Efficient, Image-Based Appearance Acquisition of Real-World Objects

Hendrik Peter Asmus Lensch

Max-Planck-Institut für Informatik
Saarbrücken, Germany

Dissertation zur Erlangung des Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing)
der Naturwissenschaftlich-Technischen Fakultät I
der Universität des Saarlandes

Eingereicht am 24. April 2003 in Saarbrücken.

Betreuender Hochschullehrer — Supervisor
Prof. Dr. Hans-Peter Seidel, MPI für Informatik, Saarbrücken, Germany

Gutachter — Reviewers
Prof. Dr. Hans-Peter Seidel, MPI für Informatik, Saarbrücken, Germany
Prof. Dr. Pat Hanrahan, Stanford University, United States
Prof. Dr. Dieter W. Fellner, Technische Universität Braunschweig, Germany

Dekan — Dean
Prof. Dr. Philipp Slusallek, Universität des Saarlandes, Saarbrücken, Germany

Datum des Kolloquiums — Date of Defense
15. Dezember 2003

Hendrik Peter Asmus Lensch
Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken, Germany
lensch@mpi-sb.mpg.de

für Viola

Abstract

Two ingredients are necessary to synthesize realistic images: an accurate rendering algorithm and, equally important, high-quality models in terms of geometry *and* reflection properties. In this dissertation we focus on capturing the appearance of real world objects. The acquired model must represent both the geometry and the reflection properties of the object in order to create new views of the object with novel illumination. Starting from scanned 3D geometry, we measure the reflection properties (BRDF) of the object from images taken under known viewing and lighting conditions. The BRDF measurement require only a small number of input images and is made even more efficient by a view planning algorithm. In particular, we propose algorithms for efficient image-to-geometry registration, and an image-based measurement technique to reconstruct spatially varying materials from a sparse set of images using a point light source. Moreover, we present a view planning algorithm that calculates camera and light source positions for optimal quality and efficiency of the measurement process. Relightable models of real-world objects are requested in various fields such as movie production, e-commerce, digital libraries, and virtual heritage.

Kurzfassung

Zur Synthetisierung realistischer Bilder ist zweierlei nötig: ein akkurates Verfahren zur Beleuchtungsberechnung und, ebenso wichtig, qualitativ hochwertige Modelle, die Geometrie *und* Reflexionseigenschaften der Szene repräsentieren. Die Aufnahme des Erscheinungbildes realer Gegenstände steht im Mittelpunkt dieser Dissertation. Um beliebige Ansichten eines Gegenstandes unter neuer Beleuchtung zu berechnen, müssen die aufgenommenen Modelle sowohl die Geometrie als auch die Reflexionseigenschaften beinhalten. Ausgehend von einem eingescannten 3D-Geometriemodell, werden die Reflexionseigenschaften (BRDF) anhand von Bildern des Objekts gemessen, die unter kontrollierten Lichtverhältnissen aus verschiedenen Perspektiven aufgenommen wurden. Für die Messungen der BRDF sind nur wenige Eingabebilder erforderlich. Im Speziellen werden Methoden vorgestellt für die Registrierung von Bildern und Geometrie sowie für die bildbasierte Messung von variierenden Materialien. Zur zusätzlichen Steigerung der Effizienz der Aufnahme wie der Qualität des Modells, wurde ein Planungsalgorithmus entwickelt, der optimale Kamera- und Lichtquellenpositionen berechnet. Anwendung finden virtuelle 3D-Modelle bespielsweise in der Filmproduktion, im E-Commerce, in digitalen Bibliotheken wie auch bei der Bewahrung von kulturhistorischem Erbe.

Summary

One central problem in computer graphics is synthesizing realistic images that are indistinguishable from real photographs. The basic theory behind rendering such images has been known for a while and has been turned into a broad range of rendering algorithms ranging from slow but physically accurate frameworks to hardware-accelerated, real-time applications that make a lot of simplifications. One fundamental building block to these algorithms is the simulation of the interaction between incident illumination and the reflective properties of the scene. The limiting factor in photo-realistic image synthesis today is not the rendering per se but rather modeling the input to the algorithms. The realism of the outcome depends largely on the quality of the scene description passed to the rendering algorithm. Accurate input is required for geometry, illumination and reflective properties. An efficient way to obtain realistic models is through measurement of scene attributes from real-world objects by *inverse rendering*. The attributes are estimated from real photographs by inverting the rendering process.

Work on acquisition of realistic 3D objects is described in this dissertation. The first algorithm is devoted to a high-precision registration of input images to a scanned 3D geometry model of the object. This automatic method is based on the silhouette of the object observed in the images but also considers texture information. The registration maps the images as textures onto the geometry in such a way that fine detail present in multiple images is precisely aligned. The resulting texture represents the object's appearance under fixed illumination.

One way to achieve realistic rendering under novel viewing and lighting conditions requires measuring the reflection properties of the surface (technically, the *bidirectional reflectance distribution function* or BRDF). It is necessary to reproduce variation in the diffuse and in the specular part of the BRDF across the surface. *Image-based BRDF measurement* estimates these properties from real images of the object from various view points under different illumination conditions. In a new algorithm we capture spatially varying BRDFs from a small set of input images using a point light source, and excluding other sources of illumination from the environment. Reliable reflection properties are obtained by fitting a BRDF model to measured samples of whole clusters of surface points belonging to the same material. Spatial variation in the diffuse and the specular part is recovered as a per-pixel linear combination of cluster BRDFs.

The quality of the estimated BRDF parameters depends on the selected camera and light source positions. We measure the quality as the uncertainty of the parameter estimation with respect to the input images. By analyzing the uncertainty it is possible to determine viewing and lighting directions that are optimal for the measurement. We integrate these insights into a view planning algorithm that captures reflection properties of real-world objects more efficiently than an

unplanned series of input images. The planning algorithm guides experts and non-experts alike through the acquisition process and ensures that the BRDF is captured with almost the same quality for all surface points.

In conclusion, we have developed several techniques to capture the appearance of real-world objects. Spatially varying per-pixel reflection properties are recovered from images in an efficient acquisition process resulting in realistic, relightable models of high quality.

Zusammenfassung

Ein zentrales Problem in der Computergraphik ist die Berechnung realistischer Bilder, die von einer Photographie kaum zu unterscheiden sind. Die grundlegende Theorie zur Berechnung solcher Bilder ist seit langem bekannt und führte zur Entwicklung einer Reihe von Algorithmen zur Beleuchtungsberechnung. Diese reichen von langsamen, aber genauen, physikalisch fundierten Methoden bis hin zu Hardware-beschleunigten Verfahren, die viele vereinfachende Annahmen treffen. Die Simulation von Reflexionen an Oberflächen ist dabei wesentlicher Bestandteil dieser Verfahren. Der beschränkende Faktor bei der Berechnung photorealistischer Bilder ist heutzutage weniger die Beleuchtungberechnung an sich als vielmehr die Modellierung der Eingabedaten. Der Realitätsnähe der erzeugten Bilder wird hauptsächlich von der Qualität der Szenenbeschreibung beeinflusst, die als Eingabe dient. Genaue Angaben über die Geometrie, die einfallende Beleuchtung und die Reflexionseigenschaften der zu berechnenden Szene sind nötig. Diese Attribute können mit *bildbasierten* Verfahren effizient aus Photographien ermittelt werden. Realistische Modelle von realen Gegenständen werden bestimmt, indem man den Syntheseprozess umkehrt (engl. *inverse rendering*). Die vorliegende Arbeit beschreibt Verfahren zur Akquisition realistischer 3D-Modelle.

Der erste Algorithmus ermöglicht eine sehr genaue Registrierung der Eingabebilder zu gemessenen 3D-Geometriemodellen. Der Algorithmus berücksichtigt dabei sowohl die Silhouette des Objekts im Bild als auch die ermittelte Textur-Information. Durch die Registrierung können die Bilder so genau als Textur auf die Geometrie abgebildet werden, dass auch sehr feine Strukturen aus unterschiedlichen Bildern nach der Projektion übereinstimmen. Die berechnete Textur repräsentiert das Erscheinen des Objekts unter den festen Beleuchtungsverhältnissen, die während der Aufnahme gewählt wurden.

Um beliebige Ansichten des Objekts unter beliebiger Beleuchtung korrekt darstellen zu können, müssen die Reflexionseigenschaften in Form der so genannten BRDF (engl. *bidirectional reflectance distribution function*) gemessen werden. Da die Reflexionseigenschaften auf der Oberfläche beliebig variieren können, ist es notwendig, Änderungen sowohl im diffusen als auch im spekularen Anteil der BRDF zu berücksichtigen. Mit Hilfe von bildbasierten BRDF-Messungen werden die Reflexionseigenschaften an Hand von Bildern des Objekts gemessen, die unter kontrollierten Lichtverhältnissen aus verschiedenen Ansichten aufgenommen wurden. In einem neuen Verfahren wird die BRDF heterogener Objekte aus wenigen Aufnahmen gemessen, bei denen eine einzige Punktlichtquelle das Objekt beleuchtet. Durch die Anpassung eines BRDF-Modells an alle Messwerte von Oberflächenpunkten, die zu dem selben Basismaterial gehören, werden die Reflexionseigenschaften zuverlässig bestimmt. Räumliche Änderungen der Reflexionseigenschaften werden als Linearkombination der Basis-BRDFs für jeden

Punkt einzeln ausgedrückt.

Die zur Aufnahme ausgewählten Positionen von Kamera und Lichtquelle beeinflussen die Qualität der berechneten BRDF. Wir bestimmen die Qualität durch Messung der Unsicherheit der ermittelten BRDF-Parameter in Abängigkeit von den Eingabebildern. Die Analyse der Unsicherheit erlaubt somit die Berechnung optimaler Blick- und Lichtrichtungen für die Messung. Hierauf basierend wurde ein Planungsalgorithmus entwickelt, mit dessen Hilfe die Reflexionseigenschaften effizienter gemessen werden können als durch eine ungeplante Aufnahmeserie. Der Planungsalgorithmus unterstützt den Anwender während des Aufnahmeprozesses und stellt sicher, dass die BRDF für alle Oberflächenpunkte des Objekts in annähernd gleicher Qualität bestimmt wird.

Zusammenfassend wurden mehrere Methoden entwickelt, um das Erscheinungsbild realer Objekte zu akquirieren. Durch einen effizienten, bildbasierten Aufnahmeprozess werden für jeden Oberflächenpunkt Reflexionseigenschaften gemessen. Die resultierenden Modelle erlauben, Ansichten des virtuellen Objekts unter beliebiger Beleuchtung in hoher Qualität zu synthetisieren.

Acknowledgements

My first thanks go to my supervisor Prof. Dr. Hans-Peter Seidel, for his support during my stay at the Max-Planck-Institut, his valuable comments, and his interest and promotion of this work. He provided an excellent research environment and surrounded me with the people and technological equipment that made this work possible.

Prof. Dr. Pat Hanrahan kindly agreed to serve as an external reviewer, which I am very grateful for. Furthermore, I would like to thank Prof. Dr. Dieter Fellner, both for acting as reviewer and for coordinating the DFG Schwerpunktprogramm V^3D^2 which partly funded this dissertation.

I would especially like to thank Prof. Dr. Wolfgang Heidrich who aroused my interest in starting my dissertation in computer graphics. He directed me towards BRDF measurements and guided me in the first year of my Ph.D.

The projects shaping the basis of this thesis would have been impossible without the co-operation, discussion and steady motivation by my colleagues of the graphics group at MPI. Their bundled knowledge, competence and creativity provide a wonderful research environment. I cannot name all of them, but I would especially like to thank the following people (in alphabetical order): Thomas Annen, Volker Blanz, Stefan Brabec, Katja Daubert, Michael Goesele, Jan Kautz, Jochen Lang, Marcus Magnor, Karol Myszkowski, Christian Rössl, Annette Scheel, Philipp Slussalek, Hartmut Schirmacher, Christian Theobalt, Jens Vorsatz, and Christel Weins. Furthermore, I would like to thank Marco Tarini who stayed at the MPI for six months, for his creative way of thinking, and Asla M. Sá working with us for two months.

Building up and maintaining a photo studio for measurement purposes requires a lot of support from non-scientists. Michael Laise and Axel Köppel always provided prompt help for many practical problems neither of us has faced before. Thanks go also to Sabine Budde for managing all bureaucratic issues.

Financial support for parts of this dissertation has been provided by three institutions, being the Max-Planck-Gesellschaft zur Förderung der Wissenschaften e.V., and the Deutsche Forschungsgemeinschaft under a subproject in the DFG-Schwerpunktprogramm Nr. 1041 V^3D^2 “Verteilte Verarbeitung und Vermittlung digitaler Dokumente”. Furthermore, the work in this thesis has contributed to the initiation of the ViHAP3D Project IST-2001-32641 “Virtual Heritage: High-Quality 3D Acquisition and Presentation” funded by the European Community.

Finally, I shall not forget the most important never-failing support I received during the last three years. I thank Viola and my family for their help whenever I needed them in order to complete this dissertation.

Contents

Chapter 1
Introduction

Synthesizing images of photo-realistic quality is a long standing goal in *computer graphics*. In order to reach this goal various physics-based rendering algorithms have been developed. They simulate light propagation through an environment based on mathematical models which describe how light interacts with matter.

The realism of the outcome of these rendering algorithms is however highly dependent on the quality of the provided scene description. 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. Hundreds of model designers and animators were involved in the productions of fully computer generated films like Toy Story[1], Shrek[2] or Final Fantasy[3].

An alternative approach to create realistic models is to capture models of real world objects. The field of estimating models from observations has so far mainly been investigated in *computer vision*. In the last decade, the requirements of computer graphics applications pushed the field towards the acquisition of high-quality 3D models. The recent availability of high-precision 3D scanners and high-resolution digital cameras makes the acquisition of these realistic models possible.

The digitization of real word objects is of increasing 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. In the context of cultural heritage, for example, the captured 3D models can serve to digitally preserve an artifact, to document and guide the restauration process, and to present art to a wide audience via the Internet.

[1] Walt Disney
[2] DreamWorks
[3] Square

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

Figure 1.1: *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).*

Depending on the application different physical properties of the object may be acquired, for example the shape of the object, the deformation of the object in reaction to applied forces, its sound, or the visual appearance of its surface. The major focus of this dissertation is on the *appearance acquisition*: This is to faithfully capture the object's reflection properties including the subtle details of the object's 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.

In this thesis, we present a set of new techniques and algorithms that capture high-quality, compact 3D models of real world objects from photographs. Special attention is paid to reconstructing the appearance based on only a small set of images which increases the efficiency of the acquisition process and leads to a relatively simple acquisition setup.

The acquisition and representation of objects including their appearance is addressed by *image-based modeling and rendering* techniques. In most of these approaches the object is represented by a very large set of images showing the object from different viewing directions usually under different illumination. The large number of images lengthen the acquisition, and they can only be recorded with an automatic setup. In contrast to that we want to capture a high-quality model from a rather sparse set of input images.

In Figure 1.1 we demonstrate the capabilities of a high-quality digitization of an object. Starting from accurate 3D geometry, the task necessary for obtaining a realistic model consists of measuring and representing the complex reflection properties of the object. Recovering the reflection properties allows us to render the object from novel view points and under novel light conditions resulting in

a correctly shaded view of the object. The reflection properties are represented by a so-called *bidirectional reflectance distribution function* (BRDF). We have to consider variation of the BRDF across the object's surface since within one object the reflection properties may change rapidly from one point to the next. Furthermore, the variation is not restricted to the diffuse color of the object but must include the specular reflectance as well. Otherwise, the bronze appearance including patina and polished areas could not have been reproduced.

Efficiency in the acquisition process and in the representation is mandatory to make the technique applicable even for a collection of numerous objects: the reconstructed model has to be small, the acquisition process has to be not too time-consuming, and the appearance has to be captured for all points on the object's surface with the same quality.

These aspects of the acquisition of realistic models are addressed in this thesis. The individual problems are matched by a set of different techniques and algorithms developed during the course of the dissertation.

1.1 Main Contributions

Parts of the outcome of this dissertation have already been published in scientific articles at different conferences and journals [Lensch00, Goesele00, Lensch01c, Lensch01a, Lensch03a, Lensch03b]. The developed techniques have been further presented in tutorials and courses at various international conferences [Lensch01e, Lensch01b, Scopigno02, Lensch02a]. This thesis builds on these publication but also includes yet unpublished work. The main contribution of this dissertation are:

- A description of the setup of a digital photo studio especially suited for the task of object acquisition and image-based BRDF measurements.
- An algorithm for the registration of 2D images to a 3D mesh based on the silhouette of the object. This registration algorithm is used all throughout the following approaches to obtain mappings between pixels of the input images to surface locations.
- An image-based BRDF measurement technique capturing spatially varying reflection properties from a sparse set of images in a controlled environment. Isotropic spatially varying BRDFs are recovered allowing variation both in the diffuse *and* in the specular part.
- A view planning algorithm based on parameter uncertainty minimization which selects the best next position for the camera and a point light source in order to obtain reliable BRDFs sampled evenly across the surface.

The above list of technical contribution presents a development from rather rough approximations of the appearance by a diffuse texture to the acquisition of more general reflection properties.

1.2 Chapter Overview

This thesis starts with a presentation of the necessary background material in Chapter 2, followed by a classification and review of related work on 3D object acquisition in Chapter 3. The setup of the photo studio is described in Chapter 4.

The algorithm presented in Chapter 5 performs hardware-accelerated image-to-texture registration. In Chapter 6, we introduce an image-based BRDF measurement technique using a point light source. The acquisition planning algorithm is described in Chapter 7. We conclude this thesis in Chapter 8 where we discuss the pros and cons of the developed techniques and raise still open issues.

Chapter 2
Background

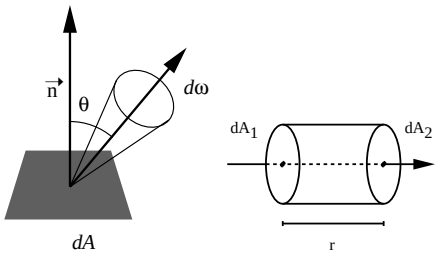
The way an object is observed depends on different factors, for example the incident light and the object's reflectance properties. Reconstruction and representation of objects, including their reflection properties, is the main focus of this thesis. We want to describe how the object interacts with light incident from a given environment. It is therefore necessary to understand the physical underpinnings of light transport. Simulating and measuring reflection properties additionally requires a model for the interaction of light with surfaces that can be evaluated efficiently. A detailed overview of light transport and light/surface interactions can be found in a review by Glassner [Glassner95].

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.

2.1 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 a number of photons shot from a light sources and traced 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 inde-



(a) (b)

Figure 2.1: *Geometrical properties of radiance.*

pendent 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 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.

2.2 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 $[W]$.

$$\Phi = \frac{dQ}{dt}. \tag{2.1}$$

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 be expressed by the radiant flux:

$$L(\vec{x}, \hat{\omega}) = \frac{d^2\Phi}{\cos\theta d\omega dA} \quad , \tag{2.2}$$

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 2.1(a).

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 2.1(b) 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

$$L_1 d\omega_1 dA_1 \quad = \quad L_2 d\omega_2 dA_2. \tag{2.3}$$

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

$$d\omega_1 = \frac{dA_2}{r^2} \quad \text{and} \quad d\omega_2 = \frac{dA_1}{r^2}. \tag{2.4}$$

Inserting this into Equation 2.3 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. The radiance reflected by a visible surface point is what will be observed when an image of a scene is taken.

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:

$$E(\vec{x}) \quad = \quad \frac{d\Phi_i}{dA} = \int_{\Omega^+} L_i(\vec{x}, \hat{\omega}_i) \cos\theta_i d\omega_i \quad \text{or} \tag{2.5}$$

$$B(\vec{x}) \quad = \quad \frac{d\Phi_o}{dA} = \int_{\Omega^+} L_o(\vec{x}, \hat{\omega}_o) \cos\theta_i d\omega_o, \tag{2.6}$$

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

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

$$I := \frac{d\Phi}{d\omega} \tag{2.7}$$

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

$$L_i = \frac{I}{r^2}. \tag{2.8}$$

With these quantities we are able to describe the radiant energy falling on or leaving a surface.

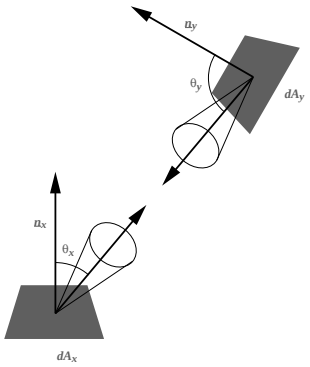


Figure 2.2: *Geometrical properties of the energy transport*

2.3 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:

$$L_o(\vec{x}, \hat{v}^{(g)}) = L_e(\vec{x}, \hat{v}^{(g)}) + \int_S f_r(\vec{x}, \hat{l}, \hat{v}) G(\vec{x}, \vec{y}) V(\vec{x}, \vec{y}) L_o(\vec{y}, \hat{l}^{(g)}) dA_y \quad (2.9)$$

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 2.2:

$$G(\vec{x}, \vec{y}) = \frac{\cos\theta_x \cos\theta_y}{\|\vec{x} - \vec{y}\|^2} \quad (2.10)$$

The integral in Equation 2.9 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:

$$L_o(\vec{x}, \hat{v}^{(g)}) = L_e(\vec{x}, \hat{v}^{(g)}) + \int_{\Omega^+} f_r(\vec{x}, \hat{l}, \hat{v}) L_i(\vec{x}, \hat{l}^{(g)})(\hat{n} \cdot \hat{l}^{(g)}) d\hat{l} \tag{2.11}$$

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.

Based on Equation 2.9 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:

$$L_o(\vec{x}, \hat{v}^{(g)}) = \int_{\Omega^+} f_r(\vec{x}, \hat{l}, \hat{v}) V(\vec{x}, \hat{l}^{(g)}) L_{env}(\hat{l}^{(g)})(\hat{n} \cdot \hat{l}^{(g)}) d\hat{l} \tag{2.12}$$

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 another 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 2.12 is reduced to a lookup of the radiance incident from the mirror direction. For some glossy BRDFs it has been shown that Equation 2.12 can be precomputed for all outgoing directions resulting in prefiltered environment map [Miller84, Greene86, Cabral99, Heidrich99a, Kautz00a, Latta02, McAllister02b]. Ramamoorthi and Hanrahan [Ramamoorthi01a, Ramamoorthi02] 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

$$L_o(\vec{x}, \hat{v}^{(g)}) = \sum_{j=0}^{n} f_r(\vec{x}, \hat{l}_j, \hat{v}) V(\vec{x}, \hat{l}_j^{(g)}) \cdot \frac{I_j}{r_j^2} (\hat{n} \cdot \hat{l}_j^{(g)}), \tag{2.13}$$

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.

2.4 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 characterize 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:

$$E(\vec{x}, \hat{v}) = \left(L_{observed}(\vec{x}, \hat{v}) - \int_{\Omega^+} f_r(\vec{x}, \hat{l}, \hat{v}) L_i(\vec{x}, \hat{l}^{(g)}) (\hat{n} \cdot \hat{l}^{(g)}) d\hat{l} \right). \tag{2.14}$$

Shape from Shading/Photometric Stereo

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, Nayar90a, 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, Nishino01b, 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 thesis are centered around inverse reflectometry. More details and related work are discussed in Chapter 3.

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. [Nishino01b] and Ramamoorthi and Hanrahan [Ramamoorthi01b] reconstructed both the material properties and the incident lighting at the same time for a given geometric model. In Section 6.8 we present an algorithm that determines the BRDF and the surface normal at the same time.

2.5 Interaction of Light with Matter

In order to simulate light interacting with surfaces or to measure appearance one has to represent the surfaces' reflective properties. A taxonomy of object appear-

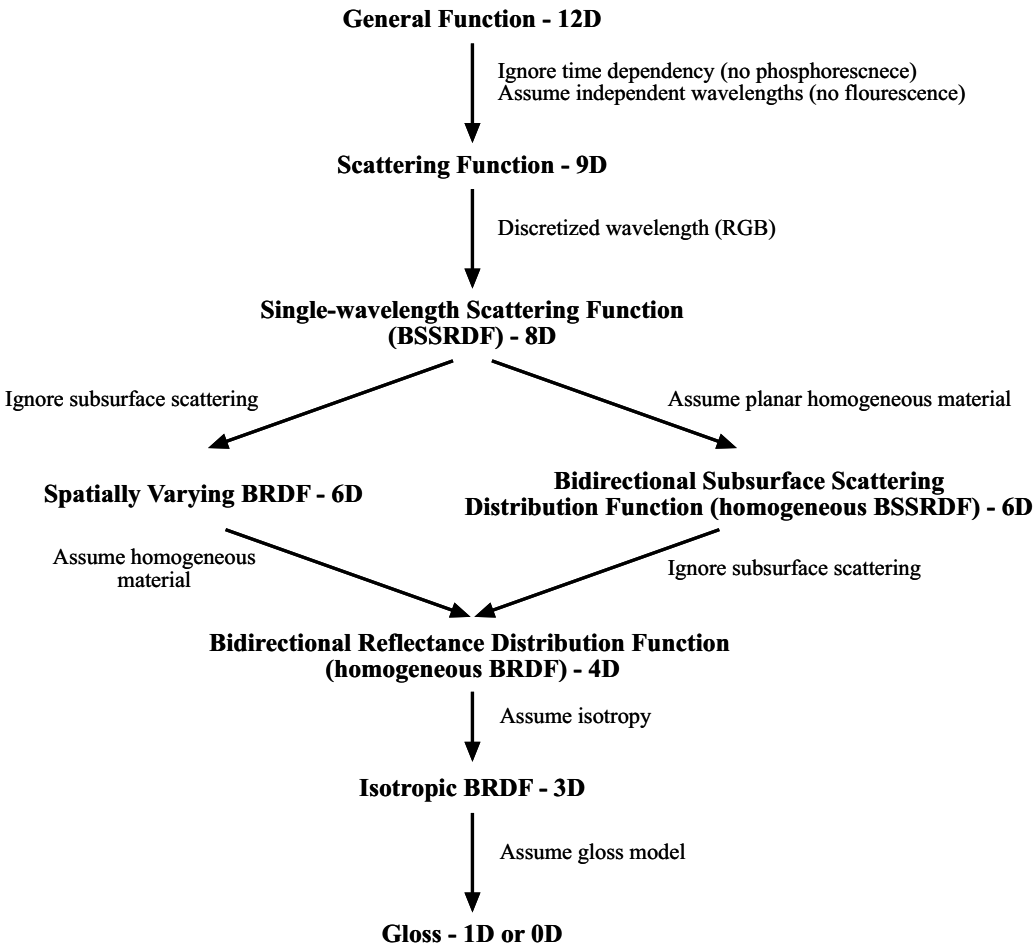


Figure 2.3: *Taxonomy of appearance measurement adapted from [Rusinkiewicz00]. The dimension of the most general description of light interacting with matter is reduced by adding more and more assumptions resulting in manageable functions like the BRDF.*

ance descriptions with different levels of abstraction is shown in Figure 2.3. 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:

$$\begin{aligned} (x, y, \theta, \phi, t, \lambda)_{in} &\rightarrow (x, y, \theta, \phi, t, \lambda)_{out}, \quad \text{or} \\ (\vec{x}_i, \hat{\omega}_i, t_i, \lambda_i) &\rightarrow (\vec{y}_o, \hat{\omega}_o, t_o, \lambda_o) \end{aligned} \tag{2.15}$$

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 *flourescence*. 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 subsurface scattering 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). This function is presented in more detail in the next section since it is the main representation for surface appearance used in this thesis. 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.

2.5.1 Bidirectional Reflectance Distribution Functions

The reflection properties of opaque surfaces can be fully specified by a *bidirectional reflection distribution function* (BRDF), which describes how incident light is reflected off the surface. The BRDF has the unit $[1/sr]$:

$$f_r(\vec{x}, \hat{\omega}_i \rightarrow \hat{\omega}_o) := \frac{dL_o(\vec{x}, \hat{\omega}_o)}{dE(\vec{x}, \hat{\omega}_i)} = \frac{dL_o(\vec{x}, \hat{\omega}_o)}{L_i(\vec{x}, \hat{\omega}_i) \cos\theta_i d\omega_i}. \tag{2.16}$$

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 $f_r(\vec{x}, \theta_i, \theta_o, \phi_0 - \phi_i)$. In the remainder of this thesis we will sometimes implicitly assume spatial variation and use the alternative writing $f_r(\hat{\omega}_i, \hat{\omega}_o)$ 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. Transparency and translucency are not handled by the measurement techniques presented in this thesis, hence we will mainly restrict ourselves to BRDFs.

2.5.2 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:

$$f_r(x, \hat{\omega}_i \rightarrow \hat{\omega}_o) = f_r(x, \hat{\omega}_o \rightarrow \hat{\omega}_i). \tag{2.17}$$

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:

$$\int_{\Omega^+} f_r(x, \hat{\omega}_i \rightarrow \hat{\omega}_o) \cos\theta_i d\omega_o \ \leq \ 1 \qquad \forall \quad \hat{\omega}_i \in \Omega^+. \tag{2.18}$$

Snell's Law

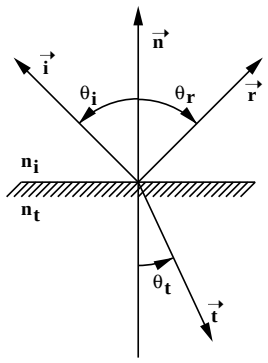


Figure 2.4: *Light ray reflection and refraction at an ideally smooth surface.*

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:

$$n(\lambda) = \frac{c}{v_\lambda}. \tag{2.19}$$

Figure 2.4 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:

$$\vec{r} = 2 < \vec{i} \mid \vec{n} > -\vec{i}. \tag{2.20}$$

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:

$$n_i(\lambda) \sin\theta_i = n_t(\lambda) \sin\theta_t. \tag{2.21}$$

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

$$\vec{t} = -\frac{n_i}{n_t}\vec{i} + \vec{n}\left(\frac{n_i}{n_t}\cos\theta_i - \cos\theta_t\right). \tag{2.22}$$

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.

$$\rho = \frac{d\Phi_r}{d\Phi_i} \qquad \text{and} \qquad \tau = \frac{d\Phi_t}{d\Phi_i}. \tag{2.23}$$

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

$$r^\perp = \frac{n_1\cos\theta_i - n_2\cos\theta_t}{n_1\cos\theta_i + n_2\cos\theta_t}, \qquad r^\parallel = \frac{n_2\cos\theta_i - n_1\cos\theta_t}{n_2\cos\theta_i + n_1\cos\theta_t}, \tag{2.24}$$

$$t^\perp = \frac{2n_1\cos\theta_i}{n_1\cos\theta_i + n_2\cos\theta_t}, \qquad t^\parallel = \frac{2n_1\cos\theta_i}{n_2\cos\theta_i + n_1\cos\theta_t}. \tag{2.25}$$

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:

$$\rho = \frac{(r^\perp)^2 + (r^\parallel)^2}{2}, \quad \text{and} \tag{2.26}$$

$$\tau = \frac{n_2\cos\theta_i}{n_1\cos\theta_t}\cdot\frac{(t^\perp)^2 + (t^\parallel)^2}{2} = 1 - \rho - \alpha. \tag{2.27}$$

In Figure 2.5 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

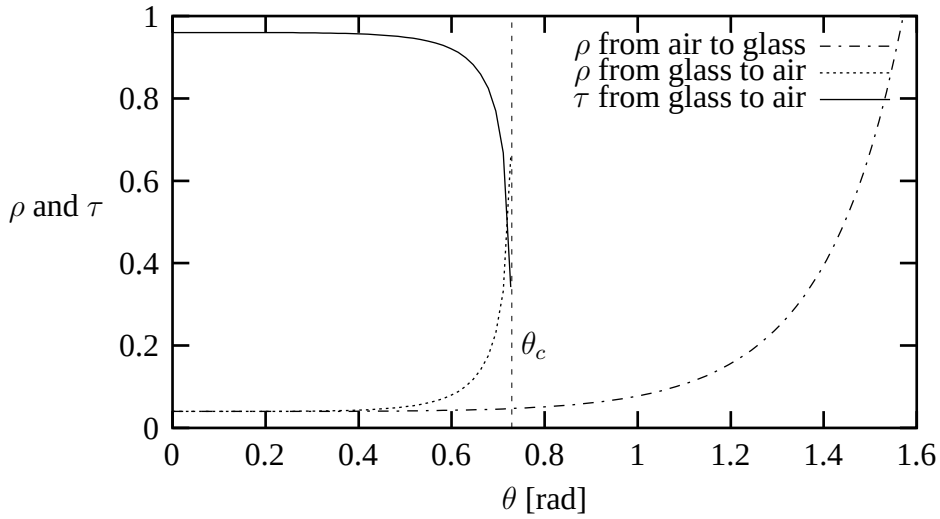


Figure 2.5: *Reflectance ρ and transmittance τ, determined by Fresnel's formulae at a surface that separates air $(n \approx 1)$ and glass $(n = 1.5)$. The critical angle is $\theta_c \approx 0.73$.*

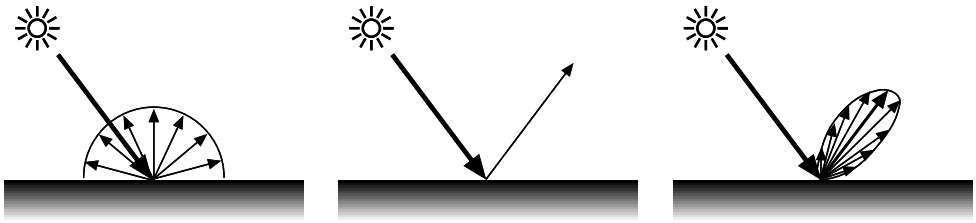
glass to air, the transmittance τ will be zero if the incident angle θ_i exceeds the critical angle θ_c. This critical angle is defined as the angle where the incident ray will be refracted in a direction parallel to the surface. Thus it is given by:

$$n_1 \sin\theta_c = n_2 \sin\frac{\pi}{2}, \quad \text{that is} \quad \theta_c = \arcsin\frac{n_1}{n_2}. \tag{2.28}$$

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

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 2.7). 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:

$$F_\lambda(\hat{h} \cdot \hat{v}) = f_\lambda + (1 - f_\lambda)(1 - (\hat{h} \cdot \hat{v}))^5 \tag{2.29}$$



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

Figure 2.6: *Qualitative description of reflection at opaque surfaces*

2.6 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 heterogenous. These attributes form three almost orthogonal classification schemes. Figure 2.7 demonstrates the appearance various different materials.

