRAPH2005

- Light Source
  - Hamamatsu SQ Xenon lamp
    - Stable emission output
    - Continuous and relatively constant radiation spectrum

Measurement

SIGGRAPH2005

- Turntable
  - Kaidan MD-19
  - Computer-controlled
- Dark Room
  - Walls painted with flat black paint
- Spherical Samples

Calibration

SIGGRAPH2005

- Geometric calibration
  - Contact digitizer
    - Faro Arm
  - Intrinsic & extrinsic camera parameters
  - Sphere center & radius
  - Light Position
    - parameterized on a circle in 3D

Measurement

SIGGRAPH2005

- 20-80 million reflectance measurements per material
- Each tabulated BRDF entails 90x90x180x3=4,374,000 measurement bins

Rendering from Tabulated BRDFs

SIGGRAPH2005

- These BRDFs are immediately useful
- Direct renderings from measurements

Nickel   Hematite   Gold Paint   Pink Felt

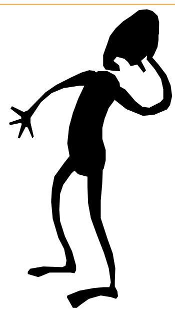
Outline

SIGGRAPH2005

- Image-based BRDF measurement
- Efficient measurement
- Validation of analytic BRDF models
- Data-driven BRDF model

IT WAS PAINFUL!!!

SIGGRAPH2005

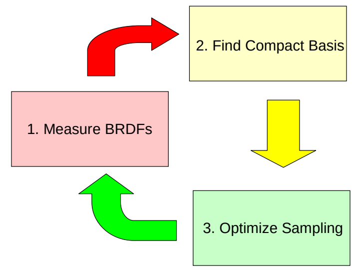


- Each ball took over 4 hours
- In 2 years we measured slightly more than 100 balls
- We repeated the whole process two times, using different light sources, and photometric calibrations
- **Can we apply what we've learned thus far to speed up the process in the future?**

Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Measurement Process

SIGGRAPH2005



Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Wavelet Analysis of BRDFs

SIGGRAPH2005

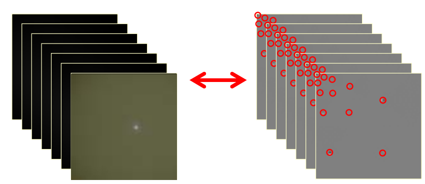
- Wavelets are a very compact representation for BRDFs
  - Schröder [95], Lalonde [97]
  - BRDFs exhibit high frequencies in very specific regions of their parameter space
  - Wavelets perform localized analysis of a larger signal

Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Wavelet Analysis of BRDFs

SIGGRAPH2005

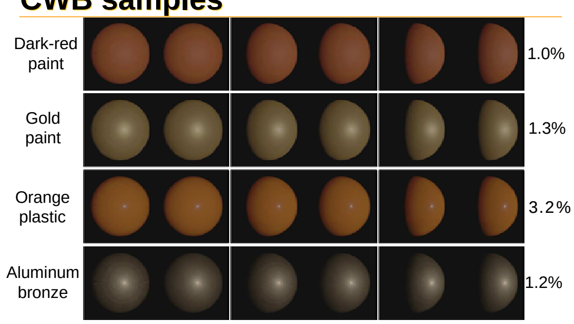
- For a given BRDF, most (+97%) of the information is in only a few wavelet coefficients
- Repeat for all BRDFs, and compute their union
  Common Wavelet Basis (CWB) = ~69,000 coefficients
- This gives 69K specific places to measure each BRDF



Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

BRDFs Reconstructed from CWB samples

SIGGRAPH2005



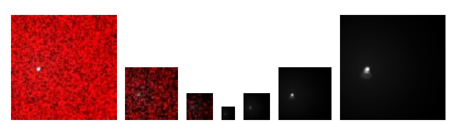
BRDFs based on 69K samples on left, dense samples on right

Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Pull-Push Reconstruction of BRDFs

SIGGRAPH2005

- Alternatively, use Pull-Push to reconstruct (Gortler, et al. 96)
- Treat as scattered data interpolation problem



Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Push-Pull Reconstructions

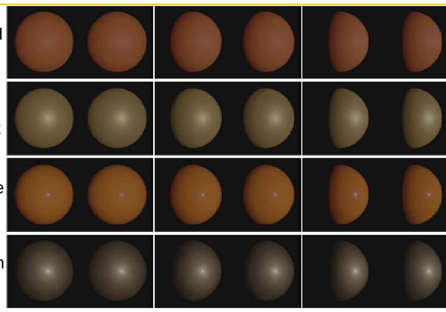
SIGGRAPH2005

Dark-red paint

Gold paint

Orange plastic

Aluminum bronze



0.6%

0.9%

2.5%

1.1%

Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Linear Combinations of BRDFs (LCB)

SIGGRAPH2005

- Can we find a linear combination of our existing BRDFs that match any new one?
- Requires only estimating 100 coefficients for source BRDFs
- Compute a set of 800 constraints that allow estimating these 100 coefficients robustly

Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Linear Combinations of BRDFs

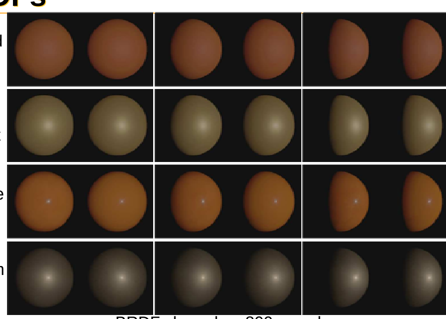
SIGGRAPH2005

Dark-red paint

Gold paint

Orange plastic

Aluminum bronze



1.8%

1.8%

4.3%

2.5%

BRDFs based on 800 samples

Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Sampling Methods Discussion

SIGGRAPH2005

- CWB and Push-Pull
  - require 69,000 measurements
  - BRDF database not required
- Linear Combination of BRDFs
  - requires fewer measurements (800)
  - requires BRDF database

Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Outline

SIGGRAPH2005

- Image-based BRDF measurement
- Efficient measurement
- Validation of analytic BRDF models
- Data-driven BRDF model

Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

BRDF Models

SIGGRAPH2005

- Phenomenological
  - Phong [75]
    - Blinn-Phong [77]
  - Ward [92]
  - Lafortune et al. [97]
  - Ashikhmin et al. [00]
- Physical
  - Cook-Torrance [81]
  - He et al. [91]

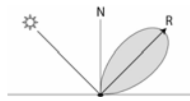
Roughly increasing computation time

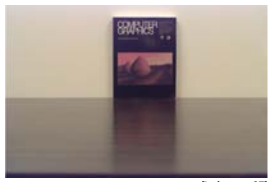
Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Differences in BRDF Models

SIGGRAPH2005

- Specular lobe definition
  - Mirror direction
  - Half vector
- Specular lobe falloff
  - $\cos^n$ (cosine lobe)
  - Gaussian
- Fresnel effect
  - surface becomes more reflective near grazing angle





[Lafortune 97]

Validation Method

SIGGRAPH2005

- BRDF Models
  - Phong, Blinn-Phong, Ward, Lafortune, Ashikhmin, Cook-Torrance
- Data fitting
  - Each material is fitted to each of the models
  - Minimize
    - L^2 error of the BRDF multiplied by the cosine of incident angle

Dark blue paint

SIGGRAPH2005

Acquired data

Environment map

Dark blue paint

SIGGRAPH2005

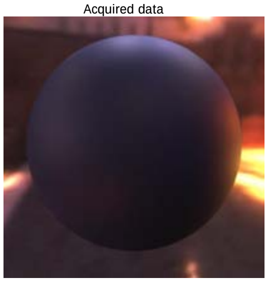
Acquired data

Cook-Torrance

Dark blue paint

SIGGRAPH2005

Acquired data



Ashikhmin

Dark blue paint

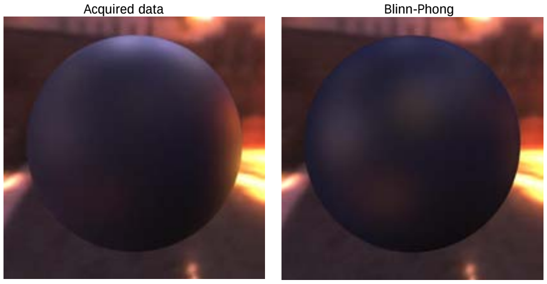
SIGGRAPH2005

Acquired data

Lafortune

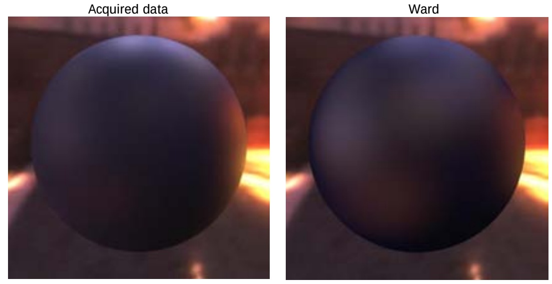
Dark blue paint

Acquired data | Blinn-Phong



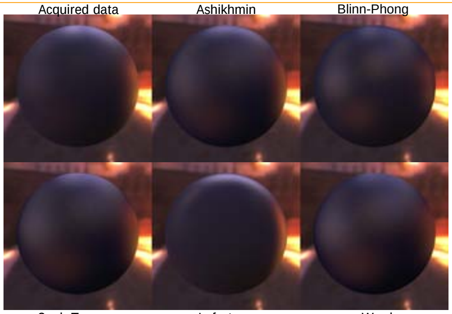
Dark blue paint

Acquired data | Ward



Dark blue paint

Acquired data | Ashikhmin | Blinn-Phong



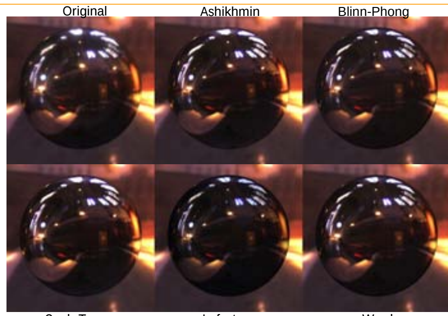
Cook-Torrance | Lafortune | Ward

Chrome

Acquired data

Chrome

Original | Ashikhmin | Blinn-Phong

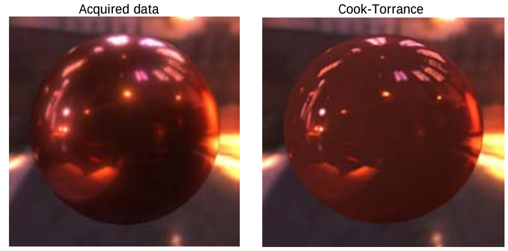


Cook-Torrance | Lafortune | Ward

Observations

- Some materials impossible to represent with a single lobe

Acquired data | Cook-Torrance



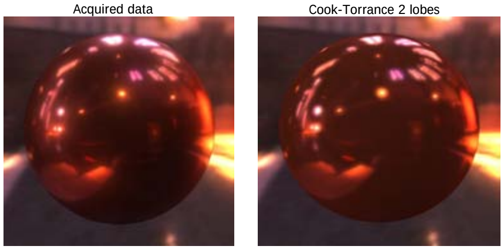
Material – Red Christmas Ball

Adding a second lobe

SIGGRAPH2005

- Some materials impossible to represent with a single lobe

Acquired data — Cook-Torrance 2 lobes



Material – Red Christmas Ball

Observations

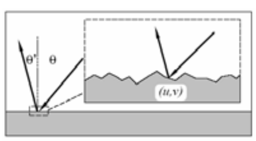
SIGGRAPH2005

- Cook-Torrance, Ashikhmin
  - Consistently outperform the other models
- Lafortune
  - High discrepancy near grazing angle
  - Shape of lobe very different
- Ward, Blinn-Phong
  - unable to reproduce Fresnel effect

Microfacet theory (T&S 1967)

SIGGRAPH2005

- Surface modeled by tiny perfect mirrors
- Value of BRDF at (L,V) = # of mirrors oriented halfway between L and V
- BRDF naturally represented by function of H
- Shape of the mirror lobe cannot be explained with any microfacet distribution



[Shirley 97]

Outline

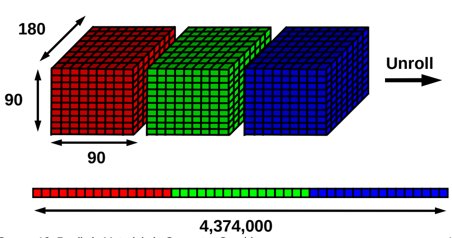
SIGGRAPH2005

- Image-based BRDF measurement
- Efficient measurement
- Validation of analytic BRDF models
- Data-driven BRDF model

BRDFs as Vectors in High Dimensional Space

SIGGRAPH2005

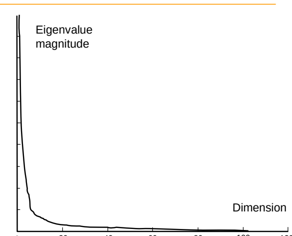
- Each tabulated BRDF is a vector in 90x90x180x3 =4,374,000 dimensional space

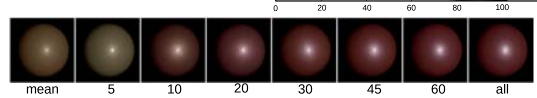


Linear Analysis (PCA)

SIGGRAPH2005

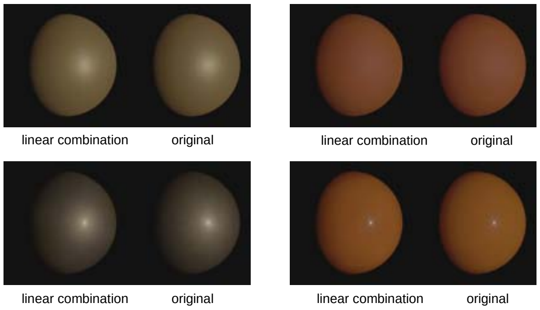
- Find optimal linear basis for our data set
- 45 components needed to reduce residue to under measurement error





Cross-Validation of Linear Model

SIGGRAPH2005



BRDF Modeling using Linear Subspace

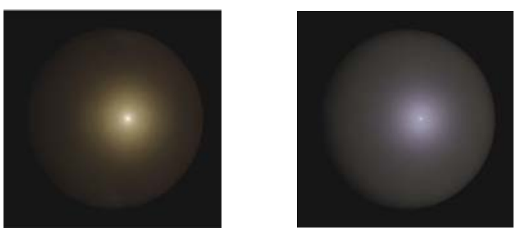
SIGGRAPH2005

- Navigation in linear subspace
  - interpolation
  - extrapolation
- All results are linear combinations of the source BRDFs

Problems with Linear Subspace Modeling

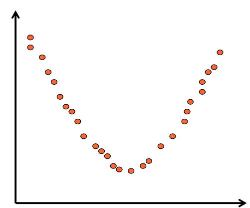
SIGGRAPH2005

- Large number of basis vectors (45)
- Some linear combinations yield invalid or unlikely BRDFs



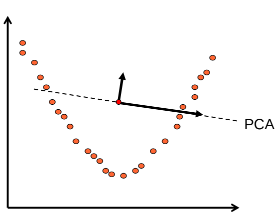
Why does linear modeling fail?

SIGGRAPH2005



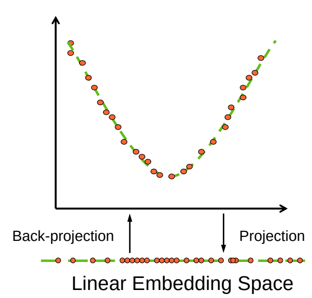
Why does linear modeling fail?

SIGGRAPH2005



Non-Linear Manifold Learning

SIGGRAPH2005

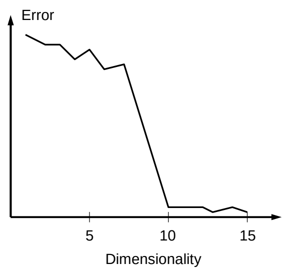


Results of Non-Linear Manifold Learning

SIGGRAPH2005

- Charting [Brand03]
- At 15 dimensions reconstruction error

  is less than 1%
- Parameter count similar to analytical models



Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Intuitive User Interface for our BRDF Model

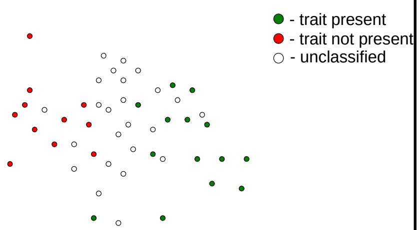
SIGGRAPH2005

- User classification of BRDFs according to a set of predefined traits
  - e.g.: *red*, *specular*, *diffuse*, *metallic*, *plastic*, *rough*, *silver*, *fabric*, *acrylic*, *greasy*, *dusty*, …
- User starts from a source BRDF and applies trait vectors to create desired model

Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Trait Vector Determination

SIGGRAPH2005

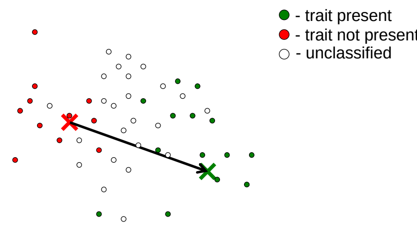


Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Trait Vector Determination

SIGGRAPH2005

- **Mean difference**

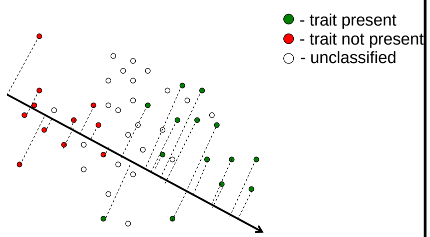


Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Trait Vector Determination

SIGGRAPH2005

- **Fisher's linear discriminant**
  - Find the axis such that distribution projected onto it are the most separable

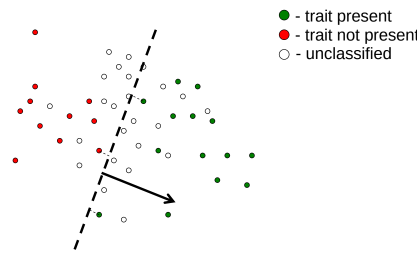


Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Trait Vector Determination

SIGGRAPH2005

- **Support Vector Machines**
  - Find optimal partitioning hyperplane, and use its normal



Course 10: Realistic Materials in Computer Graphics — Wojciech Matusik

Navigation Results

Adding Silver Trait

Navigation Results

Adding Specular Trait

Navigation Results

Adding Metallic Trait

Navigation Results

Adding Glossy Trait

Representing Physical Processes

Steel Oxidation

Outline

- Image-based BRDF measurement
- Efficient measurement
- Validation of analytic BRDF models
- Data-driven BRDF model

SIGGRAPH2005

SIGGRAPH2005

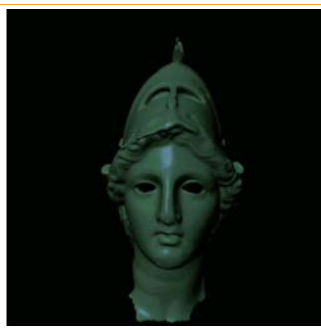
Course 10
Realistic Materials in Computer Graphics

Heterogeneous Isotropic BRDFs

Hendrik P.A. Lensch
Stanford University

Homogeneous BRDF

SIGGRAPH2005



Spatially Varying BRDF

SIGGRAPH2005

Approaches - Sampling

SIGGRAPH2005

- dense sampling for each texel
  - Reflectance Fields, BTF
- sparse sampling
  - image-based BRDF Measurement
  - combining samples from different surface points
- spatial variation
  - constant specular part vs. clustered BRDFs

Approaches - Illumination

SIGGRAPH2005

- point light
  - controlled condition
  - interreflections most often neglected
- environment maps
  - still direct illumination only
- global inverse illumination

BRDF Measurement

- Gonioreflectometer

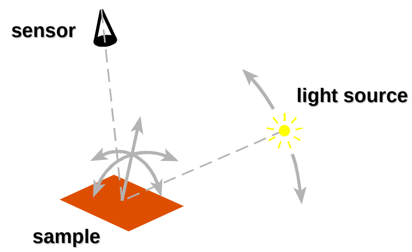
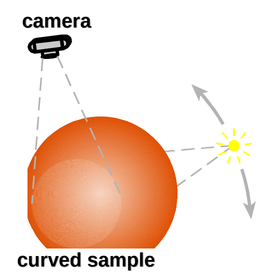


Image-Based BRDF Measurement

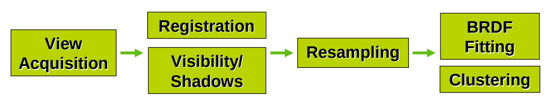
- [Marschner 1999, Lu & Koenderink 1998, …]
  - capture lots of BRDF samples at one shot by a sensor array / camera.
  - homogeneous, isotropic materials only



Acquisition Setup

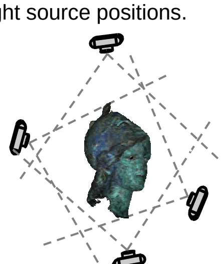
- Camera and light source are moved manually around the object.
- Positions are calibrated with respect to the object.
- The dark room reduces reflections from the environment.