2.6.1 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

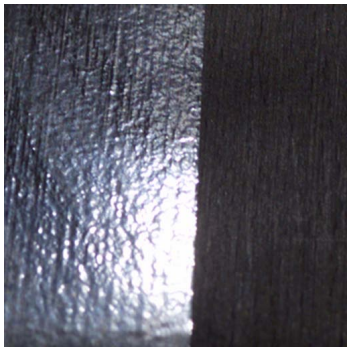
At *opaque* surfaces light is reflected directly at the surface. Examples for opaque materials are metals, chalk, and to some extent wood and plastic. There is no significant amount of light penetrating an opaque surface. A light beam illuminating an opaque surface from one direction is partially absorbed 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.

(a)

(b)

(c)

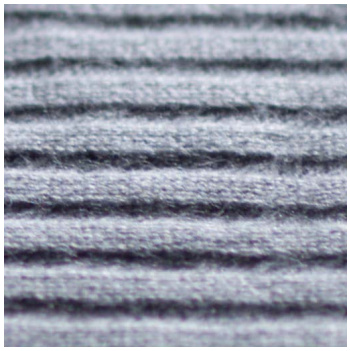
(a) Diffuse Reflection. No view-depedent highlight is visible. (b) Mirror Reflection. A clear image of the light source is reflected. (c) Glossy Reflection. An extended highlight region caused by a glossy surface.



(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 2.7: *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.*

Diffuse reflections. Some portion of light is scattered equally into all directions (Figure 2.6a) 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 2.6b).

Glossy reflections. If the surface is not perfectly smooth, light will be scattered around the mirror direction, called *glossy* reflection (Figure 2.6c). 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 2.5.2). 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

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

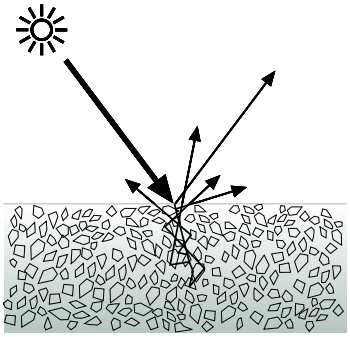


Figure 2.8: *Pigment Particles. Many surfaces obtain their diffuse color from pigment particles. Light is either reflected directly at the surface of the embedding material causing a white highlight or penetrates the surface and is diffused and filtered by the particles.*

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

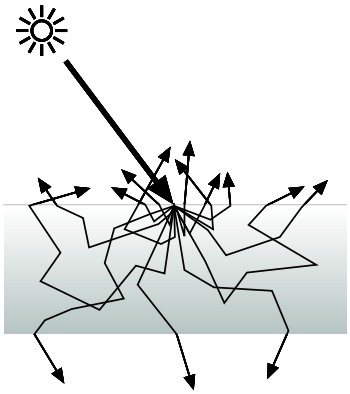


Figure 2.9: *Subsurface Scattering. For translucent materials light penetrates the surface and is scattered (multiple times) inside the material before it leaves the material probably at some other position.*

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 2.9). 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 been investigated in [Hanrahan93, Koenderink96b, Koenderink01, Jensen01, Lensch02c].

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

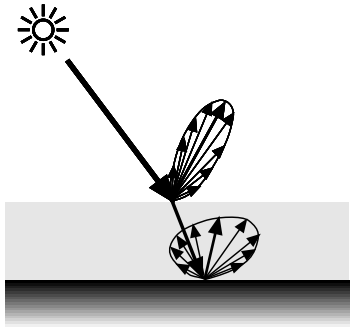


Figure 2.10: *Layered Materials. The appearance of layered materials is determined by the reflection properties at the different surfaces.*

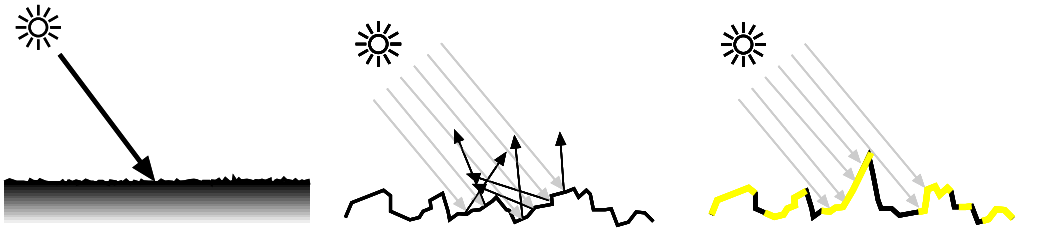
As the name suggests, layered materials consists of layers of different materials. Materials formed by multiple layers show additional effects if at least the top level is partially translucent. Typical examples are coated surfaces, like varnished wood or a thin film of oil on skin. In a simple case 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 2.10. 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.

2.6.2 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



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

Figure 2.11: *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.*

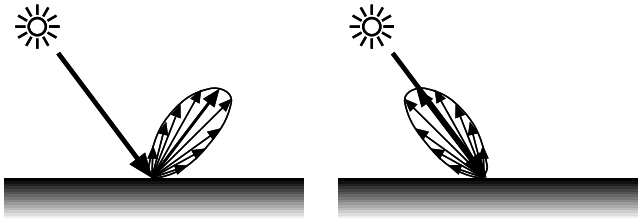
The structure of many rough surfaces can be described as a collection of a large number of so-called *microfacets* which are too small to be resolved (see Figure 2.11). Each microfacet acts as a mirror and may possess an arbitrary orientation. Given an incident light direction, the reflected radiance 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 2.12: *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 2.12d). 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 animals. Some traffic signs and cinema screens have a backward scattering layer to make better use of the incident light.

2.6.3 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

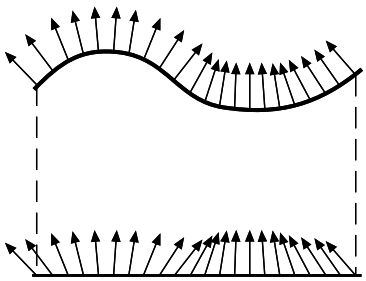


Figure 2.13: *Normal Map. The characteristic shape of a curved surface is transformed into a normal map defined on a flat surface. At each texel the original surface normal is stored.*

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 2.13).

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, Lensch02b, Daubert02].

2.6.4 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 form 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.

2.7 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.

2.7.1 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.

2.7.2 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, Ramamoorthi02, Sloan02, Kautz02].

2.7.3 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 as it is been done in this thesis 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 $\hat{h}$:

$$f_r(\hat{l}, \hat{v}) = \frac{\rho_d}{\pi} + \frac{\rho_s}{(\hat{n} \cdot \hat{l})}(\hat{h} \cdot \hat{n})^N. \tag{2.30}$$

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:

$$f(\hat{l}, \hat{v}) = \frac{\rho_d}{\pi} + \sum_i [C_{x,i}(l_x v_x) + C_{y,i}(l_y v_y) + C_{z,i} l_z v_z]^{N_i}. \tag{2.31}$$

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 $C_{z,i}$. We make use of the anisotropic Lafortune model in Chapter 6 and 7.

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 [Ashikhmin00a] 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 nor a Fresnel term:

$$f_r(\hat{l},\hat{v}) := \frac{\rho_d}{\pi} + \rho_s \frac{1}{\sqrt{(\hat{l}\cdot\hat{n})(\hat{v}\cdot\hat{n})}} \frac{1}{4\pi\alpha_x\alpha_y} \cdot \exp\left(-2 \frac{\left(\frac{\hat{h}\cdot\hat{t}}{\alpha_x}\right)^2 + \left(\frac{\hat{h}\cdot\hat{b}}{\alpha_y}\right)^2}{1+\hat{h}\cdot\hat{n}} \right). \quad (2.32)$$

The anisotropy is indicated by different values for α_x and α_y defined along the tangent $\hat{t}$ and bi-normal $\hat{b}$ 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] emulate 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.

Appearance of objects in this thesis is modeled by fitting the parameter of the Lafortune model to BRDF samples obtained from a sparse set of input images. Other parametric models may however be used as well. We preferred parametric models since the input data is too sparse for a tabulated representation. Before we detail our techniques we give an overview over alternative ways to represent a high-quality 3D model.

Chapter 3

Related Work on 3D Object Acquisition

The appearance acquisition and representation of real-world objects has recently received a lot of attention in the computer graphics and computer vision community. Following the taxonomy of Figure 3.1 the common approaches can be grouped into four different categories with regard to the complexity and flexibility of the representation: In the simplest case, diffuse textures are mapped onto the 2D surface of a geometric model parameterized by (p, q). A 4D data structure is necessary to represent view-dependent effects. Considering the viewing direction $\hat{v}$ expressed by azimuth and elevation, allows the object to be rendered from arbitrary viewpoints in a fixed environment. If the dependency on the incident illumination are to be modeled, another two dimensions are required to respect the light direction $\hat{l}$. To obtain a relightable model one can either densely sample the incident and outgoing directions and use the sampled data directly for rendering, or one may describe the reflection properties as a spatially varying BRDF estimated from a sparser sampling. The task of capturing real world objects by means of images is is called *image-based modeling and rendering*. Since the most important input to the mentioned approaches are images of the original object, they may be seen as specific image-based modeling and rendering applications. However, each category makes different assumptions on the representable effects or objects. Correlated to the assumptions, each category has a different acquisition setup, and they require drastically different numbers of input images.

The work presented in this thesis is closely related to two of these categories. Chapter 5 is related to the acquisition of diffuse textures. All subsequent techniques in this thesis deal with fitting spatially varying BRDF models to sparsely sampled data. In the following, we will first introduce the different approaches and related publications before we discuss their relative advantages and disad-

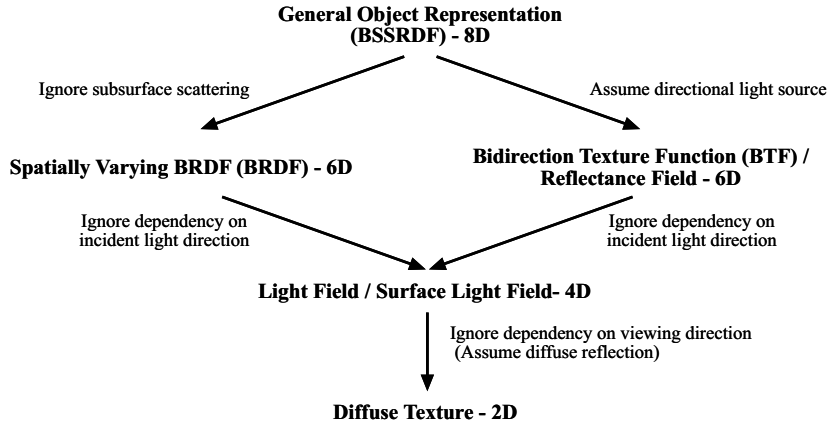


Figure 3.1: *Taxonomy of Object Representation Techniques.*

vantages at the end of this section. Other surveys on 3D object acquisition with different focus have been presented in [Bernardini02a] or [Ikeuchi01].

3.1 Acquisition of Textured 3D Models

The most widely used technique to represent the appearance of 3D objects nowadays is to map photographs as textures onto the 3D geometry of the object, considering static objects [Rushmeier98, Matsushita99, Neugebauer99, Rocchini99, Lensch00, Iwakiri00, Lensch01a, Bernardini01] or deformable objects like faces [Guenter98, Ip96, Pighin98]. To acquire a complete object including 3D geometry and texture one must run through the 3D object acquisition pipeline in Figure 3.2. Although we exemplify the pipeline in this section for the case of diffuse texture acquisition, parts of it will be picked up in the other categories capturing more complex appearance, as well.

3.1.1 3D Geometry Acquisition

The first step in the 3D acquisition pipeline is to capture the 3D geometry of the object, unless it is already provided as a CAD model. A recent survey on the field of 3D acquisition has been given by Bernardini and Rushmeier [Bernardini02a]. We will only summarize the main steps in acquiring a 3D geometry model. Geometry acquisition is typically accomplished using a 3D scanner, in our case a laser range scanner based on triangulation. The 3D scanner measures the distance from the scanner to the object's surface for a set of points. Other scanners use triangulation by projecting a structured light pattern, or measure the time of flight

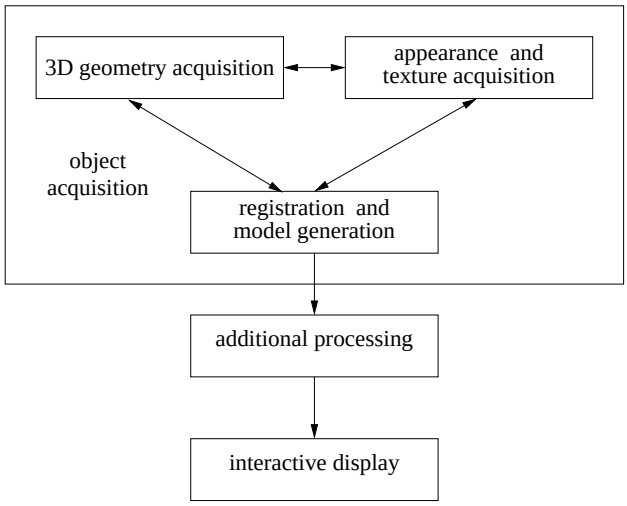


Figure 3.2: *3D Object Acquisition Pipeline. Depending on the applied techniques geometry acquisition, texture and appearance acquisition, and registration depend on each other in different configurations.*

to obtain the distance. The output of a single scan is a range map of the object from one view. Several scans are necessary to record complete information about an object.

The most complicated task in geometry acquisition is to combine, or integrate, the information of the individual scans. The integration requires registration of all range images into a common coordinate system. This task is often performed by a three-step process: A rough initial alignment is done by hand or is provided by a tracked scanner or a rotation platform. It is followed by a high-precision pair-wise alignment of successive scans. The pair-wise problem is often solved algorithms based on the *iterative closest point* (ICP), originally introduced by Besl and McKay [Besl92] where the transformation between two scans is obtained by minimizing the distances between transformed points of one scan to corresponding points on the other scan. Different methods have been proposed to determine the corresondence for robust registration [Chen92, Zhang94]. Sequential alignment of multiple views unfortunately accumulates registration errors. In a final multi-view registration the error is distributed to all scans [Turk94, Bergevin96, Pulli99].

The registration is followed by mesh *merging or integration*. One can follow basically three different approaches to obtain a single triangle mesh of multiple scans: Delaunay-based methods, surface-based methods, or volumetric methods. In the first category a point cloud is formed collecting all pixels of the range images. The final mesh is obtained as a Delaunay-triangulation of the point cloud (see [Edelsbrunner98] for a review of these techniques). An al-

ternative way creates an individual mesh for each range map and connects the surfaces of neighboring scans by local operations [Turk94]. In the last category a volumetric signed-distance field is computed based on the data of the range maps and the mesh is extracted as the iso-surface using the marching cube algorithm [Curless96, Wheeler98].

To accelerate the display of an scanned triangle mesh the mesh may be decimated and smoothed to reduce the effect of noisy vertices [Garland97, Kobbelt96]. Given the 3D geometric model we can proceed to acquire the texture of the object.

3.1.2 Imaging All Visible Surfaces

If an object's surface should be entirely digitized, the first step is to collect texture data for all visible surfaces. A set of camera positions must be determined from which every part of the surface is captured by at least one image. The process of finding a reasonable set is also known as *view planning*. For a given geometric model and a set of possible positions, Matsushita et al. [Matsushita99] determine the optimal set of required views respecting the elevation angle of the local viewing direction. The camera positions are restricted to discrete points on a sphere around the object. Stuerzlinger [Stürzlinger99] finds a minimal set of view points within the volume of all possible camera positions. He uses hierarchical visibility links to first determine optimal subregions using a simulated annealing approach, and then selects optimal points within these regions.

This is just a short list of references to view planning related to texture acquisition. A more general overview on the field of view planning is provided in Chapter 7 where we develop a view planning algorithm for the task of BRDF measurements.

3.1.3 Image-to-Geometry Registration

After recording the images, the camera position and rotation relative to the 3D model must be determined for each view. If geometry and texture are acquired at the same time with the same sensor as in [Sato97, Pulli97, Furukawa02], the images are already aligned to the range images, and a pure 3D registration algorithm like the ICP [Besl92, Bergevin96] can be applied. The use of a calibrated gantry can also replace the registration. In all other cases, one can basically follow two different approaches for the image-to-geometry registration.

The first approach selects a set of points in each image which correspond to known points on the model's surface. From these correspondences the camera transformation for the current view can be directly derived using standard camera calibration techniques, e.g., [Tsai87a, Zhang99b, Zhang99c]. However, the problem is to find these pairs of points. Depending on the object there may be geomet-

ric feature points which can be easily located in the images, and thus can be detected and assigned automatically. Kriegman et al. [Kriegman92] use T-junctions and other image features to constrain the model's position and orientation. Others attach artificial landmarks to the object's surface which are detected automatically in the images [Guenter98, Marschner99]. But these marks destroy the texture and have to be removed afterwards. If no extraordinary points can be detected automatically one may of course select corresponding pixels manually, which actually is a commonly used but tedious method [Pighin98, Rocchini99, Debevec96].

Instead of directly searching for 3D–2D point pairs, one may inspect larger image features like the contours of the object within each image. The correct camera transformation will project the 3D model in such a way that the outline of the projected model and the outline in the image match perfectly except for small errors due to imprecise geometry acquisition.

Many previous algorithms try to find the camera transformation by minimizing the error between the contour found in the image and the contour of the projected 3D model [Brunie92, Lowe91, Neugebauer99, Matsushita99, Ip96]. The error is typically computed as the sum of distances between a number of sample points on one contour to the nearest points on the other [Neugebauer99, Matsushita99]. Another approach computes the sum of minimal distances of rays from the eye point through the image contour to the model's surface which are computed using 3D distance maps [Brunie92].

To recover the different camera parameters, any kind of non-linear optimization algorithm like Levenberg-Marquardt, simulated annealing, or the downhill simplex method can be used (see [Press94] for an overview). During the optimization many different settings for the camera parameters are tested in order to guide the algorithm towards a minimum. For each test the error function has to be evaluated which is quite costly for contour-based distance measurement since the model must be projected and the point distances to the projected contour must be calculated for a sufficient number of points. In Chapter 5 we present a different, more efficient algorithm to calculate the distance between silhouettes instead of contours.

Beside geometry-based 3D–2D registration, the texture/image information itself may be used to register the different views relative to each other. For 2D–2D image registration a number of techniques have been developed [Brown92]. Based on this pairwise registration a global optimization for all incorporated views can be performed as demonstrated by Neugebauer et al. [Neugebauer99], whereas Rocchini et al. [Rocchini99] use the image information only to align the textures in those regions where different textures have to be blended during rendering. The image-to-geometry registration algorithm introduced in Chapter 5 takes into account both the silhouette and the texture itself.

3.1.4 Texture Preparation

After registration, the mapping from surface parameters to texture coordinates is known for each view. A single image can be mapped onto the object by common graphics hardware supplying projective texture mapping [Segal92]. If multiple views are incorporated one must determine which image is best to be mapped onto which part of the surface. Here, the angle between the viewing direction during acquisition and the surface normal may be considered [Rocchini99, Matsushita99]. Special care must be taken at boundaries of surface regions which are textured with data from different images. To create a smooth transition between the regions the textures must be blended appropriately. Rocchini et al. [Rocchini99] even precompute this blending into a new texture to speed up the entire rendering process.

The resulting texture may also be composed by all observations at once. Rushmeier et al. [Rushmeier97, Rushmeier98] and Bernardini et al. [Bernardini01] derive a view-independent diffuse texture from multiple images of the object lit by a point light at known positions. A diffuse texture consistent with multiple unconstrained observations has been extracted by Callieri et al. [Callieri02]. Although the optimized texture has high quality it is not possible to correctly render the digitized object in new lighting environments.

Texture Atlas

Besides the extraction of diffuse color data, the mapping of all texture data into a single image, a *texture atlas*, is important for efficient rendering. The problem of creating a texture atlas is closely related to the problem of surface parameterization. A texture atlas may simply be constructed by packing all relevant parts of the original images into one single large texture [Rocchini99] resulting in many disconnected texture patches. We will follow a similar approach in order to keep the maximum sampling rate provided by the input images.

Another way to construct texture atlases considers only the geometry of the 3D mesh. A projection of the mesh into the 2D plane is performed minimizing texture stretch and distortion on the surface, as for example proposed in [Marschner98, Piponi00, Sander01, Sorkine02]. For general meshes it is necessary to introduce cuts on the surface which may partition the surface into distinct patches.

Rendering an object represented by a polygonal mesh augmented with a diffuse texture atlas is trivial since texture mapping is supported by all 3D graphics boards nowadays.

3.1.5 Warping-Based Techniques

For diffuse objects, polygonal meshes are not the only way to represent geometry information. Instead, per-pixel geometry may be stored for each pixel in the 2D image. *Image warping* is used to render the pixels at the correct screen position for a desired view [Beymer96, McMillan95]. The warping can be performed in two ways: forward, by looping over all pixels in the input images and projecting them into the output image, or backward, by sampling for each output pixel the corresponding pixels in the input images. Efficient warping algorithms have been proposed as pure software implementations [Smith87, McMillan95, Chen99, Oliveira00] or exploiting graphics hardware [Schaufler98, Kautz01].

To represent an entire object more than a single image is required. A complex environment has been rendered by McAllister et al. [McAllister99] using multiple images with depth information. Shade et al. [Shade98] introduced *layered depth images* (LDI) which merge the information of multiple views in one data structure. At each pixel in a LDI intersection points and color of all surfaces are stored that intersect with the ray defined by the pixel. Six LDIs with a common center of projection have been combined by Oliveira and Bishop [Oliveira99] resulting in so-called *image-based objects*. *LDI trees* [Chang99] further extend the concept of LDI to represent complete environments by a multi-resolution representation with hierarchical space partitioning.

3.2 View-Dependent Representations

All previously mentioned techniques assume diffuse reflection properties. They can only reproduce view-independent shading effects. In this section we present techniques that can capture and render view-dependent effects using data that captures each point in the scene from multiple viewpoints. The most important representatives of this category are *light fields* and *lumigraphs*, explained in Section 3.2.4.

3.2.1 The Plenoptic Function

All view-dependent appearance representations can be derived from the *plenoptic function* which was introduced by Adelson [Adelson91]. It represents the radiance at every point in a scene $[x, y, z]^T$ in every direction (θ, ϕ) depending on wavelength λ and time t:

$$p = P(\theta, \phi, \lambda, x, y, z, t) \tag{3.1}$$

For static scenes the time may be neglected, and the dependency on the wavelength is frequently reduced to three RGB samples.

3.2.2 Panoramas and Mosaics

One of the first attempts to represent view-dependent information was Apple's Quicktime VR [Chen95]. It uses a number of 360° panoramic pictures of a scene and allows for reconstruction for fixed viewpoints and arbitrary viewing direction. McMillan et al. [McMillan95] augmented a similar approach in order to enable smooth transition between the discrete viewpoints in the given environment.

Similar to panoramas are *Multiple-Center-of-Projection Images* introduced by Rademacher et al. [Rademacher98]. The vertical pixel columns of such a 2D image are taken from slightly different view points along a continuous path using a slit camera. Multiple of these images taken along concentric circular paths have been combined by Shum et al. [Shum99] forming *concentric mosaics*. The captured environment can be observed from arbitrary viewpoints within the outer most circle by linear interpolation of the multiple images.

3.2.3 View-Dependent Texturing

While panoramas and mosaics are more tailored to allow free movement within an environment the following techniques concentrate more on the inspection of an object from different view points. Debevec et al. [Debevec96] model architectural scenes from airborne photographs. A coarse geometry model of the scene is textured selecting different photographs depending on the selected view point. Different metrics have been proposed to blend the pictures of multiple view points [Pulli97, Debevec98, Buehler01] to obtain the final image.

Matusik et al. [Matusik00] introduce a technique called *image-based visual hulls*. In their approach a dynamic 3D model is captured using eight video cameras. Based on silhouette information they infer 3D geometry to which view-dependent texturing is applied.

3.2.4 Light Fields

Another category of methods that aim on representing real word objects are *light fields* [Levoy96] and *lumigraphs* [Gortler96]. Compared to techniques presented in the previous section, where the space of outgoing directions is only sparsely sampled, light fields and lumigraphs typically provide a very dense sampling of the outgoing radiance.

Two-plane parameterization

A *light field* is closely related to the plenoptic function (Equation 3.1). The difference is that a light field models wavelength typically only by an RGB triple and does not depend on the time, which means that it can represent static scenes only. Therefore, it is a five-dimensional function, dependent on the location and the viewing direction.

However, the 5D representation may be reduced by one dimension in free space since radiance does not vary along one ray until it hits an occluder. Hence we may restrict all points to lie on a convex hull of the considered object, ending up with a 4D parameterization of all possible rays. A reasonable representation of this 4D function is given by the so-called *two-plane parameterization* where all possible rays are parameterized by their intersection point with two parallel planes as demonstrated in Figure 3.3. For each ray the light field stores the radiance traveling along the ray. The corresponding data structure is usually denoted *light slab*.

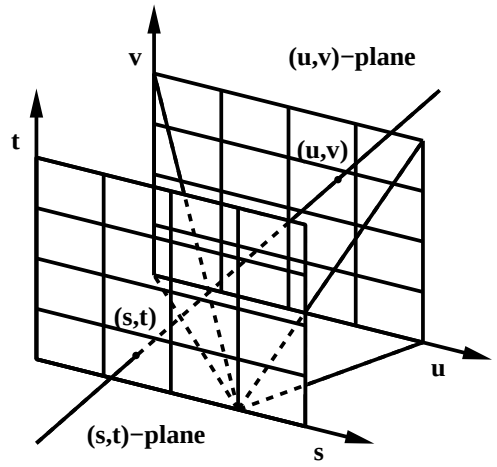


Figure 3.3: Two-plane parameterization of a light field. The eye points form the (s,t)-plane while the (u,v)-plane represents the projection plane.

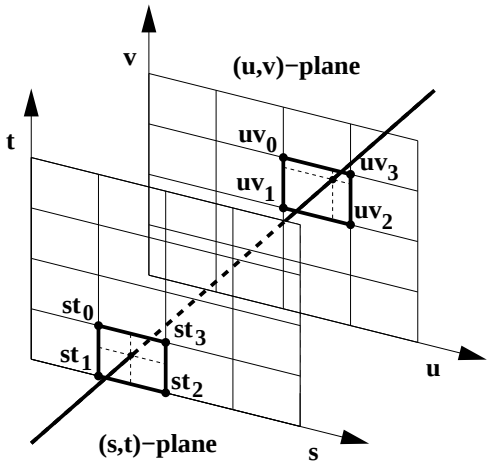


Figure 3.4: The radiance of one ray is obtained by quadri-linear interpolation of the radiance values of the 16 rays through the surrounding grid points.

As can be seen in Figure 3.3, all rays passing through one point in the (s,t)-plane sample a perspective image of the scene with (s,t) as the center of projection and the (u,v)-plane as the projection plane. One may consider a light slab as a two-dimensional array of 2D images with eye points regularly placed on the (s,t)-plane. This is why the planes are also denoted eye-plane and image-plane.

Both planes are typically discretized into a regular grid, storing radiance values only for the rays passing through the grid points. An arbitrary ray intersecting one cell on the (s,t) and another cell on the (u,v)-plane is surrounded by 16 rays through the vertices $st_0,\dots st_3$ and $uv_0,\dots,uv_3$. An example is shown in Figure 3.4. The desired radiance value is obtained by quadri-linear interpolation of these sixteen samples. In [Gortler96], Gortler et al. introduced an approach using graphics hardware to carry out the quadri-linear interpolation which significantly speeds up the rendering process. The reconstruction of a light field is basically independent of the scene's geometric complexity, For high-resolution light fields this interpolation technique works fine. However, if the resolution is decreased, only points in the (u,v)-plane will appear sharp, while blurring will increase with respect to the object's distance to the image plane. To combat this problem Gortler et al. augmented the light field by some geometric information (see again [Gortler96]). This extended light field is usually called *lumigraph*. A coarse polygon mesh approximating the object is used to determine the depth along the sampling rays. Based on this depth information, the weights for the interpolation are corrected. This way sharp contours of objects stored in a lumigraph can be achieved even for low resolutions. Other rendering techniques are based on per-pixel geometry. Here, new views are generated by warping the input images (see Section 3.1.5) and weighting the contributions of different inputs to one output pixel with respect to the selected viewing direction [Schirmacher00].

3.2.5 Surface Light Fields

Another parameterization of light fields has been introduced by Miller [Miller98]. There, the light field is closely coupled to the geometry of the reflecting object. The light field's u and v parameters are chosen to match the surface parameters, and thus every sample in the light field is associated with a location on the surface, hence the name *surface light field*. The remaining s and t parameterize the hemisphere of directions above some surface point (u,v), see Figure 3.5.

Wood et al. [Wood00] acquired and compressed surface light fields by *principle function analysis* (PFA). We will also apply PFA in Section 6.5.2 for an efficient representation of view *and* lighting-dependent data.

Unfortunately, the parameterization of surface light fields complicates the reconstruction procedure. At each vertex the viewing direction must be transformed into the local coordinate system in order to compute the s and t coordinates. These define a slice of the light field that can directly be used as a texture map applied to the surface around the point (u,v). A fully hardware accelerated rendering technique for surface light fields has been proposed by Chen et al. [Chen02]. Buehler et al. [Buehler01] present a technique that can handle both two-plane parameterized light fields and surface light fields at the same time. The technique is very

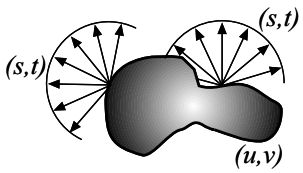


Figure 3.5: *Surface Light Field. For each position on the object's surface* (u, v) *the outgoing radiance is stored for all directions* (s, t)*.*

similar to view-dependent texturing.

Attaching the light field directly to the object's surface bears one major advantage: the contour of the surface will always appear sharp. But this is not necessarily true for the applied textures if the directional resolution is low. The resolution in s and t directly affects the quality of rendering specular highlights. Furthermore, light fields can only capture the view-dependent effects, i.e. the digitized object can be viewed from any direction, but it will always show the object in the environment where it was during the acquisition. It is not possible to adapt the incident lighting to new virtual environments.

3.3 Relightable Representations

Although light fields can represent the view-dependent appearance of an object reasonably well, including moving highlights, it is not possible to relight the object in order to show it in another environment and to account for different incident illumination. Illumination-dependent representations are often captured using a set of images showing the object in different illumination situations. A relit image is computed as a linear combination of the pre-lit images.

Belhumeur and Kriegman have investigated the number of images that are necessary to describe an object under all possible illumination conditions [Belhumeur98] in the context of object recognition. They state that the appearance of a diffuse object can be recovered by three images already. For non-Lambertian objects more input images with different illumination are necessary.

3.3.1 Scene Relighting

Early applications of scene relighting have been lighting design [Dorsey91] and interactive walkthroughs [Airey90]. In their work the intensities of a fixed set of light source could be adjusted, but arbitrary relighting is not possible.

Nimeroff et al. rendered basis images illuminated by a set of steerable basis function which then are combined linearly to approximate the effect of skylight at different times of the day [Nimeroff94]. Steerable function have also been used by Teo et al. [Teo97] in combination with principle component analysis to allow efficient re-rendering.

Loscos et al. perform interactive relighting on real scenes based on a number of photographs with known incident illumination and under the assumption of diffuse surfaces [Loscos99, Loscos00]. Marschner presented a technique where the incident illumination is recovered from a photography by inverse-lighting prior to relighting the image [Marschner98].

The relighting of transparent and mirroring objects has been addressed by Zongker et al. [Zongker99]. They introduced *environment matting* techniques. The appearance of the object is captured for different light directions by displaying various patterns on a CRT monitor. The techniques has been extended for higher accuracy, faster acquisition [Chuang00] and more complex environments [Wexler02]. The environment matting approach has also been applied to capture 3D geometry of mirroring surfaces [Tarini03].

While the techniques for general objects only acquire a sparse sampling of the incident light directions, much higher quality can be achieved by a more dense sampling.

3.3.2 Bidirectional Texture Functions

A denser sampling is for example performed with *bidirectional texture functions* (BTFs) [Dana99], although at very high storage costs. BTFs acquire the view and illumination dependent spatially varying appearance of more or less flat samples of a given complex material. A large set of images is captured showing the sample from different view points and different incident light directions using a directional light source. During acquisition the camera has to be placed at various positions in the hemisphere above the sample taking several picture with different incident light directions. BTFs actually capture spatially varying BRDFs in tabulated form.

New views of the material are rendered by interpolating the data of the nearest viewing directions as it is done in light field rendering. Changes in the incident lighting are accounted for by adding up the weighted contribution of the data for several light source directions. A large number of viewing and lighting directions have to be captured for a sufficient reconstruction quality. BTFs can even reproduce faithful renderings of materials with complex mesostructure.

McAllister [McAllister02a] fits a spatially varying Lafortune model to very densely sampled BTF data to reduce storage cost and make the task of rendering

easier. The achieved results are impressive but the technique requires flat surfaces and does not allow as complex mesostructure as when using the BTF data directly.

Several research groups are currently developing techniques to digitize art paintings. The main focus of their work is to achieve a faithful representation of the colors of the painting under various illumination conditions. Imai et al. proposed and tested several methods to estimate the spectral reflectance of paintings [Imai98, Imai01b, Imai01a]. The collected information can even be used to identify the individual color pigments used by the artist. Similar work has been performed by Tominaga et al. who additionally captured fine geometric detail and normal information for an oil painting and fitted the parameters of a reflection model [Tominaga01, Tominaga02].

Another representation of lighting dependent textures are polynomial texture maps. They have been proposed by Malzbender et al. [Malzbender01] and are suitable for hardware accelerated display of the acquire data. In order to allow sparser sampling, Liu et al. [Liu01] synthesized new samples by reconstructing the surface structure of the observed materials.