BRDF Fitting and Clustering

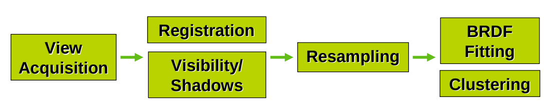


BRDF Acquisition

- Capture HDR-images from various viewpoints with different light source positions.



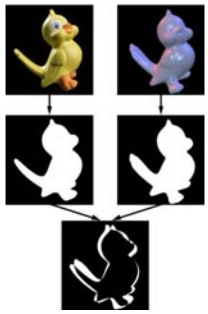
BRDF Fitting and Clustering



3D-2D Registration

SIGGRAPH2005

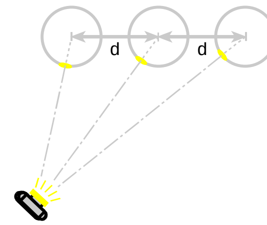
- calibrated gantry
- corresponding points
- silhouette-based method



Light Source Position

SIGGRAPH2005

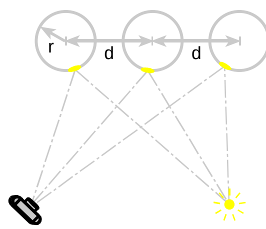
- detect highlights of ring flash reflections
- determine the position of the spheres



Light Source Position

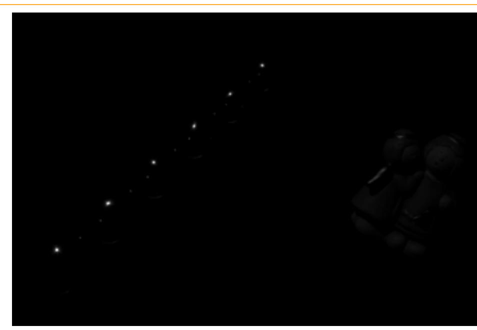
SIGGRAPH2005

- detect highlights of light source reflections
- reconstruct light source position



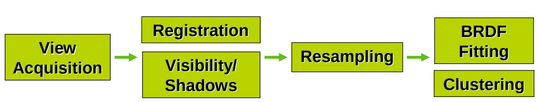
Light Source Position

SIGGRAPH2005



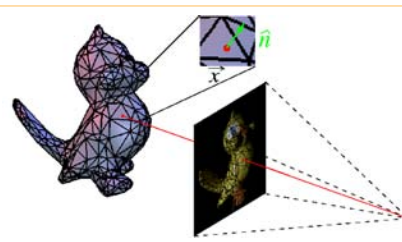
BRDF Fitting and Clustering

SIGGRAPH2005



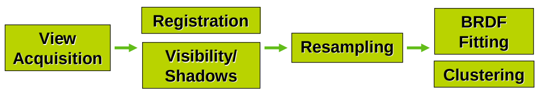
Resampling

SIGGRAPH2005



- for each point on the surface:
  - find all images where the point is visible and lit
  - take sample at corresponding pixel position

BRDF Fitting and Clustering



Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

Key Idea

- Very few radiance samples per texel
  - ⇒ no dense sampling of the BRDF
- Most real-world objects consist of a small set of distinct materials.
  - ⇒ fit a BRDF model for each basis material
  - ⇒ start with the avg. BRDF of the entire surface

Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

The Lafortune Model

$$f_r(\hat{\omega}_i, \hat{\omega}_o) = \rho_d + \sum_j \left(C_{x,j} (\omega_{ix}\omega_{ox} + \omega_{iy}\omega_{oy}) + C_{z,j}\omega_{iz}\omega_{oz} \right)^{N_j}$$

- physically plausible
- diffuse component plus a number of lobes
- 3*(1+ i*3) parameters (12 for a single lobe model)
- fit parameters to samples using Levenberg-Marquardt

Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

Fitting BRDFs to Lumitexels

- define error measure between a BRDF and a lumitexel:

$$E_{f_r}(L) = \frac{1}{|L|} \sum_{R_j \in L} \left(f_r(\hat{\omega}_{i_j}, \hat{\omega}_{o_j}) \hat{\omega}_{i_j z} - r_j \right)^2$$

= average error over all radiance samples

- perform non-linear least square optimization for a ***set*** of lumitexels using Levenberg-Marquardt
- yields a single BRDF (i.e. its parameters) per ***set*** of lumitexels

Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

Fitting Result

Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

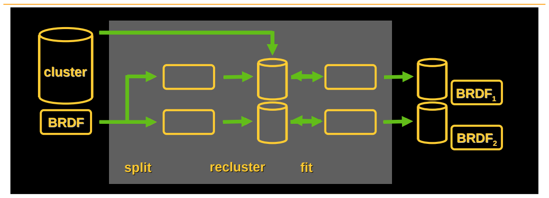
Clustering

- Goal: separate the different materials
  - similar to Lloyd iteration
  - start with a single cluster containing all lumitexels
  - split cluster along direction of largest variance
  - stop after *n* clusters have been constructed

Course 10: Realistic Materials in Computer Graphics — Hendrik Lensch

Split-Recluster-Fit Cycle

SIGGRAPH2005



- split into two BRDFs along direction of largest variance of parameters (covariance matrix)
- distribute initial lumitexels forming two new clusters
- refit new BRDFs
- repeat reclustering and fitting until clusters are stable

Clustering Results

SIGGRAPH2005

Spatially Varying Materials

SIGGRAPH2005

Projection

SIGGRAPH2005

- Goal: assign a separate BRDF to each lumitexel
  - too few radiance samples for a reliable fit
  - represent the BRDF f_π of every lumitexel by a linear combination of already determined BRDFs of the clusters $f_1, f_2, \ldots, f_m$:

$$f_\pi = t_1 f_1 + t_2 f_2 + \ldots + t_m f_m$$

  - determine linear weights $t_1, t_2, \ldots, t_m$

Projection

SIGGRAPH2005

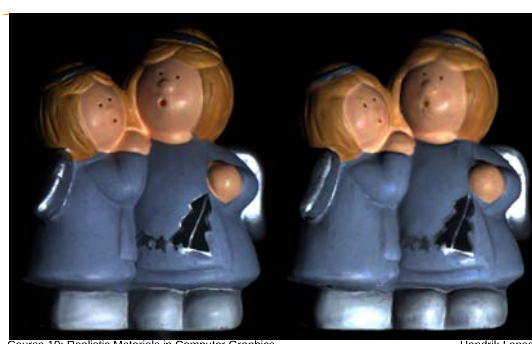
compute the pseudo-inverse using non-negative SVD to get a least squares solution for

$$\begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_{|L|} \end{pmatrix} = \begin{pmatrix} f_1(\hat{\omega}_{i1}, \hat{\omega}_{o1})\hat{\omega}_{i1,z} & f_2(\hat{\omega}_{i1}, \hat{\omega}_{o1})\hat{\omega}_{i1,z} & \cdots & f_m(\hat{\omega}_{i1}, \hat{\omega}_{o1})\hat{\omega}_{i1,z} \\ f_1(\hat{\omega}_{i2}, \hat{\omega}_{o2})\hat{\omega}_{i2,z} & f_2(\hat{\omega}_{i2}, \hat{\omega}_{o2})\hat{\omega}_{i2,z} & \cdots & f_m(\hat{\omega}_{i2}, \hat{\omega}_{o2})\hat{\omega}_{i2,z} \\ \vdots & \vdots & \ddots & \vdots \\ f_1(\hat{\omega}_{i|L|}, \hat{\omega}_{o|L|})\hat{\omega}_{i|L|,z} & f_2(\hat{\omega}_{i|L|}, \hat{\omega}_{o|L|})\hat{\omega}_{i|L|,z} & \cdots & f_m(\hat{\omega}_{i|L|}, \hat{\omega}_{o|L|})\hat{\omega}_{i|L|,z} \end{pmatrix} \cdot \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_m \end{pmatrix}$$

- it is a linear problem!

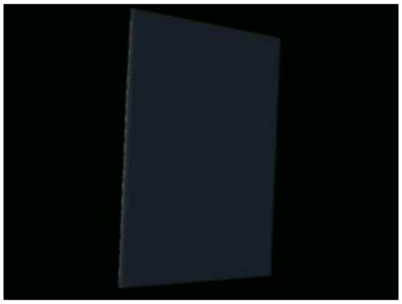
Results

SIGGRAPH2005



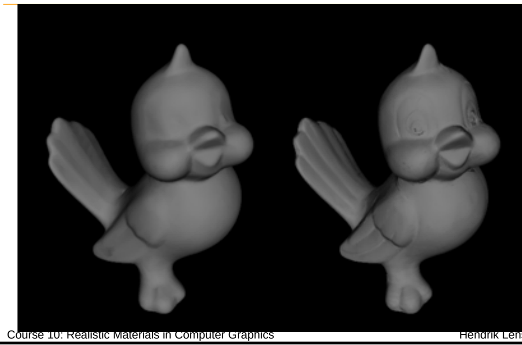
Why doing the complicated clustering?

SIGGRAPH2005



Normal Fitting

SIGGRAPH2005

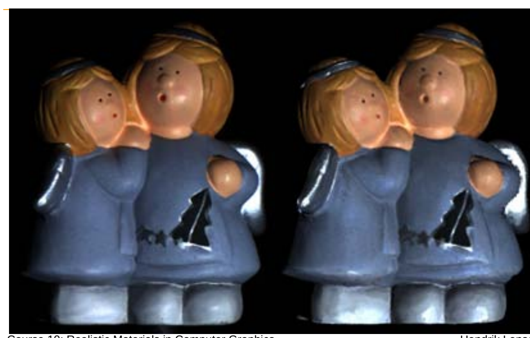


Without Normal Fitting

SIGGRAPH2005

With Normal Fitting

SIGGRAPH2005



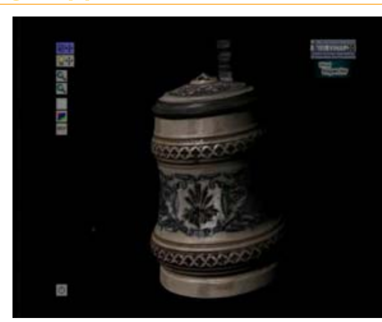
Model Exhibition for Cultural Heritage Applications

SIGGRAPH2005

- Preservation, documentation, restoration, communication, accessibility

Reflectance Models for Cultural Heritage Applications

SIGGRAPH2005



SIGGRAPH2005

---

max planck institut informatik

SIGGRAPH2005

Course 10
Realistic Materials in Computer Graphics

Translucent Materials

Michael Goesele

MPI Informatik

Translucent Objects

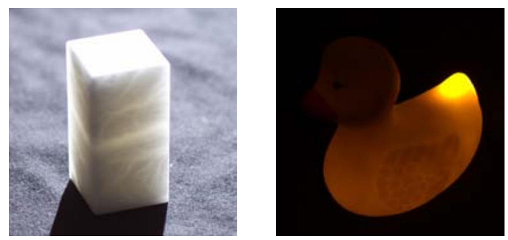
SIGGRAPH2005

- light is scattered through the object
- incident illumination smoothed due to diffuse scattering inside media

---

Inhomogeneous Translucent Objects

SIGGRAPH2005



- caused by material variation or internal structure
- required for realistic appearance

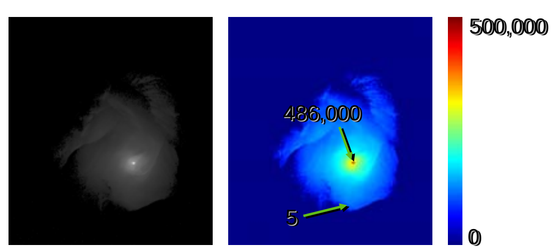
Varying Local Illumination

SIGGRAPH2005

- caused by shadows or directional light sources

Inherent High Dynamic Range

SIGGRAPH2005



Overview

SIGGRAPH2005

- **models for translucent objects**
- the BSSRDF
- dipole approximation
- hierarchical model for multiple scattering
- acquisition with DISCO

Models for Translucent Objects

SIGGRAPH2005

- basic physical properties
  - e.g., absorption and scattering cross sections σ_a and σ_s [Ishimaru78]
  - defined for the whole object volume
- rendering possible with variety of techniques such as
  - finite element methods [Rushmeier90, Sillion95, Blasi93]

Models for Translucent Objects

SIGGRAPH2005

- rendering techniques (contd.)
  - finite element methods [Rushmeier90, Sillion95, Blasi93]
  - bidirectional path tracing [Hanrahan93, Lafortune96]
  - photon mapping [Jensen98, Dorsey99]
  - Monte Carlo simulations [Pharr00, Jensen99]
  - diffusion [Stam95, Stam01]
  - precomputed radiance transfer [Sloan03a]

Models for Translucent Objects

SIGGRAPH2005

- light fields / reflectance fields
  - light fields [Levoy96,Gortler96,Miller98,Wood00,…]
  - reflectance fields [Debevec00,Malzbender01]
  - opacity hulls [Matusik03]
  - incident light fields [Masselus03]

  ⇒ no correct response to locally varying illumination

  ⇒ see remainder of the course

Models for Translucent Objects

SIGGRAPH2005

- specialized models
  - BSSRDF [Nicodemus 1977]
  - dipole approximation [Jensen et al. 2001]
    - includes measurements of physical parameters for homogeneous materials
  - DISCO system to acquire diffuse multiple scattering model R_d [Goesele et al. 2004]
    - modeling of inhomogeneous materials

Overview

SIGGRAPH2005

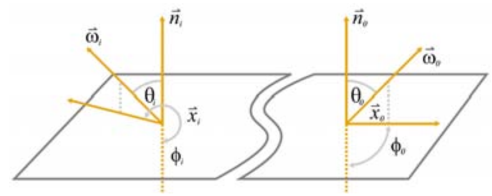
- models for translucent objects
- **the BSSRDF**
- dipole approximation
- hierarchical model for multiple scattering
- acquisition with DISCO

The BSSRDF

- bidirectional scattering-surface reflectance distribution function [Nicodemus 1977]
  - general model of light transport inside an object
  - (almost) equivalent to a reflectance field [Debevec et al. 2000]
  - ratio of reflected radiance to incident flux
  - 8 dimensional function

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

The BSSRDF



$$S(\vec{x}_i,\hat{\omega}_i;\vec{x}_o,\hat{\omega}_o) := \frac{dL^{\rightarrow}(\vec{x}_o,\hat{\omega}_o)}{d\Phi^{\leftarrow}(\vec{x}_i,\hat{\omega}_i)}$$

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

The BSSRDF

- outgoing radiance computed by integrating over the whole surface and all incoming directions

$$L^{\rightarrow}(\vec{x}_o,\hat{\omega}_o) = \int_A \int_\Omega L^{\leftarrow}(\vec{x}_i,\hat{\omega}_i)\cdot S(\vec{x}_i,\hat{\omega}_i;\vec{x}_o,\hat{\omega}_o)\langle\hat{n}_i\cdot\hat{\omega}_i\rangle d\hat{\omega}_i d\vec{x}_i$$

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Single Scattering vs. Multiple Scattering

- single scattering contribution strongly dependent on incoming light direction
  - example: honey pot illuminated by a laser from the left

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Single Scattering vs. Multiple Scattering

- multiple scattering (almost) independent of incident light direction
  - example: alabaster block illuminated by a laser from the left

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

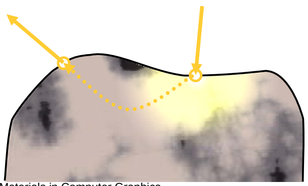
Single Scattering vs. Multiple Scattering

- often modeled independently, e.g.,
  - single scattering using ray tracing
  - multiple scattering using a less complex model with diffuse approximation

Course 10: Realistic Materials in Computer Graphics — Michael Goesele

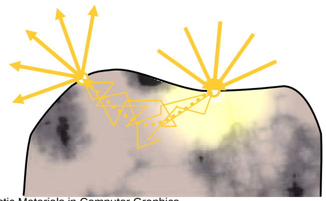
Diffuse Approximation

- BSSRDF too complex for many application
  - acquisition, storage, …
  - all combinations of directions and positions



Diffuse Approximation

- neglect directional dependence
  - frequent scattering events in optically dense media lead to diffuse scattering inside the media



Diffuse Approximation

- approximate BSSRDF by diffuse reflectance
  - only 4 dimensions
  - requires Fresnel terms at incoming and outgoing locations
  - simplifies handling drastically

$$S(\vec{x}_i, \hat{\omega}_i; \vec{x}_o, \hat{\omega}_o) = \frac{1}{\pi} F_t(\eta, \hat{\omega}_i) R_d(\vec{x}_i, \vec{x}_o) F_t(\eta, \hat{\omega}_o)$$

Overview

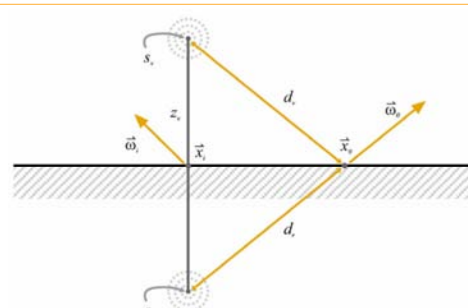
- models for translucent objects
- the BSSRDF
- **dipole approximation**
- hierarchical model for multiple scattering
- acquisition with DISCO

Dipole Approximation

- [Jensen et al. 2001]
- infinite half-space of homogeneous material
- optically dense, modeling of multiple scattering

$$R_d(\vec{x}_i, \vec{x}_o) = \frac{\alpha'}{4\pi} \left[z_r(1 + \sigma_{tr} d_r) \frac{e^{-\sigma_{tr} d_r}}{d_r^3} + z_v(1 + \sigma_{tr} d_v) \frac{e^{-\sigma_{tr} d_v}}{d_v^3} \right]$$

Dipole Approximation

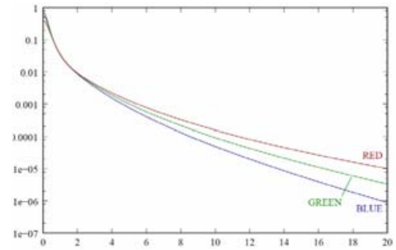


Dipole Approximation

$$\begin{aligned}
z_r &= 1/\sigma_t' \\
z_v &= z_r + 4AD \\
d_r &= \|\vec{x}_r - \vec{x}_o\|, \text{ with } \vec{x}_r = \vec{x}_i - z_r \hat{n}_i \\
d_v &= \|\vec{x}_v - \vec{x}_o\|, \text{ with } \vec{x}_v = \vec{x}_i + z_v \hat{n}_i \\
A &= \frac{1 + F_{dr}}{1 - F_{dr}} \\
F_{dr} &= -\frac{1.440}{\eta^2} + \frac{0.710}{\eta} + 0.668 + 0.0636\eta \\
D &= 1/3\sigma_t' \\
\sigma_{tr} &= \sqrt{3\sigma_a \sigma_t'} \\
\sigma_t' &= \sigma_a + \sigma_s' \\
\alpha' &= \sigma_s'/\sigma_t'
\end{aligned}$$

Dipole Approximation

- example: marble from [Jensen et al. 2001]

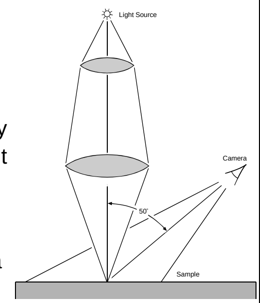


Determining Physical Parameters

- required for dipole approximation
  - scattering and absorption coefficient
  - relative index of refraction
- also required for evaluation of single scattering term

Determining Physical Parameters

- image-based measure-ment setup [Jensen et al. 2001]
  - surface point illuminated by focused beam of white light
  - object observed by digital camera
  - parameters determined via diffusion solution



Overview

- models for translucent objects
- the BSSRDF
- dipole approximation
- **hierarchical model for multiple scattering**
- acquisition with DISCO

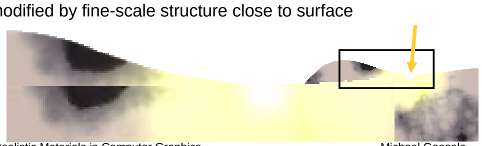
Hierarchical Model

- properties of radiosity inside object
  - exponential fall-off
  - strong variation near incident illumination
  - smooth variation far away from incident illumination
    - modified by fine-scale structure close to surface

Hierarchical Model

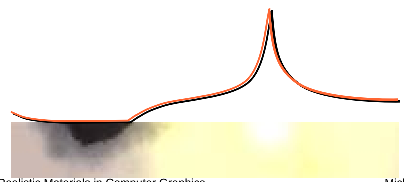
SIGGRAPH2005

- properties of radiosity inside object
  - exponential fall-off
  - strong variation near incident illumination
  - smooth variation far away from incident illumination
    - modified by fine-scale structure close to surface



Hierarchical Model

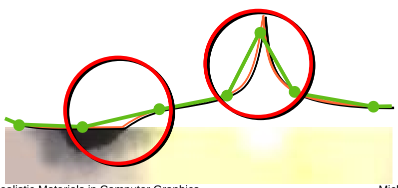
SIGGRAPH2005



Hierarchical Model

SIGGRAPH2005

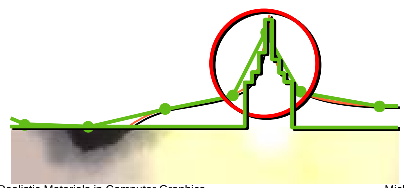
- global response as sparse transfer matrix
  - e.g., vertex-to-vertex throughput factors on mesh
  - fine scale detail preserved by a modulation texture



Hierarchical Model

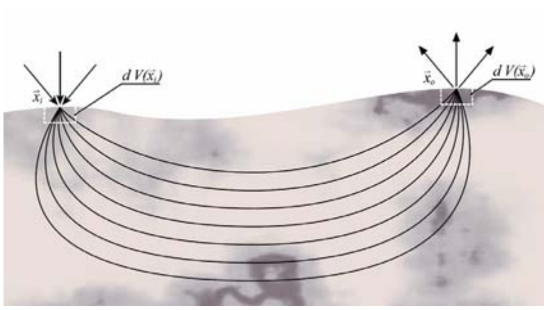
SIGGRAPH2005

- local response
  - pixel-to-pixel throughput factors in texel space
  - small neighborhood only



Hierarchical Model

SIGGRAPH2005



Overview

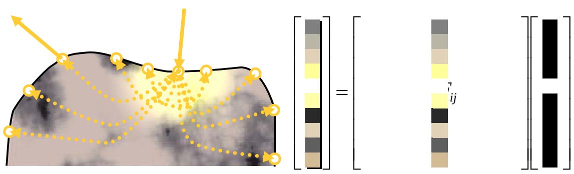
SIGGRAPH2005

- models for translucent objects
- the BSSRDF
- dipole approximation
- hierarchical model for multiple scattering
- **acquisition with DISCO**

Key Idea

SIGGRAPH2005

- directly record R_d for real objects
  - illuminate individual surface points of an object
  - record impulse response function

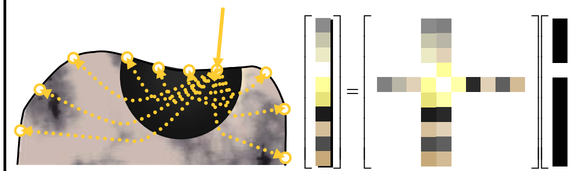


Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Key Idea

SIGGRAPH2005

- directly record R_d for real objects
  - illuminate individual surface points of an object
  - record impulse response function

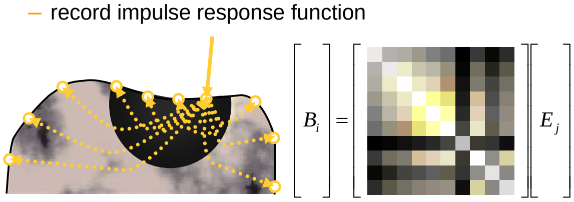


Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Key Idea

SIGGRAPH2005

- directly record R_d for real objects
  - illuminate individual surface points of an object
  - record impulse response function

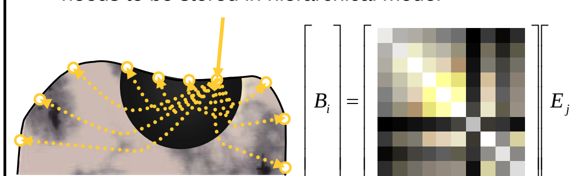


Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Key Idea

SIGGRAPH2005

- directly record R_d for real objects
  - results in huge transfer matrix
  - needs to be stored in hierarchical model

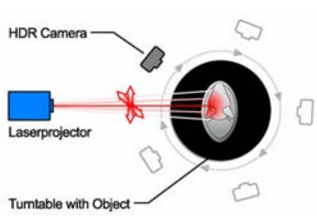


Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Acquisition Approach

SIGGRAPH2005

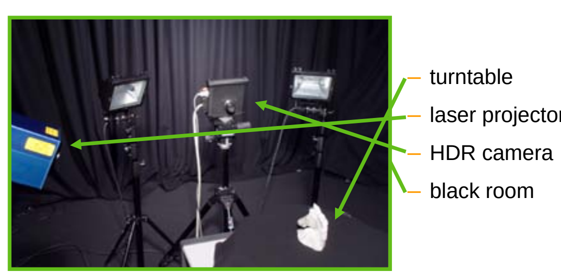
- illuminate individual surface points with laser projector
- record impulse response of the object with HDR video camera



Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Acquisition Equipment

SIGGRAPH2005

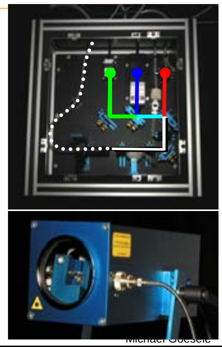


Course 10: Realistic Materials in Computer Graphics — Michael Goesele

Laser Projector

SIGGRAPH2005

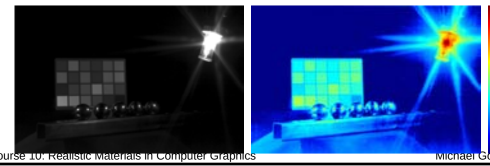
- three lasers
  - (635nm, 532nm, 476nm)
  - ~10mW optical power
- constancy
- resolution 4096x4096 steps
  - step size < 0.1 mm on object possible
- repeatable



HDR Video Camera

SIGGRAPH2005

- Silicon Vision Lars III
  - other HDR video cameras also available
- monochrome HDR
- dynamic range > 1:1,000,000
- acquisition with 15 fps



Throughput Factor Matrix

SIGGRAPH2005

- 500,000-1,000,000 images acquired
- all views where the same point is illuminated yield one column of the point-to-point matrix

• problems:
- approx $100,000^2$ entries
- still a lot of holes
- impractical to render

Resampling

SIGGRAPH2005

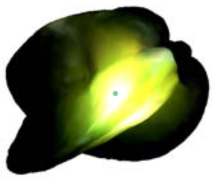
• resample images into hierarchical model
- assemble global term at vertex positions
- determine filter kernels in texture space

• problems
- incomplete sampling
  - → *missing data needs to be interpolated*
- measurement noise
- see [Goesele et al. 2004] for details

Post-Processing Steps

SIGGRAPH2005

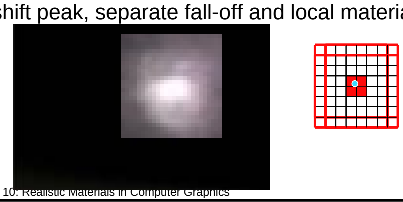
- smoothing of F_{ji} in CIE YU*V* color space
- hole filling by filtering and interpolation in log Y.
- diffusion and image impainting could also be used



Post-Processing Steps

SIGGRAPH2005

- transform to texture space and grab kernels
- laser hit point not aligned to sampling grid
  - → *assemble 7x7 filter kernel* $K_{7x7}(x,n)$ *for each of the four surrounding pixel*
- shift peak, separate fall-off and local material detail



Results

SIGGRAPH2005

photograph

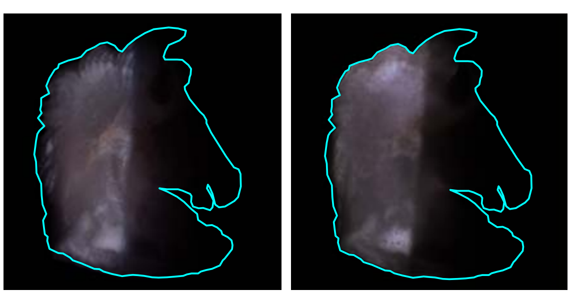
rendering

Results: Local Illumination Variation

SIGGRAPH2005

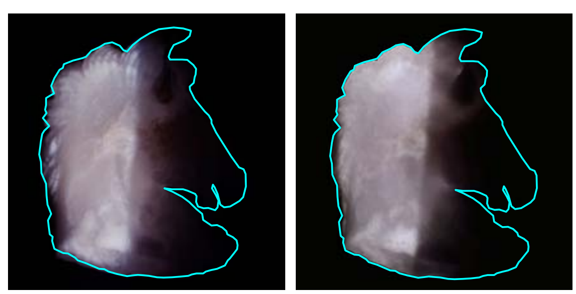
Results: Local Illumination Variation

SIGGRAPH2005



Results: Local Illumination Variation

SIGGRAPH2005



Results

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics

Michael Goesele

Overview

SIGGRAPH2005

– models for translucent objects
– the BSSRDF
– dipole approximation
– hierarchical model for multiple scattering
– acquisition with DISCO

Course 10: Realistic Materials in Computer Graphics

Michael Goesele

SIGGRAPH2005

Course 10
Realistic Materials in Computer Graphics

Environment Matting

Yung-Yu Chuang
National Taiwan University

Blue screen matting

input image   alpha composite

Problem: blue foreground

source image   alpha composite

Course 10: Realistic Materials in Computer Graphics   Yung-Yu Chuang

Two-screen matting

alpha composite

Course 10: Realistic Materials in Computer Graphics   Yung-Yu Chuang

Problem: refractive object

alpha composite

Course 10: Realistic Materials in Computer Graphics   Yung-Yu Chuang

Problem: refractive object

SIGGRAPH2005

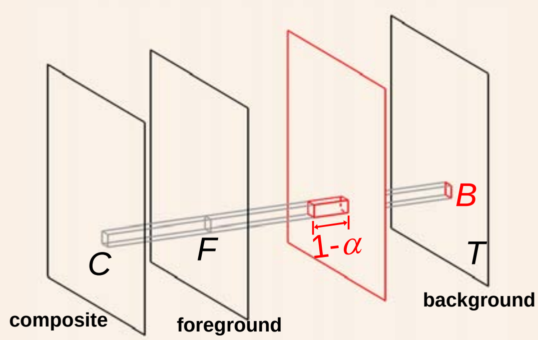
alpha composite

photograph

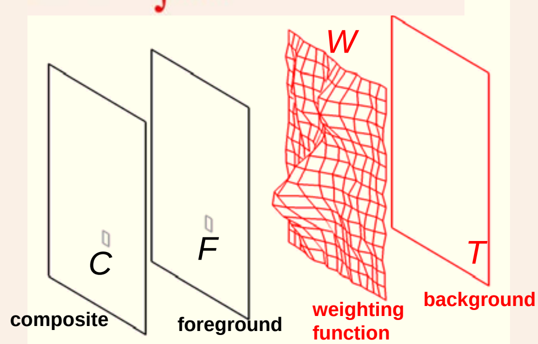
SIGGRAPH2005

Environment matting framework

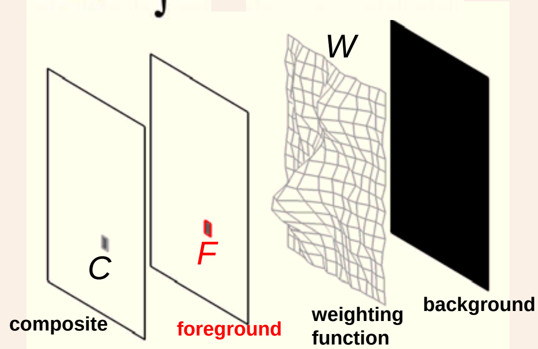
$$C = F + (1-\alpha)B$$



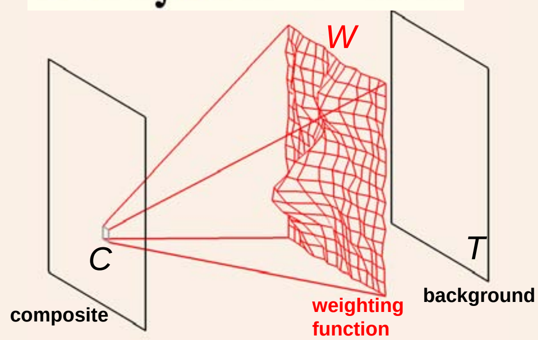
$$C = F + \int W\, T$$



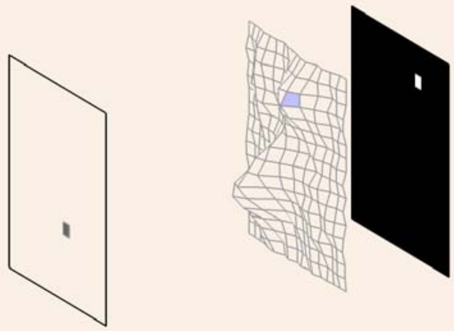
$$C = F + \int W\, 0$$



$$C = F + \int W\, T$$



Arbitrary weighting function

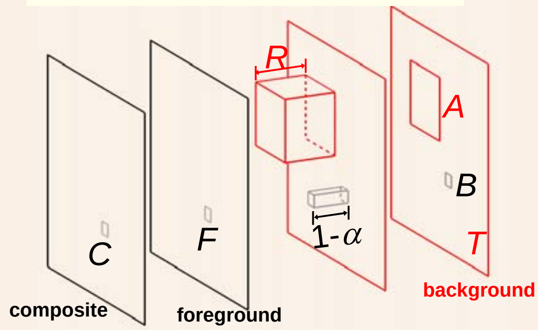


Hierarchical environment matting

Zongker et. al.