3.3.3 Reflectance Fields

While BTFs are typically used to represent a flat although complex material, reflectance fields are used to capture the viewing and lighting dependent appearance of an entire object. The acquisition technique is however quite similar. Debevec et al. [Debevec00] describe a method for acquiring the reflectance field of human faces. A video stream of the human is captured while a point light source spins around his face. A simpler setup has been proposed by Masselus et al. [Masselus02]. It allows for a manual setup of the light sources. Although the initial setup acquires the reflectance field from one view point only, the technique can easily be extended to allow for arbitrary views.

Magda et al. acquired two reflectance fields of one object to recover 3D geometry [Magda01]. For the second reflectance field, the distance of light source to the object is increased, and geometry is reconstructed based on the slightly different observations. Matusik et al. [Matusik02a] acquire the reflectance field of different even slightly translucent objects for different viewpoints and at the same time reconstruct a view-dependent geometry representation of the object, so called *opacity hulls*. The acquired data is directly used for rendering. This technique was further extended to transparent and refractive objects [Matusik02b]. Furukawa et al. [Furukawa02] also acquired the reflectance field of different objects and proposed various compression techniques exploiting the coherence of the data.

Although the reconstruction results of acquired reflectance fields allow to render the object in new environments with arbitrary lighting almost photorealistically, the number of acquired sample images for the viewing and lighting

directions is very high. Debevec et al. [Debevec00] reported around two thousand images for a single view point while Matusik et al. [Matusik02a] took several thousand images for a complete reflectance field. The number of required images lengthen the acquisition time (several hours) and occupy up to several GB of storage for a single model. Except from [Masselus02] all presented techniques build on an automated setup to obtain a dense sampling of the reflection properties.

3.4 Measurement of Bidirectional Reflectance Distribution Functions

The last category does not render directly from the acquired data but transforms the measured data by fitting a reflection model. The parameters of a reflection model can often be determined using just a sparse set of input images. However, sparser sampling comes along with a number of assumptions and restrictions probably limiting the types of materials that can be represented.

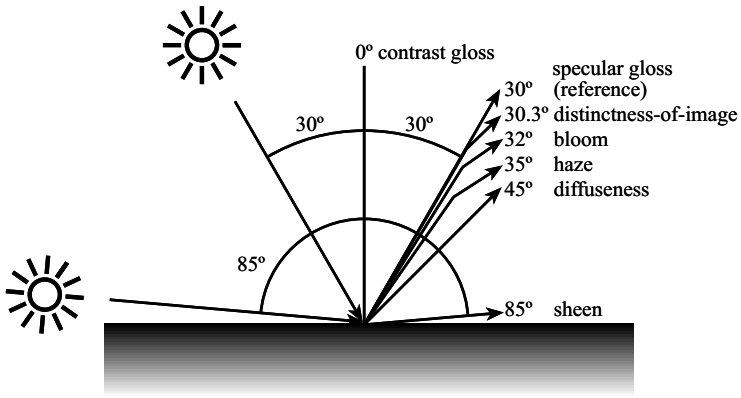


Figure 3.6: *Measurements of Gloss. Different kinds of gloss are measured as the reflected radiance at specific incident and outgoing directions.*

3.4.1 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 measurement 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 3.6.

3.4.2 Gonioreflectometer

A sparse set of discrete measurements of the reflection properties is clearly not sufficient to represent all possible effects. A dense sampling of the BRDF can be acquired using a *gonioreflectometer*. A gonioreflectometer consists of a light source and a photometric sensor. The setup is depicted in Figure 3.7(a). During one measurement light source and sensor are positioned at various angles covering the entire hemisphere above a flat sample of a homogeneous material. At each position one sample of the BRDF is measured resulting in a tabulated BRDF. Instead of moving the sensor and the light source one may keep the sensor fix and rotate the surface instead to cover the full range of directions (see Figure 3.7(b)). Goniometric measurements are often performed for a large range of wavelengths of the visible spectrum which allows for high-quality color reproduction.

Gonioreflectometers are designed to obtain reproducible results of high precision. For the purpose of measuring the appearance of an object gonioreflectometers are not suitable for two reasons: Gonioreflectometers are relatively slow since the sensor and the light source have to be repositioned for every pair of incident

and outgoing direction, yielding a single BRDF sample. In addition, gonioreflectometers measure the BRDF of only a single material at a time. The device would have to be positioned at every point on the surface separately to capture spatially varying materials.

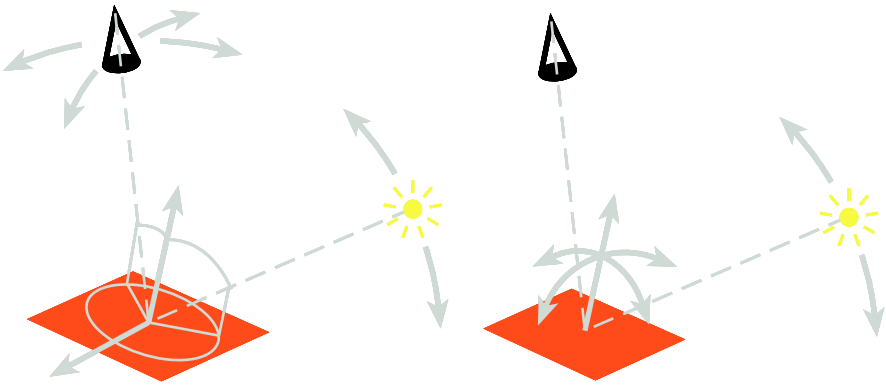


Figure 3.7: *Principle Setup of a Gonioreflectometer. (a) A light source and a photometric sensor take BRDF samples for each pair of directions of the hemisphere above a flat homogeneous surface sample. (b) Alternative setup: the sensor is fixed and the sample is tilted instead.*

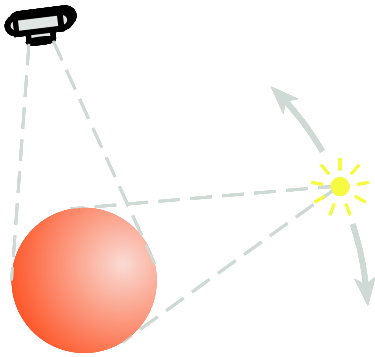


Figure 3.8: *Image-Based BRDF Measurements. Images of a curved homogeneous object lit from various directions provide a huge number of BRDF samples due to the varying surface normal.*

3.4.3 Image-Based Measurements of Homogeneous BRDFs

More recently, image-based approaches have been proposed which significantly speed up the measurement of BRDFs. These methods are able to acquire a large number of samples at once by using a digital camera instead of a single sensor. Each pixel of a camera image may capture a different BRDF sample. For example, Ward [Ward Larson92] uses a hemispherical mirror to sample the exitant hemisphere of light with a single image. Instead of using curved mirrors, it is also possible to use curved geometry to obtain a large number of samples with a single image, as demonstrated in Figure 3.8. Images of curved objects will show for each pixel a slightly different normal direction and thus a different viewing and lighting direction in the local tangential coordinate frame of every point. This approach is taken by Lu et al. [Lu98], who applied cylindrical surfaces. Marschner et al. [Marschner98, Marschner99] propose a method to measure the BRDF of arbitrarily shaped objects. The geometry of these objects is measured in a preprocessing step using a range scanner.

While the previous techniques allow to measure the BRDF only in a controlled environment where all light incident on the object is emitted by a single light source with known position Ramamoorthi and Hanrahan [Ramamoorthi01b] and Gibson et al. [Gibson01] proposed inverse rendering algorithms that do not need a carefully controlled environment. The reflection properties and the incident light field are reconstructed at the same time allowing arbitrary incident illumination. The only requirement on the incident illumination is that enough high frequencies, e.g., point light sources, are present.

3.5 Sparsely Sampled Spatially Varying BRDFs

The reflection properties of a single material can only partially represent the appearance of a 3D model since most objects consist of more than a single material and may show subtle variation of the reflectance even within a single material, e.g., caused by imperfections, dust, scratches, and so on. Realistic representations can only be obtained if spatial variation is taken into account.

Sato et al. [Sato97] take several images of an object from different view points with different lighting and fit a Torrance-Sparrow model [Torrance67] to the data. They consider high-frequency variations for the diffuse part of the BRDF but only per-triangle variations for the specular part. This is also the case for the work by Marschner et al. [Marschner00] who recover the BRDFs of faces.

Yu and Malik [Yu98] capture the reflection properties of architectural scenes

respecting the illumination by the sun, skylight and even from the environment. To reconstruct indoor scenes Yu et al. setup some point lights at known positions and consider the indirect illumination in an iterative process [Yu99]. Again only the diffuse part varies freely, but they allow anisotropic reflections. In [Gibson01] incident illumination and BRDFs are reconstructed at the same time by investigating shadows similar to [Sato99].

Indoor scenes with anisotropic patches are also reconstructed by Boivin and Gagalowicz [Boivin01] based on the observations from just a single image. Several constraints are enforced to fit the parameters. Although the method is able to reproduce the input image reasonably well, the estimated BRDFs lack precision due to undersampling.

The appearance of objects with initially unknown illumination has been addressed by Ramamoorthi and Hanrahan [Ramamoorthi01b] and Nishino et al. [Nishino01b]. Nishino et al. reprojected specular highlights into an environment map to estimate incident illumination. In [Ramamoorthi01b] the reconstruction is performed using a spherical harmonics representation of both the incident illumination and the BRDF. The authors also derive conditions under which the combined inverse problem is well-posed: The BRDF can only be recovered up to the highest frequency of the incident illumination and vise-versa. In both approaches the specular part is assumed to be constant over the entire object.

In Chapter 6 we present a technique that recovers per-pixel specular variation across the object's surface under the illumination of a known point light source. To obtain reliable BRDFs, surface points belonging the same base material are grouped by a clustering process. Later, per-pixel BRDFs are obtained as a linear combination of the basis materials. The technique has been taken up by Li et al. [Li02] who apply a clustering approach to single-image observations. The acquired BRDF samples of different pixels contribute to one another's estimate depending on their relative location on the surface and on the similarity of their reflection properties.

3.6 Normal Maps

Reflection properties together with measured photometric data can also be used to derive geometric information of the original object. Photometric stereo approaches (see [Zhang99a] for an overview) have been developed to extract geometric information from a set of pictures with different lighting conditions. The shading information is used to compute surface normals and depth values.

Rushmeier et al. calculate normal directions at every visible surface point from a set of images showing the same view of an object illuminated by a point light source placed at different but known positions for each image

[Rushmeier97, Rushmeier98, Bernardini01]. Besides the normal direction the diffuse color (albedo map) is reconstructed by solving a linear system of equation for each point. The technique assumes however the surface to be purely diffuse (Lambertian) and simply discards the brightest radiance samples at each point to circumvent specular highlights. Hereby valuable data is ignored. Additionally, as the surface gets more and more glossy it is hard to guarantee that all remaining samples show a purely diffuse reflection which is required to compute the correct normal direction. In Section 6.8 we use measured non-Lambertian reflection properties to compute normal maps for arbitrary materials.

3.7 Discussion

Each of the above techniques for appearance acquisition has its specific strengths and weaknesses. In Table 3.1 we compare the main representatives of each category with respect to various criteria.

In the first six rows one can observe a steady increase in the number of views that are captured to obtain a model. This increase correlates to the complexity of the representable appearance. In order to acquire a diffuse texture rather few views are sufficient, representing view-dependent effects adds almost one order of magnitude, and another increase is necessary if relightable models for various viewing directions are captured. While light field and reflectance field techniques are able to represent almost arbitrary objects, they also complicate the acquisition process and create a huge amount of data which poses serious problems during rendering. The last row in Table 3.1 indicates however that the number of necessary views can be drastically reduced if the reflection properties of the object are assumed to be representably by a low-parameter BRDF model. The appearance of an opaque object can to some extent be recovered spending roughly the same number of views necessary to acquire a diffuse texture.

Since the goal of this dissertation is to create visually convincing models we need to take illumination effects into account. Techniques that only recover diffuse texture of an object or light field techniques are therefore clearly not sufficient. In Chapter 5 we anyway present a method to capture a textured 3D models. The main contribution of the chapter is on high-quality image-to-geometry registration which forms the basis for all subsequent techniques presented in this thesis.

Besides representing illumination-dependency, we aim at an efficient acquisition regarding the cost of equipment, the required time and the amount of data produced. Reflectance fields and opacity hulls require an automatic setup with computer controlled gantries to perform the acquisition. In order to avoid a complicated setup we concentrate on techniques that recover spatially varying BRDFs from a rather sparse set of input images.

Technique	Acquisition Environment	#Views	View-dependent	Relighting	Comment
texture mapped geometry	diffuse illumination	10-20	-	-	- hard to remove highlights
light fields	arbitrary environments	100-500	+	-	+geometry independent - ghosting
surface light fields	arbitrary environments	100-500	+	-	+ better quality through explicit geometry
BTFs	automated point light and surface orientation	500-10000	+	+	+ arbitrary materials - flat samples only
reflectance fields	automated point light	2000-10000	+	+	+geometry independent + arbitrary materials
opacity hulls	automated point light and object orientation	20000-60000	+	+	+ arbitrary complex geometry through view-dependent geometry
spatially varying BRDFs from few images	various setups	10-40	+	+	- restrictions on types of materials - restrictions on specular variation

Table 3.1: *Comparison of 3D Object Acquisition Techniques. The quality of the representation and the visual effects that can be reproduced is correlated to the number of views that have to be acquired, and thus to the acquisition time. Some methods require an automated setup, otherwise the acquisition process would take too long.*

Technique	single view	environment	interreflections	inverse lighting	specular variation	isotropic	anisotropic
[Sato97]	-	PL	-	-	patch	+	-
[Lu98]	-	PL	-	-	object	+	+
[Yu98]	-	E	+	-	object	+	-
[Yu99]	-	E	+	-	patch	+	+
[Marschner00]	-	PL	-	-	patch	+	-
[Gibson01]	-	E	+	+	patch	+	-
[Boivin01]	+	E	+	-	patch	+	+
[Ramamoorthi01b]	-	E	-	+	object	+	-
[Nishino01b]	-	E	-	+	object	+	-
[Li02]	+	PL	-	-	pixel	+	-
Chapter 6	-	PL	-	-	pixel	+	-

Table 3.2: *Comparison of algorithms acquiring spatially varying BRDFs from sparse sets of images. BRDF measurements are performed using point lights (PL) or allowing arbitrary environments (E). Major differences are with respect to the variation of the specular part which is frequently restricted to be constant over some surface area.*

In this category various techniques have been developed prior or concurrently to this dissertation. The new technique in Chapter 6 removes some of the restrictions imposed by previous algorithms. As can be seen in Table 3.2, all previous techniques permit spatial variation in the diffuse part only while the specular part is kept constant for one patch or for the entire object. Our technique presented in Chapter 6 recovers spatial variation of the specular part on a per-pixel level.

In our measurement technique we do not perform a complete global inverse illumination but determine the BRDF based on the local illumination. Interreflections within the object are currently ignored. They may however be incorporated easily as a further step at the end of the acquisition pipeline.

Except of [Ramamoorthi01b], none of the previous techniques analyze the dependency of the quality of the resulting BRDFs on the selected views. In Chapter 7 we perform such an analysis and derive a view planning algorithm for the task of BRDF measurement.

Before we turn to the actual algorithms we first describe the equipment and basic techniques that are applied for the measurements.

Chapter 4
Acquisition Setup

A special purpose digital photo studio was built to generate high quality real world input data for appearance measurements. Special attention was paid to carefully control the lighting conditions in order to be able to acquire exact data about the surface properties of objects using readily available digital camera technology.

This chapter discusses the specific demands and requirements that arise from these goals for the equipment. We describe our requirements for each component of the photo studio and to what extend they are met by the equipment we use. The photo studio consists mainly of a digital camera, a 3D scanner, almost point shaped light sources, and a low reflective (i.e., black) room. After describing the equipment we will discuss basic algorithms that are used in our measurements to derive high quality input data from the captured images.

4.1 The Camera

A camera is the central piece of hardware for any image-based measurement lab. Analog cameras are not practical for this purpose, mostly due to the long development cycle of the film, and the necessity to digitize the image afterwards, which is either tedious or, if done by a photo lab, out of control of the researcher. This leaves digital cameras as the most feasible option. The choice of a particular model has to be made based on a variety of criteria that have to be fulfilled for different applications. These criteria, and the tradeoff between them will be described in the following.

4.1.1 General Requirements

When using a camera as a measurement device, we demand that it records high resolution image information with good accuracy (10-12 bit per color channel).

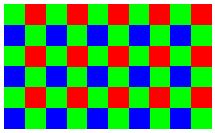


Figure 4.1: *Arrangement of individual colored pixels in a Bayer pattern sensor.*

The camera should be able to deal with a variety of different lighting situations, especially those that require long exposure times. Additionally, objects of different sizes will be captured under various conditions which is greatly facilitated if the lenses are interchangeable. Since the camera will be used for measurement purposes, we would like to avoid introducing artifacts due to lossy compression techniques such as JPEG. Finally, a very important criterion for simplifying the actual measurements is that the camera should be completely controllable remotely from a computer. The measurements often require us to take series of images with different exposure or aperture settings. Manual adjustment at the camera bears the danger of slight changes in camera position and orientation and slows down the measurement process considerably.

4.1.2 The Choice of Camera

As a result of the above discussion, we have decided for a high-end digital camera intended for professional photography. In particular, we have chosen the Kodak DCS 560, a single lens reflex (SLR) camera based on a 35 mm shutter Canon EOS-1N body. It has all the usual features of current SLR cameras like an auto focus and auto exposure system and can be used with most Canon EF auto focus lenses. The 3040×2008 pixel imaging sensor produces images with 12 bit color depth. It is possible to control the camera remotely via a serial IEEE 1394 FireWire connection. In this case most of the settings can be changed by the host computer and images can be acquired and downloaded via FireWire.

The camera uses a single CCD chip with color filters in front of the individual pixels arranged in the Bayer pattern [Bayer76], as shown in Figure 4.1. This way, a color image can be obtained with a single exposure, but every pixel will only have information about *one* color channel corresponding to the color of its filter. This means that the missing channels have to be reconstructed in order to get an RGB tripled per pixel. This reduces the spatial resolution and can introduce artifacts. The reconstruction is especially error prone at the borders of high contrast regions or at small but very bright features exciting only a single pixel on the CCD.

4.2 3D Scanning Devices

The other measuring device that is used in our applications is a 3D scanner. In most cases the 3D model of an object is not available and we therefore have to acquire it ourselves. For this task we used a Steinbichler Tricolite 3D scanner. More recently, a Minolta Vi910 3D scanner was purchased including a rotary platform. The latter scanner is self calibrating and thus easier to use. Although the acquisition is still not completely automatic, the use of a rotary platform further simplifies the geometry acquisition since it reduces the effort for rough initial alignment (see Section 3.1.1). For mesh registration, merging, and editing the commercial package Polyworks by InnovMetric was applied. Software developed at MPI has been applied for mesh smoothing and simplification.

Two major problems were encountered when capturing the 3D geometry of an object for the purpose of measuring reflective properties: Due to noise in the single scans, the registration, mesh merging, and smoothing some geometric detail is lost. The resulting mesh does often not provide sufficiently exact normals. If the normal at a surface point given by the mesh does not match the original surface normal the estimated BRDF will be affected quite severely.

The reflection properties we want to measure also cause another problem: Both scanners produce the best results for homogeneous, diffuse, and almost white surfaces. If the reflection properties vary or contain a significant specular component the estimated depth values can be wrong or the scanner even provides no data at all. As a consequence, we had to paint or powder some of our models or some duplicate of them for 3D scanning, and cleaned them afterwards before measuring the reflection properties.

4.3 The Lamps

For measurement purposes the applied light source has to be selected carefully as well. The decision for a specific light source is based on the emitter geometry and the lamp's photometric properties.

4.3.1 Light Source Geometry

For many image-based rendering and vision algorithms the solid angle formed by the incident light rays should be well defined and as small as possible for any given surface point. This can be achieved by either using a parallel light source or a point light source.

The most common parallel light source is a collimated laser beam. It is often used in gonioreflectometers to measure the BRDF of a surface. But it is hard to

illuminate a large object with collimated light as a lens with the same diameter as the object is needed. Furthermore, a laser emits light of a single wavelength which makes it impossible to record the color of the surface. Due to the coherence of laser diffraction patterns are formed.

A point light source can be approximated by a nearly point shaped light source or by a virtual point light source created by adding an aperture, a lens system, or a reflector to a regular lamp. Placing a small aperture in front of a non-point shaped light source leads to two problems. Only a small fraction of the emitted light will pass the aperture, and the angular distribution of the light is rather inhomogeneous as the aperture forms a pin hole camera which projects an image of the light source onto the object. For a lens based light system the effective size of the light source equals the size of the lens which is usually quite large. The same holds for a system with a reflector unless it contains an ideal point shaped light source and the light is always reflected in the direction it is coming from.

If a non-ideal light source is placed at a large distance from the object the solid angle formed by the incident light rays becomes very small. As long as the emitted light is constant over the solid angle covered by the object even non-ideal light sources may be used to approximate point light sources.

4.3.2 Photometric Requirements

The irradiance at each surface point must be known in order to interpret the reflection properties based on the light reaching the camera from that point. It can be determined by a calibration procedure for each setup. Nevertheless, it is much more desirable to use a homogeneous point light source, emitting equally in all directions.

In order to acquire high quality color information the spectrum of the emitted light should be constant during a single photo session but also for several sessions over the course of weeks or even months. Furthermore, the spectrum should be approximately constant over the visible spectrum to allow for a good color reproduction [Osram GmbHa].

For practical reasons a point light source must be very bright since most of the light (i.e. usually more than 90%) is not emitted in the direction of the object.

4.3.3 Real Lamps

Most of the photometric requirements in the previous section are also common for photographic and cinematographic applications. But a point light source is usually not desired as it produces very hard shadows and is rather unpleasant to look at for human subjects.

Figure 4.2: *A HMI bulb mounted in a lamp. An electric arc is produced between the two electrodes in the inner glass ball. The effective light source diameter is smaller than 1 cm.*

In photographic applications flash lamps are often applied. For BRDF measurement purposes they are however not ideal, since they are not well suited for high dynamic range imaging. Without continuous modeling light it is hard to predict which parts of the surface will be illuminated for a selected position of the flash light.

Standard tungsten lamps often used in video applications are not ideal for our purpose. Their low color temperature leads to poor color reproduction and yellowish images which have to be corrected using filters or white balancing techniques. Due to their low efficiency these lamps have to be rather strong producing a large amount of heat.

These disadvantages can be overcome using HMI discharge lamps [Osram GmbHb]. In these lamps a luminous arc burns between two electrodes (see Figure 4.2). It produces a very bright light with a near daylight spectrum. The efficiency is about 3 times the efficiency of tungsten lamps.

We bought Joker-Bug 800 lamps produced by the French company K5600. They contain a special 800 W HMI bulb with a coating that shields the UV radiation. In addition we have several reflectors and diffusor boxes which can be mounted on the lamps to generate diffuse lighting situations commonly used by photographers.

4.4 The Room

To perform the acquisition in a controlled environment a special room was set up. It is divided into two parts – a small computer lab with additional storage facilities for the equipment and the actual photo studio. People are able to work in the computer lab, e.g., remote controlling the camera while the photo studio is

Figure 4.3: *The photo studio. This image was generated from a high dynamic range image by applying a tone-mapper.*

used to take images under controlled lighting conditions.

The most important requirement for the photo studio was that no light bouncing of the walls, the floor, or the ceiling should illuminate the scene. This means that as much light as possible must be absorbed by the walls, the floor, and the ceiling of the room, and the applied materials should be as diffuse as possible, to avoid disturbing highlights. Furthermore, no light from the outside (i.e., the computer lab) should get inside the photo studio. As a result of this considerations the walls and the ceiling were covered with thick black felt and a black needle fleece carpet was put in.

Figure 4.3 shows an image of the interior of the photo studio in a working situation. A high dynamic range image was generated from a series of images with different exposure times as explained in the next section. By applying a suitable tone-mapping operator (e.g., [Tumblin93, Ward94, Larson97]) the dynamic range is compressed and more details can be seen.

Besides a controlled environment and suitable equipment, we applied a number of standard algorithms to further increase the quality of the images that are used as input to our measurements. These algorithms concern image rectification, noise removal and photometric calibration.

4.5 Geometric Camera Calibration and Image Rectification

Most algorithms assume that images are acquired using a perspective projection. This is true for pin hole cameras but not necessary for cameras with a lens system

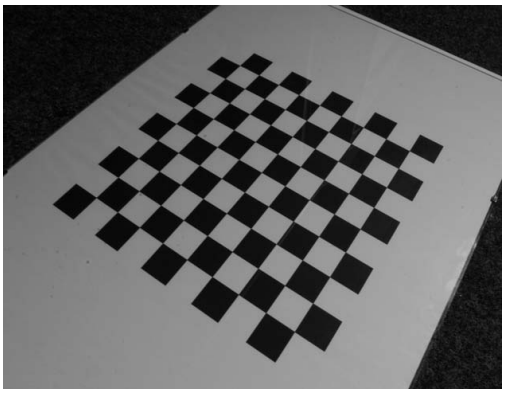


Figure 4.4: *Image of a calibration object taken with a 14 mm lens.*

which often show distortion effects. Good lens design can reduce the distortions within the limits of physics, but especially for wide angle lenses, a geometric correction needs to be applied to the acquired images.

The actual transformation of the camera's lens system is described by its intrinsic parameters. In order to measure the parameters several images of a checker board calibration object (see Figure 4.4) are taken with the same lens settings (i.e., object distance, focal length) that are used to capture the actual images of the object. An implementation of the Harris detector [Harris88] included in Bouguet's camera calibration toolbox [Bouguet00] is used to extract the feature points from the calibration images. As the geometry of the calibration object is known, it can be used to extract the intrinsic data such as the focal length and radial and tangential distortion coefficients using standard camera calibration techniques [Heikkila97, Tsai87b, Zhang99c, Bouguet00]. With this information the input images can be rectified resembling a perfect perspective projection.

4.6 Noise Reduction

Another issue to increase the quality of the input images is to remove noise. At room temperature uncooled CCD sensors can produce a significant amount of noise for exposure times larger than 1 s (see Figure 4.5). This noise seems to be due to "hot pixels" on the chip which collect charge even when no light is hitting them. The effect of fixed pattern noise for long exposures can be captured by a series of long exposed dark frame images. We used the technique presented by Goesele et al. [Goesele01b] to reduce the fixed pattern noise.

4.7 Photometric Camera Calibration

BRDF measurements require photometrically calibrated input data to reproduce an object's reflection properties faithfully. The input images should correctly represent the radiance reflected by the object. This implies three things: one would like to obtain linear response to the incoming radiance, the images should cover a high dynamic range, and the reported color values should be in a well defined color space independent from the applied light sources.

4.7.1 High Dynamic Range Imaging

The dynamic range of a camera, i.e., the ratio between the brightest and the darkest luminance sample that can be captured in a single image, is for most cameras quite small (on the order of $10^2 - 10^3$). As the dynamic range of an object with specular reflection properties can be much higher (e.g., about 10^6 between highlight and shadow regions), some techniques have to be used to capture the full dynamic range.

Several manufacturers have developed CMOS cameras that are capable of capturing a sufficiently large dynamic range by either combining multiple exposures or by the use of special imaging sensors. These cameras are typically video cameras and provide only a limited resolution. Furthermore, the measured values are quantized to 8–12 bits per pixel and color channel leading to a rather low precision. Since the applied Kodak camera does not provide such high dynamic range images, the full range of most scenes has to be captured by multiple images.

In the computer graphics community, several authors proposed methods to extend the dynamic range of digital images by combining multiple images of the same scene that differ only in exposure time. Madden [Madden93] assumes linear response of the imaging sensor and selects for each pixel an intensity value from the brightest non-saturated image. Debevec and Malik [Debevec97] and Robertson et al. [Robertson99] recover the response curve of the imaging system and linearize the input data before combining them into a single high dynamic range image. We used this approach for most of the measurements in the Chapter 6 and 7.

4.7.2 Color Issues

For a digital camera, the recorded color of an object depends on multiple factors: the spectral response of the object, the color of the light source, the properties of the optical system, the sensor, and the image processing steps applied by the camera itself or other software. The goal is to faithfully record the object's color independently of all these factors.

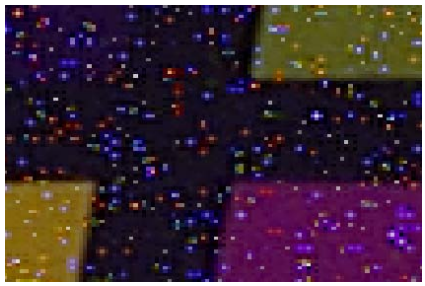


Figure 4.5: *CCD noise. Excerpt from an image of a GretagMacBeth ColorChecker taken with the DCS 560 at a film speed of 80 ASA with 25 s exposure time. The image shows four constant colored patches separated by a black cross. The noise is mainly due to "hot pixels" on the CCD chip.*

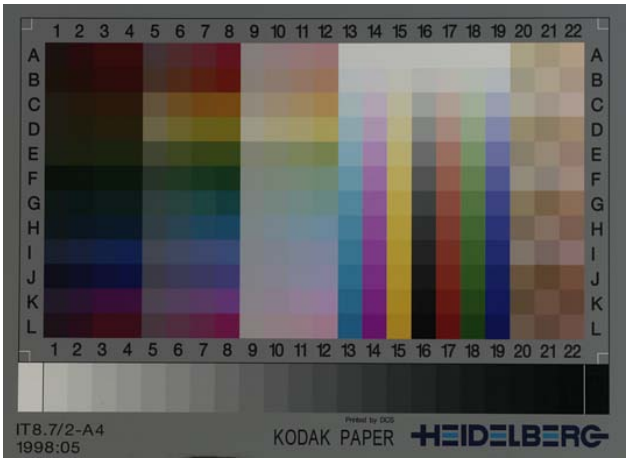


Figure 4.6: *IT8.7/2 target used to capture the color properties of an imaging system in order to generate an ICC profile.*

In an ideal case, one would like to accurately measure the continuous spectrum of the visible light. Measurement devices such as a spectrophotometer perform therefore a very dense sampling of the spectrum. In contrast to that, most digital cameras record only three color values per pixel (tristimulus values) [Hunt95] obtained by integrating the incident spectrum with the response curve of each CCD sensor. Since different spectra can result in the same tristimulus values (metamerism) [Luther27] color measurements done with a tristimulus device are always an incomplete representation of the actual spectrum.

Color Management Systems

In order to relate the recorded color to well defined standards, color management systems have become a standard tool. Hereby, an image of a test target with well known properties such as the IT8.7/2 target (see Figure 4.6) or the Macbeth ColorChecker is taken and processed in the same way as all later images. The target should ideally be a good representation of the colors in the considered scene. The relation between the color values of the test target patches and the color values reported by the camera is analyzed and used as calibration data. The International Color Consortium (ICC) introduced the ICC profiles [ICC98, Wallner] as a standard way to store this information. ICC profiles map input color to a linear standard color space, the profile connection space (PCS). Other profiles are used to convert data from the PCS into the color space of display or output devices. Goesele et al. [Goesele01a], have shown that this color space can be used to generate color calibrated high dynamic range images.

White Balance

A simpler calibration approach accounts only for the influence of the applied light source. Depending on the selected light source the tristimulus values are multiplied with constant factors so that the color of the light source is recorded as white. The influence of the light source on the recorded color of an object is hereby minimized. If the weights are not known beforehand they can be captured using a standard gray card as reference.

4.8 Conclusion

The main challenge in building the photo studio was to find equipment that suites our needs. As our requirement are in many cases different from the requirements of traditional photographers we sometimes had to search for a long time until we found a good solution. With the current equipment and the basic techniques explained above we are able to capture the appearance of 3D models with very high quality. How to use the equipment to acquire textures for real-world objects is explained in the next chapter.

Chapter 5
Texture Registration

In this chapter a system is presented that automatically registers and stitches textures acquired from multiple photographic images onto the surface of a given corresponding 3D model. Within this process the camera position, direction, and field of view must be determined for each of the images. For this registration, which aligns a 2D image to a 3D model we present an efficient hardware-accelerated silhouette-based algorithm working on different image resolutions that accurately registers each image without any user interaction. Besides the silhouettes, also the given texture information can be used to improve accuracy by comparing one stitched texture to already registered images resulting in a global multi-view optimization. After the image-to-geometry registration for each part of the 3D model's surface the view is determined which provides the best available texture. Textures are blended at the borders of regions assigned to different views.

5.1 Introduction

Throughout the past years 3D rendering solutions have advanced in rendering speed and realism. Because of this, there is also an increased demand for models of real world objects, including both the object's geometry and its surface texture. Precise geometry is typically acquired by specialized 3D scanners while detailed texture information can even be captured by consumer quality digital cameras. Only a few 3D scanning devices are designed to capture 3D geometry and 2D textures at the same time. Even if texture acquisition is supported it may be required to take the images under controlled lighting conditions with a special sensor implying that the object of interest has to be placed in a fully controllable environment while taking the pictures. In cases where photos and geometry are not acquired by the same sensor, the images must be registered with the 3D model afterwards in order to connect geometry and texture information.

For this registration task we present a hardware-accelerated algorithm that aligns an image to the 3D model as well as to other already registered images. All stages of the algorithm can run completely automatically. Alternatively, the user can skip some steps in the algorithm providing a rough alignment.

5.2 Overview / Contributions

Out of the set of the different tasks necessary to acquire a complete texture mentioned in the previous section, we present new solutions for the following ones:

- single view registration based on silhouettes (Section 5.4 and Section 5.5)
- global registration of multiple views with respect to image features (Section 5.7)
- view-independent assignment of surface parts to the images providing the best texture for the single part (Section 5.6)
- blending between textures at assignment boundaries (Section 5.6)

Although we briefly explain all necessary steps from image acquisition to rendering of the textured model, the main focus within this chapter is on novel techniques for image registration.

5.3 Camera Transformation

During registration we have to determine the camera settings for each image that maps it correctly onto the 3D model (Figure 5.1). In our system a pinhole camera model is assumed. Up to seven camera parameters are recovered: the field of view which is the only intrinsic parameter and is related to the focal length, and six extrinsic parameters describing the camera pose and orientation. All other intrinsic parameters like aspect ratio, principal point, or radial lens distortion are assumed to be constant and known since they can be obtained easily using common camera calibration tool kits (see Section 4.5), or they are simply ignored and set to reasonable approximate values.