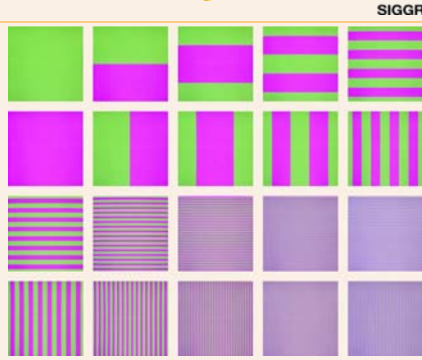
SIGGRAPH 1999

$$C = F + (1-\alpha)B + R\mathcal{M}(T, A)$$

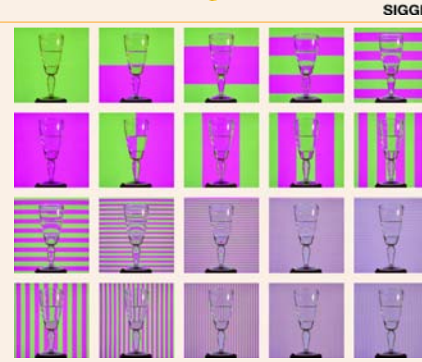


Acquisition setup

Hierarchical backgrounds



Hierarchical backgrounds



Computing R and F from alpha

SIGGRAPH2005

$$C = F + (1-\alpha)X + RX$$

$$C' = F + (1-\alpha)X' + RX'$$

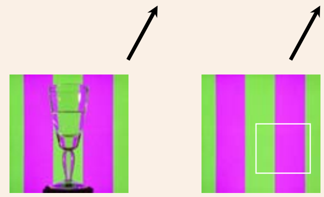
$$R(\alpha) = \frac{C - C'}{X - X'} - (1-\alpha)$$

$$F(\alpha) = C - (1 - \alpha + R(\alpha))X$$

Searching for α and *A*

SIGGRAPH2005

$$E = \sum \left\| C_{observed} - C_{computed}(\alpha, A) \right\|^2$$



hypothesize d α and *A*

Separate *x* and *y* extent searches

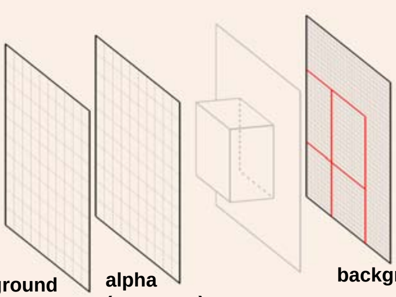
SIGGRAPH2005

(α, l, r) (α, t, b)

Environment matte composite

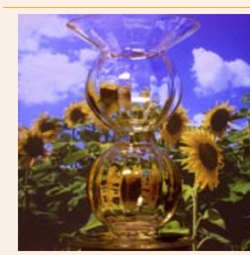
SIGGRAPH2005

$$C = F + (1-\alpha)B + R\mathcal{M}(T, A)$$

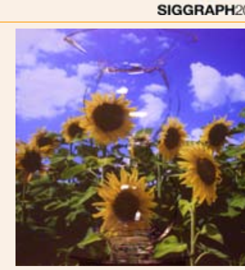


Results and comparisons

SIGGRAPH2005



environment matte composite



alpha matte composite

Results and comparisons

SIGGRAPH2005

environment matte composite

photograph

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Results and comparisons

SIGGRAPH2005

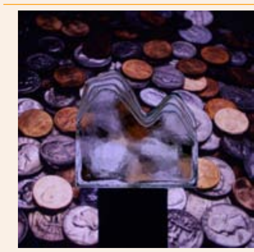
environment matte composite

alpha matte composite

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Results and comparisons

SIGGRAPH2005



environment matte composite

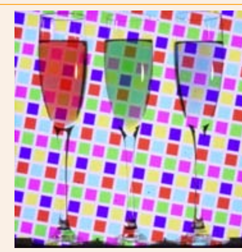
photograph

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Results and comparisons

SIGGRAPH2005

environment matte composite



alpha matte composite

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Results and comparisons

SIGGRAPH2005

environment matte composite

photograph

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

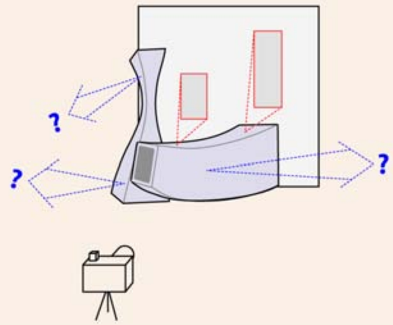
Reflective objects

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Many rays not captured

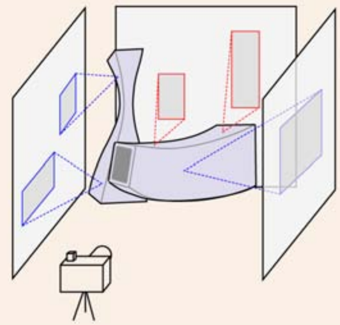
SIGGRAPH2005



Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Add sidedrops to capture these rays

SIGGRAPH2005



Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Capturing multiple sides

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

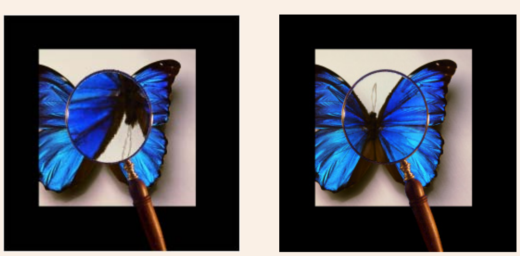
Contributions from multiple sides

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Depth correction

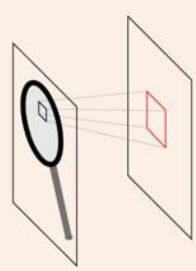
SIGGRAPH2005



Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Capturing at a single depth

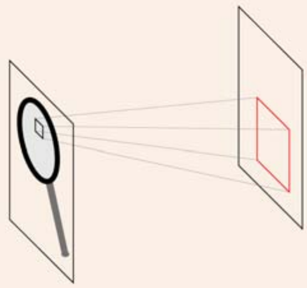
SIGGRAPH2005



Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Capturing a second depth

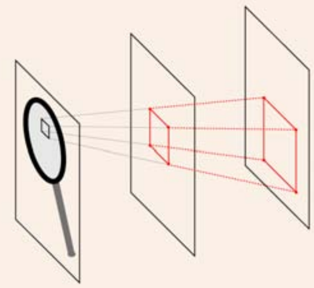
SIGGRAPH2005



Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Constructing the 3D beam

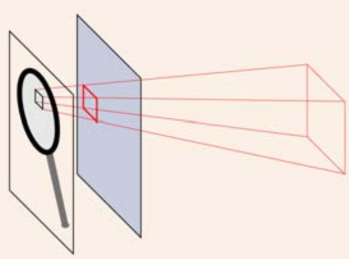
SIGGRAPH2005



Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Rendering at novel depths

SIGGRAPH2005



Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Rendering at novel depths

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Problem: glossy surface

SIGGRAPH2005

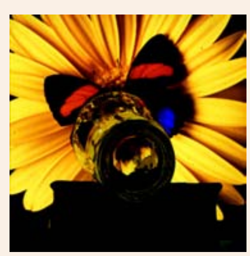
environment matte composite

photograph

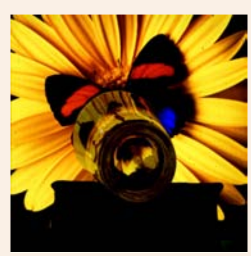
Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Problem: multiple mappings

SIGGRAPH2005



environment matte composite



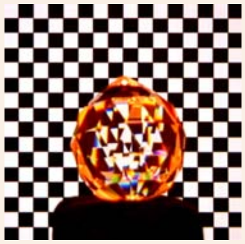
photograph

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Problem: color dispersion

SIGGRAPH2005

environment matte composite



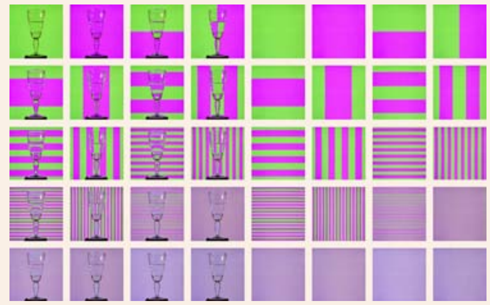
photograph

Course 10: Realistic Materials in Computer Graphics

Yung-Yu Chuang

Problem: many photographs needed

SIGGRAPH2005

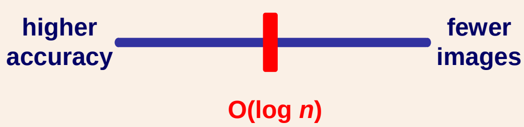


Course 10: Realistic Materials in Computer Graphics

Yung-Yu Chuang

Environment matting

SIGGRAPH2005

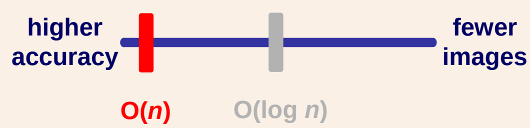


Course 10: Realistic Materials in Computer Graphics

Yung-Yu Chuang

Towards higher accuracy

SIGGRAPH2005

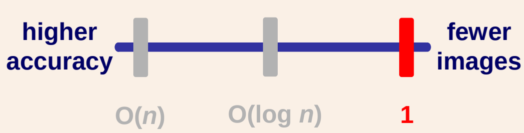


Course 10: Realistic Materials in Computer Graphics

Yung-Yu Chuang

Towards real-time capture

SIGGRAPH2005



Course 10: Realistic Materials in Computer Graphics

Yung-Yu Chuang

SIGGRAPH2005

Real-time environment matting

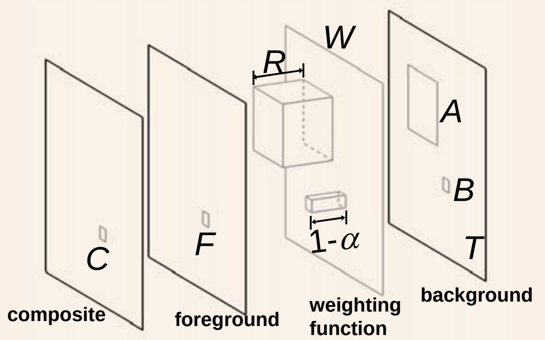
Chuang et. al.

SIGGRAPH 2000

Course 10: Realistic Materials in Computer Graphics

Yung-Yu Chuang

$$C = F + (1-\alpha)B + R\mathcal{M}(T,A)$$



$$C = F + (1-\alpha)B + R\mathcal{M}(T,A)$$

3 3 1 3 4

3 observations
11 variables

- A, R
- α
- F

$$C = R\mathcal{M}(T,A)$$

3 3 4

3 observations
7 variables

- A, R
- α
- F

$$C = \rho\mathcal{M}(T,A)$$

3 1 4

3 observations
5 variables

- A, R ⟶ A, ρ colorless
- α
- F

$$C = \rho\, T(c_x, c_y)$$

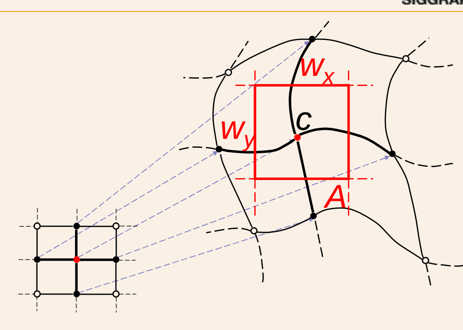
3 1 1 1

3 observations
3 variables

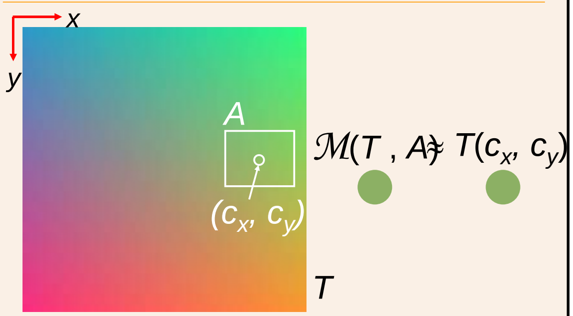
- A, R ⟶ A, ρ ⟶ c_x, c_y, colorless specularly refractive
- α
- F

Estimate w_x, w_y

SIGGRAPH2005

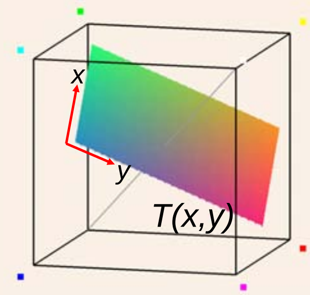


Stimulus function



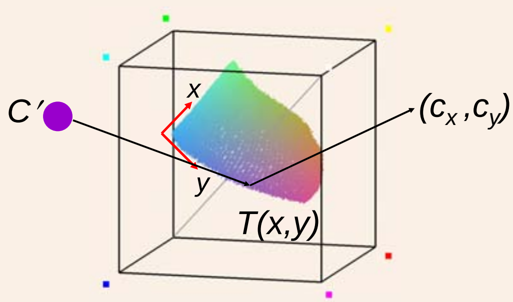
Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

Ideal plane in RGB cube



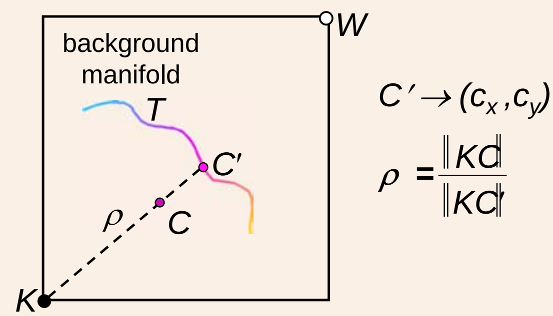
Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

Calibrated manifold in RGB cube



Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

Estimate c_x ,c_y and ρ

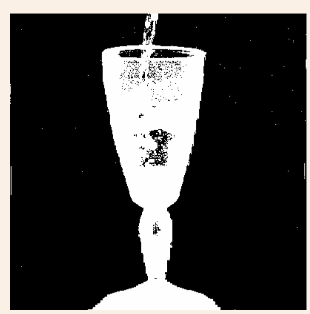


Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

Input image

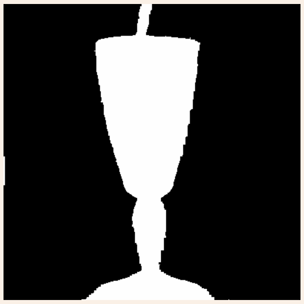
Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

Difference thresholding



Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

Morphological operation



Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

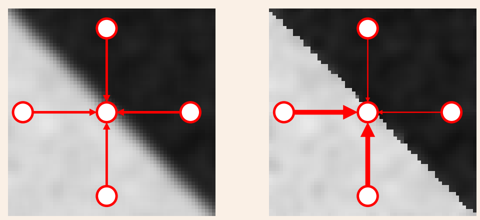
Feathering

Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

Problem: noisy matte

Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

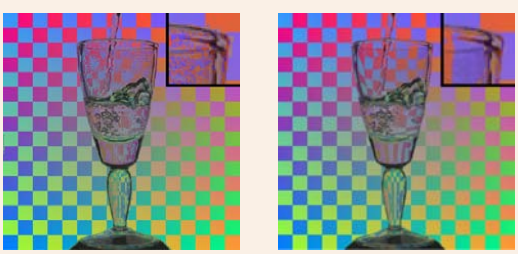
Edge-preserving filtering



isotropic filter anisotropic filter

Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

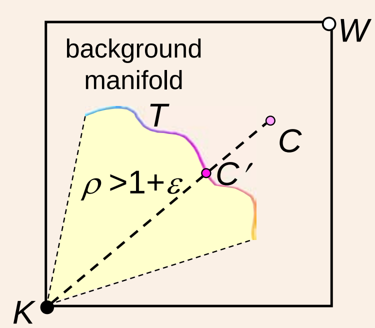
Edge-preserving filtering



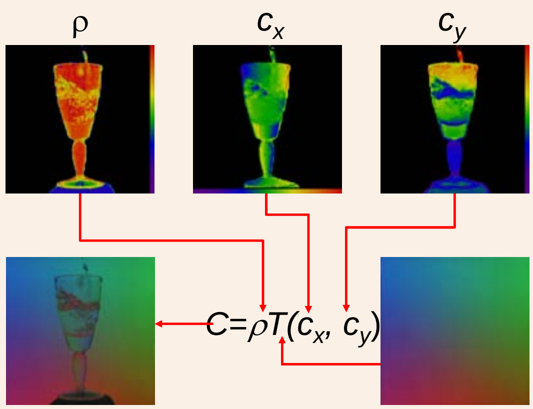
without filtering with filtering

Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang

Heuristics for specular highlights



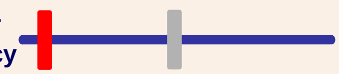
Course 10: Realistic Materials in Computer Graphics Yung-Yu Chuang



Heuristics for specular highlights

Composite with highlights

Towards higher accuracy

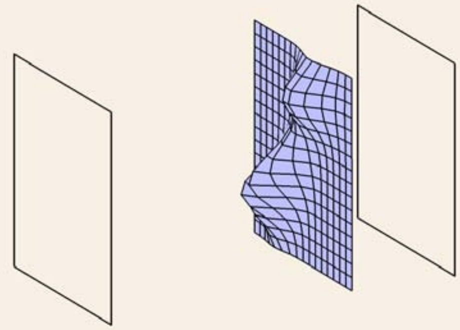


Gaussian environment matting

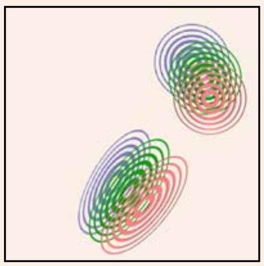
Chuang et. al.

SIGGRAPH 2000

Multimodal oriented Gaussian

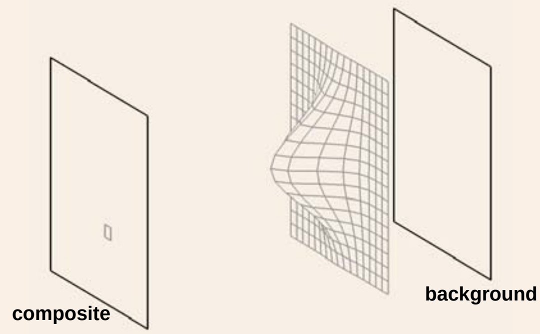


Multimodal oriented Gaussian

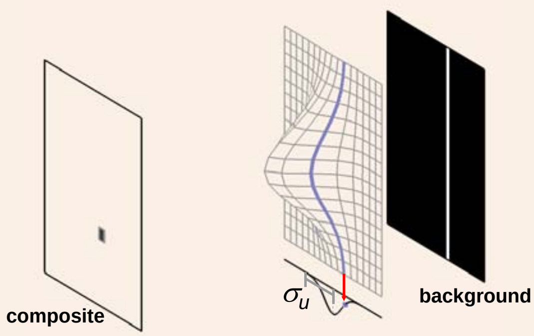


- Better BRDF approximation
- Multiple mappings
- Wavelength-coupled variation

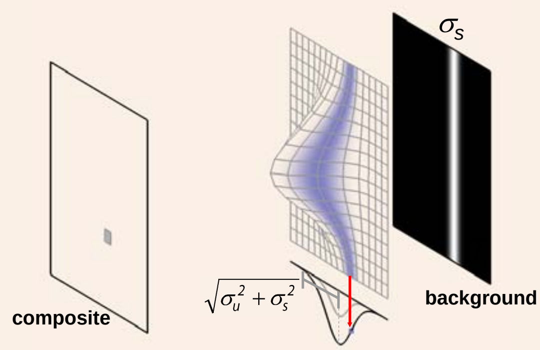
Unimodal axis-aligned Gaussian



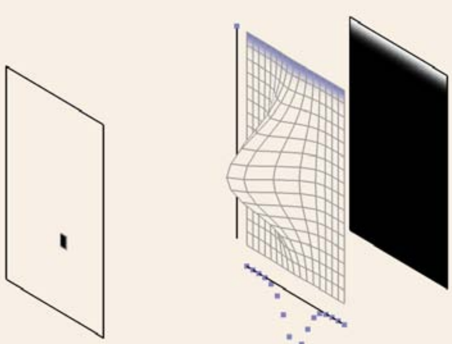
Unimodal axis-aligned Gaussian



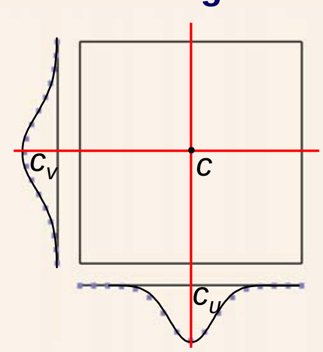
Unimodal axis-aligned Gaussian



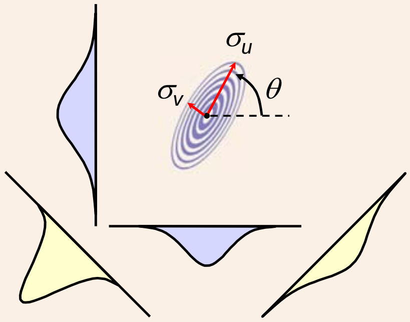
Unimodal axis-aligned Gaussian



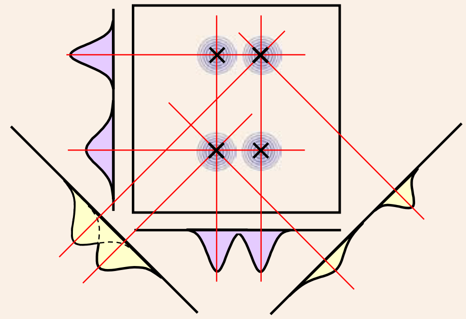
Unimodal axis-aligned Gaussian



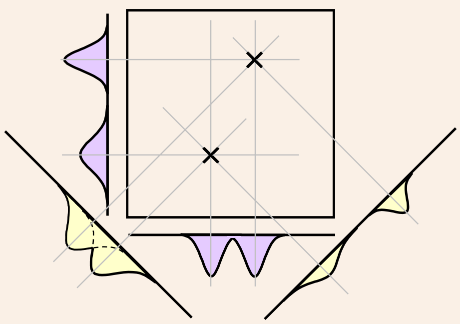
Unimodal oriented Gaussian



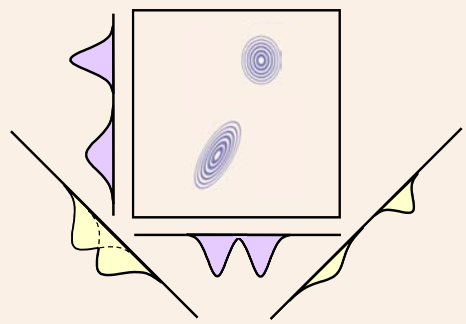
Multimodal oriented Gaussian



Multimodal oriented Gaussian



Multimodal oriented Gaussian



Glossy surface

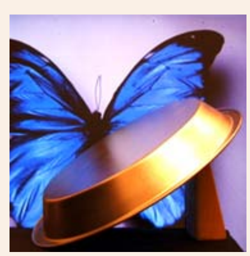
SIGGRAPH2005

SIGGRAPH 99

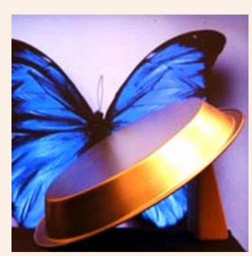
photograph

Glossy surface

SIGGRAPH2005



higher accuracy algorithm



photograph

Oriented Gaussian


without orientation


photograph

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Oriented Gaussian


with orientation


photograph

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Multiple mappings


SIGGRAPH 99


photograph

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Multiple mappings

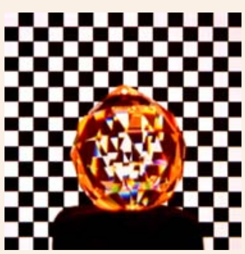

higher accuracy algorithm


photograph

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

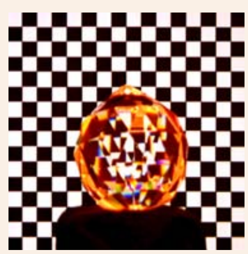
Color dispersion

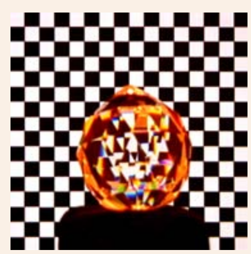

SIGGRAPH 99


photograph

Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Color dispersion


higher accuracy algorithm


photograph

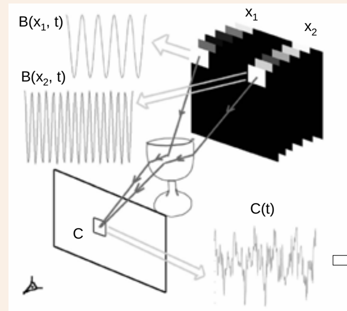
Course 10: Realistic Materials in Computer Graphics — Yung-Yu Chuang

Frequency-based environment matting

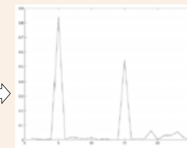
Zhu et. al.

Pacific Graphics 2004

Frequency-based environment matting



spectrum of C

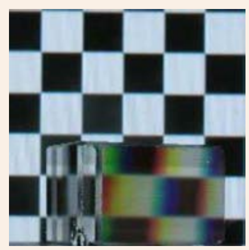


Results: refraction

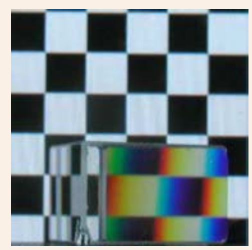
frequency-based environment matting

photograph

Results: color dispersion

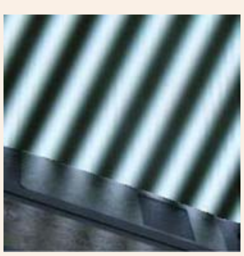


frequency-based environment matting

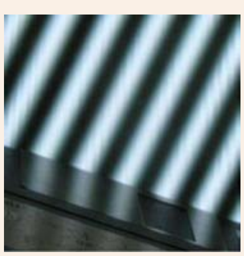


photograph

Results: oriented

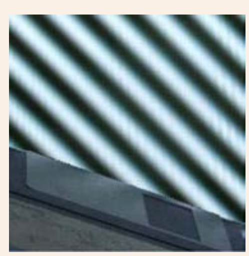


frequency-based environment matting



photograph

Results: oriented



frequency-based environment matting

photograph

SIGGRAPH2005

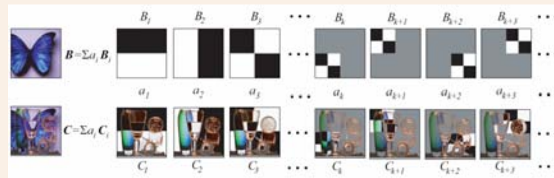
Wavelet environment matting

Peers et. al.

EGSR 2003

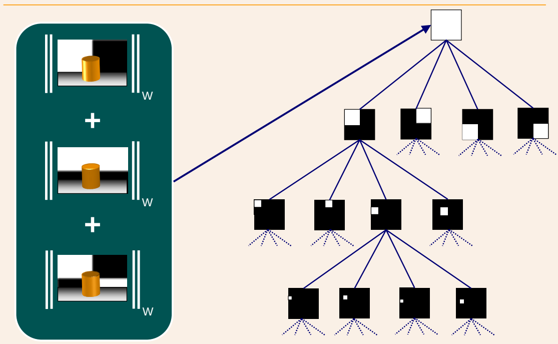
Wavelet environment matting

SIGGRAPH2005



Error tree

SIGGRAPH2005



Results: number of basis images

SIGGRAPH2005

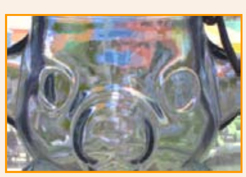
reference image

1200 basis images

Results: number of basis images

SIGGRAPH2005

reference image



1200 basis images

Results: wavelet patterns

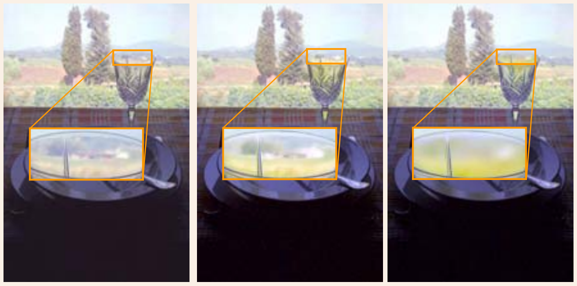
SIGGRAPH2005

reference image

1000 Haar patterns

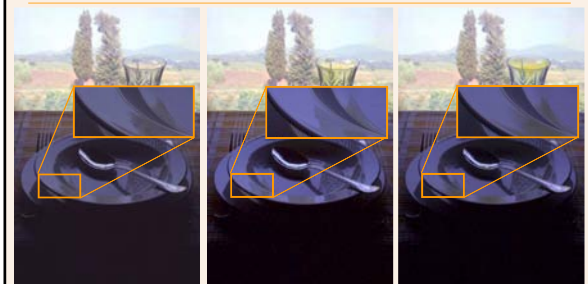
1000 Daubechies (9,7) patterns

Results: wavelet patterns



reference image | 1000 Haar patterns | 1000 Daubechies (9,7) patterns

Results: wavelet patterns



reference image | 1000 Haar patterns | 1000 Daubechies (9,7) patterns

Results: diffuse materials

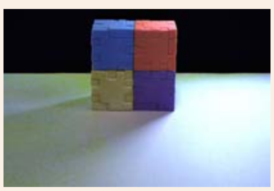


Image-based environment matting

Wexler et. al.

EWSR 2002

Image-based environment matting

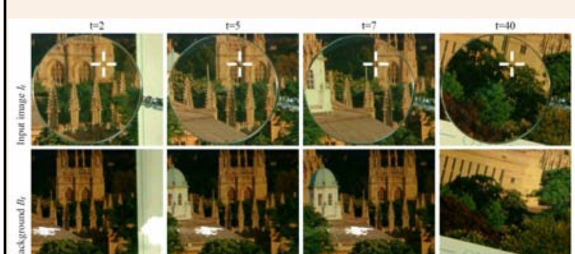


Image-based environment matting

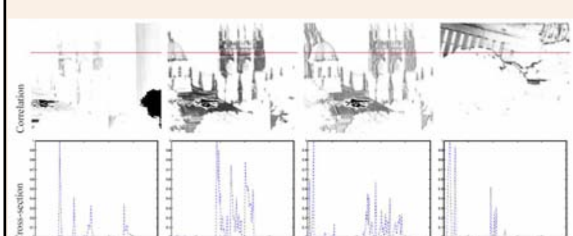
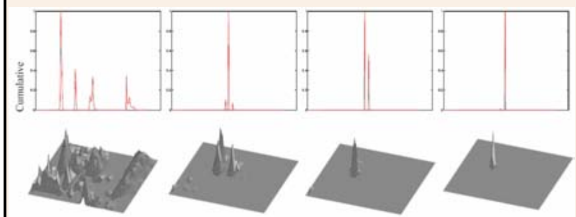


Image-based environment matting

SIGGRAPH2005



Results

SIGGRAPH2005

Results

SIGGRAPH2005

SIGGRAPH2005

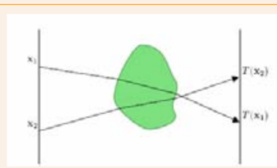
Refractive Optical Flow

Agarwal et. al.

ECCV 2004

Refractive optical flow

SIGGRAPH2005

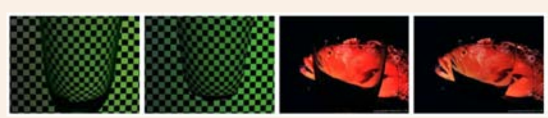


$$\mathbf{c}(T(\mathbf{x}),t)^\top \mathbf{J}^{-\top}(T(\mathbf{x}))\left[\nabla I(\mathbf{x},t) - I(\mathbf{x},t)\nabla\beta(\mathbf{x})\right] + I_t(\mathbf{x},t) = 0$$

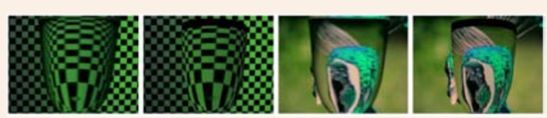
refractive optical flow equation

Results

SIGGRAPH2005



without water



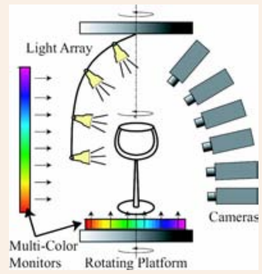
with water

Acquisition and Rendering of Transparent and Refractive Objects

Matusik et. al.

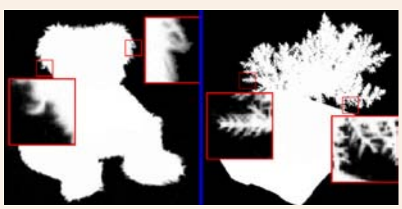
EWSR 2002

The system



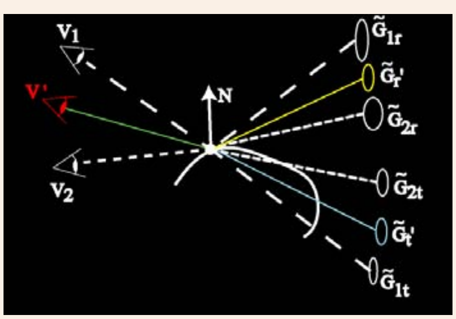
Geometry - opacity hull

- Visual hull augmented with view-dependent opacity
- Store the opacity of each observation at each point on the visual hull.



Results

Interpolation



Acknowledgements

Douglas Zongker, Dawn Werner, Brian Curless, David Salesin, Joel Hindorff, Richard Szeliski, Jiayuan Zhu, Yee-Hong Yang, Pieter Peers, Philip Dutre, Yonatan Wexler, Andrew Fitzgibbon, Andrew Zisserman, Sameer Agarwal, Satya Mallick, David Kriegman, Serge Belongie, Wojciech Matusik, Hanspeter Pfister, Remo Ziegler, Addy Ngan, Leonard McMillan

SIGGRAPH2005

Course 10
Realistic Materials in Computer Graphics

Scattering from fibers

Steve Marschner
Cornell University

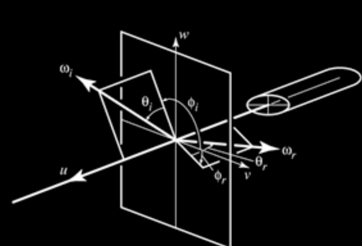
Overview

- Reflection from fibers
  - *reflection geometry*
  - *radiometry*
- Scattering from human hair
  - *the nature of hair*
  - *hair scattering phenomena*
  - *a model for scattering from hair*

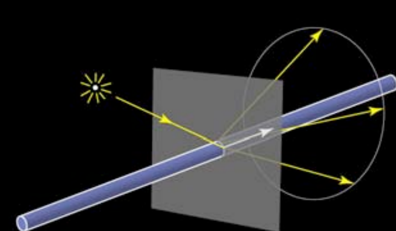
Fiber reflection

- Fibers are nearly 1D
- Choice: model as 1D or 3D
- 1D model leads to different radiometry from surfaces

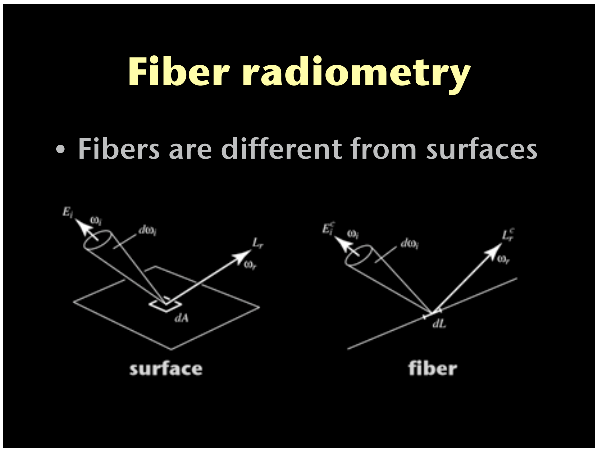
Scattering geometry



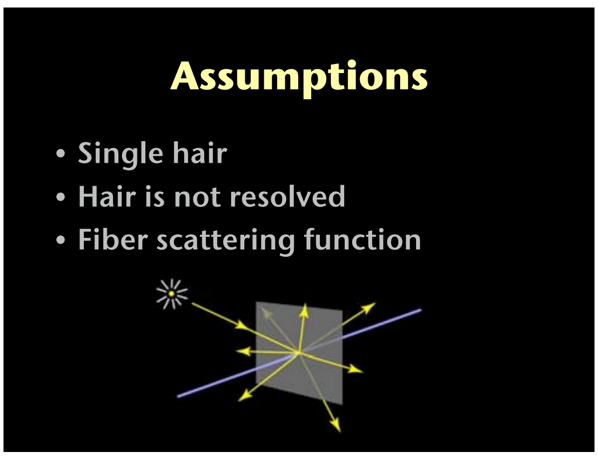
Kajiya-Kay model

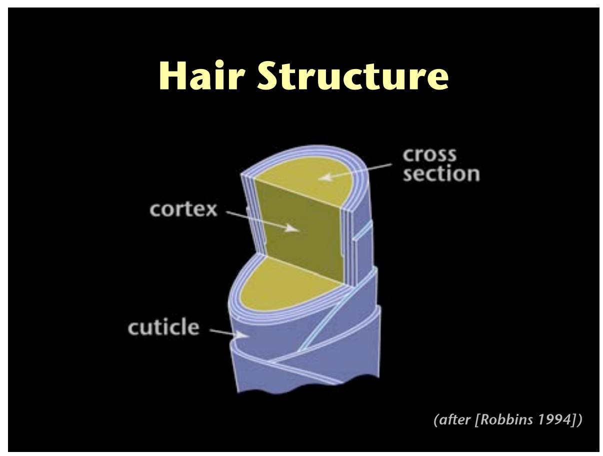


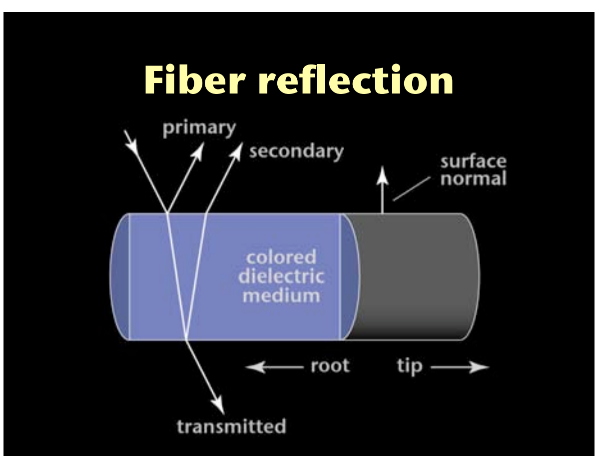
Is hair really like this?