The camera position is expressed by the translation vector $\vec{t_c} \in \mathbb{R}^3$, while the orientation of the camera is described by (ϕ_x, ϕ_y, ϕ_z), the rotation angles about the coordinate axes, which form a 3×3 rotation matrix R. These extrinsic parameters determine a rigid body transformation that maps a point in world coordinates $\vec{x}_w \in$

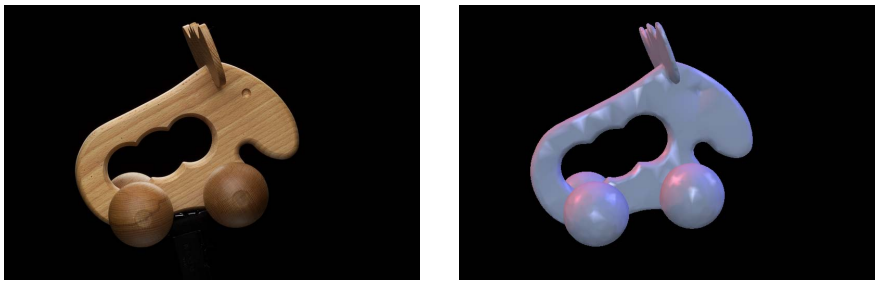


Figure 5.1: *Recovering the camera parameters for one image (left) allows to render the corresponding 3D model from exactly the same view (right).*

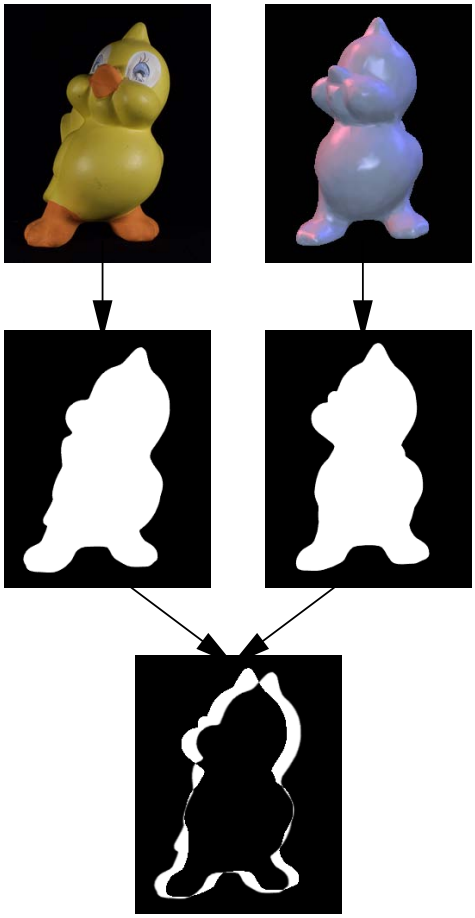


Figure 5.2: *Measuring the difference between the photo (left) and one view of the model (right) by the area occupied by the XOR-ed foreground pixels.*

$\mathbb{R}^3$ into camera coordinates $(x_c, y_c, z_c)^T$:

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = R\vec{x}_w + \vec{t_c} \tag{5.1}$$

For a camera far away from the object this representation has the disadvantage that a small rotation around the camera results in a large displacement of the object in camera coordinates. If the point $\vec{x}_w$ is instead rotated around the center of gravity $\vec{g}$ of the object the effects of rotation and translation are much easier to distinguish, thus simplifying the optimization [Neugebauer99]. The translation is now given by $\vec{t} = R\vec{g} + \vec{t_c}$ which actually is the position of the center of gravity in camera coordinates.

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = R(\vec{x}_w - \vec{g}) + R\vec{g} + \vec{t_c} = R(\vec{x}_w - \vec{g}) + \vec{t} \tag{5.2}$$

To fully describe the camera transformation the points $(x_c, y_c, z_c)^T$ are further mapped to 2D image space (u, v):

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} + \frac{1}{z_c} \begin{pmatrix} \frac{F}{\alpha} x_c \\ F y_c \end{pmatrix}, \tag{5.3}$$

where (u_0, v_0) is the principle point (in our case the center of the image), α the aspect ratio of width to height which must be provided by the user, F the focal length. F is related to the field of view f by $F = \cot \frac{f}{2}$. Thus, the camera transformation is determined by f and the vector $\pi = (\phi_x, \phi_y, \phi_z, t_x, t_y, t_z)^T$. For each image these seven parameters have to be recovered by a non-linear optimization of a similarity measure comparing the projected model to the object found in the image.

5.4 Similarity Measure

Since we want to optimize seven parameters $(\phi_x, \phi_y, \phi_z, t_x, t_y, t_z, f)$ we define a function $s : \mathbb{R}^7 \rightarrow \mathbb{R}$ which returns a scalar value for the specified camera transformation expressing the similarity of the projected model and an image, i.e., with a small value indicating high similarity while the value increases when the projected model and the image are misaligned. As this function s will be evaluated quite often during the optimization process it is necessary that it can be computed very quickly.

At first, we have to define in which way we want to measure the similarity, which feature space to be used. Since the 3D geometric model does not yet carry any color information we are restricted to geometric properties. In contrast to Neugebauer et al. [Neugebauer99] and Matsushita et al. [Matsushita99] who compared the contour of the projected model to the contour in the image, we decided to directly compare the silhouettes, which requires less computation. A silhouette is the object projected onto a plane filled with uniform color while a contour is the outline of the silhouette.

5.4.1 Segmentation

When rendering the model for a given view the silhouette can be generated simply by choosing a uniform white color in front of a black background. Extracting the contour instead of the silhouette would require further processing which we can avoid.

To compare silhouettes the second silhouette must be extracted from the image data. If the object is captured in front of a black background the image can be segmented automatically by histogram-based thresholding. The threshold is chosen directly after the first peak in the histogram which corresponds to the number of very dark pixels. If the contrast between the object and the background is too low (like in less controllable environments) other image processing techniques must be applied. For example the semi-automatic algorithm presented by Mortensen and Barret [Mortensen95] may be used to trace the contour in the image which afterwards can be filled automatically. This segmentation has to be done only once for each image before starting the actual optimization and thus user interaction seems acceptable in rare cases.

5.4.2 Silhouette Comparison

After extracting the silhouettes some kind of distance measurement between the silhouettes has to be defined. The technique presented here can be carried out completely by use of commonly available graphics hardware supporting histogram evaluation.

The first step renders the silhouette of the projected 3D model into the framebuffer. The result is then combined with the segmented image using a per-pixel XOR-operation. This process is visualized in Figure 5.2 where the silhouettes are computed for the photo and for one view of the 3D model and combined afterwards. After the XOR-operation exactly the pixels between the outlines remain white. Their number can be counted by simply evaluating the histogram of the combined image which is computed very efficiently by the graphics hardware.

For exact matches a value close to zero will be returned while the number of remaining pixels will be much larger if the rendered view of the model is different from that in the photo.

The computation time for the similarity measure is dominated by two quantities. The more important one is the resolution selected for rendering since each pixel of the XORed image will be processed during the computation of the histogram. The other quantity is the complexity of the 3D model in terms of the number of geometric primitives that have to be rendered to produce the model's silhouette.

5.4.3 Blurred Silhouettes

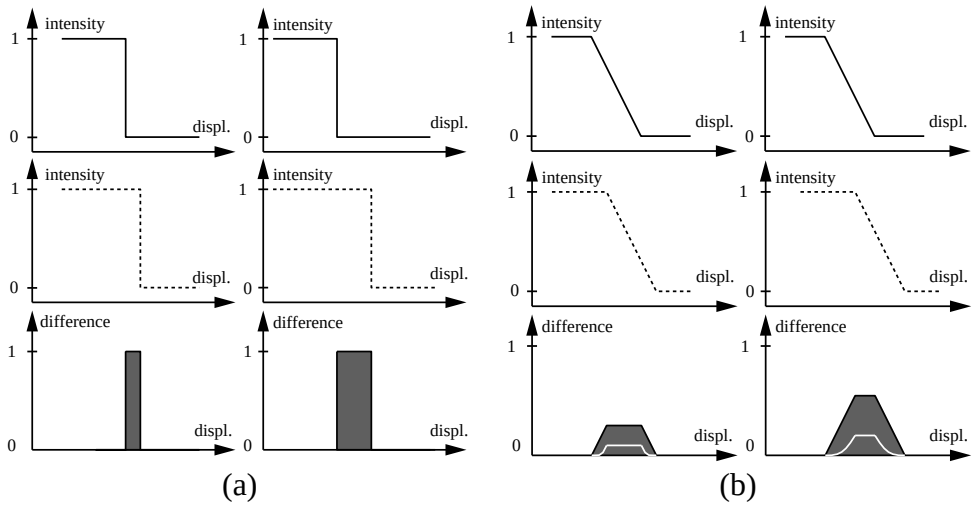


Figure 5.3: *(a) The integral of differences between a sharp intensity edge and the same edge slightly displaced (dashed) is proportional to the displacement. (b) Blurred edges also produce a linear distance measure. However, the differences between blurred edges can be squared before integration approximating a quadratic measurement (white line).*

Until now, we have considered monochromatic silhouette images with a sharp transition between the intensity of pixels belonging to the object and those belonging to the background. Suppose two sharp intensity transitions that are slightly displaced as depicted in Figure 5.3a. With increasing displacement, the integral of the differences of the two curves grows linearly while the differences are either one or zero. This exactly corresponds to the result of the presented similarity measurement based on XORed monochromatic silhouettes.

However, a measurement that is proportional to the squared distance between points on the outlines would be more desirable. In particular, we would like to

have a measure that introduces some notion of distance of points on one silhouette to points on the other silhouette, penalizing points far away from the other silhouette much more than points close to it. For arbitrarily large distances we cannot compute this error term efficiently, as this would require an extensive search for the closest point on the other silhouette (one-sided Hausdorff distance). For smaller distances, we can compute the squared distance metric simply by blurring the silhouette and then performing the same operations as above, counting the various distances for the individual pixels using a histogram. As can be seen in Figure 5.3b, even for blurred transitions the integral of the differences between the curves is proportional to the displacement. However, in this case the magnitude of the differences is also linear to the displacement in regions where the transitions overlap. These differences can be squared prior to the integration. This way, a quadratic distance measurement is approximated for edges as long as the displacement of the edges is smaller than the size of the filter kernel applied to blur the edges. Larger displacements are emphasized compared to smaller ones. This behavior can guide the optimization algorithm faster to the minimum. However, computing the differences between blurred images is slightly more expensive than just applying the XOR-operation and one can decide if it is worth the cost (see Section 5.8).

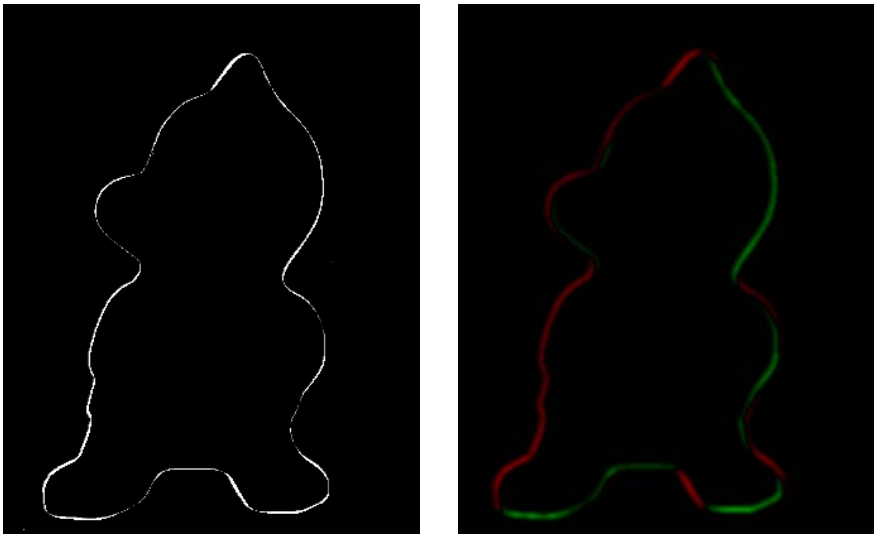


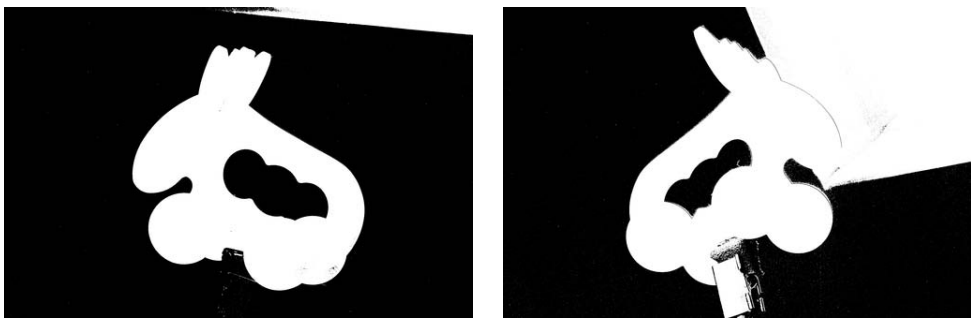
Figure 5.4: *XORed sharp silhouettes (left) and subtracted blurred silhouettes (right).*

To blur the silhouettes an $n \times n$ low-pass filter is applied (e.g. Gaussian). While this is no problem with respect to the photo since it is done before the optimization, the silhouette of the projected 3D model must be filtered again for

each view. Although convolution can be computed by the graphics hardware, it requires processing the entire framebuffer and thus slows down the evaluation of the similarity measure. After blurring the silhouettes the absolute difference values between them must be computed on a per-pixel basis. A special OpenGL extension allows to compute the positive difference of the framebuffer contents and an image by specifying a particular blending equation. Since only positive values are computed while negative values are clamped against zero we first render the silhouette of the 3D model minus the photo into the red channel and then the photo minus the 3D model's silhouette into the green channel of the framebuffer as can be seen in Figure 5.4. The histogram of the red and of the green channel are then combined to obtain the sum of the absolute values, and the approximate quadratic distance is computed.

5.4.4 Erroneous Pixels

For real photos the defined similarity measure will always return values much larger than zero no matter how close the determined view comes to the original view of the photo. There are always some pixels of the silhouette in the photo which are not covered by the projected 3D model or vice versa, originating from different sources of error. On one hand, the 3D model may be somewhat imprecise due to the acquisition. There may be even parts of the object visible in the image which are not part of the 3D model. On the other hand some pixels in the image may be wrongly classified by the automatic segmentation due to unfavorable lighting conditions. Additionally, in some views, parts of the object will be hidden by other objects.



(a) (b)

Figure 5.5: *Large regions of wrongly segmented pixels apart from the silhouette (a) and penetrating the silhouette (b).*

There are several possible ways to deal with these erroneous pixels: If the regions of erroneous pixels do not penetrate the silhouette of the object like holes

within the silhouette or bright regions in the image apart from the object (Figure 5.5a) the optimization is not affected since these pixels only add a constant bias to the histogram. If erroneous pixels disturb the outline they may lead to slight misregistration (Figure 5.5b). But the error may be corrected afterwards by comparing the registered texture to the texture of other views as explained in Section 5.7, or it may simply be ignored if it is only small. In cases where larger regions of the silhouettes are corrupted the erroneous pixels can be masked out and the histogram is evaluated only over those regions which provide reliable information. However, masking out the bad regions requires user interaction and thus should be avoided.

5.5 Non-linear Optimization

Let us assume a similarity measure s as defined in the previous section. To recover the correct transformation for a given image we have to find the pair $p_{min} = (\pi_{min}, f_{min})$ that minimizes s. Since s typically is non-linear and possesses several local minima, an appropriate optimization method must be applied. We chose and extended the *downhill simplex method* as it is presented in [Press94]. Other optimization techniques may be used as well but we found the simplex method easy to control. Furthermore, it does not require any partial derivatives, which makes it very efficient, even if the cost for evaluating the similarity measure is high, as in our case.

Since the original simplex method tends to converge too fast to local minima we extended it by some aspects similar to simulated annealing.

The simplex method for N dimensions takes $N + 1$ sets of camera parameters p_i (the vertices of a simplex) and evaluates the similarity measure $s(p_i)$ for each set. Then it tries to find a better solution for the currently worst p_{hi} by testing a set of positions: p_{hi} is reflected through the opposite face of the simplex and even further displaced in the same direction if a better result can be achieved. If this fails p_{hi} is gradually pulled towards the center of the simplex. If the solution could not be improved by these steps Press et al. [Press94] propose a contraction of all points towards the best point which unfortunately makes the method converge too fast to a local minimum. Therefore, we instead place p_{hi} randomly within an adaptively decreasing radius around the best point based on the distance between the remaining vertices. This radius corresponds to the temperature in simulated annealing. The system is cooled down quite slowly thus decreasing the speed of the optimization while increasing its stability.

5.5.1 Hierarchical Optimization

The algorithm still converges very quickly to a minimum that is not necessarily the global minimum. In order to find the global minimum we restart the optimization process several times. Hereby, the minimum found by the previous optimization is used as the next starting point. The radius of the initial simplex is of course reduced before each iteration to speed up the convergence. The optimization is stopped if it converged to the same minimum for the second time.

Another method to speed up the optimization and even to increase robustness is to run the optimization at different image resolutions. As pointed out in Section 5.4 the evaluation time for our similarity measurement depends on the used image resolution since the histogram has to count all pixels. Starting with low resolution the view can be approximated roughly but very quickly. For accurate registration the resolution is increased. At the same time also the tessellation of the object can be varied to gain a speedup. In our implementation we just used two different resolutions.

5.5.2 Generating a Starting Point

For the optimization it is important to have an appropriate starting point. A starting value for the field of view can be derived directly from the focal length of the applied lens which is reported by some digital cameras. This typically will not be the correct focal length since it is slightly changed by selecting the focal distance. Assuming that the entire object is visible in the image, an initial guess for the distance can be computed using the field of view and the size of the object. The x and y displacements are initially assumed to be neglectable.

What remains is to make a guess for the orientation. This is done by sparsely sampling the space of possible angular directions. We try three different angles for ϕ_x and four for both ϕ_y and ϕ_z yielding 48 samples. From each of these samples we start the simplex algorithm running at a rather low resolution and stop already after a few evaluations of the similarity measure. The best five results are selected and further optimized, this time allowing more evaluations at the same resolution. It turned out that the best of the computed minima is already quite close to the one we are searching for. With this value the final optimization can be started.

Of course the generation of the starting point takes some time, but it does not require any user interaction. Especially, there is no need to select pairs of corresponding points. However, time can be saved by manually moving and rotating the 3D model very roughly into a position similar to the photo.

For a fixed field of view the following steps are performed to recover the trans-

lation vector t and the rotation R given by ϕ_x, ϕ_y, ϕ_z: generate a starting position automatically like described above or select it manually, run the simplex method twice at low image resolution and then twice at the final resolution.

5.5.3 Optimizing the Field of View

Given the optimized parameter set π, the field of view f_{start} obtained from the camera and the result of the similarity measure $s(\pi, f_{start})$ we now try to find the best field of view f_{min} that further minimizes $s(\pi', f_{min})$ where π' is only slightly changed compared to the previous π. This problem is a search in only one dimension for which we implemented a simple algorithm.

Let us start with f set to f_{start}. At first, f is increased by an amount d yielding a new f'. All other parameters are simply copied from π to π'. Then the distance t_z in π' is updated to compensate for the change in the field of view in such a way that the size of the projected object approximately remains the same for the new f'. To π' and f' the simplex method is applied allowing only a few evaluations of the similarity measure. This yields an optimized parameter set π'_{opt}. This optimization is necessary to slightly correct π' since a wrong field of view will lead to a wrong registration in the other parameters, too. If by increasing f a better field of view was found ($s(\pi'_{opt}, f') < s(\pi, f)$) the field of view is increased and the algorithm is repeated, starting with (π'_{opt}, f'). Otherwise, we divide the increment d by two, step back to the predecessor of the last field of view and proceed with the search until d is sufficiently small. If no better field of view can be found by increasing f_{start} the algorithm is just applied into the other direction, decreasing f_{start}.

Using this algorithm it is possible to determine the best field of view for each photograph independently. This allows selecting a different focal distance or even different lenses for each view in contrast to previous approaches in which the field of view had to be fixed [Neugebauer99, Matsushita99].

5.6 Texture Stitching

After determining the correct viewing parameters for an image, it can be stitched as a texture onto the surface of the 3D model. In this section a triangular mesh is assumed although the presented ideas can easily be adapted to other surface representations as well.

5.6.1 Single View Processing

Given the viewing transformation the set of visible vertices of the 3D model can easily be determined either by casting a ray from the view point to the vertex and

testing for occlusion or by a simple z-buffer depth test. For all visible vertices a texture coordinate into the image is computed by projecting the vertex into the image plane using the recovered camera transformation. Additionally, the viewing angle (angle between the vertex normal and the viewing direction) is determined for each vertex using the averaged normal of the surrounding triangles. From this data the set of usable vertices is derived. A vertex is declared valid only if the viewing angle is smaller than 85 degrees, the depth variation around that point is not too steep, and the point does not lie exactly on the outline of the projected object. Using this criterion texture mapping artifacts can be avoided when viewing the textured object from other views than the determined one.

Based on the set of valid vertices those triangles can be selected for which reliable texture information is available. A triangle is used only if all its vertices are valid.

5.6.2 Combining Multiple Textures

If multiple images are involved, the sets of valid triangles will overlap and the best assignment of triangles to images must be determined. A static decision can be again made by inspecting the angle under which the triangle is seen in each image. Each triangle is assigned the texture from that image in which it possesses the largest viewing angle.

There will be triangles that are assigned to one image while an adjacent triangle is assigned to another image (Figure 5.6a). This often results in a visible discontinuity in the texture even if the images are taken without changing the lighting conditions. A smooth transition is achieved by blending between the textures across the border triangles. This requires all boundary triangles to be valid also for adjacent textures. To ensure this the set of valid triangles for each image is reduced prior to the assignment to the images. All those triangles are invalidated which have at least one invalid triangle as their neighbor.

Next, the triangles must be determined across which to blend. All triangles containing a boundary vertex are possible candidates for the blending (Figure 5.6b). They are rendered once for each adjacent texture using appropriate alpha values at the vertices to gain correct blending. The assignment of alpha values for each vertex for each image is as follows. For each boundary vertex it is decided in which image it is best represented. For the best image the vertex is assigned an alpha value of one, while for all other images it is set to zero. For all surrounding vertices, that are not boundary vertices the alpha value is set to one if the vertex belongs to a triangle that was previously assigned to the current texture (Figure 5.6c).

Rendering the textured triangles with these alpha values results in a smooth transition. Unfortunately, the blending takes place across the width of only one

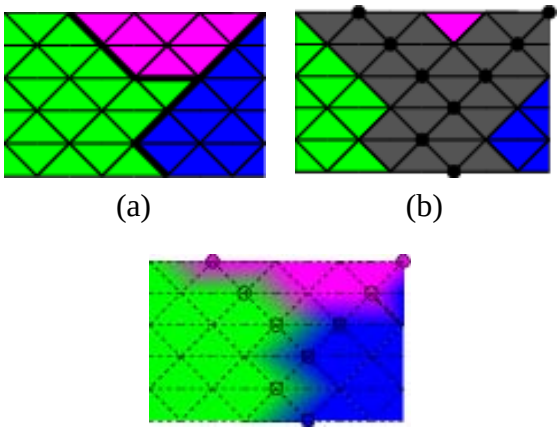


Figure 5.6: *(a) Adjacent triangles textured using different images. (b) Possibly blended triangles shaded grey. (c) Each boundary vertex is assigned to one image and textures are blended.*

Figure 5.7: *All image information of three images packed into a single texture. The texture consists of the relevant rectangular regions of the original images and of the blended textures for single triangles. (Although the final texture consists of 15 images, only three pictures were considered in this case to better visualize the layout of original image parts and blended triangles.)*

triangle. If the object is finely tessellated the blending area will become rather small and contrasting textures are still not sufficiently separated. This problem can be solved by computing the blending on an object with coarser tessellation and assigning interpolated alpha values to the vertices of the fine subdivided mesh.

5.6.3 Organizing the Texture

Blending between the different images during rendering is of course cumbersome because it must be recomputed for every new view. Some of the triangles have to be rendered up to three times with three different textures requiring different sets of texture coordinates for each texture. Therefore, it is more practical to precompute the blended texture for each border triangle and store it into a new texture allowing to render all triangles within just one pass.

This leads to a considerable number of small images, one image per blended triangle. Instead of using a large number of images, meaning a large number of textures, we copy the relevant parts of the original images and the blended texture triangles into a large texture as demonstrated in Figure 5.7. A similar technique has also been applied by Rocchini et al. [Rocchini99].

Unfortunately, the texture patches belonging to triangles adjacent in the mesh are not necessarily neighboring in the resulting texture. To enable correct bilinear texture filtering across the shared edges of the triangles during rendering one must providing a border of at least one additional pixel around each texture triangle.

In addition to this filtering problem, the shared vertices need two different sets of texture coordinates for the triangles. Since rendering is much easier when only one texture coordinate is assigned to each vertex we double those vertices and assign different texture coordinates to each of the copies. Then, the first triangle is constructed using the original vertices and the second using the copies. Introducing copies of vertices unfortunately destroys the topology of the mesh.

By selecting the relevant rectangular regions of the original images and by intelligent packing the size of the resulting texture can be significantly decreased. Additionally, having just one texture facilitates representing the textured object using standard formats as VRML.

5.7 Multiple View Registration

When the texture is combined from multiple views a slightly misaligned image can produce visible artifacts since image features blended between two images may not be aligned. The circumstances which can lead to misalignment when only one view is considered are mentioned in Section 5.4. If we have multiple already registered views an additional similarity measurement s_{tex} can be defined which does not compare silhouettes but the texture of one view to the texture obtained by another view. This results in a global optimization taking into account all views.

5.7.1 Texture Comparison

Given the parameters (π_1, f_1) and (π_2, f_2) of two registered views and the sets T_1 and T_2 of valid triangles, the quality of the registration can be measured by comparing the textures mapped, one in turn, onto the set of overlapping triangles $T_1 \cap T_2$. The triangles are rendered from the view specified by the averaged parameters $\left(\frac{\pi_1+\pi_2}{2}, \frac{f_1+f_2}{2}\right)$. Choosing the averaged view results in similar loss of quality due to distortion and resampling in both textures.

In the case of a perfectly diffuse surface the textures mapped onto $T_1 \cap T_2$ will look identical, whereas specularity leads to view-dependent highlights which occur in different locations on the surface for different views. To get less view-dependent textures the color images are transformed into the HSV color space which separates the brightness (value) from the hue and the saturations. Only the hue and/or saturation-channel are used for comparison avoiding the influence of view-dependent brightness. Of course, also other methods can be applied to create view-independent textures like the one presented in [Neugebauer99], but they tend to be more expensive. However, the hue channel of the two textures can now be compared like the intensity values of two different blurred silhouettes in Section 5.4. At first, the positive difference of the first texture minus the second texture is rendered into the red channel of the frame buffer and then the reversed difference is rendered into the green channel. Summing up the histogram weighted by the difference values yields a value that becomes minimal when the two views are perfectly aligned. This measure $s_{tex}(\pi_1, f_1, \pi_2, f_2)$ allows to register multiple views with respect to each other.

5.7.2 Iterative Global Optimization

A registration of multiple views starts with the separate registration of each view based on the silhouette as described in Section 5.5. After the single-view registration the sets of valid triangles are determined and texture coordinates are computed for the vertices. For each pair of views (i, j) the set of overlapping triangles $T_i \cap T_j$ is determined and the averaged parameters (π_{ij}, f_{ij}) are calculated. For these pairs an initial measurement $s_{ij} = s_{tex}(\pi_i, f_i, \pi_j, f_j)$ is evaluated.

Successively each view i is selected and the set of other views V_i is determined which are sharing overlapping triangles with i. We can now optimize the following function:

$$s_{\text{multiview}}(\pi_i, f_i) = \sum_{j \in V_i} \frac{s_{tex}(\pi_i, f_i, \pi_j, f_j)}{s_{ij}} \tag{5.4}$$

Again, the extended downhill simplex method presented in Section 5.5 can be applied, this time calculating new texture coordinates and evaluating

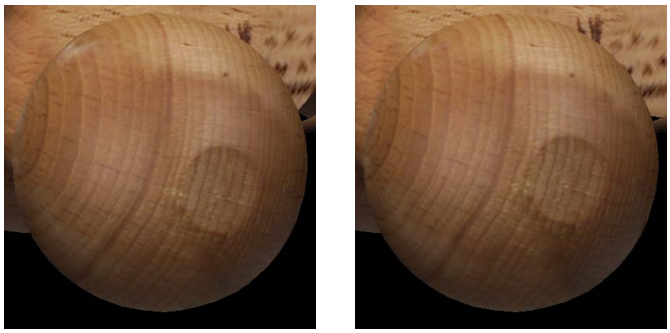


Figure 5.8: *Comparing registration results: Due to an inexact 3D model or disturbed silhouettes, the purely silhouette-based registration leads to artifacts in the texture (left). Global optimization using texture comparison results in a better alignment of the features (right). Note that the entire elk model was considered during both the silhouette-based registration and the global optimization.*

$s_{\text{multiview}}(\pi_i, f_i)$ for each try. Since the changes in π_i are expected to be rather small a simplex with small radius is constructed around π_i and the optimization is already stopped after a few evaluations of $s_{multiview}$. Iterating this process several times over all views until no further updates are performed will produce the best possible registration regarding the surface textures.

One result of this approach is shown in Figure 5.8 where you can see the result of the purely silhouette-based registration on the left and the globally optimized registration using texture comparison on the right. Note the better quality of the dark line across the wheel which is now continuous.

5.8 Results

The presented methods were applied to two different objects, a bird and an elk. The models were acquired using a Steinbichler Tricolite 3D scanner. The bird's model consists of around 7000 triangles while the elk is tessellated more finely with nearly 11000 triangles. The images were taken with a Kodak DCS 560 digital camera that yields an image resolution of 3040x2008 pixels which we reduced to 1024x676 since the applied graphics hardware cannot deal with larger textures. We run the optimization on a SGI Octane equipped with a MXE graphic board containing 8MB of texture RAM.

In Figure 5.9 the results after automatic registration and stitching of several images onto the models are shown and compared to real photos that were not used for generating the texture. The elk texture consist of 15 different images taken with two different lenses and at different object distances. The bird was textured

photo texture

Figure 5.9: *Novel viewpoint. Left column: photo that was not used to generate the texture. Right column: synthetic model rendered with the generated textures.*

Figure 5.10: *Texture Alignment. View of the right front wheel. Several textures are so accurately aligned that even fine lines in the wood's structure are preserved.*

using only 10 images.

mode	x·y	value	t_x	t_y	t_z	ϕ_x	ϕ_y	ϕ_z	sec.
XOR	500x330	avg	7.4756	-5.5169	704.69	-118.956	-43.5465	-119.326	40
		var	0.0059	0.0065	1.9769	0.4664	0.0864	0.2826	
XOR	1000x660	avg	7.5548	-5.5596	706.19	-119.32	-43.519	-119.719	130
		var	0.0034	0.0030	2.4188	0.2151	0.0938	0.1547	
blurred	500x330	avg	7.2606	-5.6441	706.57	-117.479	-43.0215	-118.237	39
		var	0.0007	0.0042	0.1653	0.0381	0.0163	0.03054	
blurred	1000x660	avg	7.3034	-5.6764	706.66	-117.667	-43.0383	-118.386	104
		var	0.0041	0.0009	0.3010	0.1582	0.0676	0.1501	

Table 5.1: *Average value and variance value of the recovered camera parameters and the required time applying the XORed and blurred silhouette matching algorithm for different resolutions. The optimization has been started several times from different positions.*

The synthetic results compare really well to the photos although two kinds of artifacts are visible. At the top of the antler some triangles are not textured because they are too close to the outline in each incorporated image. Due to imprecise geometry some of their vertices are wrongly classified to be outside of the silhouettes and the triangles are discarded.

The other artifacts are due to the non-diffuse surface reflectance. Even though the position of the lights was not changed during the acquisition, specular highlights result in brightness differences among the acquired images as can clearly be seen in Figure 5.10. To further reduce these lighting artifacts a purely diffuse texture, or even better the BRDF, would have to be computed incorporating samples from all acquired pictures.

The precision of the presented algorithm is visualized in Figure 5.10 where the right front wheel of the elk is shown. The wheel is actually textured by at least six different images. Although the texture of the wheel is composed using several different views, the fine lines of the wood's structure are completely preserved, indicating a very accurate registration. This accuracy may also be achieved using outline-based algorithms [Neugebauer99, Matsushita99] since outlines and unblurred silhouettes carry almost the same information.

	image proc.	start pos.	opt.	FoV	stitching	total
10 images	235	826	239	365	12	1677
average	23.5	82.6	23.9	36.5	1.2	155
15 images	359	1660	536	1250	21	3826
average	23.9	110.6	35.7	83.3	1.4	255

Table 5.2: *Registration timings (in seconds) for the bird (top rows) and the elk (bottom rows).*

When comparing the XOR and blurred matching methods, it can be seen from Table 5.1 that the blurred silhouette method leads to superior results. The variance of the recovered parameters is generally decreased, often by one order of magnitude. From our experiments, it could also be observed that although the computation of the similarity measure is computationally more expensive, the optimization converges more quickly for non-ideal starting points.

Table 5.2 lists the time (in seconds) needed for the registration task of the bird and elk models. The registration of the bird took around 28 minutes, while the elk took 64 minutes since more texture information and a more complex geometric model were used and the resolution used for the final optimization was increased (bird: 500x300, elk: 800x528). The images are first loaded and processed to extract the silhouettes, then a starting point for the optimization is generated, the optimization is run for recovering the position and orientation, the field of view is determined, and finally the textures are stitched onto the model. Most of the time is spent for finding an appropriate starting position and for determining the field of view. Still the silhouette-based algorithm 3 to 5 times faster than previously presented outline-based algorithms [Neugebauer99, Matsushita99] which do not optimize for the field of view.

Time could be saved by manually selecting a good starting position. However, it turned out that the optimization of the pose and orientation after manual alignment consumed more time (around one minute) since the starting point for the optimization is not as precise as the automatic method. Further time could be saved by fixing the field of view during the acquisition, since in that case the field of view has to be determined only once.