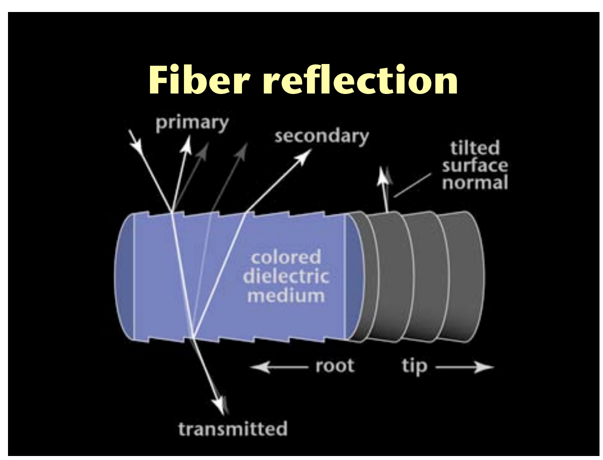

Fiber radiometry
• Fibers are different from surfaces
surface
fiber

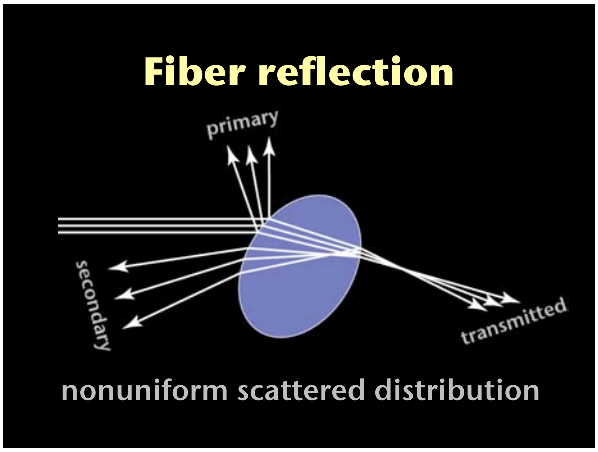

Real Hair
[Robbins 1994]


Assumptions
• Single hair
• Hair is not resolved
• Fiber scattering function


Hair Structure
cross section
cortex
cuticle
(after [Robbins 1994])

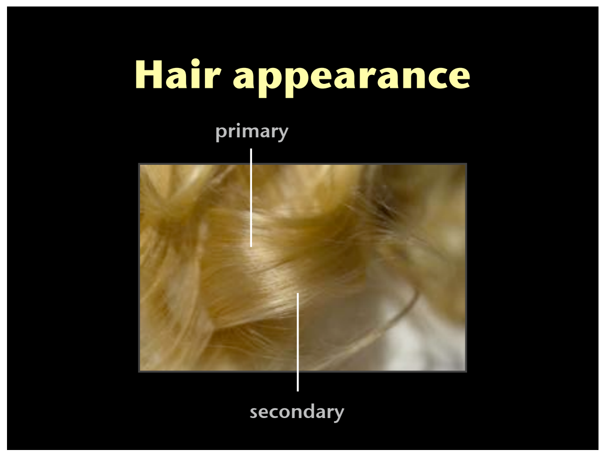

Fiber reflection
primary
secondary
surface normal
colored dielectric medium
root
tip
transmitted

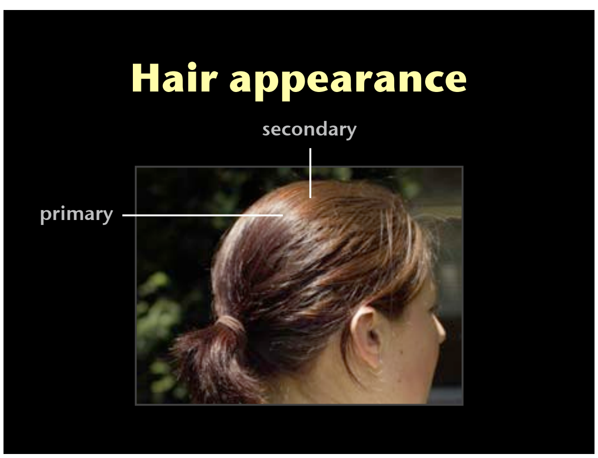

Fiber reflection
primary
secondary
tilted surface normal
colored dielectric medium
root
tip
transmitted


Fiber reflection
primary
secondary
transmitted
nonuniform scattered distribution

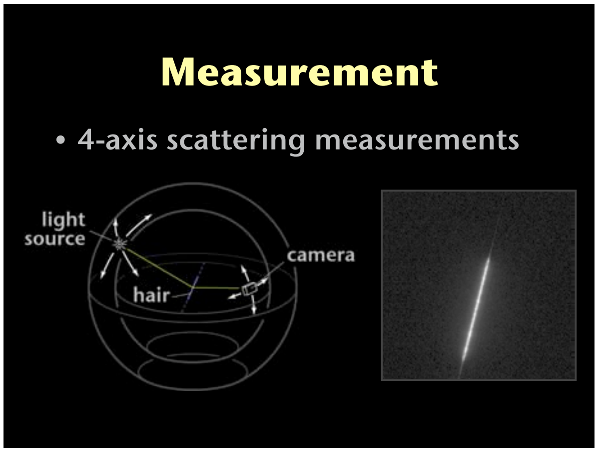

Hair appearance
primary
secondary

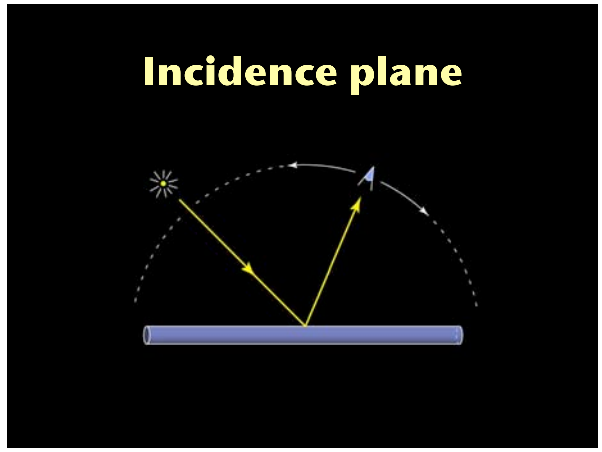

Hair appearance
primary
secondary

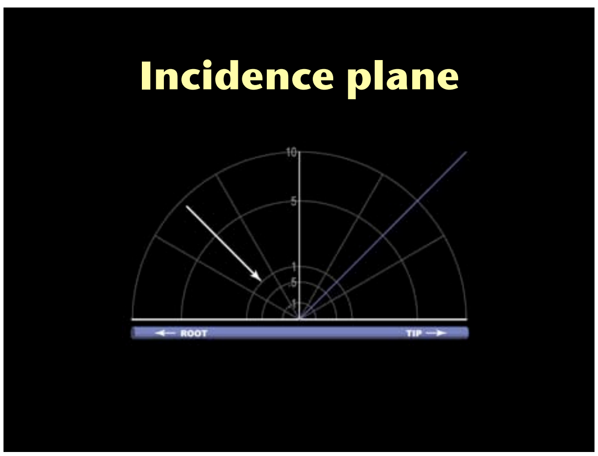

Hair appearance
primary
secondary

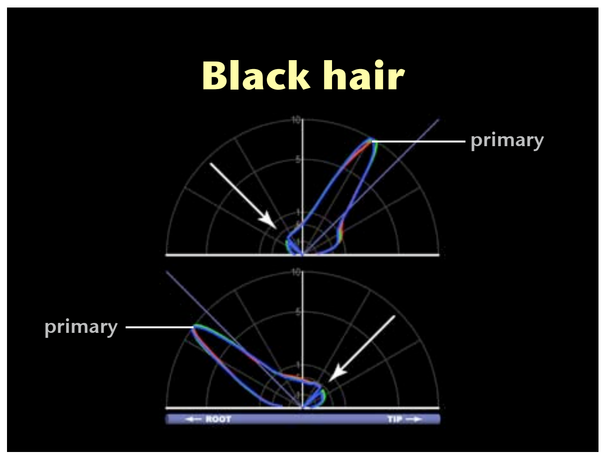

Hair appearance
secondary
primary

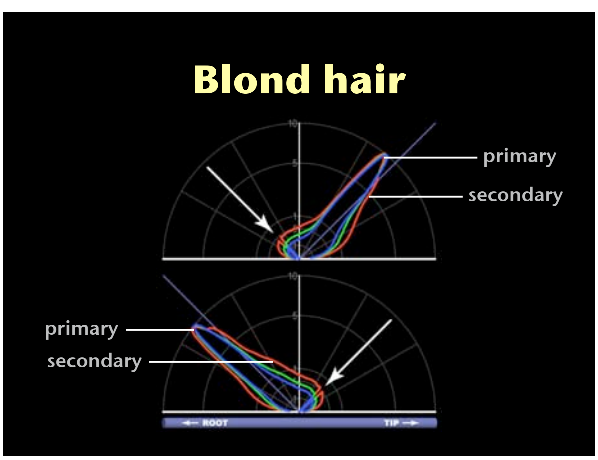

Hair appearance
secondary
glints
(secondary)
primary

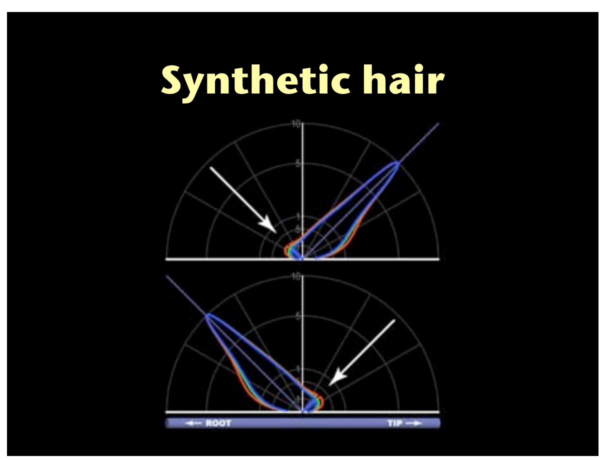

Measurement
• 4-axis scattering measurements
light source
camera
hair


Incidence plane

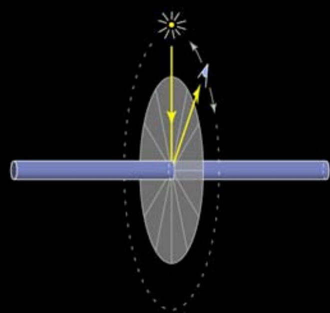

Incidence plane


Black hair
primary
primary

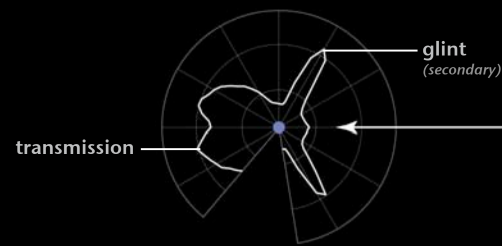

Blond hair
primary
secondary
primary
secondary

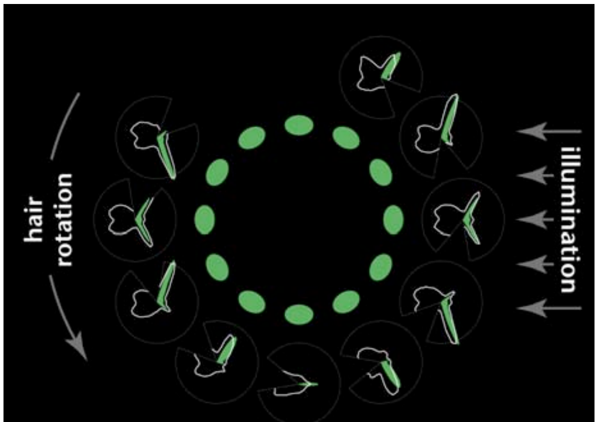

Synthetic hair

Normal plane



Blond hair

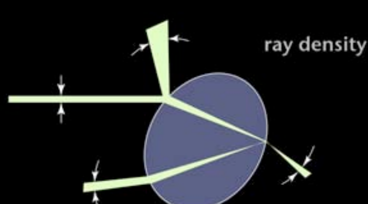




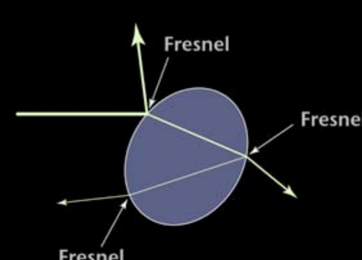
Optical model

- Based on structure of hair
- Explains:
  - double highlight
  - off-axis glints
  - effect of eccentricity
- Core is cross-section scattering

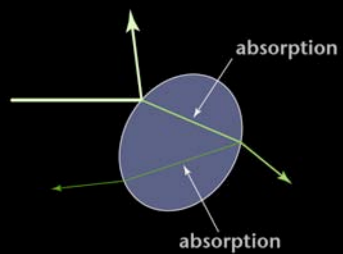
Cross-section scattering



Cross-section scattering



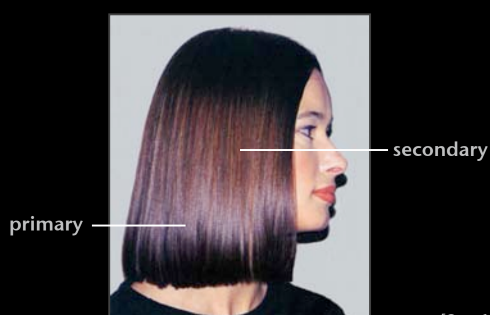
Cross-section scattering



Rendering

- Approximate scattering model
  based on circular fiber + 1st order eccentricity correction
- Ray trace fiber assemblies
- Compare to photograph
- Evaluate against Kajiya-Kay

Photograph

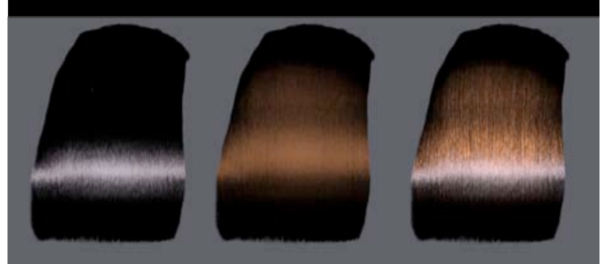


[Gray 1997]

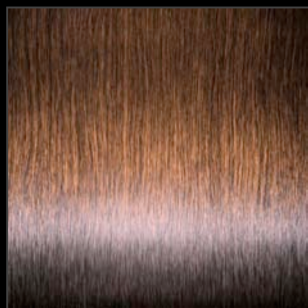
Photo vs. Kajiya-Kay

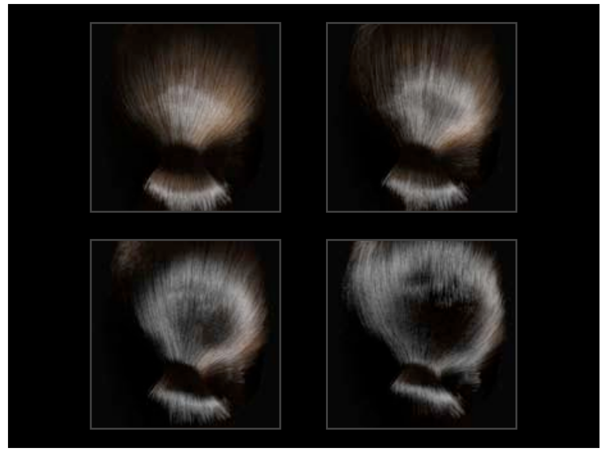
Photo vs. new model

Components of model



Hair texture





SIGGRAPH2005

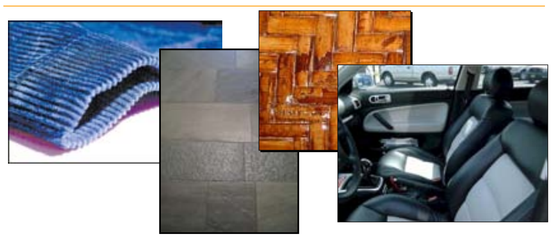
Course 10
Realistic Materials in Computer Graphics

Bidirectional Texture Functions

Gero Müller

University of Bonn

Motivation



- Visualization of complex materials in real-time for interactive applications

---

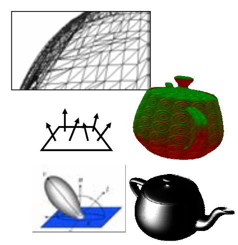
Current modeling approaches in CG

1. Classic hierarchical approach: Modeling on different scales

Macroscopic scale (geometry i.e. triangles)

Mesoscopic scale (textures, bump-maps)

Microscopic scale (BRDF modeling)



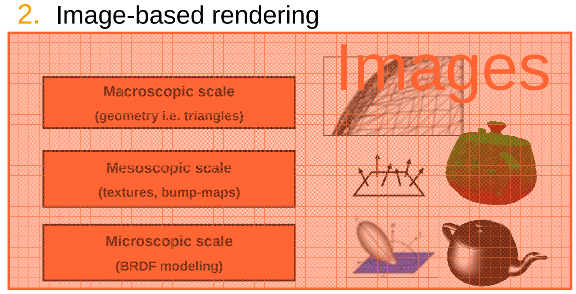
Current modeling approaches in CG

1. Classic hierarchical approach: Modeling on different scales
   - Extremely optimized hardware available
   - Artistic freedom to model almost everything
   - Interaction and alteration of model possible

- BUT: Extremely difficult to model reality accurately

Current modeling approaches in CG

2. Image-based rendering



Current modeling approaches in CG

SIGGRAPH2005

2. Image-based rendering

Render geometry, material detail and reflection behavior from images

- Comparably small effort
  to create images as realistic as photographs
- Very restricted interaction and alteration
- Data explosion

Current modeling approaches in CG

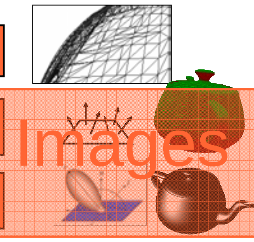
SIGGRAPH2005

3. Rendering materials using measured material data

Macroscopic scale (geometry i.e. triangles)

Mesoscopic scale (textures, bump-maps)

Microscopic scale (BRDF modeling)



Current modeling approaches in CG

SIGGRAPH2005

3. Rendering materials using measured material data

Model Geometry using triangles

Render material from images, measurements (Textures, BTFs, DSRFs, etc.)

- Create photorealistic images
- Complex interaction possible (Relighting, Deformation etc.)