Most of the results presented so far were computed without using the texture-based multi-view optimization (see Section 5.7). It turned out that the purely geometry-based registration already produces results of very high accuracy if the input silhouettes and the 3D model are precise.

However, in cases where some of the input images are misaligned due to imprecise 3D geometry or partially wrong silhouettes (see Figure 5.8), it is worth spending additional time on the texture-based matching since it corrects for inaccuracies in the silhouettes if the texture contains enough features.

Comparing textures may also help to correctly register the textures of symmetric objects where the silhouette-based optimization can fail, e.g. for registering a sphere some manual work is required at least to provide a meaningful starting point for the optimization.

5.9 Conclusions

We have described a number of novel techniques to register and to stitch 2D images onto 3D geometric models. The camera transformation for each image is determined by an optimization based on silhouette comparison. If the resulting alignment is not accurate enough, further optimization based on texture information is possible. Using the recovered camera transformation, the image is stitched onto the surface. Finally, for multiple views, an algorithm is presented that produces smooth transitions between textures assigned to adjacent surface regions on the model. The presented methods do not require any user interaction during the entire processing. The methods work efficiently, exploit graphics hardware features and result in very accurately aligned textures.

Further aspects of obtaining surface textures from input images are increasing the quality of the extracted texture with respect to bi-linear interpolation applied during rendering [Hakura01] or to synthesize detail in regions where the input images only provide low resolution due to a steep viewing angle [Ismert03]. During 2D-3D registration Kurazume et al. [Kurazume02] even investigated the alignment of image features to geometric features.

One issue that will be addressed in the remainder of this thesis is that differences in the brightness of input images due to specularity are still visible. To further improve the quality of the results, the reflective properties of the surfaces would also have to be considered which is done in the next chapter. The techniques for image-to-geometry registration presented in this chapter are used in the following approaches as well.

Chapter 6

BRDF Measurement by Clustering

Based on the insights of the previous chapter realistic appearance can only be reproduced if the reflection properties of the object are measured. Real-world objects are usually composed of a number of different materials that often show subtle changes even within a single material. Photorealistic rendering of such objects requires accurate measurements of the reflection properties of each material, as well as the spatially varying effects. We present an image-based measuring method that robustly detects the different materials of real objects and fits an average bidirectional reflectance distribution function (BRDF) to each of them. In order to model local changes as well, we project the measured data for each surface point into a basis formed by the recovered BRDFs leading to a truly spatially varying BRDF representation. Real-world objects often also have fine geometric detail that is not represented in an acquired mesh. To increase the detail, we derive normal maps even for non-Lambertian surfaces using our measured BRDFs. A high quality model of a real object can be generated with relatively little input data. The generated model allows for rendering under arbitrary viewing and lighting conditions and realistically reproduces the appearance of the original object.

6.1 Introduction

In this chapter we concentrate on the acquisition of realistic materials. In particular, we describe an acquisition process for spatially varying BRDFs that is efficient, reliable, and requires little manual intervention. Other methods described in the literature (see Chapter 3 for an overview) either focus on homogeneous materials, or make assumptions on the type of material to be measured (e.g. human

faces). In our work, we measure spatially varying BRDFs without making any additional assumptions. Further, we use the derived reflection properties to compute normal maps even for non-Lambertian surfaces. In particular, our contributions are

- efficient, reliable, and mostly automatic calibration schemes for the light source position relative to the geometry,
- a robust and efficient BRDF fitting process that clusters the acquired samples into groups of similar materials and fits a Lafortune model [Lafortune97] to each group,
- a method that projects every sample texel into a basis of BRDFs obtained from the clustering procedure. This projection accurately represents the material at that point and results in a compact representation of a truly spatially varying BRDF,
- an algorithm that uses the reconstructed BRDF at every point together with the measured reflectance samples to optimize the orientation of the surface normal yielding a normal map.

We require only a relatively small number of high dynamic range photographs (about 15-25 images for one object), thereby speeding up the acquisition phase.

As a result of the fitting, clustering, and projection process, we obtain a compact representation of spatially varying materials that is well suited for rendering purposes (see Figure 6.12 for an example). The method works both for objects consisting of a mixture of distinct materials (e.g. paint and silver, see Figure 6.14), or for smooth transitions between material properties.

6.2 Acquisition

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 the measurement lab described in Chapter 4 (see Figure 6.1) using a Kodak DCS 560 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

Figure 6.1: *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.*

a) b) c)

Figure 6.2: *Image series captured for one camera/lighting position. a) Silhouette image and reconstructed silhouette. b) Two images for recovering the light source position (see Section 6.3). c) Photograph samples with varying exposure time for HDR image reconstruction.*

however easily possible since only a few different views are required. Several constraints should be met when selecting the views to obtain the best quality:

- all surface points should be visible in more than one image,
- the position of the camera and the light source should be varied in order to provide different pairs of viewing and lighting directions, and
- at least one highlight should be observed in each material.

These considerations may slightly increase the number of images that are required

to reproduce objects with a larger number of different basis materials. In our experience their number is typically small.

For each view we acquire three sets of images: one image of the object's silhouette to register the 3D model with the image (Figure 6.2a), and two images to recover the light source position as explained in Section 6.3 (Figure 6.2b). 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 time [Debevec97, Robertson99] (Figure 6.2c). We use an entropy-based dark frame subtraction algorithm [Goesele01b] to remove dark current CCD noise.

Once per session a 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 [Zhang99b] is used to recover the intrinsic camera parameters.

To register the images with the 3D model we use the silhouette-based method presented in the previous chapter that yields the camera position relative to the object for a single view. Given the 3D model of the object the registration is performed by aligning the captured silhouette with the silhouette from a virtual view of the 3D model. The final view is found by minimizing the difference between the two silhouettes.

6.3 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 resting on a metal bar, which was very precisely manufactured using a CNC milling machine. This ensures that the centers of the spheres are on a straight line and that their respective distances are known.

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 by automatically fitting an ellipse to them.

Given the intrinsic parameters of the camera, the pixel coordinates of the reflections of the ring flash define rays $\hat{r}_i$ in space on which the centers of the spheres $\vec{p}_i$ are located (see Figure 6.3). Knowing the distance $\|\vec{d}\|$ between the spheres one can triangulate their positions relative to the camera $C\vec{O}P$ by computing the least squares solution to the following system of equations:

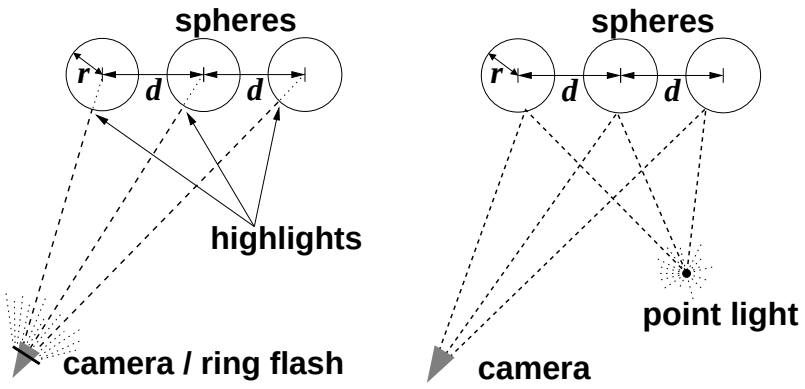


Figure 6.3: *Left: the ring flash mounted on the camera yields a highlight in the center of the spheres. Right: rays from the camera to the light source highlights will be reflected to the point light.*

$$\vec{p_i} = \vec{p_0} + i \cdot \vec{d} \tag{6.1}$$

$$\vec{p_i} = C\vec{O}P + n_i \cdot \vec{r_i} \tag{6.2}$$

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 reflected rays (see Figure 6.3). To increase the stability of this method we use more than the three necessary spheres and compute a linear least squares approximation.

6.4 Resampling of Reflectance Values

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 all available information into a data structure we call a *lumitexel*.

One lumitexel, denoted by $\mathcal{L}$, is generated for every visible surface point. Each lumitexel stores the geometric and photometric data of one point, i.e., its position $\vec{x}$ and the normal $\hat{n}$ in world coordinates. Linked to the lumitexel is a list of reflectance samples $\mathcal{R}_i$, each representing the measured reflectance r of the surface point captured by one image plus the direction of the light $\hat{l}^{(g)}$ and the viewing direction $\hat{v}^{(g)}$ which are further transformed into the local coordinate frame of the surface point spanned by $\hat{n}$ and a deterministically constructed tangent and bi-normal yielding $\hat{l}$ and $\hat{v}$.

A lumitexel can be seen as a very sparsely sampled BRDF. We define the error between a given BRDF f and a lumitexel $\mathcal{L}$ as:

$$E_f(\mathcal{L}) = \frac{1}{|\mathcal{L}|} \sum_{\mathcal{R}_i \in \mathcal{L}} s \cdot I(f(\hat{l}_i, \hat{v}_i) l_{i,z}, r_i) + D(f(\hat{l}_i, \hat{v}_i) l_{i,z}, r_i), \tag{6.3}$$

where $|\mathcal{L}|$ stands for the number of reflectance samples linked to the lumitexel, $I(r_1, r_2)$ is a function measuring the intensity difference, and $D(r_1, r_2)$ measures the color-difference. We introduce the weight s, to be able to compensate for noisy data (e.g., a slightly wrong normal resulting in a wrong highlight). In order to emphasize the color-difference we always set $s \leq 1$. Note that the cosine between the normal and the local light direction l_z is already included in our reflectance samples r such that the BRDF f has to be multiplied by it.

6.4.1 Assembling Lumitexels

Collecting all reflectance samples for a lumitexel requires a resampling of the input images for the particular point on the surface. First, one has to determine the set of surface points for which a lumitexel should be generated. In order to obtain the highest quality with respect to the input images, the sampling density of the surface points must match that of the images.

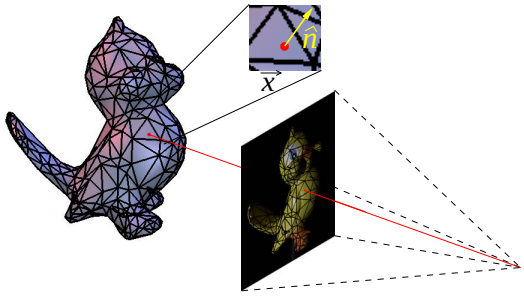


Figure 6.4: *The correspondence between pixel position and point position $\vec{x}$ on the object is computed by tracing a ray through the image onto the object. At every $\vec{x}$ a local normal $\hat{n}$ can be computed from the triangle's vertex normals.*

Every triangle of the 3D model is projected into each image using the previously determined camera parameters. The area of the projected triangle is measured in pixels and the triangle is assigned to the image I_{best} in which its projected area is largest. For every pixel within the triangle in I_{best} a lumitexel is generated.

The position $\vec{x}$ of the surface point for the lumitexel is given by the intersection of the ray from the camera through the pixel with the mesh (see Figure 6.4). The normal $\hat{n}$ is interpolated using the triangle's vertex normals.

A reflectance sample $\mathcal{R}_j$ is now constructed for each image I_j in which $\vec{x}$ is visible from the camera position and the surface point is lit by the point light source. The vectors $\hat{l}_j$ and $\hat{v}_j$ can be directly calculated. The associated reflectance is found by projecting $\vec{x}$ onto the image plane and retrieving the color c_j at that point using bilinear interpolation. Note, that for I_{best} no bilinear interpolation is necessary and c_{best} can be obtained without resampling since $\vec{x}$ exactly maps to the original pixel by construction. The reflectance r_j of the reflectance sample $\mathcal{R}_j$ is obtained by scaling c_j according to the brightness of the light source and the squared distance from the light source to $\vec{x}$.

6.4.2 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. 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. Reflectance samples would then be assigned to completely wrong surface points or lumitexels.

Furthermore, since a sensor element of the camera always integrates over a finite area, the reflectance 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.

The depth discontinuities are detected using the following approach: A depth map of the registered 3D model is rendered and subsequently blurred using an averaging filter. This changes the depth values of pixels near depth discontinuities while pixels showing a flat surfaces will not be affected. Regions where the filtered depth map deviates more than a small threshold from the original one will not be considered for further processing. The threshold can be computed given the filter size and the difference of two adjacent depth values that should be detected as a discontinuity.

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. They can be determined by a filtered shadow map.

The results of removing samples at depth discontinuities are displayed in Figure 6.5. Note that the dark stripes on the dress and across the hand have been removed by this step.

Figure 6.5: *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.*

Figure 6.6: *Problems due to non-modeled geometry. Parts of the object's surface may be occluded or shadowed by geometry that is not accounted for in the 3D mesh. It results in a disturbed texture, see the dark regions at the bottom.*

6.4.3 Problems With Non-modeled Geometry

Another problem arise with real geometry not represented by the triangle mesh. In Figure 6.6 the problem is apparent. The supporting rod is not represented in the geometric model. Furthermore, the relative position of the rod with respect to the object may be different in each captured view. Thus it is not possible to automatically detect occlusions and shadows cast by non-modeled geometry. One solution to this problem is to invalidate the affected regions in the input images and to ignore the corresponding reflectance samples.

6.5 BRDF Fitting

In this section we first detail the Lafortune BRDF model [Lafortune97] that we fit to our given lumitexels. Then we explain how this fit is performed using Levenberg-Marquardt optimization.

6.5.1 Lafortune Model

BRDFs are four-dimensional functions that depend on the local viewing and light direction. The dependence on wavelength is often neglected or simply three different BRDFs are used for the red, green, and blue channel. We use the latter approach.

Instead of representing a measured BRDF as a 4D table, the measured samples are in our case approximated with a parameterized BRDF model. This has two advantages: Firstly, the BRDF requires much less storage since only the parameters are stored and secondly, we only require a sparse set of samples that would not be sufficient to faithfully represent a complete tabular BRDF.

Many different BRDF models have been proposed (see Section 2.7, e.g., [Torrance67, Ward Larson92]) with different strengths and weaknesses. Our method may be used together with any parameterized BRDF model. We have chosen the computationally simple but general and physically plausible Lafortune model [Lafortune97] in its isotropic form, i.e., the orientation of the tangent and bi-normal within the tangential plane has no influence on the resulting reflectance (compare to Equation 2.31):

$$f(\hat{l},\hat{v}) = \rho_d + \sum_i [C_{xy,i}(l_x v_x + l_y v_y) + C_{z,i} l_z v_z]^{N_i}. \tag{6.4}$$

This model uses only a handful of parameters: $\hat{l}$ and $\hat{v}$ are the local light and viewing directions, ρ_d is the diffuse component, N_i is the specular exponent, the ratio between $C_{xy,i}$ and $C_{z,i}$ indicates the off-specularity of lobe i of the BRDF. The

sign of $C_{xy,i}$ makes the lobe i either retro-reflective (positive $C_{xy,i}$) or forward-reflective (negative $C_{xy,i}$). The albedo of the lobe i is given by the magnitude of the parameters $C_{xy,i}$ and $C_{z,i}$. From now on we will denote the BRDF with $f(\vec{\beta}; \hat{l}, \hat{v})$, where $\vec{\beta}$ subsumes all the parameters of the model, i.e., ρ_d, $C_{xy,i}$, $C_{z,i}$, and N_i. In the case of only one lobe $\vec{\beta}$ is 12-dimensional (4 parameters for each color channel).

6.5.2 Non-Linear Fitting

The Lafortune BRDF is non-linear in its parameters, which means that we have to use a non-linear optimization method to fit the parameters to the given data. As in the original work by Lafortune et al. [Lafortune97], we use the Levenberg-Marquardt optimization [Press94] to determine the parameters of the Lafortune model from our measured data.

We ensure that the fitting process works well and does not get stuck in undesired local minima by initializing the fitting routine with parameters that correspond to an average BRDF.

The Levenberg-Marquardt optimization outputs not only the best-fit parameter vector $\vec{\beta}$, but also a co-variance matrix of the parameters, which provides a rough idea of the parameters that could not be fit well. This information is used in our splitting and clustering algorithm, as explained in the next section.

6.6 Clustering

In this section we will explain how we cluster the given lumitexels so that each cluster C_i corresponds to one material of the object. Given a set of BRDFs $\{f_i\}$, each cluster C_i consists of a list of all the lumitexels $\mathcal{L}_i$ for which f_i provides the best approximation. Determining these clusters is a problem closely related to vector quantization [Gersho92] and k-means clustering [Lloyd82, MacQueen67], both of which work in affine spaces. Unfortunately, we do not have an affine space when clustering BRDF samples since there is no meaningful distance measure for BRDF samples with arbitrary viewing and lighting directions. Therefore we are employing a modified Lloyd [Lloyd82] iteration method.

The general idea is to first fit a BRDF f to an initial cluster containing all the data. Then we generate two new BRDF models f_1 and f_2 using the co-variance matrix from the fit (explained in more detail below) representing two new clusters. The lumitexels $\mathcal{L}_i$ from the original cluster are then distributed according to the errors $E_{f_1}(\mathcal{L}_i)$ and $E_{f_2}(\mathcal{L}_i)$ into the new clusters. We then recursively choose another cluster, split it, and redistribute the lumitexels and so on. This is repeated until the desired number of materials is reached, as detailed in Section 6.6.4.

6.6.1 Lumitexel Selection

The fitting procedure described in Section 6.5 performs a relatively large number of operations per reflectance sample. Thus, it is expensive to fit a BRDF using all lumitexels (and all reflectance samples contained in the lumitexels) generated by the assembling procedure. Instead, it is sufficient to consider only a few thousand lumitexels at the beginning. Later on, we increase the number for an accurate fit.

A first, naive approach to choosing this subset for fitting selects every n-th lumitexel regardless of its reliability or possible contribution. However, as stated in [Yu99] and [Schirmacher99], for a robust estimation of the specular part of a BRDF it is very important to include reflectance samples within the specular lobe of the material. Unfortunately, these brightest pixels statistically also carry the largest error.

Following these ideas we select more lumitexels in areas where a highlight is likely to occur. These areas are determined by the surface normal, the light source position and a synthetic BRDF with a broad highlight.

6.6.2 Splitting

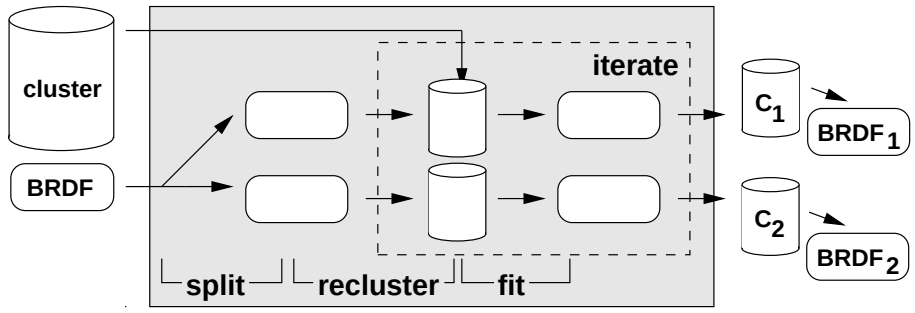


Figure 6.7: *Split-recluster-fit process (SRF). The initial BRDF is split into two new BRDFs using the co-variance matrix. The lumitexels from the initial cluster are distributed according to their distance to the BRDFs. Then we fit the BRDF again to each new cluster. We now iterate the reclustering and fitting until the resulting BRDFs and clusters have converged.*

Fitting just a single BRDF to the initial cluster of course is not sufficient if the concerned object consists of more than one material. Rather, we have to recursively split the clusters to account for the different materials comprising the object. We decide which cluster to split, by computing the following error for all clusters C_j:

$$E(C_j) = \sum_{\mathcal{L}_i \in C_j} E_f(\mathcal{L}_i). \tag{6.5}$$

The cluster C_j with the largest error will be split into two new clusters each with a different BRDF. Further materials can be extracted by further splitting the clusters.

But how do we split a cluster? The BRDF fit to a cluster represents the average material of the lumitexels in that cluster. Fitting the BRDF using the Levenberg-Marquardt algorithm (see Section 6.5) will also provide us with the co-variance matrix of the parameters. The eigenvector belonging to the largest eigenvalue of this matrix represents the direction in which the variance of the samples is highest, and is therefore a good choice for the direction in which the parameter space is to be split.

Let $\vec{\boldsymbol{\beta}}$ be the fit parameter vector of the BRDF $f(\vec{\boldsymbol{\beta}}; \hat{l}, \hat{v})$ for cluster C. Vector $\vec{e}$ denotes the eigenvector belonging to the largest eigenvalue λ of the corresponding co-variance matrix. We then construct two new BRDFs:

$$f_1(\vec{\boldsymbol{\beta}} + \tau\lambda\vec{e}; \hat{l}, \hat{v}) \quad \text{and} \quad f_2(\vec{\boldsymbol{\beta}} - \tau\lambda\vec{e}; \hat{l}, \hat{v}), \tag{6.6}$$

where τ is a scaling factor to adapt λ to a moderate value. Two new clusters C_1 and C_2 are generated by distributing every lumitexel $\mathcal{L}_i$ of cluster C either to C_1 if $E_{f_1}(\mathcal{L}_i) < E_{f_2}(\mathcal{L}_i)$, or to C_2 otherwise. In the next step, f_1 and f_2 are fit to best approximate the lumitexels in the new clusters.

6.6.3 Reclustering

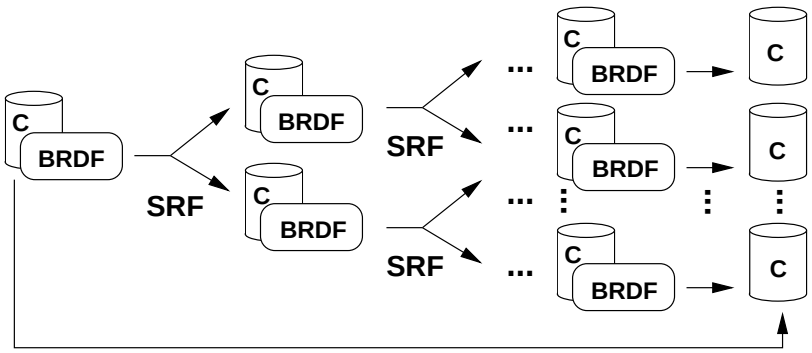


Figure 6.8: *The complete splitting and reclustering algorithm including the global reclustering, which is similar to the recluster-fit iteration, only that all lumitexels are distributed among all clusters.*

Because the parameters of a BRDF fit to a multi-material cluster are not necessarily the center of the parameters of the contained materials and due to improper scaling of λ and other reasons like noise, the performed split will not be optimal and the two new clusters may not be clearly separated, e.g. in the case of two distinct materials some lumitexels belonging to one material may still be assigned to the cluster of the other material.

A better separation can be achieved by iterating the procedure of distributing the lumitexels $\mathcal{L}_i$ based on $E_{f_1}(\mathcal{L}_i)$ and $E_{f_2}(\mathcal{L}_i)$, and then fitting the BRDFs again. The iteration stops when the number of lumitexels in the generated cluster does not change any more. In our experiments this reclustering operation leads to a clear separation of materials and is done after each split. The split-recluster-fit (SRF) process is visualized in Figure 6.7.

When more than two clusters have been generated by successive binary splits and a new material is clearly distinguished, it is helpful to clean the other clusters, which were not involved in the last split, from all lumitexels belonging to the newly discovered material. This can be done in a global reclustering step by redistributing all initial lumitexels $\mathcal{L}_i$ to the cluster C_j with

$$j = \underset{k}{\text{argmin}}\ E_{f_k}(\mathcal{L}_i). \tag{6.7}$$

And again, the BRDFs of all involved clusters have to be refit. This global reclustering is repeated several times to clearly separate the materials. We stop this iteration when the percentage of change is smaller than some ϵ, or a maximum number of iterations is reached. The complete splitting and reclustering algorithm is depicted in Figure 6.8 and the processing on a real model is shown in Figure 6.9.

6.6.4 Termination of the Splitting Process

We still have to decide when to stop the splitting process. To do this we require the user to input the estimated number of materials $|M|$. We stop the splitting and clustering process after at least $|M|$ clusters have been created. More clusters can be generated to compensate for the often noisy and not absolutely accurate reflectance samples (e.g. slightly wrong normals, noise in the images, misregistration, etc.).

This means that we do not necessarily have a one to one mapping between actual materials and clusters. This is not crucial since the projection, which we will present in the next section, uses a weighted sum of several BRDFs to accurately represent every lumitexel.

6.7 Projection

The representation of an object by a collection of only a few clusters and BRDFs makes the virtual object look artificial as can be seen in Figure 6.10. The main reason for this is that real surfaces exhibit changes in the reflective properties even within a single material. These changes cannot be represented by a single BRDF per cluster since all lumitexels within the cluster are assigned the same BRDF parameters.

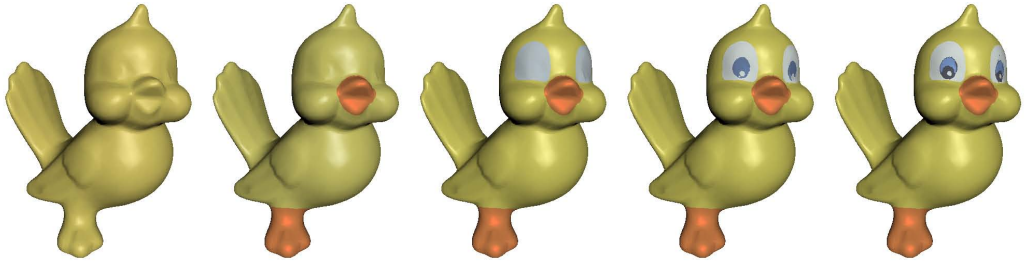


Figure 6.9: *The clustering process at work. In every image a new cluster was created. The object was reshaded using only the single BRDFs fit to each cluster before the projection into a basis of multiple BRDFs.*

Figure 6.10: *Left: The result of the clustering process still does not look realistic since there is no variation of the material within one cluster. Right: Spatial variation derived by projection the reflectance samples of each lumitexel in a basis formed by the clustered materials.*

To obtain truly spatially varying BRDFs we must find a specific BRDF for each lumitexel. But the sparse input data does not allow to fit a reliable or even meaningful BRDF to a single lumitexel because each lumitexel consists of only a few reflectance samples. In addition, we would need to acquire a highlight in every lumitexel to reliably determine the specular part, as already explained in Section 6.6.1.

The solution is to project each lumitexel into a basis of BRDFs (see Section 6.7.1). The BRDF $f_{\pi i}$ of a lumitexel $\mathcal{L}_i$ is then represented by a linear combination of m BRDFs $f_1, f_2, \ldots, f_m$:

$$f_{\pi i} = t_1 f_1 + t_2 f_2 + \ldots + t_m f_m, \tag{6.8}$$

with $t_1, t_2, \ldots, t_m$ being positive scalar weights. This forces the space of solutions (i.e., the possible BRDFs for a pixel) to be plausible since the basis BRDFs are already fit reliably to a large number of reflectance samples.

Given the BRDFs, the weights have to be determined for each lumitexel. Let $r_{j=1\ldots|\mathcal{L}_i|}$ be the reflectance values of the lumitexel $\mathcal{L}_i$. The weights are found by a least square optimization of the following system of equations using singular-value decomposition:

$$\begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_{|\mathcal{L}_i|} \end{pmatrix} = \begin{pmatrix} \tilde{f}_1(\hat{l}_1, \hat{v}_1) & \tilde{f}_2(\hat{l}_1, \hat{v}_1) & \cdots & \tilde{f}_m(\hat{l}_1, \hat{v}_1) \\ \tilde{f}_1(\hat{l}_2, \hat{v}_2) & \tilde{f}_2(\hat{l}_2, \hat{v}_2) & \cdots & \tilde{f}_m(\hat{l}_2, \hat{v}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{f}_1(\hat{l}_{|\mathcal{L}_i|}, \hat{v}_{|\mathcal{L}_i|}) & \tilde{f}_2(\hat{l}_{|\mathcal{L}_i|}, \hat{v}_{|\mathcal{L}_i|}) & \cdots & \tilde{f}_m(\hat{l}_{|\mathcal{L}_i|}, \hat{v}_{|\mathcal{L}_i|}) \end{pmatrix} \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_m \end{pmatrix}, \tag{6.9}$$

with $\tilde{f}(\hat{l}, \hat{v}) := f(\hat{l}, \hat{v}) l_z$. Compared to the non-linear fitting of BRDF model parameters (see Section 6.5.2), we now have a linear problem to solve with a smaller degree of freedom and even more constraints. Equation 6.9 shows only the system for one color channel, whereas the weights t_i have to be the same for all channels. In contrast to this, BRDF parameters would require a distinct set of parameters per channel.

We find the weights t_i as the non-negative least square solution of the system of equations. Negative values are avoided because they may result in an oscillating BRDF that represents only the given reflectance samples accurately but will produce unpredictable values for other viewing and light directions.

6.7.1 Basis BRDFs

The next question is how to determine the set of basis BRDFs. Since the changes of the surface properties within one material tend to be small, a distinct set of basis BRDFs is assigned to each cluster. It is therefore sufficient to store just the scalar weights per lumitexel instead of the full set of BRDF parameters.

Finding the optimal set of BRDFs $f_1, f_2, \ldots, f_m$, that minimizes the error

$$E_\pi(C) = \frac{1}{|C|} \sum_{\mathcal{L}_i \in C} E_{f_{\pi i}}(\mathcal{L}_i) \tag{6.10}$$

for a cluster C, where $f_{\pi i}$ denotes the least square projection of the lumitexel $\mathcal{L}_i$ as defined in Equation 6.8, is a problem of principal function analysis (PFA)

(see [Wood00]). Principal function analysis is closely related to principal component analysis (PCA) with the important difference that functions f_m are optimized instead of vectors. Unfortunately, the PFA does not reduce to a simple eigenvalue problem as PCA does. To minimize $E_\pi(C)$, we again perform a least square optimization using the Levenberg-Marquardt method, this time fitting m BRDFs simultaneously. Within each iteration we recompute the projection $f_{\pi i}$ of lumitexel $\mathcal{L}_i$ into the currently estimated basis.

As for every optimization problem the initial parameters (BRDFs) are quite important. For a given cluster C, we use the following BRDFs as a basis:

- f_C, the BRDF fit to the cluster C,
- the BRDFs of spatially neighboring clusters to match lumitexels at cluster boundaries,
- the BRDFs of similar clusters with respect to the material, and
- two BRDFs based on f_C, one with slightly increased and one with decreased diffuse component ρ_d and exponent N.

In our experiments it turned out that this initial basis together with the projection already produces very good results with small errors. In most cases the PFA computed almost negligible changes to the initial BRDFs. This is to be expected because the initially chosen basis constructed through splitting and clustering already approximates the material properties quite well.

6.8 Acquiring Normal Maps

The reconstructed spatially varying BRDFs can further be used to add geometric detail in the form of normal maps. The resolution of the acquired geometry of an object is typically limited by the applied 3D scanning device. Additional processing of the 3D data like combining multiple scans, smoothing the surface to remove noise and mesh simplification to reduce the complexity of the model further erases fine scale geometric detail.

Although our projection method compensates somewhat for imprecise normals, one can observe in Figures 6.11 and 6.14 that much geometric detail was lost during the geometry acquisition, especially in the hair region.

Our method of measuring reflection properties as described in Section 6.7 can easily be extended to measure normal maps even for surfaces that are not perfectly diffuse.

In Section 6.4 the normal $\hat{n}$ and a deterministically constructed tangent and binormal are used to transform the lighting and viewing direction, $\hat{l}^{(g)}$ and $\hat{v}^{(g)}$ from

world coordinates into the local coordinate frame, yielding $\hat{l}$ and $\hat{v}$ at each surface point. In the case of isotropic materials the direction of the normal is sufficient to define this transformation since the BRDF is independent of the orientation of the tangent within the tangential plane by definition. Thus, the transformation into the local coordinate frame can be carried out by just two rotations about the y and z axis:

$$\hat{l} = R_y(-\alpha)R_z(-\beta)\hat{l}^{(g)}, \quad \text{and} \tag{6.11}$$

$$\hat{v} = R_y(-\alpha)R_z(-\beta)\hat{v}^{(g)}, \tag{6.12}$$

where α and β are azimuth and zenith of the normal.

As already mentioned in Section 6.4 an initial estimate of the normal at every surface point/lumitexel is provided by the triangle mesh. Based on these inexact normals the basic materials of the object are separated by the clustering process (Section 6.6). Subsequently, for each lumitexel the weighting coefficients for the basis materials are determined by projection to obtain starting values for these coefficients.

Given that enough reflectance samples are provided at every point (more than two) it is possible to extract the direction of the normal (α, β) for every lumitexel. This is done by minimizing the error between the measured reflectance samples r_i and the evaluated reflectance values

$$f_\pi(R_y(-\alpha)R_z(-\beta)\hat{l}_i^{(g)}, R_y(-\alpha)R_z(-\beta)\hat{v}_i^{(g)}), \tag{6.13}$$

for which once again the Levenberg-Marquardt algorithm is applied. Both the direction of the normal and the optimal weights for the basis BRDFs can be found in the same step. However, since the overall distribution of the normals is altered by computing normal maps one has to recompute the BRDF for the single clusters afterwards and has to perform the projection once again. If desired, the process of normal fitting, recomputation of the BRDF, and projection can be iterated. In our experience this is normally not necessary.

The quality of the results is presented in Figure 6.11 where normal maps have been recovered for all materials. Unfortunately, the method produces some artifacts in concave regions where interreflections become important (lower right of Figure 6.11). Since interreflections are not yet considered in our algorithm the normal directions are noisy and the BRDFs are not very accurate. A feature sensitive smoothing of the normals followed by a BRDF projection may slightly improve the results.

Since a non-linear optimization is performed for every lumitexel the recovery of the normal map is a time-consuming step. For the bird model it took around three hours but was completely automatic.