The Bidirectional Texture Function

SIGGRAPH2005

- Measured 6D-slice of Reflectance Field of a texture parameterized over planar surface S

$$BTF(\mathbf{x},\mathbf{v},\mathbf{l}) = \int_S RF_{rgb}(\mathbf{x}_i,\mathbf{x}_r,\mathbf{v},\mathbf{l})d\mathbf{x}_i$$

- Integrates:
  - Occlusions (shadowing, masking)
  - Interreflections and subsurface scattering

General Overview

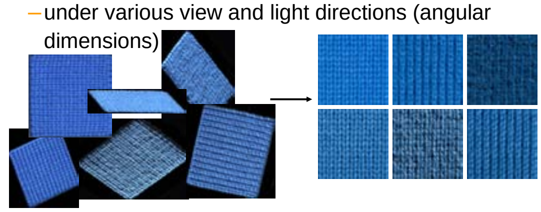
SIGGRAPH2005

- Acquisition
- Compression
- Rendering

BTF Acquisition

SIGGRAPH2005

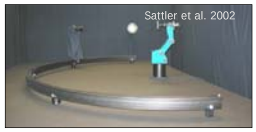
- Sampling a 6D-function
  - Take pictures… (spatial dimension)
  - under various view and light directions (angular dimensions)



Measurement setups

SIGGRAPH2005

- Gonioreflectometer-like
  - Advantages
    - fully automatic, flexible
    - dense sampling
  - Problems
    - measurement time: ~14h
    - moving parts:
      camera, light, sample carrier

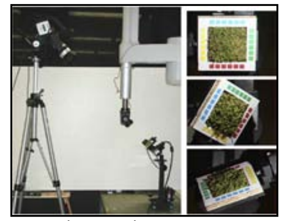


Course 10: Realistic Materials in Computer Graphics — Gero Müller

Measurement setups

SIGGRAPH2005

- Using video cameras (Koudelka et al. 2003)
  - Advantages
    - easy acquisition
    - dense sampling
  - Problems
    - low resolution
    - measurement time: 10h
    - Variation of light direction across the sample

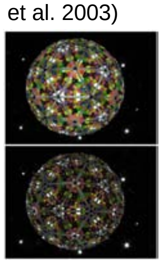


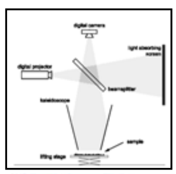
Course 10: Realistic Materials in Computer Graphics — Gero Müller

Measurement setups

SIGGRAPH2005

- Using Mirrors (Han et al. 2003)
  - Advantages
    - Massively parallel
    - Fast
    - No moving parts
  - Problems
    - small resolution
    - non-perfect mirrors



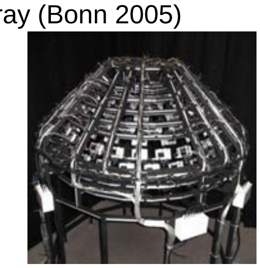


Course 10: Realistic Materials in Computer Graphics — Gero Müller

Measurement setups

SIGGRAPH2005

- Using a camera array (Bonn 2005)
  - Advantages
    - no moving parts
    - acquisition time
      (< 1 hour)
    - dense sampling
  - Problems
    - fixed angular sampling



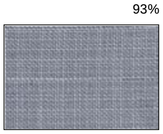
Course 10: Realistic Materials in Computer Graphics — Gero Müller

Gritty Details

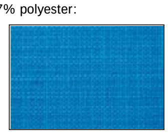
SIGGRAPH2005

- In principle need of spectral measurements
  (see EG2002 STAR on spectral rendering for an overview)

93% wool / 7% polyester:



direct daylight - outdoor



diffused neonlight & daylight - indoor

- Light source
  - daylight-like spectrum
  - in combination with color management system

Course 10: Realistic Materials in Computer Graphics — Gero Müller

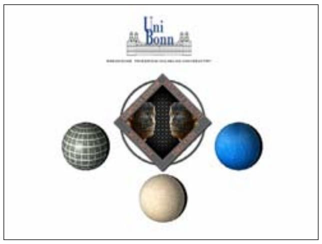
Post-processing

SIGGRAPH2005

- Motivation:
  - Extraction of consistent ROI
  - Texel-to-texel correspondence
  - Compensation for point-light source
- Requirements:
  - Fully automatic
  - Robust
  - Low positioning error (minimize jittering, smoothing)

Course 10: Realistic Materials in Computer Graphics — Gero Müller

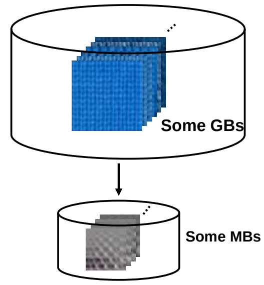
BTF Database Bonn



www.cg.cs.uni-bonn.de/btf

General Overview

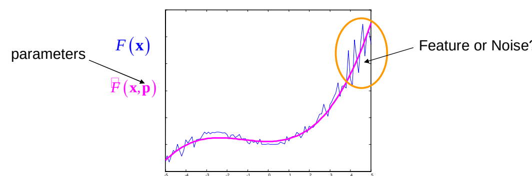
- Acquisition
- Compression
- Rendering



Compression

- Preferable properties:
  - asymmetric compression
    - Fast (real-time), random access decompression
  - preservation of visual important features
  - maximum of a few MBs
    - depends on the current graphics card memory and material complexity of scene

Fitting Analytical Functions



- $|\mathbf{p}|$ small, evaluation of $\tilde{F}(\mathbf{x},\mathbf{p})$ cheap

➔ High compression and fast rendering

Fitting Analytical Functions

- Fixing spatial position

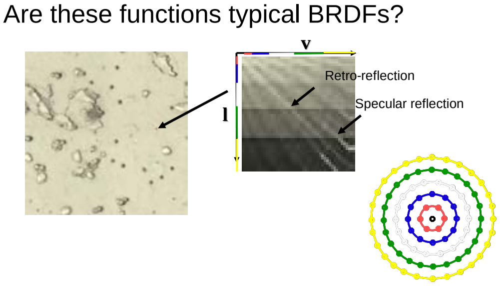
$$B_{\mathbf{x}}(\mathbf{v},\mathbf{l}) := BTF(\mathbf{x},\mathbf{v},\mathbf{l})$$

- 4D-function depending on incoming light direction $\mathbf{l}$ and outgoing view direction $\mathbf{v}$

Apply techniques from BRDF modeling

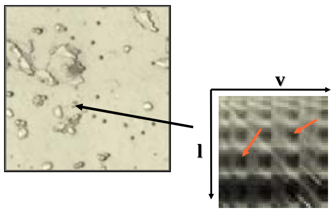
Fitting Analytical Functions

- Are these functions typical BRDFs?



Fitting Analytical Functions

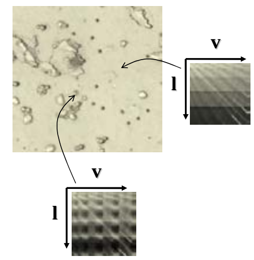
- Are these functions typical BRDFs?



Course 10: Realistic Materials in Computer Graphics — Gero Müller

Fitting Analytical Functions

- ABRDF = Apparent BRDF [Wong et al. 97]
  - Contains also influence from neighborhood:
    - Self-Shadowing
    - Self-Occlusion
    - Sub-Surface Scattering
    - Resampling artefacts
    - …



Course 10: Realistic Materials in Computer Graphics — Gero Müller

Fitting BRDF-Models

- Generalized Cosine-Lobe Model [Lafortune et al. 97] :

$$\Rightarrow BTF(\mathbf{x},\mathbf{v},\mathbf{l}) \approx f_{\mathbf{x}}(\mathbf{v},\mathbf{l},\mathbf{p}) = \rho_{d,\mathbf{x}} + \sum_{i=1}^{k} \rho_{s,\mathbf{x}} \cdot \left(\mathbf{v}^T \mathbf{D}_{\mathbf{x},i} \mathbf{l}\right)^{n_{\mathbf{x},i}}$$

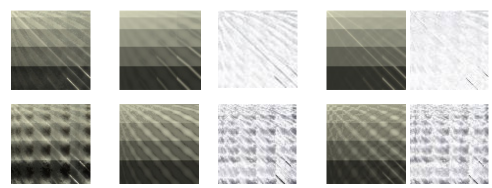
diffuse color — specular color — specular lobe

- Non-linear least-squares fitting (Levenberg-Marquardt)
- $\mathbf{k}$ typically around 2
- Improvement [Daubert et al. 2001]: view-dependent scale factor per texel to account for shadowing effects

Course 10: Realistic Materials in Computer Graphics — Gero Müller

Fitting BRDF-Models

- Results (cosine lobe model)



Course 10: Realistic Materials in Computer Graphics — Gero Müller

Summary: BRDF-Models

- Advantages
  - High compression
  - Simple evaluation
- Problems
  - Limited quality (loss of depth impression)
  - Non-linear fitting
  - expensive
  - depends on initialization

McAllister 2002

Course 10: Realistic Materials in Computer Graphics — Gero Müller

Fitting (Hemi-) Spherical Functions

- Approximate hemispherical slices (fixing e.g. view direction) of per-texel ABRDF separately and blend

$$\Rightarrow BTF(\mathbf{x},\mathbf{v},\mathbf{l}) \approx \sum_{v \in N(\mathbf{v})} w_{\mathbf{v},v} HSF_{\mathbf{x},v}(\mathbf{l})$$

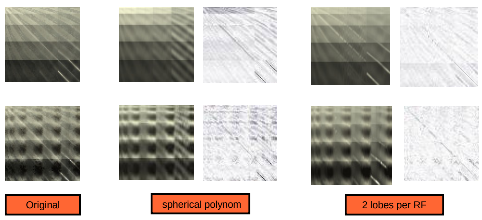
Hemispherical Function (2D) — Interpolation weights

- Meseth et al. used polynomials and cosine lobes
  Sloan et al. used spherical harmonics
  (consider also [Masselus et al. EGSR04])

Course 10: Realistic Materials in Computer Graphics — Gero Müller

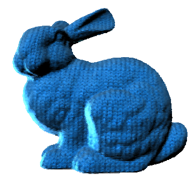
Fitting Hemispherical Functions

- Fitting results



Summary: Fitting Hemispherical Functions

- Advantages
  - Better preservation of ABRDF features
- Problems
  - Chosen function basis may introduce artifacts
  - Memory consuming (Apply clustering $\Rightarrow$ quantization artifacts)
  - View-interpolation required



Meseth et al. 2003

Statistical Data Analysis

- Motivation
  - Assuming general basis functions (polynomials, lobes, etc…) suboptimal for a given measured BTF-dataset
- Idea
  - Find customized basis functions adapted for the actual data set
  - ⇔Learn small set of "features" describing the data set best

Statistical Data Analysis

- In practice:
  - Organize measured data in matrix $\mathbf{M}$
  - Apply a Singular Value Decomposition (SVD) to $\mathbf{M}$
  - Keep only the first c components with highest singular values

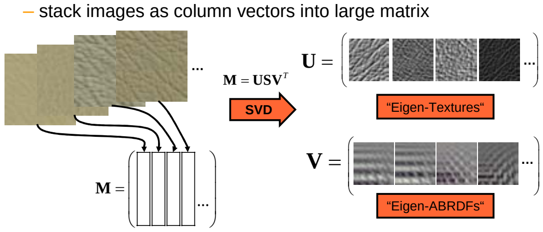
$$\mathbf{M} = \mathbf{USV}^T = \sum_j^N \sigma_j \mathbf{u}_j \mathbf{v}_j^T \approx \tilde{\mathbf{M}} = \sum_j^{c<<N} \sigma_j \mathbf{u}_j \mathbf{v}_j^T$$

→Reconstruction on GPU possible

$\tilde{\mathbf{M}}$ is best rank-c approximation of $\mathbf{M}$ (in the least-squares sense)

Full BTF-Factorization

- Straight forward approach [Koudelka et al. 2003]
  - stack images as column vectors into large matrix



Full BTF-Factorization

- Write the BTF as sum of products of two functions

$$\Rightarrow BTF(\mathbf{x},\mathbf{v},\mathbf{l}) = \sum_j^N \sigma_j \cdot t_j(\mathbf{x}) b_j(\mathbf{v},\mathbf{l}) \approx \sum_j^c \sigma_j \cdot t_j(\mathbf{x}) b_j(\mathbf{v},\mathbf{l})$$

"Eigen-Textures" "Eigen-ABRDFs"

- Note: smooth reconstruction requires interpolation

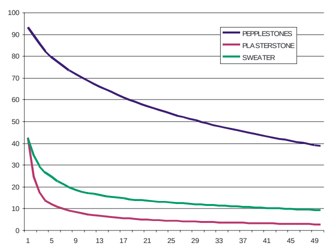
Full BTF-Factorization

SIGGRAPH2005

- Number of terms

RMS-Error

$$\varepsilon = \sqrt{\sum_{i=c+1}^{N} \sigma_i^2}$$

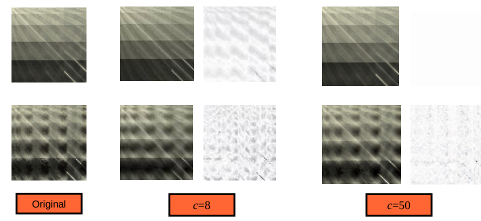


Course 10: Realistic Materials in Computer Graphics | Gero Müller

Full BTF-Factorization

SIGGRAPH2005

- Analysis result



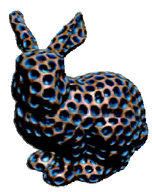
Course 10: Realistic Materials in Computer Graphics | Gero Müller

Summary: Full BTF-Factorization

SIGGRAPH2005

- Advantages
  - simple and straight-forward
  - good compression
- Problems
  - complex materials require many terms

  → not suitable for real-time reconstruction
  - Implementation: SVD of large BTFs



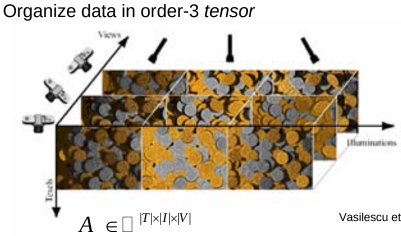
Liu et al. TVCG2004

Course 10: Realistic Materials in Computer Graphics | Gero Müller

Full BTF-Factorization

SIGGRAPH2005

- Multi-Linear SVD [Vasilescu et al. 2004]
  - Organize data in order-3 *tensor*



$$A \in \mathbb{R}^{|T|\times|I|\times|V|}$$

Vasilescu et al. 2004

Course 10: Realistic Materials in Computer Graphics | Gero Müller

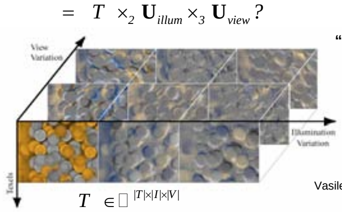
Full BTF-Factorization

SIGGRAPH2005

- Apply SVD to different tensor *modes*

$$A = Z \times_1 \mathbf{U}_{texel} \times_2 \mathbf{U}_{illum} \times_3 \mathbf{U}_{view}$$
$$= T \times_2 \mathbf{U}_{illum} \times_3 \mathbf{U}_{view} ?$$

"Tensor Textures"



$$T \in \mathbb{R}^{|T|\times|I|\times|V|}$$

Vasilescu et al. 2004

Course 10: Realistic Materials in Computer Graphics | Gero Müller

Full BTF-Factorization

SIGGRAPH2005

- „Strategic“ dimensionality reduction (possibility to choose c_1 and c_2 for different tensor dimensions)
- Better perceptual quality than standard SVD despite of worse RMS-error

Vasilescu et al. 2004

Course 10: Realistic Materials in Computer Graphics | Gero Müller

Per-Texel ABRDF Factorization

- Chained-Matrix Factorization [Suykens et al. 2003]
  - Generalization of common BRDF-Factorization techniques [Kautz&McCool 1999],[McCool et al. 2001]
  - Idea:
    - Apply chain of factorizations (SVD) to reparameterized data
    - Each parameterization accounts for certain ABRDF-features

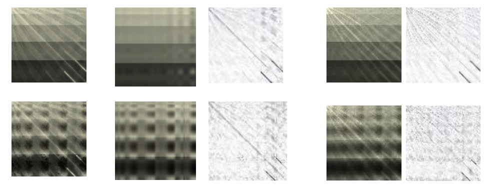
$$BTF(\mathbf{x},\mathbf{v},\mathbf{l}) \approx \prod_j^d \sum_k^{c_j} P_{\mathbf{x},j,k}\left(\pi_{j,1}(\mathbf{v},\mathbf{l})\right)\cdot Q_{\mathbf{x},j,k}\left(\pi_{j,2}(\mathbf{v},\mathbf{l})\right)$$

Different parameterizations for each factor j

Course 10: Realistic Materials in Computer Graphics | Gero Müller

Per-Texel ABRDF Factorization

- Reconstruction Results

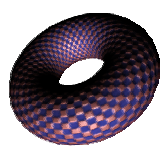


Original | SVD (c=2) | CMF (d=2) GSHD+IO

Course 10: Realistic Materials in Computer Graphics | Gero Müller

Summary: Per-Texel ABRDF Factorization

- Advantages
  - Suitable for real-time rendering: Combination of few factors on GPU
- Problems
  - Resampling artifacts
  - Memory consumption (authors propose clustering of factors ⇒ quantization artifacts)



Suykens et al. 2003

Course 10: Realistic Materials in Computer Graphics | Gero Müller

Per-View Factorization

- Apply SVD to BTF-slices with fixed view direction (Reflectance Fields)
- Idea
  - Increase quality of low-term factored approximations by factorizing subsets of the data

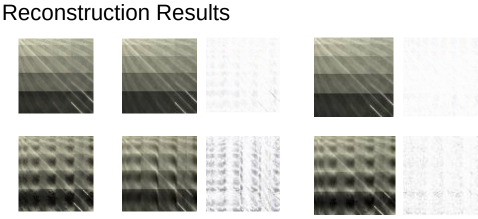
$$\Rightarrow BTF(\mathbf{x},\mathbf{v},\mathbf{l}) \approx \sum_{v\in N(\mathbf{v})} w_v \sum_j^c r_{v,j}(\mathbf{l})\, t_{v,j}(\mathbf{x})$$

"Eigen-Reflectance-Functions" | "Eigen-Textures" (per view)

Course 10: Realistic Materials in Computer Graphics | Gero Müller

Per-View Factorization

- Reconstruction Results



Original | c=3 | c=8

Course 10: Realistic Materials in Computer Graphics | Gero Müller

Summary: Per-View Factorization

- Advantages
  - Low-term factorization enables high-quality interactive rendering on graphics hardware
- Problems
  - Memory consumption
  - Coherence between different views not exploited

Sattler et al. 2003

Course 10: Realistic Materials in Computer Graphics | Gero Müller

Per-Cluster Factorization

SIGGRAPH2005

- Factorize variable number of arbitrarily shaped BTF-slices (clusters)
- Clustering BTF-texels (ABRDFs) leads to

$$\Rightarrow BTF(\mathbf{x},\mathbf{v},\mathbf{l}) \approx \sum_{j}^{c} t_j(\mathbf{x}) b_{k(\mathbf{x}),j}(\mathbf{v},\mathbf{l})$$

"Eigen-Textures" (segmented)

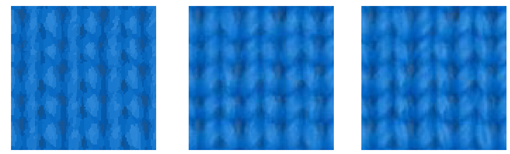
"Eigen-BRDFs" (per cluster)

- Clustering with generalized Lloyd-algorithm and re-construction error as distance metric (Local PCA)

Per-Cluster Factorization

SIGGRAPH2005

- Factorization per cluster reduces clustering artifacts

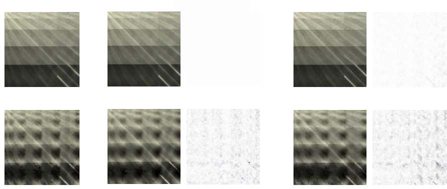


c=1 c=3 c=8

Per-Cluster Factorization

SIGGRAPH2005

- Reconstruction Results

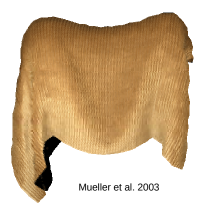


Original | *k=1, c=50* | *k=32, c=8*

Summary: Per-Cluster Factorization

SIGGRAPH2005

- Advantages
  - Low-term factored representation suitable for GPU implementation
  - Good compression
  - Reconstruction per cluster reduces quantization artifacts
- Problems
  - Expensive fitting



Mueller et al. 2003

Storage Requirements

SIGGRAPH2005

Model	Storage (|L|,|V|,|T|)	|V|=|L|=81, |T|=256² 8-Bit per channel
Raw BTF	|L|*|V|*|T|	1.2 GB
Analytical BRDF-Model	*f*(k)*|T|	(k=2) 2.4 MB
Hemispherical Function	*f*(k)*|V|*|T|	(k=2) 95 MB
BTF Factorization	c*(|V|*|L| + |T|)	(c=40) 8.6 MB
ABRDF Factorization	d*(|V|+|L|)*|T|	(d=2) 63 MB
Per-View Factorization	|V|*c*(|L|+|T|)	(c=4) 64 MB
Per-Cluster Factorization	c*(k*(|V|*|L|) + |T|)	(k=32, c=8) 6.6 MB

General Overview

SIGGRAPH2005

- Acquisition
- Compression
- Rendering

Rendering

SIGGRAPH2005

- Determine color / visible radiance for every point

„Exitant Radiance = Emitted Rad. + Reflected Rad.“

$$L_r(\mathbf{x},\mathbf{v}) = L_e(\mathbf{x},\mathbf{v}) + L_{ref}(\mathbf{x},\mathbf{v})$$

„Reflected Rad. = Incoming Rad. combined with reflection properties“

$$L_{ref}(\mathbf{x},\mathbf{v}) = \int_{\Omega_i} \rho_{\mathbf{x}}(\mathbf{v},\mathbf{l}) \cdot L_i(\mathbf{x},\mathbf{l}) \cdot \langle \mathbf{n}|\mathbf{l}\rangle \, d\mathbf{l}$$

Software Rendering

SIGGRAPH2005

- Raytracing
  - Implement decompression-stage on CPU
  - Integrate into favorite raytracing engine

Created with OpenRT from InTrace (Saarbrücken)

Model courtesy of DaimlerChrysler

Software Rendering

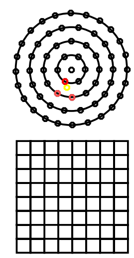
SIGGRAPH2005

- Global Illumination

General Implementation Issues

SIGGRAPH2005

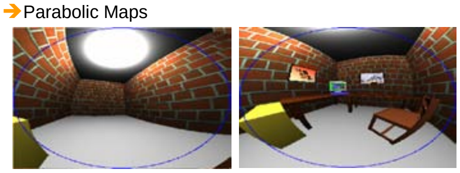
- Measured BTFs
  - Evaluation for directions not in the measured set
    - Interpolation in angular domain
  - Interpolation rather expensive
    - ➔graphics hardware
    - ➔Interpolation from regular samples



Hardware Supported Interpolation

SIGGRAPH2005

- Reparameterization
  - Approximately uniform sampling of hemisphere
  - Suitable for hardware filtering
  - ➔Parabolic Maps

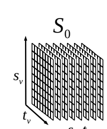


Heidrich + Seidel 1998

Hardware Supported Interpolation

SIGGRAPH2005

- 2D Data
  - Bilinear filtering on graphics hardware
- 4D Eigen-ABRDFs
  - Quadrilinear filtering
  - Hardware: trilinear filtering
  - ➔Trilinear filtering of s_v,t_v,s_l
    - ➔3D textures S_l for fixed t_l
  - ➔Manual interpolation of t_l



Anti-Aliasing

SIGGRAPH2005

- Mip-Mapping compressed BTFs
  - In general BTF depends non-linear on compression parameters
  - Standard Mip-Mapping not possible

➔ Mip-Mapping on uncompressed data

➔ Compression of each individual Mip-Map level

Point- and Directional Light Sources

SIGGRAPH2005

- Finite number of light directions

$$L_{ref}(\mathbf{x},\mathbf{v}) = \int_{\Omega_i} \rho_{\mathbf{x}}(\mathbf{v},\mathbf{l}) \cdot L_i(\mathbf{x},\mathbf{l}) \cdot \langle \mathbf{n} | \mathbf{l} \rangle d\mathbf{l}$$

$\downarrow$

$$L_{ref}(\mathbf{x},\mathbf{v}) = \sum_{i=1}^{n} \rho_{\mathbf{x}}(\mathbf{v},\mathbf{l}_i) \cdot \hat{L}_i(\mathbf{x},\mathbf{l}_i) \cdot \langle \mathbf{n} | \mathbf{l}_i \rangle$$

$\downarrow$ BTF with foreshortening term included

$$L_{ref}(\mathbf{x},\mathbf{v}) = \sum_{i=1}^{n} BTF(\mathbf{x},\mathbf{v},\mathbf{l}_i) \cdot \hat{L}_i(\mathbf{x},\mathbf{l}_i)$$

BRDF Models per Texel

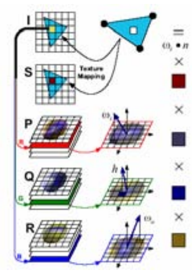
SIGGRAPH2005

- Store parameters in textures
- Evaluation of model in pixel shader
- Optional: cluster look-up
- Hemispherical function:
  - Implement view-interpolation

Factorization Methods

SIGGRAPH2005

- ABRDF-Factorization
  - Store scaled 2D-factors in parabolic maps
  - Optional: cluster look-up
  - Combine in pixel shader

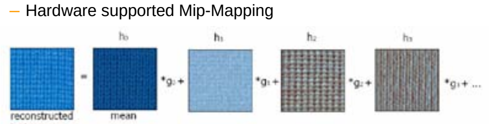


Suykens et al. 2003

Factorization Methods

SIGGRAPH2005

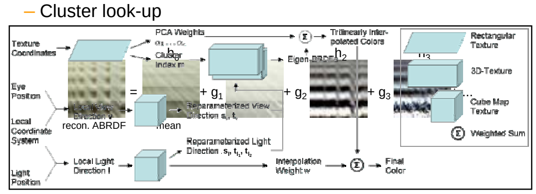
- Per-View Factorization
  - Combine weighted Eigen-Textures on GPU
  - Light-interpolation by computing interpolated weights on the CPU per vertex
  - View-interpolation using multi-pass rendering (alpha-blending)
  - Hardware supported Mip-Mapping



Factorization Methods

SIGGRAPH2005

- Full-BTF Factorization/Per-Cluster Factorization
  - Store 4D ABRDFs in 3D texture
  - Use 4D interpolation and combine in pixel shader
  - Cluster look-up



Results

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics Gero Müller

Image-Based Lighting

SIGGRAPH2005

- Environment Map
  - Standard, high-dimensional representation as pixels

Course 10: Realistic Materials in Computer Graphics Gero Müller

Image-Based Lighting

SIGGRAPH2005

- Infinitely distant Lighting Environment
  - No spatial dependence

$$L_{ref}(\mathbf{x},\mathbf{v}) = \int_{\Omega_i} \rho_{\mathbf{x}}(\mathbf{v},\mathbf{l}) \cdot L_i(\mathbf{x},\mathbf{l}) \cdot \langle \mathbf{n}|\mathbf{l}\rangle d\mathbf{l}$$

$$\downarrow$$

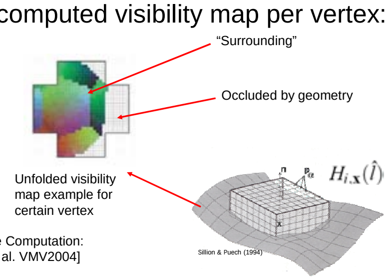
$$L_{ref}(\mathbf{x},\mathbf{v}) = \int_{\Omega_i} \rho_{\mathbf{x}}(\mathbf{v},\mathbf{l}) \cdot L_i(\mathbf{l}) \cdot \langle \mathbf{n}|\mathbf{l}\rangle d\mathbf{l}$$

Course 10: Realistic Materials in Computer Graphics Gero Müller

Per-View Factorization

SIGGRAPH2005

- Precomputed visibility map per vertex:



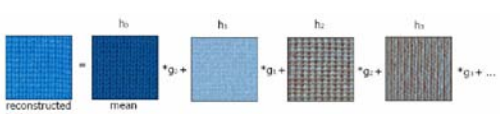
Real-Time Computation:
[Sattler et al. VMV2004]

Course 10: Realistic Materials in Computer Graphics Gero Müller

Per-View Factorization

SIGGRAPH2005

- Integrate incoming radiance over hemisphere including visibility information on the CPU per vertex
- Combine and interpolate weighted Eigen-Textures on GPU



Course 10: Realistic Materials in Computer Graphics Gero Müller

Per-View Factorization

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics Gero Müller

Image-Based Lighting with Spherical Harmonics

SIGGRAPH2005

- SH encoded Lighting Environment

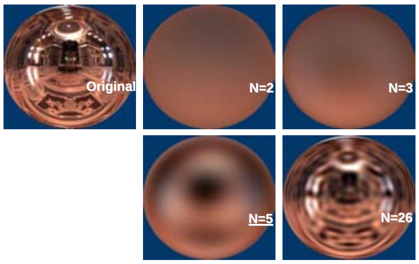


Image-Based Lighting with Spherical Harmonics

SIGGRAPH2005

- Extensions: precompute shadows and inter-reflections (Precomputed Radiance Transfer)

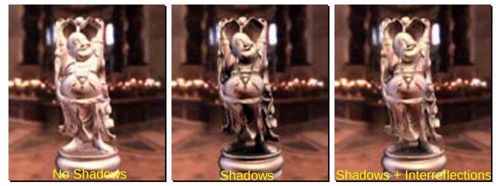


Image-Based Lighting of BTFs

SIGGRAPH2005

- Radiance Transfer Texture (RTT)
  [Sloan et al. 2003]
  - Extend per-vertex Precomputed Radiance Transfer to handle BTFs
  - Encode every texel in SH-Basis
  - Lighting dot-products in pixel shader
  - SH-encoded BTF requires about 300 MB per 256^2 pixel BTF

Image-Based Lighting of BTFs

SIGGRAPH2005

- Clustered Precomputed RTT
  [Müller et al. 2004]
  - Compress both SH-encoded BTF and precomputed transfer using clustered factorization
  - Storage size independent of BTF size and geometry size
  - High quality setting:
    - 16.9 MB
  - Good quality:
    - 4.2 MB

Image-Based Lighting of BTFs

SIGGRAPH2005

HDR environments wood, beach, kitchen, building and uffizi from www.debevec.org

Conclusions

SIGGRAPH2005

- BTFs capture 6D-slice of the reflectance field of a complex material
- Represents the “look-and feel” of a material
- Several high-quality acquisition setups
- Effective and appearance preserving compression algorithms available
- Real-time rendering possible with point light sources and image-based lighting

SIGGRAPH2005

Course 10
Realistic Materials in Computer Graphics

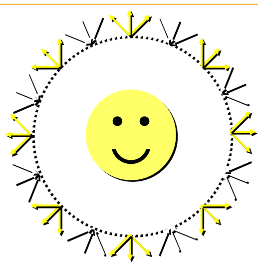
Reflectance Fields

Tim Hawkins

USC Institute for Creative Technologies

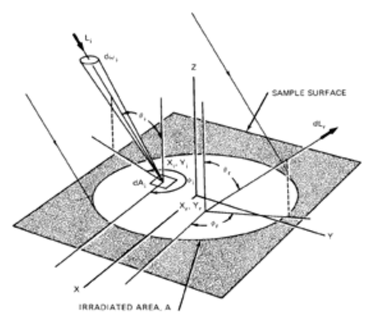
The Reflectance Field
(Debevec et al, 2000)

- Describes how an object transforms an incident light field to a radiant light field

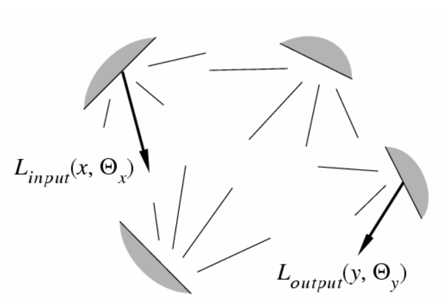


$R_i(u_i, v_i, \theta_i, \phi_i\ ;\ u_r, v_r, \theta_r, \phi_r)$

BSSRDF
(Nicodemus et al, 1977)



GRDF
(Lafortune et al, 1996)

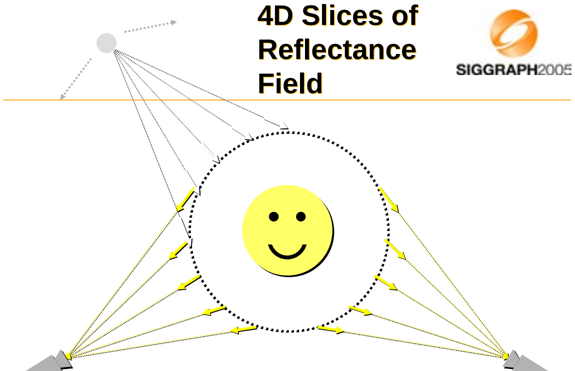


Reflectance Field

- Simple geometry, just a sphere
- Complex reflectance, 8 dimensional
- Rendering becomes “convolution” of incident light field with reflectance field, yielding outgoing light field—ray tracing and visibility calculations are not needed

4D Slices of Reflectance Field

SIGGRAPH2005



$R_i(u_i,v_i,\theta_i,\phi_i\ ;\ u_r,v_r,\theta_r,\phi_r)$

SIGGRAPH2005

Light Stage 1

(Debevec et al, 2000)

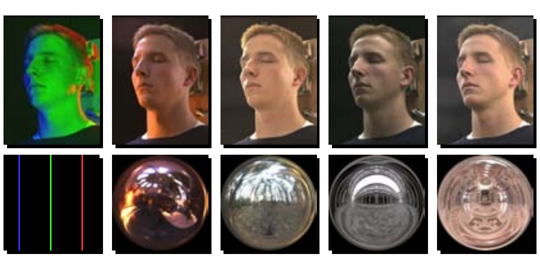
Light Stage Data

Original Resolution: 64x32

Lighting through image recombination: Haeberli '92, Nimeroff '94, Wong '97

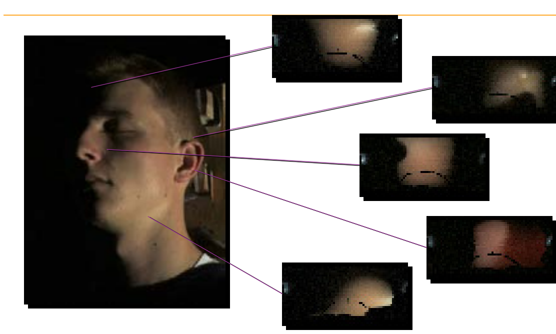
Relighting Results

SIGGRAPH2005



Reflectance Functions

SIGGRAPH2005



Light Stage 2
(Hawkins et al, 2004)

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics — Tim Hawkins

Light Stage 2

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics — Tim Hawkins

Six Viewpoints

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics — Tim Hawkins

Captured Expressions

SIGGRAPH2005

Course 10: Realistic Materials in Computer Graphics — Tim Hawkins

Animatable Model

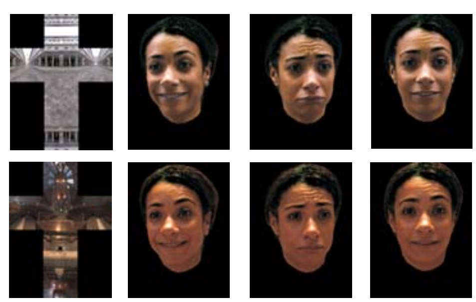
SIGGRAPH2005

- Registering all expressions and views gives a morphable model across expression, viewpoint, and lighting
- Rough geometric model for view interpolation
- Raw data size is 60 expressions x 6 views x 480 lights x 1 MB per image = 160 GB

Course 10: Realistic Materials in Computer Graphics — Tim Hawkins

Results

SIGGRAPH2005



Course 10: Realistic Materials in Computer Graphics — Tim Hawkins

Opacity Hull
(Matusik et al, 2002)

SIGGRAPH2005

- Rotating arc of lights
- Static arc of cameras
- Rotating platform
- Two plasma monitors
- Captures 6D reflectance field + rough geometry + opacity

Opacity Hull

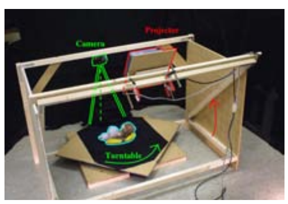
SIGGRAPH2005



Relighting with Incident Light Fields (Masselus et al, 2003)

SIGGRAPH2005

- Captures a basis for illumination by any incident light field



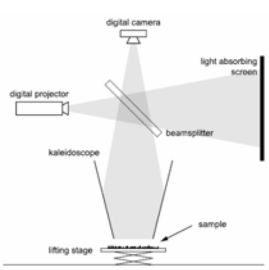
Relighting with Incident Light Fields

SIGGRAPH2005


- Relit with synthetic incident light fields

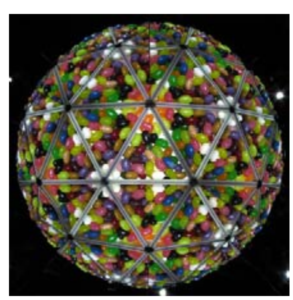
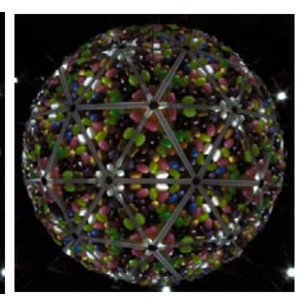
Kaleidoscopic Reflectometry (Han and Perlin, 2003)

SIGGRAPH2005



Kaleidoscopic Reflectometry (Han and Perlin, 2003)

SIGGRAPH2005

Conclusion

SIGGRAPH2005

- Reflectance fields provide a unifying framework for many reflectance capture techniques
- Full 8 dimensional capture remains daunting

Course 10: Realistic Materials in Computer Graphics   Tim Hawkins