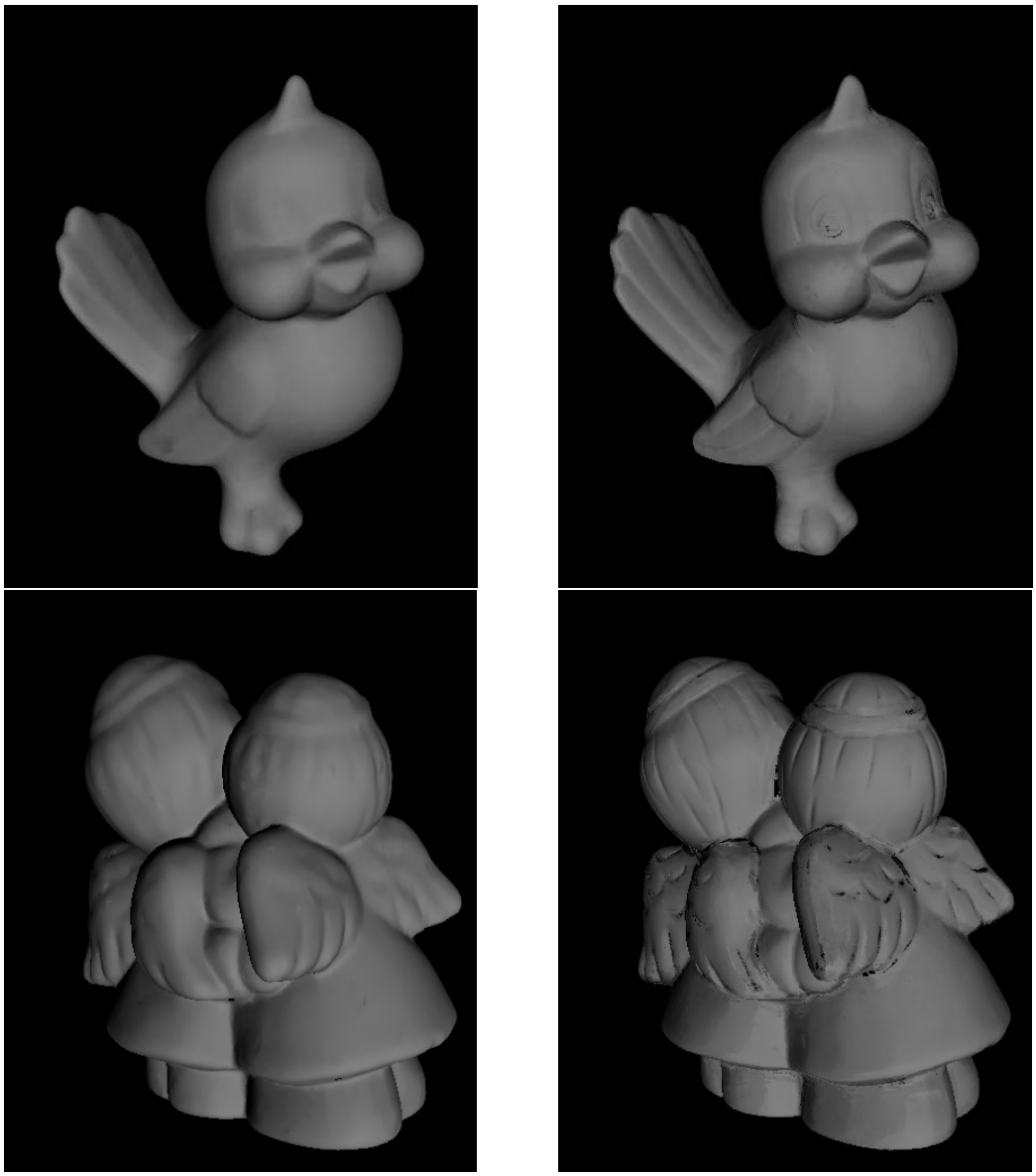


Figure 6.11: *Left: Normals of the original mesh. Right: Normals optimized using spatially varying BRDFs*

6.9 Rendering

As explained in Section 6.4.1, we know the position of every lumitexel, as well as the triangle it belongs to and the 2D coordinates within that triangle.

This information can then be used to generate an index texture for the full object. For every texel, that texture contains an index to the cluster it belongs to.

Then we generate a weight texture map for every cluster that stores the weights resulting from the projection into the basis BRDFs. The parameters for the basis BRDFs of every cluster are stored in a small table. Additionally, we may have a normal map that was reconstructed using the technique from Section 6.8.

Raytracing such an object is very simple, since for every point on the object that is raytraced we can simply look up the cluster the texel belongs to. Then we evaluate the basis BRDFs for the local light and viewing direction and compute the weighted sum using the weight texture map. So rendering simply is a matter of evaluating a few Lafortune BRDFs per pixel, and weighting the results.

If no normal map was reconstructed, mip-mapping can easily be used. Since the weighted sum is just a linear operation, the weights of neighboring texels can simply be averaged to generate the next coarser mip-map level.

Another, more accurate method is to explicitly construct lumitexels at coarser mip-map levels that combine the reflectance samples of the finer ones. The linear weights are computed to best fit these larger collections of radiance samples. This also allows to reconstruct normal maps for every mip-map level.

If the original images are of high resolution and hence the object is sampled very densely, point-based rendering using forward projection is a viable alternative. It completely avoids the generation of texture maps and the resulting data can be used with almost no further processing.

Hardware-Accelerated Rendering

In order to speed up the rendering of objects with spatially varying BRDFs illuminated by a point light source we have also implemented hardware-accelerated algorithms based on register-combiner [NVI02a] or fragment-programs [NVI02b]. We basically render the entire object for each cluster BRDF separately and modulate the result with the per-pixel weights represented as multiple texture maps.

The reduced instruction set of register-combiner prevents us from directly implementing the Lafortune model. Calculating the diffuse part is trivial. For the specular part, we can use the register-combiner to evaluate the weighted scalar product, but we cannot carry out the exponentiation. Following the technique proposed by Kautz and Seidel [Kautz00b], we can nonetheless precompute the exponentiation into a texture map and perform a dependent texture lookup to carry it out. The implementation was done on a *n*VIDIA NV30 where four texture lookups can be performed within the register-combiner. We can blend up to two cluster BRDFs wihtin one rendering pass. Depending on the number of cluster we obtain framerates of approximately 100 fps for the bust and around 20 fps for the angel model since it contains more clusters.

The implementation using fragment-programs is much easier since the Lafortune model can be coded directly. On an ATI Radeon 9700 graphics card we

managed to evaluate and blend up to six BRDFs within one fragment program yielding superior performance.

In the future we would like to investigate whether more complex illumination represented by environment maps can be considered using our representation, similar to the work presented by McAllister et al. [McAllister02b] who prefiltered environment maps for the Lafortune model.

model	T	V	L	R	C	B
angels	47000	27	1606223	7.6	9	6
bird	14000	25	1917043	6.3	5	4
bust	50000	16	3627404	4.2	3	4
elk	50000	25	1659945	5.4	4	4

Table 6.1: *This table lists the number of triangles (T) of each model, the number of views (V) we used to reconstruct the spatially varying BRDFs, the number of acquired lumitexels (L) and the average number of reflectance samples (R) per lumitexel, the number of partitioned material clusters (C), and the number of basis BRDFs (B) per cluster.*

6.10 Results

We applied our algorithm to four different objects consisting of different materials with varying reflection properties in both the diffuse and the specular part. The model of the angels was generated by extracting an isosurface of a computer tomography scan. The geometry of all other models was captured using a structured light 3D scanner. Some statistics about the meshes and the number of acquired views are listed in Table 6.1. Acquisition of 20 views (each needing about 15 photographs) takes approx. 2.5h. This is the only step that requires user input. The high dynamic range conversion and the registration with the 3D model takes about 5h but is a completely automated task. The clustering and the final projection takes about 1.5h for all models, and is again automatic. An additional 3h (angels), 3h (bird), 4h (bust), 2.5h (elk) were needed for the normal fitting. While timings for the conversion is dependent on the number of input images, the clustering time mainly depends on the number of texels in the selected subset. The remaining steps of projection and normal fitting depend on the number of overall reflectance samples.

In Figure 6.9 one can see how five successive split operations partition the lumitexels of the bird into its five materials. The splits were performed as described in Section 6.6. Only the per-cluster BRDFs determined by the clustering process

Figure 6.12: *A bronze bust rendered with a spatially varying BRDF acquired with our reconstruction method (without normal fitting). The reconstructed BRDFs allow for rendering the model with arbitrary lighting.*

Figure 6.13: *A comparison of a photograph (left) of a wooden elk with the reconstructed model (right). Four clusters have been generated which faithfully reproduce the appearance. Note how the wooden structure is preserved.*

model	1-RMS	C-RMS	P-RMS	F-RMS	NF-RMS
angels	.2953	.1163	.1113	.1111	0.0703
bird	.1513	.0627	.0387	.0387	0.0269
bust	.1025	.0839	.0583	.0581	0.0113
elk	.0768	.0556	.0275	.0274	0.0202

Table 6.2: *This table lists for each model the RMS error for a single average BRDF (1-RMS), the RMS error when using per-cluster BRDFs, the RMS error after projecting every lumitexel into the basis of BRDFs, the RMS error after doing a PFA on the basis BRDFs and projecting every lumitexel into the new basis, and finally the RMS error after fitting the normals.*

are used for shading, making the object look rather synthetic. After performing the projection step every lumitexel is represented in a basis of four BRDFs, now resulting in a much more detailed and realistic appearance, see Figure 6.10.

The bust in Figure 6.12 shows another reconstructed object with very different reflection properties. The bronze look is very well captured.

Another model is shown in Figure 6.13 where the reconstruction of a wooden elk using 25 views is compared to an actual photograph. The image-to-geometry registration explained in Chapter 5 worked well enough to reproduce the fine detail wooden structure. Unfortunately, the bad quality of the geometric model caused some noticeable differences at the rim of the antler. It also causes the darker stripe right behind the eye, where applying a 10x10 filter to the shadow map could not remove enough points around the shadow edge. These artifacts will disappear if a better quality geometry model is available. Due to the slightly more complex geometry 25 different views are actually not enough to cover the entire surface. This is why some holes appear close to the wheels. Clearly, more images would remove these problems.

In Figure 6.14 you can see another comparison between an object rendered with an acquired BRDF (using the projection method) and a photograph of the object. They are very similar, but differences can be seen in highlights and in places where not enough reflectance samples were captured. Capturing more samples will increase the quality. The difference in the hair region in the right picture is due to missing detail in the triangle mesh. This detail can be recovered by computing normal maps as explained in Section 6.8, the result is shown in Figure 6.14 bottom.

Another difference is due to the fact that the diffuse color of one lumitexel may not be represented in any of the constructed clusters because the number of lumitexels belonging to the same material can be so small that they nearly vanish in the mass of lumitexels of the cluster they are currently assigned to. This effect

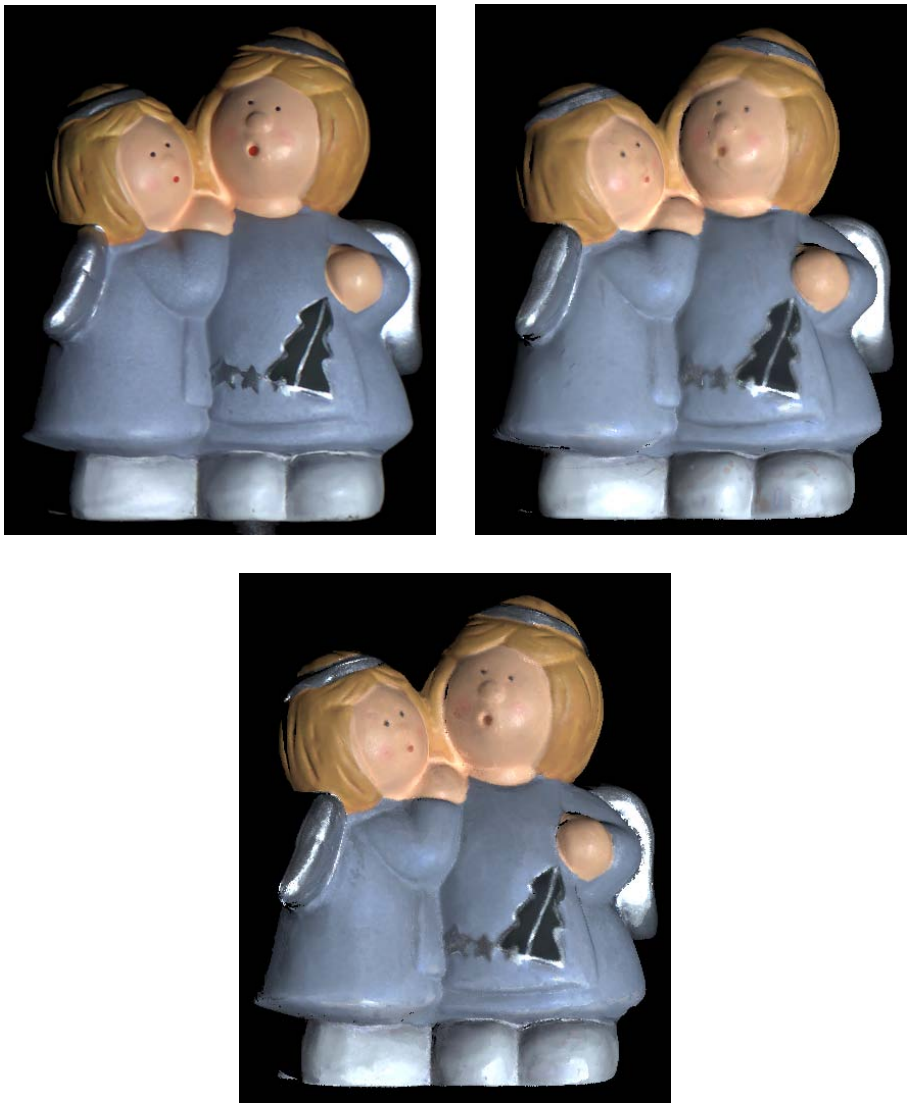


Figure 6.14: *Left side: Photograph of model. Right side: Model with acquired BRDF rendered from the same view with similar lighting direction. The difference in the hair region is due to missing detail in the triangle mesh. Bottom: After computing a normal map the missing detail has been recovered. Note how the highlights around the Christmas tree and on the left wing matches the original.*

can for example be observed at the mouth of the larger angel which in reality exhibits a much more saturated red, see Figure 6.14.

In Table 6.2 we list RMS errors computed between all the reflectance samples of a model and the reconstructed BRDFs. You can see that the error considerably decreases when going from one average BRDF to per-cluster BRDFs and then to per-pixel BRDFs (using projection). As already mentioned the PFA only slightly changes the RMS error.

Generally, it can be said that for all the models only a few clusters were needed to accurately represent all the materials since the projection takes care of material changes. In our experiments even Lafortune BRDFs consisting of a single lobe were sufficient to form good bases for the clustering and projection.

The fitting of normal maps works very well using the reconstructed BRDFs. Fine geometric detail could be recovered and the quality of the overall model was increased even for non-Lambertian surfaces. This is also indicated by the RMS error in Table 6.2 which has been noticeably reduced by the normal fitting step.

One can however observe noise in regions where too few reflectance samples or views have been acquired. Although three reflectance samples should be enough to determine the normal direction in the ideal case, more samples have to be provided to get reliable results.

Furthermore, in concave regions interreflections become very important. Since interreflections are not yet considered in our algorithm the presented method unfortunately produces noisy normals and inaccurate BRDFs in those regions. This problem may be solved by applying techniques capable of dealing with interreflections such as [Nayar90b, Nayar91b] or [Forsyth89].

To show that the presented method can distinguish materials with the same diffuse color but varying specularity, we initially experimented with artificially generated data. We generated samples for five different Lafortune BRDFs with the same

$$\begin{aligned} k_d &= (0.15, 0.3, 0.1), \quad \text{and different exponents} \\ N &= 36.2, 42.2, 48.2, 54.2, 60.2, \end{aligned}$$

where $-C_{xy} = C_z = \sqrt[N]{(N+2)/2\pi}$ corresponding to the modified Phong model [Lewis93]. To each reflectance sample we added up to 5% noise. Overall there were about 46000 reflectance samples (on average 11 per texel) with random lighting directions and five different viewing directions. Our clustering algorithm was able to clearly distinguish these five materials although they had the same color but different specular lobes. The resulting k_d's and exponents had less than 0.2% error.

In Figure 6.15 we also show a real example of a measured bathroom tile where the design gets visible only for special viewing and lighting directions. Since the

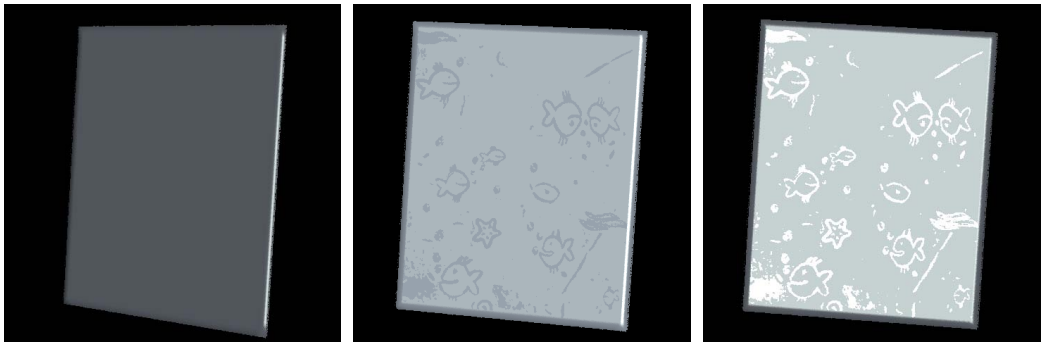


Figure 6.15: *Measured bathroom tile. The entire tile has the same diffuse color but distinct specular components for the background and the design. Left: No pattern is visible if the diffuse reflection is dominant. Middle: The differences come out only in highlight regions. Right: Close to the mirror direction the narrow specular lobe of the pattern is brighter than the background. The patterns is reverted.*

tile is made of the same material through and through, the pattern has the same diffuse color but a different specular component. The design comes out only in highlight regions.

Reliable classification of these materials requires however that part of a highlight is visible in at least one of the reflectance sample at every surface point. Depending on the narrowness of the specular lobe and the geometry of the object, a lot of images may be required to achieve this for real objects. For example, on the angels model the specular part of the blue skirt was overestimated at some texels due to undersampled highlights. The problem of appropriate sampling will be addressed in the next chapter.

6.11 Conclusions and Future Work

We have presented an algorithm and demonstrated a system for reconstructing a high-quality spatially varying BRDF from complex solid objects. The BRDFs of the basic materials is measured precisely by collecting and clustering the samples of larger surface regions with similar reflection properties. Spatial variation in the diffuse and the specular part is added by calculating the best linear combination of the basis materials that matches the measured reflectance samples of each texel.

For the measurements only a small number of images is required showing the object illuminated by a point light. The same input images are also used to improve geometric detail. Combining the resulting optical and geometric data allows for accurately shaded, photorealistic rendering of these objects from new viewpoints and under arbitrary lighting conditions.

Furthermore, representing the spatially varying BRDFs as texture maps allows us to modify the object's geometry after the acquisition. Since the BRDF is not changed with the geometry the object can be altered or animated while preserving the material properties and thus the realistic appearance.

Several objects consisting of different materials have been acquired to demonstrate the quality and accuracy of our approach. The resulting spatially varying BRDFs accurately represent the original materials. The normal fitting algorithm recovered geometric detail that was not represented by the original mesh obtained from the 3D scanner.

The reconstruction of a new model is a relatively simple task and requires only a moderate amount of human work, mainly during the acquisition process. Except for this, all the data processing and fitting algorithms are automatic. Fortunately, the number of input views required by our algorithm is rather small.

Compared to previous approaches for representing real-world objects, such as surface light fields or reflectance fields which need several hundred or thousand images [Wood00] our method requires less input data and even the size of the output data ($\sim$25MB) is considerably smaller. All we have to store per texel are the linear blending weights for the basic materials.

As our acuqisition setup is quite similar to previous approaches for measuring homogeneous [Marschner98, Lu98] or diffusely varying BRDFs [Sato97, Yu98] on 3D geometric objects we can achieve the same quality for those materials. But in contrast to previous methods, our method can reproduces spatial variation considering the diffuse *and* the specular part of the reflection properties. To represent this variation is very important since it can be observed for many real world objects, i.e., if an object is composed by multiple materials each material typically has a different diffuse and specular component.

In the next chapter we analyze the quality of the obtained results with regards to the viewing and lighting direction selected for capturing the input images.

Chapter 7
Acquisition Planning

As explained in the previous chapter, measuring reflection properties of a 3D object involves capturing images for numerous viewing and lighting directions. We now present a method to select advantageous measurement directions based on analyzing the estimation of the bi-directional reflectance distribution function (BRDF). The selected directions minimize the uncertainty in the estimated parameters of the BRDF. As a result, few measurements suffice to produce models that describe the reflectance behavior well. Moreover, the uncertainty measure can be computed fast on modern graphics cards by exploiting their capability to render into a floating-point frame buffer. This forms the basis of an acquisition planner capable of guiding experts and non-experts alike through the BRDF acquisition process. We demonstrate that spatially varying reflection properties can be captured more efficiently for real-world applications using the acquisition planner.

7.1 Introduction

In the field of 3D object acquisition progress has been made both in the area of geometry and appearance acquisition. Appearance or reflection properties are in most approaches measured by capturing a number of samples of the BRDF of the object. The samples are commonly acquired by a sensor (a digital camera in our set-up) and a point-light source. One pair of light source and camera position (called a view collectively in the remainder of this thesis) captures a single reflectance sample for each point that is visible and lit.

A number of researchers have built special gantries to perform a robot controlled dense sampling of the reflection properties [Debevec00, Matusik02a, Furukawa02, McAllister02a]. Others position the camera and the light source manually [Marschner98, Marschner99, Lensch01c, Lensch03a].

Figure 7.1: *Comparison of Spatially Varying BRDF Models from Unplanned / Planned Series. The model on the left contains holes in the BRDF due to undersampling. The undersampling results in holes (black) or wrongly estimated BRDFs (bright highlight at the bottom of the skirt). The model on the right obtained from the same number of views suggested by our planner samples the surface evenly.*

The basic question for both, the automatic and the manual approach is: How to sample the reflection properties in an efficient way? The acquisition of reflection properties needs to be planned in order to measure efficiently, failing to plan may result in insufficient data for the modeling task or lead to highly redundant oversampling. Measurement is typically an involving task and efficiency in the process is of paramount importance.

In this chapter we present a method that assesses the uncertainty in the parameters of the Lafortune model [Lafortune97] (The underlying method is however also applicable to other parametric models and may be adapted to non-parametric models as well.) Based on this uncertainty measure we develop an acquisition planning algorithm that computes from where to sample next in order to minimize the uncertainty in the parameters, i.e., where to place the camera *and* the light source with respect to a set of previously acquired views. For 3D objects, we have to evaluate and combine the predicted uncertainty of each single surface point. A good set of views will measure each point on the surface several times with varied viewing and lighting directions sampling a highlight at each point. The view planning is influenced by a number of further constraints including the 3D shape. The shape limits the number of visible and lit surface points in a view.

One of the goals of acquisition planning is to perform measurements efficiently. Time spent on the planning itself therefore has to be reasonable. We

compute the uncertainty measure in modern graphics hardware with floating-point frame buffers. The evaluation is performed directly on the texture atlas of the object.

The measurement theory behind our approach is well established in other fields; physicists and other natural scientists apply it quite routinely to their measurement tasks. Our contribution in this area is to adapt some of the natural sciences' measurement theory to the task of measuring the BRDF for computer graphics. Our chapter makes three main contributions:

- the definition of a function to measure the reduction in uncertainty added by one view (camera and light source position),
- a view planning algorithm that combines this function with geometric constraints imposed by a 3D object to predict the next best view for efficient measurement, and
- a hardware-accelerated implementation for evaluation of the objective function directly on the texture atlas.

In the next section we discuss related work before we present an overview over the acquisition planning in the measurement process in Section 7.3.

7.2 Related Work in Acquisition Planning

Work related to the automated acquisition of reflection properties of complete objects can be found in different fields including computer graphics, computer vision, robotics and visual metrology. We start our review with a brief summary of work in computer vision, followed by a discussion of automatic scanning of 3D models including some theoretical issues, and we conclude our review with work on BRDF acquisition and representation in computer graphics.

In computer vision, the active vision paradigm [Aloimonos87, Bajcsy88] motivates purposive viewpoint control. Viewpoint control in minimizing uncertainty of 3D object representations [Whaite97] and in scene exploration [Kutulakos94, Marchand99] relates directly to the task of exploring unknown spatially-varying BRDF of an object. Related to BRDF acquisition are also visual metrology tasks which are reviewed by Tarabanis and Tsai [Tarabanis95]. In metrology the planning task is to position a sensor to satisfy some sensing quality criterion, e.g., work by Cowan and Kovesi [Cowan88] and Mason and Grün [Mason95]. The quality criterion of interest in our work is the certainty in the acquired BRDF. We aim at achieving high quality by choosing advantageous viewpoints of camera and light source positions.

Planning the placement of cameras [Triggs95, Gonzalez-Banos97] with actuated devices is part of motion planning in robotics. Chaumette et al. [Chaumette96] and Marchand and Chaumette [Marchand99] plan camera motions to recover geometric primitives using a structure-from-controlled-motion technique. Geometric models of an object have also been acquired with a robotic facility [Reed99, Lang00] and with an automated commercial scanner [Pito99]. In practice, view planning is often started after an initial rough acquisition of the object's geometry from pre-set viewpoints, as in the geometry model acquisition system by Reed and Allen [Reed99, Reed00]. The task in this situation is to fill holes in the existing model stemming from tight visibility restrictions. Filling holes is an example where an automated planner can be very beneficial.

The incremental *Next Best View* planning strategy of Whaite and Ferrie [Whaite97] is closely related to ours. Their strategy explores the *geometry* of a 3-D model with *a priori* unknown shape. They apply a synthesis approach which is based on a probabilistic model of an object's geometry to be explored. The approach minimizes uncertainty of a parametric object model. Objects are represented by sets of superquadrics. Using an active sensing strategy a next view is selected. The selected view minimizes the current uncertainty of the object model. For an object consisting of multiple superquadrics, an uncertain view for each superquadric is found. Next, a strategy selects the most uncertain view amongst the views for the individual superquadric models. The algorithm is applied to explore the environment of a robotic agent enabling object recognition and manipulation.

An alternative approach to view planning is to delegate the (computationally complex) task to a human but to provide real-time feedback. This has been demonstrated successfully by Rusinkiewicz et al. [Rusinkiewicz02] for geometry scanning of objects. Planning the acquisition of reflectance properties in this fashion is however impossible since humans can generally not reason about the four-dimensional BRDF on the surface of complex objects in real-time.

In computational geometry, visibility in polygonal environment is considered [Chvátal75] with applications in geographical data processing, security and military [Marengoni00]. Acquiring realistic reflectance properties of a 3D object requires imaging its complete surface. This completeness constraint restricts the set of possible solutions in acquisition planning. Finding the minimal set of views which cover the complete surface of an object is a NP-hard problem [Tarbox95]. A planned set of views may still fail in practice due to positioning or modeling errors and off-line plans need to consider uncertainty to ensure coverage of the complete surface when actually executed [Tarbox95, Scott01].

In computer graphics, the imaging of visible surfaces is relevant in radiosity, ray-tracing, scene walk-throughs and texturing of surfaces [Matsushita99, Stürzlinger99] as well as for image-based rendering [Hlavac96, Chai00, Vázquez02]. In the BRDF measurement and representation field, quite a number

of articles have been published but we are not aware of any systematic investigation of the estimation quality of the BRDF. McAllister computes the number of samples required to densely cover the hemisphere above a point with a light source [McAllister02a]. Ramamoorthi and Hanrahan [Ramamoorthi01b] examine the problem of global inverse illumination within a signal processing framework. Existing BRDF measurement approaches can be classified by the number of images necessary and the generality of the estimated BRDF model.

A family of methods measure reflectance properties by performing a very dense sampling. Some of them are designed for flat surface samples only [Dana99, McAllister02a] while others can deal with 3D objects [Debevec00, Masselus02, Matusik02a, Matusik02b, Furukawa02]. Since a dense sampling allows to render directly from the measured data, the reflection properties are unrestricted in those approaches. The quality of the outcome when no BRDF model is fit mainly depends on the sampling density. Another set of techniques estimate the appearance of an object from a sparse set of images. Measurements in these approaches are typically performed either by using a point light source [Kay94, Sato97, Marschner99, Marschner00, Nishino01a, Lensch01c, Lensch03a, Li02] or by performing inverse global illumination [Yu98, Gibson01, Boivin01, Ramamoorthi01b, Nishino01b]. Using just a sparse set of input images, the quality of the outcome is very much dependent on the selected views and illumination.

In this chapter, we again follow the sparse sampling approach using a point light source. We introduce our method which is able to select a sparse set of measurements and ensures the quality of the BRDF model at the same time. The method may also be applied to dense sampling approaches in order to perform an adaptive sampling, reducing the redundance in the acquired data.

7.3 Acquisition Loop

The measurement of reflection properties is executed as a number of successive steps which are shown in Figure 7.2. Prior to the acquisition of reflectance properties the 3D geometry of the object has to be acquired. The acquisition starts with the planning of the first view. Each new view is planned based on the current estimate of the BRDF parameters and the visibility and shadowing constraints imposed by the 3D geometry of the object.

In the second step the planned view is acquired. Next, the view is registered with the 3D object (see Chapter 5 and Section 6.3), since the real camera and light source position may deviate from the proposed view. The recovered positions are used to determine the regions of the object's surface which are visible and lit. The valid pixels are resampled into a texture atlas.

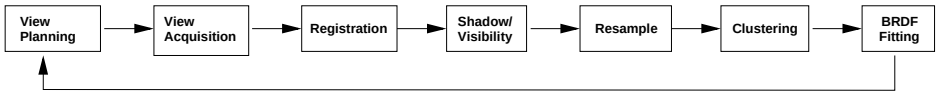


Figure 7.2: *View planning interacts with the entire pipeline of appearance measurements: An optimal view is proposed and captured. Manual placement of camera and light source requires registration with the 3D object. Visibility and shadows are computed and the data is resampled. From the resampled data of all views the BRDF parameters per cluster are updated which again influence the planning of the next view.*

For the planning a coarse texture atlas is used to speed up the process. Local viewing and lighting direction are computed per pixel. From the resampled data a new set of BRDF parameters is estimated. As the number of measurements for a single point are too few to obtain credible BRDF parameters, clusters of points are used to fit the parameters. Points can be clustered either based on their diffuse color or using the method presented in the previous chapter. The estimated spatially varying BRDF parameters are then used in the next execution of the planning step.

After the capturing process is complete, all measurement data is resampled again using a high resolution texture atlas and a final spatially varying reflection model is estimated.

Planning is performed by minimization of an objective function that takes into account the previously acquired data, the geometry of the object and the currently estimated BRDF parameters. The objective function is based on co-variance matrices as an uncertainty measure. The co-variance matrices are compact and summarize all necessary information about the views acquired so far. Their storage cost and the computation time of the planning algorithm is constant and independent of the number of acquired views. In particular, we do not have to store each local viewing and lighting direction per pixel for each measurement.

In the next section we detail the relationship between the co-variance matrix and measurement uncertainty with an object consisting of a single pixel as a tutorial example.

7.4 One-Pixel Objects

Measuring the BRDF of a single point on an object is already a task which involves some effort. It is necessary to understand how measurements of the reflectance of a single pixel influence the reliability of parameters of the BRDF model we are going to fit. We briefly summarize background material on parameter estimation

and measurement theory related to our fitting approach. We conclude that not all possible measurements contribute in the same way to the quality of the fitted model parameters.

For the purpose of the discussion in this and the next section, the prior information I for our measurement task is that we would like to estimate a Lafortune model f_r with one isotropic specular lobe (see Equation 7.1) for an object consisting of one-pixel.

$$f_r(\boldsymbol{\beta};\boldsymbol{\omega}) = \rho_d + (C_x\omega_{i_x}\omega_{o_x} + C_y\omega_{i_y}\omega_{o_y} + C_z\omega_{i_z}\omega_{o_z})^N \tag{7.1}$$

where ρ_d denotes the diffuse reflectance, N the exponent of the specular lobe, C_x, C_y and C_z the weighting coefficients of the dot product between ω_i and ω_o; ω_i is the incident light direction and ω_o the exitant lighting direction. In the following, we denote these parameters collectively as $\boldsymbol{\beta}$. The model $M(\boldsymbol{\beta},\boldsymbol{\omega})$ calculates the reflectance for a given ω_i and ω_o (collectively $\boldsymbol{\omega}$). Equation 7.2 shows the standard regression problem for m measurements with an additive error term ε resulting in m reflectance samples R_m. For the purpose of this discussion, we assume that the error terms are independent and identically distributed (iid) samples from a distribution centered at 0.

$$R_m = M(\boldsymbol{\beta},\boldsymbol{\omega}_m) + \varepsilon_m \tag{7.2}$$

A principled approach to solve this model fitting task is by employing Bayes theorem

$$P(M|DI) = \frac{P(M|I)P(D|M)}{P(D|I)}. \tag{7.3}$$

Bayes theorem describes how to obtain the posterior probability distribution of the reflectance model $P(M|DI)$. It depends on our prior belief of possible model parameters $P(M|I)$, the measurement or predictive probability of a measurement $P(D|I)$ with acquired data D and our model of the measurement process $P(D|M)$ which here is Equation 7.2. Measurements are a principled way to change one's prior beliefs. If the observations provide strong evidence, the data term dominates while with a lack of evidence the prior remains unchanged. The certainty in the model is described by the full distribution $P(M|I)$. The distribution is in the dimensions of the model and requires a summary for interpretation. The most probable parameter value and confidence intervals are common summaries. (See, e.g., Bretthorst [Bretthorst88] or Hastie et al. [Hastie02] for a more complete introduction to Bayesian model estimation).

An approach to find the most probable model is minimization. The sum of squared errors

$$Q = \sum_m \left(R_m - M(\boldsymbol{\beta},\boldsymbol{\omega_m})\right)^2 \tag{7.4}$$

is the most common error measure to minimize. This error measure coincides with the most probable model in the Bayesian approach given the model is linear in the parameters β, the noise ε in the measurements is Gaussian and our prior beliefs are uninformative (flat priors) [Hastie02]. Under these circumstances, the uncertainty in the most probable model parameters depend linearly on the co-variance matrix $\boldsymbol{CoV}$. The co-variance matrix $\boldsymbol{CoV}$ is the inverse of the Hessian matrix $\boldsymbol{H}$ with entries $\boldsymbol{H}_{i,j} = \frac{\partial^2 Q}{\partial \boldsymbol{\beta}_i \partial \boldsymbol{\beta}_j}$ (see Appendix A for the derivatives). The singular values σ of the co-variance matrix define the length of the major axes of the hyperellipsoid of a given Q. In linear models this hypervolume bounded by the hyperellipsoid for a given quadratic error measure Q is directly related to the posterior probability.

In the non-linear Lafortune reflectance model the simple relationship between the least-squares residual Q and the uncertainty in the parameters does not hold. The most popular way to proceed is to employ a non-linear least-square solver to minimize Q but analyze the fit with the co-variance matrix $\boldsymbol{CoV}$. The co-variance matrix is only strictly valid for linear models, however, employing $\boldsymbol{CoV}$ is justifiable if Q is well approximated by a quadratic near the minimum [Press94]. We performed some Monte-Carlo bootstrap analysis [Efron86, Carpenter00] in order to confirm the validity of the linear approximation when fitting the non-linear Lafortune model. In summary, our conclusion is that the non-linearity of the Lafortune reflectance model prevents us from stating confidence intervals based on the linear approximation. However, the co-variance matrix is a good indicator of parameter uncertainty and if we would like to obtain measurements in a way such that we are most confident in the estimated parameters, minimizing the co-variance matrix is a sensible strategy. This conclusion is also consistent with the reasoning of Whaite and Ferrie [Whaite97]. The co-variance matrix of the Lafortune model depends only on the chosen incident and exitant light direction given a fixed estimate $\bar{\boldsymbol{\beta}}$. A recipe of how to choose the light and viewing directions for the one-pixel object is described in the next section.

7.5 Uncertainty Minimization

The uncertainty in the estimated parameters for a set of views is minimum if the co-variance matrix is minimal. If we are exploring unknown reflectance properties, we only learn our model parameters as we acquire new views, gaining information incrementally. The certainty gain of a view is therefore the reduction in the co-variance matrix from the previous view to the current one and hence, the objective function

$$F = \|\boldsymbol{CoV}(\bar{\boldsymbol{\beta}}_v, \boldsymbol{\omega}_{1,\ldots,v})\| - \|\boldsymbol{CoV}(\bar{\boldsymbol{\beta}}_v, \boldsymbol{\omega}_{1,\ldots,v+1})\|. \tag{7.5}$$

We would like to maximize the certainty gain, i.e., we have to maximize F. A greedy strategy selects after each view v the one which maximizes the expected gain of the next view $v+1$. This greedy strategy is optimal in the linear case with fixed estimates $\bar{\boldsymbol{\beta}}$ because then the co-variance is independent of the order of views. In our scenario this approximation becomes more appropriate as the certainty in the estimates increases.

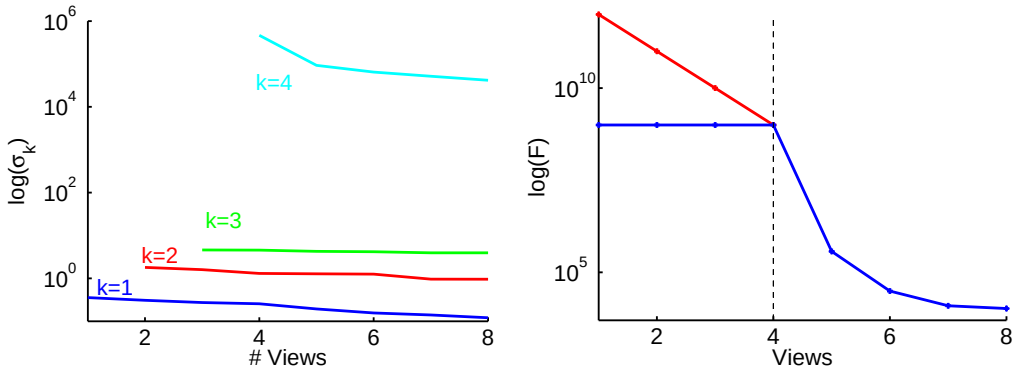
The selection strategy must also deal with a rank-degenerate Hessian matrix $\boldsymbol{H}$ during the views $v = 1 \dots L$, where L is the number of parameters in the Lafortune model f_r, in our case $L = 4$. These initial views have to be chosen in order to arrive at a small $||\boldsymbol{CoV}||$ after view $v = L$. While $||\boldsymbol{CoV}|| = \infty$ for $v < L$, we can calculate the Pseudo-Inverse of the Hessian matrix instead. Figure 7.3(a) shows the singular values σ_k of the pseudo-inverse which approximately increase exponentially with k. We maximize the information gain based on this pseudo co-variance until the Hessian matrix reaches full rank. The gain can be considered infinite with each rank gained, i.e., a choice of $F(k \leq L) = Inf - \sigma_k$ suggests itself. We illustrate this strategy in Figure 7.3 for the one-pixel object. (We pick $Inf = 10^9$ and the parameters $\boldsymbol{\beta} = (\rho_d = 0.3, N_s = 10, C_x = C_y = -0.8, C_z = 0.8)$ for the Lafortune model (Equation 7.1)). A better continuation of the objective function towards the first view can be obtained with $F(k \leq L) = 10^{L-k} * (Inf - \sigma_k)$ which is shown in Figure 7.3(b). The difference in the two considered choices is only of importance in multi-pixel objects when the information gain at one pixel has to be compared to the information gain at another pixel. Under these circumstances, the exponentially increasing function will strongly favour low rank updates over high rank updates.

7.5.1 Maximization

The task of view planning is to maximize the objective function in Equation 7.5. Our observation is that the objective function is partially smooth depending on the visibility of a given view. Globally, it can have many discontinuities due to shadows and visibility. In order to maximize the objective function we apply a two-step procedure with randomization:

- Search for the best pair of initial positions of camera and light source on a discretized sphere. Randomize the orientation of the discretization before each new view.
- Use a local continuous optimizer to improve the discrete solution found.

The random rotation of the sphere discretization improves the discrete search, eventually evaluating all views. This procedure allows us to use a coarse dis-



(a) σ_k of CoV (b) Obj. Function F

Figure 7.3: *Objective Function F for Infinite Co-Variance. The singular values σ_k of the $\boldsymbol{CoV}$ show an exponential increase with k. Figure 7.3(b) shows two choices for F when $k \leq L$: $F(k \leq L) = 10^{L-k} * (Inf - \sigma_k)$ in red and $F(k \leq L) = Inf - \sigma_k$ ($Inf = 10^9$) in blue.*

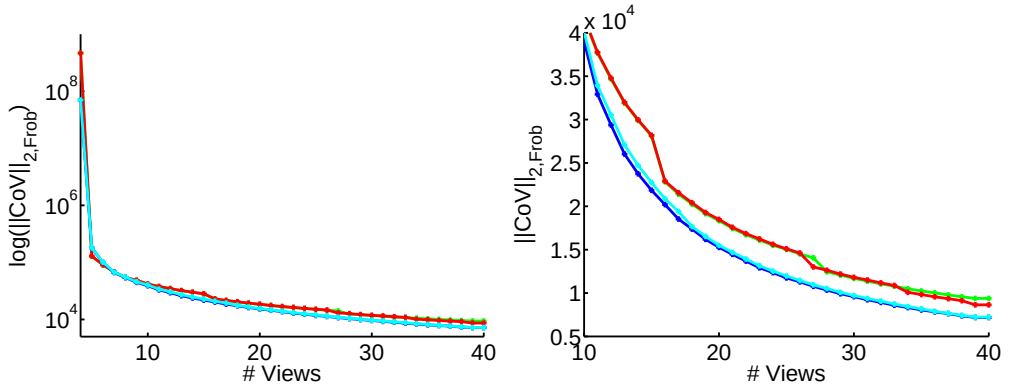


Figure 7.4: *Norm of the Co-Variance Matrix during Optimization. Green $||\boldsymbol{CoV}||_2$ and red $||\boldsymbol{CoV}||_{Frob}$ are obtained with Powell's method, while blue $||\boldsymbol{CoV}||_2$ and cyan $||\boldsymbol{CoV}||_{Frob}$ are obtained by the simplex method. The optimization is applied to the one-pixel object.*

cretization in the discrete search achieving acceptable coverage at least over multiple views.

We have tested two downhill non-gradient-based optimizers: the Simplex method and Powell's method with non-gradient based linear search [Press94]. The Simplex method is initialized with the starting value of the discrete search plus random points within its neighborhood. Although both methods may con-

verge to local maxima, we still achieve good results due to the discrete initialization. In Figure 7.4, we show the norm of the co-variance matrix achieved by maximizing F in Equation 7.5 with both, the Simplex and Powell's method. We have observed that Powell's method requires typically less function evaluations but achieves slightly worse results than the Simplex method. Figure 7.4 also compares the behavior of the optimization with the $2-$norm $||\boldsymbol{CoV}||_2$ versus the Frobenius norm $||\boldsymbol{CoV}||_{Frob}$. The difference is negligible due to the exponential rate of decay in the size of the singular values of the co-variance matrix.

Besides planning the next best view, the objective function can be employed to determine when to stop acquiring more views. Observe that the norm decays roughly exponentially as the number of samples of the one-pixel object increases. It indicates that one can stop the optimization after the gain in confidence is below a pre-defined threshold.

7.6 Multi-Pixel Objects

We are now generalizing our insights from the one-pixel object to 3D objects. The reflectance of real-world 3D objects cannot be modeled with sufficient accuracy by a single point. The complete surface area of a 3D object is not visible from any single given camera and light source position. Thus, the geometric shape influences the objective function and plays an important role in the uncertainty minimization.

7.6.1 Homogeneous vs. Spatially Varying BRDFs

Calculation of model uncertainty for a real 3D object requires the computation of the the Hessian matrices $\boldsymbol{H}_i$ considering all previous measurements at each point i on the surface. Assuming all points on the surface have identical reflectance then the overall co-variance matrix of all surface points is

$$\|\boldsymbol{CoV}\|_{Homogenous} = \| \left(\sum_i \boldsymbol{H}_i \right)^{-1} \|. \tag{7.6}$$

In this norm there is no relationship between samples and surface location. It does not ensure an even sampling of the surface and thus is only applicable if the object's surface consists of a single homogeneous material without the slightest spatial variation.

An approach more applicable to real-world 3D objects is to treat each surface point individually. The uncertainty measure in this case is Equation 7.7.

$$\|\boldsymbol{CoV}\|_{obj} = \sum_i \|\boldsymbol{H}_i^{-1}\|. \tag{7.7}$$

This uncertainty measure can only be decreased by imaging each surface point repeatedly. This is required if the goal of the measurement process is to model each surface point with a different BRDF but as well if surface points are eventually to be clustered and represented by a smaller set of BRDFs.

7.6.2 Real-World Constraints

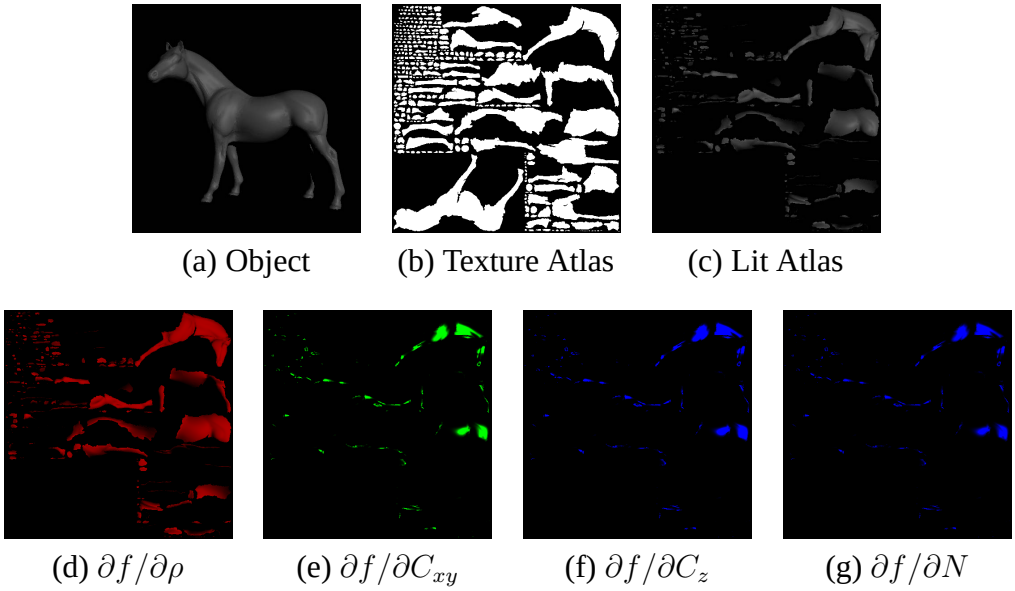
While performing a real measurement one has to take into account a number of additional constraints. The planning algorithm must prevent placement of light source and camera at the same location. In order to achieve this, we simply set the objective function to zero in these cases. Furthermore, the reliability of collected samples is not independent of the viewing angles for real measurements. Samples at grazing angles are typically hard to measure since the registration of the 3D object with the 2D image is not be perfect. We follow the approach by Lafortune et al. [Lafortune97] and weight the influence of each sample by the cosine of the angle between the surface normal and the viewing direction and between the normal and the lighting direction. The weight is set to zero for all points where one of the angles is larger than 80 degrees.

7.7 Implementation

We have defined the objective function F in Section 7.5 and constraints in the previous section. Here, we detail the implementation of an efficient optimization which results in a usable view planning algorithm. The view planning has to simulate the next view in order to evaluate the objective function. We calculate the norm of the co-variance matrices for each point on the surface. This computation has to be very efficient since it is executed several hundred times during one step of the optimization. We achieve this efficiency by exploiting newly available graphics hardware with floating-point precision frame buffers.

7.7.1 Texture Atlas

Since all quantities need to be computed for all surface points we represent the object's surface by a texture atlas (see Figure 7.5). We construct the texture atlas from the 3D mesh. All subsequent calculations are performed directly on the texture atlas. Using a vertex program it is very easy to perform calculations on the texture atlas: the final vertex position is set to the texture coordinates of the vertex while the original vertex position is used for other computations, e.g. lighting and shadowing.



(a) Object (b) Texture Atlas (c) Lit Atlas

(d) $\partial f/\partial \rho$ (e) $\partial f/\partial C_{xy}$ (f) $\partial f/\partial C_z$ (g) $\partial f/\partial N$

Figure 7.5: *Texture Atlas. Lighting and the derivatives of an isotropic Lafortune BRDF (see Appendix A) are computed in a texture atlas.*

7.7.2 Visibility and Shadows

The texture atlas shows all points on the object's surface at the same time, but the derivatives may only be computed for those points which are visible and lit for a given view. We accomplish this by first computing two depth maps of the 3D object, one depth map for the camera view and one from the view of the light source. As in traditional shadow mapping the point is visible and lit if the transformed pixel has the same or smaller depth than the corresponding points in the depth maps. Figure 7.5(c) shows the resulting texture atlas of all visible and lit points of the corresponding view Figure 7.5(a).

7.7.3 Derivatives and Matrix Norms

Given the 3D mesh and some BRDF Parameters we setup a fragment program which computes the derivatives of the BRDF model with respect to its parameters for the valid texels on the graphics board (see Figure 7.5(d)–(g)). We take the spatially varying parameters obtained by the clustering into account. The calculated derivatives are then downloaded from a floating-point frame buffer to main memory. Hessian matrices for the current view are calculated and added to the accumulated Hessian matrices of previous views. The result is inverted using singular value decomposition to obtain the norms of the co-variance matrices. The final value of the objective function is then computed as the sum of the objective

Task	$\partial f/\partial \boldsymbol{\beta}$	Download	$\boldsymbol{H}$	SVD	Total
Time $[s]$	0.119	0.477	0.021	0.523	1.138

Table 7.1: *Time consumed for computing the derivatives, to download the results to main memory, to add to the Hessian matrices and to perform the remaining calculations (SVD) in software in order to evaluate the objective function of one view on a 512x512 texture atlas. Note that on average only one fourth of the 147444 valid pixels were visible and lit.*

functions of each pixel. All software computations are done only at those pixels in the texture atlas which are visible and lit. A total of approximately 440 evaluations of the objective function are performed in the optimization of one view.

Table 7.7.3 lists how much time is spend for each of the computation steps. A considerable amount of time is unfortunately consumed by downloading the frame buffer. To save bandwidth we currently use a monochromatic isotropic Lafortune BRDF model with one lobe (4 parameters) for the view planning. The method can however be easily extended to work with more complex or multi-lobe models.

7.8 Measurement Results

We compare the views selected by our method to the views selected by a human expert, i.e., comparing to the results of the last chapter. The 3D object for which the spatially varying BRDF is acquired, are the angels shown in Figure 7.1. The

Rank k	# Pixels Planner	# Pixels Expert	$\bar{\sigma}_{L-k+1}$ Planner	$\bar{\sigma}_{L-k+1}$ Expert
0	10	857	-	-
1	300	2013	0.06587	0.04482
2	1755	973	2.3804	0.9823
3	1412	583	39.170	75.075
4	5767	4818	745.16	2308.94

Table 7.2: *Comparison of $\|\boldsymbol{CoV}\|$ Obtained by Planner and Human Expert after 27 views. Shown are the singular values averaged over all pixels. The planner acquired more pixels with higher rank and higher confidence. (Note the considerably smaller singular value in the last row).*

reflectance model of the angels has previously been captured with a set of 27 views (not shown). Figure 6.14 shows the images acquired with our planner. The planner selects camera and light source position in order to collect samples of each surface point under various angles. A comparison of the resulting uncertainty

Figure 7.6: *Comparison of Planner and Human Expert. First and third rows show results with the planner, while results obtained by the human expert are shown in the second and forth row. (Images shown are after every* 3^{rd} *view.) Notice how the planner selects samples to cover the object's surface evenly. The human expert acquires redundant samples of some surface areas (front) while other areas remain undersampled (back and bottom). The images show* $log(\|\boldsymbol{CoV}\|)$ *color-coded in* matlab jet *style. Blue means high confidence in the estimated BRDF parameters while the uncertainty increases towards red. Black regions have not been updated to full rank so far.*

$log(\|\boldsymbol{CoV}\|)$ shows that the planner covers the surface of the object quite evenly (see Figure 7.6). The human expert samples the front of the model frequently but misses areas on the rear, bottom and sides of the model (see Figure 7.1). The even sampling with our planner is even more apparant by studying the smallest singular value σ_1 of the $\boldsymbol{CoV}$ in Figure 7.8. The average singular values are compared in Table 7.2. The planner increases the rank of the $\boldsymbol{CoV}$ per pixel evenly. Table 7.2 shows that after 27 views the $\boldsymbol{CoV}$ has reached full rank at 62.4% of pixels. The lower rank (pseudo) $\boldsymbol{CoV}$ have already mostly gained rank 2 or 3. The $\boldsymbol{CoV}$ obtained by the human expert are mainly either, rank 4 (52.1%) or still rank 1 and rank 0 (unobserved). Additionally, the norms $\|\boldsymbol{CoV}\|$ with high rank are considerably larger than the number obtained with the planner, i.e., the measurements result in higher uncertainty.

Highlights have to be observed in order to update the rank of one pixel beyond rank 1 because three of the derivatives evaluate to nearly zero for non-highlight directions (Appendix A and Figure 7.5). For a human it is hard to keep track of the observed highlight areas and to reason on how to place the camera and the light source in order to observe a highlight at a specific surface region. The planner automatically considers this by maximizing the proposed objective function (see Figure 7.9).

The objective function evaluated after each view is shown in Figure 7.7. The series with the planner has been continued in simulation to 50 views. Note, that while local minima are encountered by the planner, it successfully reduces the error measure in each step and the steps decrease approximately exponentially as expected. (Note, that $log(F_{obj})$ in Equation 7.7 is a difference and the numbers have to be accumulated for absolute values.) The optimization per view took less than 2 minutes, computed on a 2.4GHz Pentium PC with an ATI Radeon 9700. The computations were carried out on a texture atlas with resolution 128x128. This time is constant and does not depend on the number of previously seen views, since the information is accumulated in the per-pixel Hessian matrix $\boldsymbol{H}$.

In summary, the planner helps to sample the angels' surface more evenly resulting in higher certainty in the BRDF parameters. The number of views (27) is generally too low for being able to judge the fit of BRDF parameters at each pixel. Using the clustering method of Chapter 6.5.2 the results however are visually already quite pleasing over the entire surface (Figure 7.1). The simulation suggests that after 48 views on more than 90% of the surface the $\boldsymbol{CoV}$ would reach full rank. Over 90% of the surface have already been imaged once after 5 views ($> 99\%$ after 10 views) in the actual measurements. The actual capturing process has been slightly slowed down by the planning but mainly because view registration has to be performed during acquisition when using the planner. The registration also revealed that some additional cues would be very helpful in setting up camera and light source. Besides the numerical positions, two shaded

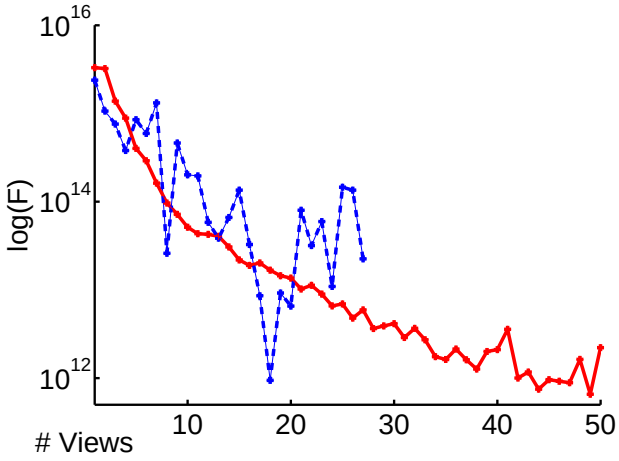


Figure 7.7: *Objective Function for Angel Acquisition. The red curve is obtained with the uncertainty minimization planner while the blue dashed curve is the objective function evaluated for the series taken by a human expert.*

images are currently provided to the user, one rendered from the proposed camera position and another from the light source. A low quality real-time registration algorithm would integrate nicely into the planner.

7.9 Conclusion

We have presented a novel method to analyze and plan the acquisition of realistic reflection models of 3D objects. Central part of the method is a measure of uncertainty which allows one to assess the quality of the sampling so far and to select from where to view the object. This uncertainty measure can be evaluated efficiently in graphics hardware with floating point precision and has been integrated into a view planner for BRDF acquisition. The performance of the planner compares favorably to the view selection by a human expert. It is extremely hard for humans to reason about the 4D BRDF on the surface of 3D objects. Consequently, it is very difficult for them to select good camera and light source positions in order to obtain a high-quality reflectance model. Our view planning algorithm can assist experts and enables novices to measure the BRDF of 3D objects, and will make automatic measurements more efficient.

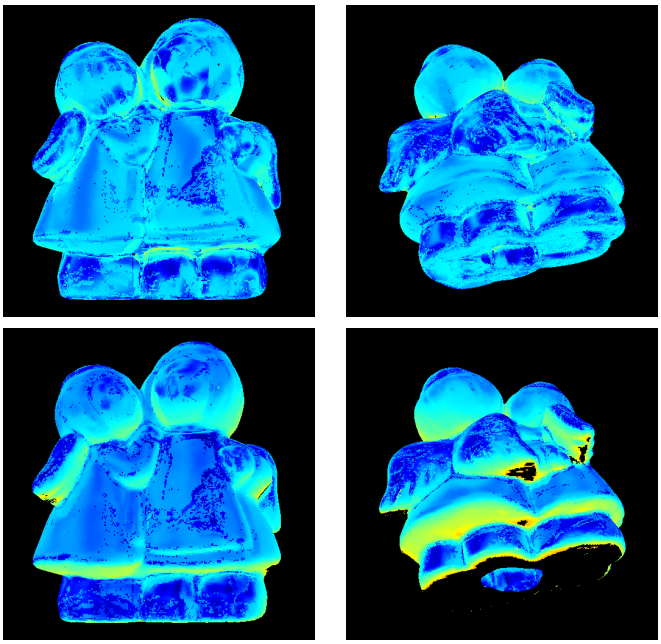


Figure 7.8: *Smallest Singular Value: Comparison of Planner and Human Expert. Top row show results with the planner, while the bottom row shows results obtained by the human expert. Notice how the planner selects samples to cover the object's surface evenly. The human expert acquires redundant samples of some surface areas (front) while other areas remain unobserved (back and bottom). The images show* $log(\|\sigma_1\|)$ *after* 27 *views color-coded in* matlab jet *style.*

Figure 7.9: *Views Planned for the Angels. Our algorithm positions both the camera and the light source to minimize BRDF uncertainty. As a result a highlight is observed at each surface point.*

Chapter 8
Discussion and Conclusion

The main goal of this thesis is the acquisition and representation of the appearance of real-world objects, in particular their reflective properties. Resulting 3D models can be rendered from novel view points under arbitrary lighting conditions. Several subproblems have been identified and solutions have been proposed: High-precision image-to-geometry registration, BRDF measurement in controlled environments, as well as a view-planning algorithm.

Efficient acquisition of high-quality models is made possible by the proposed techniques since they require only a small number of input images. Furthermore, efficiency is increased by planning the view points and light source positions. Highly detailed textures are recovered by precisely aligning the input images, and models of spatially varying reflection properties are inferred. These models faithfully capture the appearance of objects with a uniform surface material but also of objects composed of different materials. The estimated BRDF for an object may vary in the diffuse and the specular part. Measurement of this variation was not possible by any previous technique based on a sparse set of input images.

8.1 Summary

In the following, we briefly summarize our algorithms, discuss their advantages and drawbacks and show the advance over existing techniques.

8.1.1 Texture Registration

A fundamental building block of the presented method is the faithful alignment of input images to scanned 3D geometry. In Chapter 5, a silhouette-based registration algorithm is proposed which exploits graphics hardware to efficiently optimize the extrinsic camera parameters for each image with respect to a 3D model.

Given a segmented silhouette image the algorithm works automatically. The application of graphics-hardware allows us to perform faster optimization compared to other contour-based algorithms. Highly accurate alignment of detailed texture features is achieved by an additional texture-based matching step.

Since the algorithm is based on silhouettes no additional markers have to be attached to the object which would disturb the object's texture. A bottleneck however is the creation of reliable silhouettes. Although separate images are acquired in front of a white background to automatically extract the silhouette, the silhouettes typically include pieces created by the object's support. Geometry that is not explicitly represented by the 3D mesh may further occlude parts of the object or cast shadows. If necessary, the affected image regions may be marked manually.

8.1.2 BRDF Measurement

An algorithm for reconstructing a high-quality spatially varying BRDF from complex solid objects using only a small number of images has been proposed. In Chapter 6, the measurements are performed using a point light source in an otherwise black room. The same input images that are used for the BRDF measurements are also used to improve geometric detail by normal fitting. Combining the resulting optical and geometric data allows for accurately shaded, photo-realistic rendering of these objects from new viewpoints and under arbitrary lighting conditions. Several objects consisting of different materials have been acquired to demonstrate the performance of our approach. The resulting spatially varying BRDFs accurately represent the original materials. The normal fitting algorithm recovers geometric detail that was not represented by the original mesh obtained from the 3D scanner.

Representing the spatially varying BRDFs as texture maps allows to modify the object's geometry after the acquisition. Since the BRDF is not changed with the geometry the object can be altered or animated while preserving the material properties and thus the realistic appearance.

Compared to previous approaches for representing real-world objects, like surface light fields [Wood00] or reflection fields [Debevec00, Matusik02a] which needed up to several thousand images our methods require far less input data and even the size of the resulting model is considerably smaller. Having acquired a large number of images, however, image-based rendering can be performed directly. Reflectance fields can therefore represent objects with highly complex geometry including volumetric structures, and, to some extent, also subsurface scattering materials. Our approach currently relies on explicit geometry acquired by a 3D scanner in order to be able to fit a BRDF model to the measured data, which restricts the class of representable materials. Nevertheless, we have shown that our method applies to a broad range of objects.

One shortcoming of our method is that interreflections are not yet taken into account. In concave regions we will not recover the true BRDF of the surface. Instead, the apparent BRDF is approximated by fitting to a BRDF model. It depends on the BRDF model and on the actual configuration how well this approximation can work. A general solution would be to take interreflections into account in an iterative process: first, recover the BRDF based on direct illumination, and afterwards perform inverse global illumination for the affected regions, while taking the previously estimated BRDF as an initial solution.

The advantages of our method are the efficient acquisition, the simple setup, the high quality of the recovered spatially varying BRDFs, and the compact representation.

8.1.3 View-Planning

We described a view-planning algorithm in Chapter 7 which further increases the efficiency in the acquisition. It first analyses the quality of the estimated BRDF parameters in the form of per-pixel co-variance matrices given the current set of input images. The co-variance matrices are related to the uncertainty of the estimated parameters which mainly depends on the selected positions of the camera and of the light source. Based on the compact and efficiently evaluated uncertainty measure, a next view of the objects is proposed leading to an almost optimal set of views.

Only a small amount of work related to the analysis of the process of BRDF measurement has been done. Ramamoorthi and Hanrahan [Ramamoorthi01b] analyzed under which conditions inverse rendering problems are well-posed and McAllister [McAllister02a] calculated the number of samples necessary to cover the entire hemisphere by a light source with small extent. In other related appearance measurement approaches the problem of quality control regarding sparse sampling is either not addressed at all, e.g., [Sato97, Boivin01, Marschner99], or circumvented by very dense sampling [Dana99, Debevec00, Matusik02a, Furukawa02]. Appearance reconstruction from a sparse set of images and from densely sampled data may both benefit from our proposed uncertainty measure and view-planning algorithm.

8.2 Future Work

Although we have derived a set of new solutions, the investigation of these problems has raised many new questions to be explored.

In Chapter 6, we measured the appearance by clustering of surface points and fitting a low parameter BRDF model. Is it also possible to derive even a spatially

varying *non-parametric* BRDF from a sparse set of images? Measuring a tabulated BRDF instead of fitting a BRDF model would lift the restrictions implicitly imposed by the model. This would permit the accurate reconstruction of even more complex materials, e.g., velvet or layered materials. One solution could be to combine and adopt the measured BRDF samples of surface points with similar reflection properties, in a fashion comparable to what is done during the current clustering process.

Another future research direction is to extend the clustering method to work with arbitrary illumination, still recovering spatial variation in the diffuse and the specular part. This would permit to perform the measurements of real world objects in more natural environments, not necessarily in a highly controlled photo studio. Furthermore, one may measure dynamic objects instead of static ones. The image-to-geometry registration has to be adjusted to work on dynamic geometry. New algorithms have to be developed to determine reliable shadows and visibility in the presence of dynamic objects and scene geometry that is not entirely captured by the model.

The view planning algorithm proposed in Chapter 7 is designed to work with a single point light source. It is worth investigating if the planning algorithm can also be applied for less constrained environments. At present the cost of such an algorithm would be dominated by the convolution of the spatially varying BRDF with the environment. Efficient, accurate rendering algorithms for objects with dynamically varying per-pixel BRDFs are required.

The presented methods currently require a geometric model of the object in order to resample the reflectance samples for each surface point. The appearance of volumetric materials such as hair of fur, where no explicit geometric model is known, can so far only be acquired by dense sampling approaches. Combining inverse reflectometry and photometric stereo may help to infer both geometry and reflection properties of objects with complex geometry even from a sparse set of images.

Most approaches for appearance acquisition, including those presented in this thesis, are proof-of-concept or research prototypes and none of them is currently widely used in commercial applications. Only the simplest form, diffuse texture acquisition, is nowadays commercially available bundled with 3D scanning devices. As we have seen in Chapter 5 there are several problems when constructing a diffuse texture. Integration of measurement techniques for spatially varying BRDFs into standard 3D scanning equipment would make high-quality appearance acquisition of real-world objects available to a larger number of users.

Appendix A
Hessian Matrix

The Hessian matrix for Lafortune model (Equation 7.1) is given by the following equation:

$$\frac{\partial^2 Q}{\partial \beta^2} = -2 \left(\frac{\partial L}{\partial \beta} \right)^T \frac{\partial L}{\partial \beta}. \tag{A.1}$$

The derivatives $\frac{\partial L}{\partial \beta}$ of the single specular lobe Lafortune model are $\partial L / \partial \rho_d = 1$,

$$\begin{aligned}
\frac{\partial L}{\partial C_{xy}} &= (C_{xy}\omega_{i_x}\omega_{o_x} + C_{xy}\omega_{i_y}\omega_{o_y} + C_z\omega_{i_z}\omega_{o_z})^{N-1} N(\omega_{i_x}\omega_{o_x} + \omega_{i_y}\omega_{o_y}), \\
\frac{\partial L}{\partial C_z} &= (C_{xy}\omega_{i_x}\omega_{o_x} + C_{xy}\omega_{i_y}\omega_{o_y} + C_z\omega_{i_z}\omega_{o_z})^{N-1} N(\omega_{i_z}\omega_{o_z}), \text{ and} \\
\frac{\partial L}{\partial N} &= (C_{xy}\omega_{i_x}\omega_{o_x} + C_{xy}\omega_{i_y}\omega_{o_y} + C_z\omega_{i_z}\omega_{o_z})^{N} N \cdot \\
&\quad \log(C_{xy}\omega_{i_x}\omega_{o_x} + C_{xy}\omega_{i_y}\omega_{o_y} + C_z\omega_{i_z}\omega_{o_z}).
\end{aligned}$$

Bibliography

[Adams99] ANSEL ADAMS. *The Negative*. Little, Brown and Company, 6th paperback printing edition, 1999.

[Adelson91] E. H. ADELSON AND J. R. BERGEN. *Computational Models of Visual Processing*, chapter 1 (The Plenoptic Function and the Elements of Early Vision). MIT Press, Cambridge, MA, 1991.

[Airey90] J. AIREY, J. ROHLF, AND F. BROOKS. Towards Image Realism with Interactive Update Rates in Complex Virtual Building Environments. In *Symposium on Interactive 3D Graphics*, pages 41–50, February 1990.

[Aloimonos87] J. Y. ALOIMONOS, I. WEISS, AND A. BANDYOPADHYAY. Active Vision. *International Journal of Computer Vision*, 1:333–356, 1987.

[Ashikhmin00a] M. ASHIKHMIN, S. PREMOZE, AND P. SHIRLEY. A Microfacet-based BRDF Generator. In *Proc. SIGGRAPH*, pages 65–74, July 2000.

[Ashikhmin00b] M. ASHIKHMIN AND P. SHIRLEY. An Anisotropic Phong BRDF Model. *Journal of Graphics Tools*, 5(2):25–32, 2000.

[Bajcsy88] R. BAJCSY. Active Vision. *Proceedings of the IEEE*, 76:996–1005, 1988.

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

[Bayer76] BRYCE E. BAYER. Color Imaging Array. US patent number 3,971,065, July 1976.

[Beasley85] J.E. BEASLEY. An exact two-dimensional non-gauillotine cutting tree search procedure. *Operation Research*, 33(1):49–64, 1985.

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

[Belhumeur98] PETER N. BELHUMEUR AND DAVID J. KRIEGMAN. What Is the Set of Images of an Object Under All Possible Illumination Conditions? *International Journal of Computer Vision*, 28(3):245–260, July 1998.

[Bergevin96] R. BERGEVIN, M. SOUCY, AND H. GAGNON. Towards a General Multi-View Registration Technique. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(5):540–547, 1996.

[Bernardini01] FAUSTO BERNARDINI, IOANA M. MARTIN, AND HOLLY RUSHMEIER. High-quality texture reconstruction from multiple scans. *IEEE Transactions on Visualization and Computer Graphics*, 7(4):318–332, October - November 2001. ISSN 1077-2626.

[Bernardini02a] FAUSTO BERNARDINI AND HOLLY RUSHMEIER. The 3D Model Acquisition Pipeline. *Computer Graphics Forum*, 21(2):149–172, 2002.

[Bernardini02b] FAUSTO BERNARDINI, HOLLY RUSHMEIER, IOANA M. MARTIN, JOSHUA MITTLEMAN, AND GABRIEL TAUBIN. Building a Digital Model of Michelangelo's Florentine Piet. *IEEE Computer Graphics & Applications*, 22(1):59–67, January 2002.

[Besl92] P.J. BESL AND N.D. MCKAY. A Method for Registration of 3D Shapes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[Beymer96] D. BEYMER AND T. POGGIO. Image Representations for Visual Learning. *Science*, 272:1905–1909, 1996.

[Blinn76] J. BLINN AND M. NEWELL. Texture and Reflection in Computer Generated Images. *Communicatoins of the ACM*, 19:542–546, 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.

[Bouguet00] JEAN-YVES BOUGUET. Camera Calibration Toolbox for Matlab. See http://www.vision.caltech.edu/bouguetj, 2000.

[Bretthorst88] G.L. BRETTHORST. *Bayesian spectrum analysis and parameter estimation*. Springer-Verlag, New York, 1988.

[Brown92] LISA GOTTESFELD BROWN. A Survey of Image Registration Techniques. *ACM Computing Surveys*, 24(4):325–376, Dec 1992.

[Brunie92] L. BRUNIE, S. LAVALLÉE, AND R. SZELISKI. Using Force Fields Derived from 3D Distance Maps for Inferring the Attitude of a 3D Rigid Object. In Giulio Sandini, editor, *Proceedings of Computer Vision (ECCV '92)*, volume 588 of *LNCS*, pages 670–675. Springer, Mai 1992.

[Buehler01] C. BUEHLER, M. BOSSE, L. MCMILLAN, AND S. GORTLER. Unstructured Lumigraph Rendering. In *Proceedings SIGGRAPH*, pages 425–432, aug 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.

[Callieri02] M. CALLIERI, P. CIGNONI, AND R. SCOPIGNO. Reconstructing Textured Meshes from Multiple Range RGB Maps. In *Proceedings of Vision, Modeling and Visualization*, pages 419–426, Erlangen, Germany, 2002.

[Carpenter00] J. CARPENTER AND J. BITHELL. Bootstrap confidence intervals: when, which, what? A pratical gude for medical statisticians. *Statistics in Medicine*, 19:1141–1164, 2000.

[Chai00] J.X. CHAI, S.C. CHAN, X. TONG, AND H.Y. SHUM. Plenoptic sampling. In *Proceedings SIGGRAPH*, 2000.

[Chang99] CHUN-FA CHANG, GARY BISHOP, AND ANSELMO LASTRA. LDI Tree: A Hierarchical Representation for Image-Based Rendering. In *Proceedings of SIGGRAPH 99*, Computer Graphics Proceedings, Annual Conference Series, pages 291–298, August 1999.

[Chaumette96] F. CHAUMETTE, S. BOUKIR, P. BOUTHEMY, AND D. JUVIN. Structure From Controlled Motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(5):492–504, 1996.

[Chen92] Y. CHEN AND G. G. MEDIONI. Object Modeling by Registration of Multiple Range Images. *Image and Vision Computing*, 10(3):145–155, 1992.

[Chen95] SHENCHANG ERIC CHEN. QuickTime VR - An Image-Based Approach to Virtual Environment Navigation. In *Computer Graphics Proceedings, Annual Conference Series (Proc. SIGGRAPH '95)*, pages 29–38, 1995.

[Chen99] B. CHEN, F. DACHILLE, AND A. KAUFMAN. Forward Image Warping. In *IEEE Visualization*, pages 89–96, October 1999.

[Chen02] W.-C. CHEN, J.-Y. BOUGUET, M. H. CHU, AND R. GRZESZCZUK. Light Field Mapping: Efficient Representation and Hardware Rendering of surface Light Fields. In *Proceedings SIGGRAPH*, pages 447–456, 2002.

[Chuang00] Y.-Y/ CHUANG, D. E. ZONGKER, J. HINDORFF, B. CURLESS, D. H. SALESIN, AND R. SZELISKI. Environment

Matting Extensions: Towards Higher Accuracy and Real-Time Capture. In *Proceedings of ACM SIGGRAPH 2000*, Computer Graphics Proceedings, Annual Conference Series, pages 121–130, July 2000.

[Chvátal75] V. CHVÁTAL. A combinatorial theorem in plane geometry. *J. Combin. Theory (B)*, 18:39–41, 1975.

[CIE86] Colorimetry. Publication CIE No. 15.2, 1986.

[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.

[Cowan88] C. K. COWAN AND P. D. KOVESI. Automatic Sensor Placement from Vision Task Requirements. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3):407–416, 1988.

[Curless96] B. CURLESS AND M. LEVOY. A volumetric method for building complex models from range images. In *Proceedings SIGGRAPH*, pages 303–312, August 1996.

[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.

[Daubert01] KATJA DAUBERT, HENDRIK P. A. LENSCH, WOLFGANG HEIDRICH, AND HANS-PETER SEIDEL. Efficient Cloth Modeling and Rendering. In *Rendering Techniques 2001: Proceedings of the 12th Eurographics Workshop on Rendering*, pages 63–70, London, Great Britain, 2001. Eurographics, Springer.

[Daubert02] K. DAUBERT AND H. SEIDEL. Hardware-Based Volumetric Knit-Wear. *Computer Graphics Forum*, 21(3):575–784, 2002.

[Debevec96] PAUL E. DEBEVEC, CAMILLO J. TAYLOR, AND JITENDRA MALIK. Modeling and Rendering Architecture from

Photographs: A Hybrid Geometry- and Image-Based Approach. In *Proceedings of SIGGRAPH 96*, pages 11–20, August 1996.

[Debevec97] P. DEBEVEC AND J. MALIK. Recovering High Dynamic Range Radiance Maps from Photographs. In *Proc. SIGGRAPH*, pages 369–378, August 1997.

[Debevec98] PAUL E. DEBEVEC, YIZHOU YU, AND GEORGE D. BORSHUKOV. Efficient View-Dependent Image-Based Rendering with Projective Texture-Mapping. *Eurographics Rendering Workshop 1998*, pages 105–116, June 1998.

[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.

[DeYoung97] J. DEYOUNG AND A. FOURNIER. Properties of Tabulated Bidirectional Reflectance Distribution Functions. In *Graphics Interface*, pages 47–55, May 1997.

[DiCiccio96] T.J. DICICCIO AND B. EFRON. Bootstrap Confidence Intervals. *Statistical Science*, 11(3):198–212, 1996.

[Dietrich00] S. DIETRICH. Elevation Maps. Technical report, NVIDIA Corporation, 2000.

[Dorsey91] J. DORSEY, F. SILLION, AND D. GREENBERG. Design and Simulation of Opera Lighting and Projection Effects. In *Proceedings SIGGRAPH*, pages 41–50, Las Vegas, Nevada, July 1991.

[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.

[Edelsbrunner98] H. EDELSBRUNNER. Shape reconstruction with denaunay complex. In *Latin'98: Theoretical Informatics. Third Latin American Symposium, Campinas, Brazil*, pages 119–132. Lecture Noes in Computer Science, LNCS 1380, New York: Springer, 1998.

[Efron86] B. EFRON AND R. TIBSHIRANI. Bootstrap Methods for Standard Errors, Confidence Intervals, and other Measures of Statistical Accuracy. *Statistical Science*, 1(1):54–75, 1986.

[Fellner98a] D. FELLNER. Generalized Electronic Documents. In *Computer Graphics International 1998*, June 1998.

[Fellner98b] D. W. FELLNER, S. HAVEMANN, AND G. MÜLLER. Modeling of and navigation in complex 3D documents. *Computers & Graphics*, 22(6):647–653, December 1998.

[Fellner00] DIETER FELLNER, JÖRG HABER, SVEN HAVEMANN, LEIF KOBBELT, HENDRIK LENSCH, GORDON MÜLLER, INGMAR PETER, ROBERT SCHNEIDER, HANS-PETER SEIDEL, AND WOLFGANG STRAER. Beitrge der Computergraphik zur Realisierung eines verallgemeinerten Dokumentbegriffs. *Informationstechnik und Technische Informatik*, 42(6):8–18, December 2000.

[Forsyth89] DAVID A. FORSYTH AND ANDREW ZISSERMAN. Mutual Illumination. In *Proceedings of Computer Vision and Pattern Recognition (CVPR '89)*, pages 466–473. IEEE Computer Society Press, 1989.

[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.

[Furukawa02] R. FURUKAWA, H. KAWASAKI, K. IKEUCHI, AND M. SAKAUCHI. Appearance based object modeling using texture database: Acquisition compression and rendering. In *Thirteenth Eurographics Workshop on Rendering*, pages 267–276, June 2002.

[Garland97] M. GARLAND AND P. HECKBERT. Surface Simplification Using Quadric Error Metrics. In *Proc. SIGGRAPH*, pages 209–216, August 1997.

[Gersho92] A. GERSHO AND R. GRAY. *Vector Quantization and Signal Compression*. Kluwer Acad. Publishers, 1992.

[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.

[Goesele00] MICHAEL GOESELE, WOLFGANG HEIDRICH, HENDRIK P.A. LENSCH, AND HANS-PETER SEIDEL. Building a Photo Studio for Measurement Purposes. In *Proceedings of Vision, Modeling, and Visualization*, pages 231–238, November 2000.

[Goesele01a] MICHAEL GOESELE, WOLFGANG HEIDRICH, AND HANS-PETER SEIDEL. Color Calibrated High Dynamic Range Imaging with ICC Profiles. In *9th Color Imaging Conference*, pages 286–290, November 2001.

[Goesele01b] MICHAEL GOESELE, WOLFGANG HEIDRICH, AND HANS-PETER SEIDEL. Entropy Based Dark Frame Subtraction. In *Image Processing, Image Quality, Image Capture Systems Conference (PICS)*, pages 293–298, April 2001.

[Gonzalez-Banos97] H.H. GONZALEZ-BANOS, L.J. GUIBAS, J.C. LATOMBE, S.M. LAVALLE, D. LIN, R. MOTWANI, AND C. TOMASI. Motion planning with visibility constraints: building autonomous observers. In *Robotics Research - The* 8^{th} *International Symposium*, pages 95–101, Oct. 1997.

[Gortler96] S. GORTLER, R. GRZESZCZUK, R. SZELINSKI, AND M. COHEN. The Lumigraph. In *Proc. SIGGRAPH*, pages 43–54, August 1996.

[Greene86] N. GREENE. Environment Mapping and Other Applications of World Projections. *IEEE Computer Graphics & Applications*, 6(11):21–29, nov 1986.

[Guenter98] BRIAN GUENTER, CINDY GRIMM, DANIEL WOOD, HENRIQUE MALVAR, AND FRÉDÉRIC PIGHIN. Making Faces. *Proceedings of SIGGRAPH 98*, pages 55–66, July 1998.

[Hakura01] ZIYAD SAMI HAKURA. *Inverse Rendering Methods for Hardware-Accelerated Display of Parameterized Image Spaces*. PhD thesis, Stanford University, 2001.

[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.

[Harris88] C. HARRIS AND M.J. STEPHENS. A Combined Corner and Edge Detector. In *Alvey88*, pages 147–152, 1988.

[Hastie02] T. HASTIE, R. TIBSGIRANI, AND J.H. FRIEDMAN. *The elements of statistical learning: data mining, inference and prediction*. Springer-Verlag, 5 edition, 2002.

[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.

[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.

[Heikkila97] J. HEIKKILA AND O. SILVEN. A Four-Step Camera Calibration Procedure With Implicit Image Correction. In *CVPR97*, 1997.

[Hlavac96] V. HLAVAC, A. LEONARDIS, AND T.WERNER. Automatic selection of reference views for image-based scene representations. In *eccv*, pages 525–526, 1996.

[Horn86] B. HORN. *Computer Vision*. MIT Press, Cambridge, Mass., 1986.

[Hunt95] ROBERT W. G. HUNT. *The reproduction of colour*. Fountain Press, 5. ed. edition, 1995.

[Hunter87] RICHARD S. HUNTER AND RICHARD W. HAROLD. *The measurement of appearance*. Wiley, 2. ed., 5. print. edition, 1987.

[ICC98] Specification ICC.1:1998-09, File Format for Color Profiles. available from http://www.color.org, 1998.

[Ikeuchi01] K. Ikeuchi and Y. Sato, editors. *Modeling from Reality*. Kluwer Academic Publishers, Boston, 2001.

[Imai98] F.H. IMAI AND R.S. BERNS. High-resolution multi-spectral image archives: A hybrid approach. In *Proceedings of the Sixth Color Imaging Conference: Color Science, Systems, and Applications*, pages 224–227. IS&T, 1998.

[Imai01a] F.H. IMAI AND R.S. BERNS. Spectral estimation of artist oil paints using multi-filter trichromatic imaging. In *Proceedings of AIC Colour 01*, 2001.

[Imai01b] F.H. IMAI, M.R. ROSEN, AND R.S. BERNS. Multi-spectral imaging of van Gogh's self-portrait at the National Gallery of Art, Washington, D.C. In *Proceedings of IS&T's PICS Conference*, pages 185–189. IS&T, 2001.

[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.

[Ip96] HORACE H. S. IP AND LIJUN YIN. Constructing a 3D individualized head model from two orthogonal views. *The Visual Computer*, 12(5):254–268, 1996.

[Ismert03] RYAN M. ISMERT, KAVITA BALA, AND DONALD P. GREENBERG. Detail Synthesis for Image-based Texturing. Technical Report PCG-03-1, Program of Computer Graphics Technical Report, Cornell, 2003.

[Iwakiri00] YUYA IWAKIRI, YUUICHI OMORI, AND TOYOHISA KANEKO. Practical Texture Mapping on Free-form Surfaces. In *8th Pacific Conference on Computer Graphics and Applications*, pages 97–104. IEEE, October 2000. ISBN 0-7695-0868-5.

[Jensen96] H.W. JENSEN. Global Illumination using Photon Maps. In *7th Eurographics Workshop on Rendering*, pages 21–30, jun 1996.

[Jensen01] H. W. JENSEN, S. MARSCHNER, M. LEVOY, AND P. HANRAHAN. A Practical Model for Subsurface Light Transport. In *Proceedings SIGGRAPH*, pages 511–518, August 2001.

[Kajiya86] JAMES T. KAJIYA. The rendering equation. In *Computer Graphics (SIGGRAPH '86 Proceedings)*, 20(4):143-150, August 1986.

[Karner96] K. KARNER, H. MAYER, AND M. GERVAUTZ. An Image based Measurement System for Anisotropic Reflection. *Computer Graphics Forum*, 15(3):119–128, August 1996. ISSN 1067-7055.

[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] J. KAUTZ AND H.-P. SEIDEL. Towards Interactive Bump Mapping with Anisotropic Shift-Variant BRDFs. In *Eurographics/SIGGRAPH Hardware Workshop*, pages 51–58, August 2000.

[Kautz01] J. KAUTZ AND H.-P. SEIDEL. Hardware Accelerated Displacement Mapping for Image Based Rendering. In *Graphics Interface*, pages 61–70, June 2001.

[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.

[Kay94] GREG KAY AND TERRY CAELLI. Inverting an illumination model from range and intensity maps. *CVGIP: Image Understanding*, 59(2):183–201, March 1994.

[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.

[Kodak Company98] KODAK COMPANY. Kodak Professional DCS 500 Series Digital Cameras – User's Guide, 1998.

[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.

[Kriegman92] D. J. KRIEGMAN, B. VIJAYAKUMAR, AND J. PONCE. Constraints for Recognizing and Locating Curved 3D Objects from Monocular Image Features. In *Proceedings of Computer Vision (ECCV '92)*, volume 588 of *LNCS*, pages 829–833. Springer, mai 1992.

[Kurazume02] RYO KURAZUME, KO NISHINO, ZHENGYOU ZHANG, AND KASUSHI IKEUCHI. Simultaneous 2D images and 3D geometric model registration for texture mapping utilizing reflectance atrtribute. In *ACCV 2002: The 5th Asian Conference on Computer Vision*, January 2002.

[Kutulakos94] K.N. KUTULAKOS AND C.R. DYER. Recovering Shape by Purposive Viewpoint Adjustment. *International Journal of Computer Vision*, 12:113–136, 1994.

[Kutulakos00] K.N. KUTULAKOS AND S.M. SEITZ. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.

[Lafortune93] E. LAFORTUNE AND Y. WILLEMS. Bidirectional Path Tracing. In *Computergraphics*, pages 95–104, 1993.

[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.

[Lang00] J. LANG AND M.R.M. JENKIN. Active object modeling with VIRTUE. *Autonomous Robots*, 8(2):141–159, 2000.

[Larson97] GREG WARD LARSON, HOLLY RUSHMEIER, AND C. PIATKO. A Visbility Matching Tone Reproduction Operator for High Dynamic Range Scenes. *IEEE Transactions on Visualization and Computer Graphics*, 3(4):291–306, 1997.

[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] HENDRIK LENSCH, WOLFGANG HEIDRICH, AND HANS-PETER SEIDEL. Automated Texture Registration and Stitching for Real World Models. In *Pacific Graphics '00*, pages 317–326, October 2000.

[Lensch01a] HENDRIK LENSCH, WOLFGANG HEIDRICH, AND HANS-PETER SEIDEL. Silhouette-based Algorithm for Texture Registration and Stitching. *Graphical Models*, 63(4):245–262, July 2001.

[Lensch01b] HENDRIK P. A. LENSCH, MICHAEL GOESELE, AND HANS-PETER SEIDEL. A Framework for the Acquisition, Processing and Interactive Display of High Quality 3D Models – Tutorial Notes of the DAGM 2001. Research report, Max-Planck-Institut für Informatik, Munich, Germany, 2001.

[Lensch01c] 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.

[Lensch01d] HENDRIK P. A. LENSCH, JAN KAUTZ, MICHAEL GOESELE, AND HANS-PETER SEIDEL. 3D Model Acquisition Including Reflection Properties. In Dieter W. Fellner, Norbert Fuhr, and Ian Witten, editors, *Proceedings of the ECDL Workshop Generalized Documents*, pages 1–6, Darmstadt, Germany, 2001.

[Lensch01e] HENDRIK P. A. LENSCH, JAN KAUTZ, MICHAEL GOESELE, AND HANS-PETER SEIDEL. A Framework for the Acquisition, Processing, Transmission, and Interactive Display of High Quality 3D Models on the Web – Tutorial Notes of the Web3D Conference 2001. Paderborn, Germany, 2001.

[Lensch02a] HENDRIK P. A. LENSCH. Measuring spatial variation with complex BRDFs. In Steve Marschner and Ravi Ramamoorthi, editors, *Acquiring Material Models Unsing Inverse Rendering (Course 39)*, volume 39 of *ACM Siggraph Course Notes*, chapter 5, pages 80–113. ACM SIGGRAPH, New York, USA, July 2002.

[Lensch02b] 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.

[Lensch02c] 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.

[Lensch03a] 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*, 2:27, April 2003.

[Lensch03b] HENDRIK P. A. LENSCH, JOCHEN LANG, ASLA M. SA, AND HANS-PETER SEIDEL. Planned Sampling of Spatially-Varying BRDFs . *Computer Graphics Forum*, 2003. accepted for publication at EUROGRAPHICS 2003.

[Leung99] T. LEUNG AND J. MALIK. Recognizing Surfaces using Three-Dimensional Textons. In *IEEE Int. Conf. on Computer Vision*, sep 1999.

[Levoy96] M. LEVOY AND P. HANRAHAN. Light Field Rendering. In *Proc. SIGGRAPH*, pages 31–42, August 1996.

[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.

[Liu01] XINGUO LIU, YIZHOU YU, AND HEUNG-YEUNG SHUM. Synthesizing Bidirectional Texture Functions for Real-World Surfaces. In *Proceedings of ACM SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 97–106. ACM Press / ACM SIGGRAPH, August 2001. ISBN 1-58113-292-1.

[Lloyd82] S. LLOYD. Least squares quantization in PCM. *IEEE Trans. on Information Theory*, IT-28:129–137, 1982.

[Loscos99] C. LOSCOS, M. C. FRASSON, G. DRETAKIS, B. WALTER, X. GRANIER, AND P. POULIN. Interactive Virtual Relighting and Remodeling of Real Scenes. In *10th Eurographics Workshop on Rendering*, pages 329–340, 1999.

[Loscos00] C. LOSCOS AND G. DRETTAKIS. Low-cost Photometric Calibration for Interactive Relighting. In *Frist French-British International Workshop on Virtual Reality*, 2000.

[Lowe91] DAVID G. LOWE. Fitting Parameterized Three-Dimensional Models to Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):441–450, 1991.

[Lu98] R. LU, J. KOENDERINK, AND A. KAPPERS. Optical Properties (bidirectional reflectance distribution functions) of velvet. *Applied Optics*, 37(25):5974–5984, September 1998.

[Luther27] R. LUTHER. Aus dem Gebiet der Farbreizmetrik. *Zeitschrift fr technische Physik*, 8(12):540–558, 1927.

[MacQueen67] J. MACQUEEN. Some methods for classification and analysis of multivariate observations. In *Proc. of the 5th Berkeley Symp. on Mathematical Statistics and Probability*, volume 1, 1967.

[Madden93] BRIAN C. MADDEN. Extended Intensity Range Imaging. Technical report, University of Pennsylvania, GRASP Laboratory, 1993.

[Magda01] S. MAGDA, T. ZICKLER, D. KRIEGMAN, AND P. BELHUMEUR. Beyond Lambert: Reconstructing Surfaces with Arbitrary BRDFs. In *ICCV*, pages 391–398, 2001.

[Malzbender01] TOM MALZBENDER, DAN GELB, AND HANS WOLTERS. Polynomial Texture Maps. In *Proceedings of ACM SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 519–528. ACM Press / ACM SIGGRAPH, August 2001. ISBN 1-58113-292-1.

[Marchand99] E. MARCHAND AND F. CHAUMETTE. Active Vision for Complete Scene Reconstruction and Exploration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(1):65–72, 1999.

[Marengoni00] M. MARENGONI, B. DRAPER, A. HANSON, AND R. SITARAMAN. Placing Observers to Cover a Polyhedral Terrain in Polynomial Time. *Vision and Image Computing*, 18(10):773–780, 2000.

[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.

[Marschner00] 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.

[Mason95] S.O. MASON AND A. GRÜN. Automatic Sensor Placement for Accurate Dimensional Inspection. *Computer Vision and Image Understanding*, 61(3):454–467, 1995.

[Masselus02] V. MASSELUS, P. DUTRÉ, AND F. ANRYS. The Free-form Light Stage. In S. Debevec, P. Gibson, editor, *Proc. EuroGraphics Workshop on Rendering*, pages 257–265, Pisa, Italy, June 26–28 2002.

[Matsushita99] KENJI MATSUSHITA AND TOYOHISA KANEKO. Efficient and Handy Texture Mapping on 3D Surfaces. *Computer Graphics Forum*, 18(3):349–358, September 1999.

[Matusik00] WOJCIECH MATUSIK, CHRIS BUEHLER, RAMESH RASKAR, STEVEN J. GORTLER, AND LEONARD MCMILLAN. Image-Based Visual Hulls. In *Proceedings of ACM SIGGRAPH 2000*, Computer Graphics Proceedings, Annual Conference Series, pages 369–374, July 2000.

[Matusik02a] W. MATUSIK, H. PFISTER, A. NGAN, P. BEARDSLEY, AND L. MCMILLAN. Image-Based 3D Photography Using Opacity Hulls. In *Proceedings SIGGRAPH*, pages 427–437, July 2002.

[Matusik02b] W. MATUSIK, H. PFISTER, R. ZIEGLER, A. NGAN, AND L. MCMILLAN. Acquisition and Rendering of Transparent and Refractive Objects. In *Thirteenth Eurographics Workshop on Rendering*, pages 277–288, June 2002.

[McAllister99] D. MCALLISTER, L. NYLAND, V. POPESCU, A. LASTRA, AND C. MCCUE. Real-Time Rendering of Real-World Environments. In *10th Eurographics Rendering Workshop*, pages 153–168, June 1999.

[McAllister02a] D. MCALLISTER. *A Generalized Representation of Surface Appearance*. PhD thesis, University of North Carolina, 2002.

[McAllister02b] D. MCALLISTER, A. LASTRA, AND W. HEIDRICH. Efficient Rendering of Spatial Bi-directional Reflectance Distribution Functions. In *Proceedings Graphics Hardware*, September 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.

[McMillan95] L. MCMILLAN AND G. BISHOP. Plenoptic Modeling: An Image-Based Rendering System. In *Proceedings SIGGRAPH*, pages 39–46, August 1995.

[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.

[Miller98] G. MILLER, S. RUBIN, AND D. PONCELEON. Lazy Decompression of Surface Light Fields for Precomputed Global Illumination. In *9th Eurographics Workshop on Rendering*, pages 281–292, June 1998.

[Mortensen95] ERIC N. MORTENSEN AND WILLIAM A. BARRETT. Intelligent Scissors for Image Composition. In *Proceedings of SIGGRAPH 95*, pages 191–198, August 1995.

[Nayar90a] S. NAYAR, K. IKEUCHI, AND T. KANADE. Determining Shape and Reflectance of Hybrid Surfaces by Photometric Sampling. *RA*, 6(4):418–431, 1990.

[Nayar90b] SHREE K. NAYAR, KATSUSHI IKEUCHI, AND TAKEO KANADE. Shape from Interreflections. In *International Conference on Computer Vision (ICCV '90)*, pages 2–11, Osaka, Japan, 1990.

[Nayar91a] S. NAYAR, K. IKEUCHI, AND T. KANADE. Recovering Shape in the Presence of Interreflections. In *IEEE Int. Conf. on Robotics and Automation*, pages 1814–1819, 1991.

[Nayar91b] S. NAYAR, K. IKEUCHI, AND T. KANADE. Surface reflection: physical and geometrical perspectives. *IEEE Trans. PAMI*, 13(7):611–634, 1991.

[Neugebauer99] PETER J. NEUGEBAUER AND KONRAD KLEIN. Texturing 3D Models of Real World Objects from Multiple Unregistered Photographic Views. *Computer Graphics Forum*, 18(3):245–256, September 1999.

[Nicodemus77] F. NICODEMUS, J. RICHMOND, J. HSIA, J. GINSBERG, AND T. LIMBERIS. *Geometric Considerations and Nomenclature for Reflectance*. Monograph 161. National Bureau of Standards (US), oct 1977.

[Nimeroff94] J. NIMEROFF, E. SIMONCELLI, AND J. DORSEY. Efficient Re-rendering of Naturally Illuminated Environments. In *Fifth Eurographics Workshop on Rendering*, pages 359–373, June 1994.

[Nishino01a] K. NISHINO, Y. SATO, AND K. IKEUCHI. "Eigen-Texture Method: Appearance Compression and Synthesis based on a 3D Model". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1257–1265, nov 2001.

[Nishino01b] 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.

[NVI02a] NVIDIA Corporation. *NVIDIA OpenGL Extension Specifications*, 2002. Available from http://www.nvidia.com.

[NVI02b] NVIDIA Corporation. *NVIDIA Texture Shaders Presentation*, 2002. Available from http://developer.nvidia.com.

[Oh01] BYONG MOK OH, MAX CHEN, JULIE DORSEY, AND FRÉDO DURAND. Image-Based Modeling and Photo Editing. In *Proceedings of ACM SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 433–442, August 2001.

[Oliveira99] M. OLIVEIRA AND G. BISHOP. Image-Based Objects. In *1999 ACM Symposium on Interactive 3D Graphics*, pages 191–198, April 1999.

[Oliveira00] M. OLIVEIRA, G. BISHOP, AND D. MCALLISTER. Relief Texture Mapping. In *Proceedings SIGGRAPH*, pages 359–368, July 2000.

[Oren94] M. OREN AND S. NAYAR. Generalization of Lambert's Reflectance Model. In *Proceedings SIGGRAPH*, pages 239–246, July 1994.

[Osram GmbHa] OSRAM GMBH. Halogen-Metalldampflampen Foto Optik.

[Osram GmbHb] OSRAM GMBH. Lighting Program '98/99 Photo Optics.

[Philip94] A. G. Davis Philip, Kenneth A. Janes, and Arthur R. Upgren, editors. *New Developments in Array Technology and Applications. Proceedings of the 167th Symposium of the International Astronomical Union*, 1994.

[Phong75] B.-T. PHONG. Illumination for Computer Generated Pictures. *Communications of the ACM*, 18(6):311–317, June 1975.

[Pighin98] FRÉDÉRIC PIGHIN, JAMIE HECKER, DANI LISCHINSKI, RICHARD SZELISKI, AND DAVID H. SALESIN. Synthesizing Realistic Facial Expressions From Photographs. In *Proceedings of SIGGRAPH 98*, pages 75–84, July 1998.

[Piponi00] DAN PIPONI AND GEORGE D. BORSHUKOV. Seamless Texture Mapping of Subdivision Surfaces by Model Pelting and Texture Blending. In *Proceedings of ACM SIGGRAPH 2000*, Computer Graphics Proceedings, Annual Conference Series, pages 471–478, July 2000.

[Pito99] R. PITO. A Solution to the Next Best View Problem for Automated Surface Acquisition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(10):1016–1030, 1999.

[Poulin90] P. POULIN AND A. FOURNIER. A Model for Anisotropic Reflection. In *Proceedings SIGGRAPH*, pages 273–282, July 1990.

[Press94] WILLIAM H. PRESS, SAUL A. TEUKOLSKY, WILLIAM T. VETTERLING, AND BRIAN P. FLANNERY. *Numerical recipes in C: the art of scientific computing*. Cambridge Univ. Press, 2nd ed. edition, 1994.

[Pulli97] KARI PULLI, MICHAEL COHEN, TOM DUCHAMP, HUGUES HOPPE, LINDA SHAPIRO, AND WERNER STUETZLE. View-based Rendering: Visualizing Real Objects from Scanned Range and Color Data. In *Eurographics Rendering Workshop 1997*, pages 23–34. Springer Wien, June 1997.

[Pulli99] K. PULLI. Multiview registration for large data sets. In *Proceedings of the 2nd International Conferrence on 3D Digital Imaging and Modeling*, pages 160–168, October 1999.

[Rademacher98] PAUL RADEMACHER AND GARY BISHOP. Multiple-Center-of-Projection Images. In *Proceedings of SIGGRAPH 98*, Computer Graphics Proceedings, Annual Conference Series, pages 199–206, July 1998.

[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.

[Ramamoorthi02] R. RAMAMOORTHI AND P. HANRAHAN. Frequency Space environment Map Rendering. In *Proceedings SIGGRAPH*, pages 517–526, July 2002.

[Reed99] M.K. REED AND P.K. ALLEN. 3-D Modeling from Range Imagery: An Incremental Method with a Planning Component. *Image & Vision Computing*, 17:99–111, 1999.

[Reed00] M.K. REED AND P.K. ALLEN. Constraint-Based Sensor Planning for Scene Modeling. *IEEE Trans. on Pattern Recognition and Machine Intelligence*, 22(12):1460–1467, 2000.

[Robertson99] M. A. ROBERTSON, S., AND R. L. STEVENSON. Dynamic Range Improvement Through Multiple Exposures. In *Proc. of the Int. Conf. on Image Processing (ICIP'99)*, pages 159–163. IEEE, October 1999.

[Rocchini99] C. ROCCHINI, P. CIGNONI, AND C. MONTANI. Multiple Textures Stitching and Blending on 3D Objects. In *Eurographics Rendering Workshop 1999*. Eurographics, June 1999.

[Rushmeier97] H. RUSHMEIER, G. TAUBIN, AND A. GUÉZIEC. Applying Shape from Lighting Variation to Bump Map Capture. In *8th Eurographics Workshop on Rendering Workshop*, pages 35–44, June 1997.

[Rushmeier98] HOLLY RUSHMEIER, FAUSTO BERNARDINI, JOSHUA MITTLEMAN, AND GABRIEL TAUBIN. Acquiring Input for Rendering at Appropriate Levels of Detail: Digitizing a Pietà. *Eurographics Rendering Workshop 1998*, pages 81–92, June 1998. ISBN 3-211-83213-0. Held in Vienna, Austria.

[Rusinkiewicz00] S. RUSINKIEWICZ AND S. MARSCHNER. Measurement I - BRDFs. Script of Course CS448C: Topics in Computer Graphics, held at Stanford University, October 2000.

[Rusinkiewicz02] S. RUSINKIEWICZ, O. HALL-HOLT, AND M. LEVOY. Real-time 3D model acquisition. In *Proceedings SIGGRAPH*, pages 438–446, 2002.

[Sander01] PEDRO V. SANDER, JOHN SNYDER, STEVEN J. GORTLER, AND HUGUES HOPPE. Texture Mapping Progressive Meshes. In *Proceedings of ACM SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 409–416, August 2001.

[Sato97] YOICHI SATO, MARK D. WHEELER, AND KATSUSHI IKEUCHI. Object Shape and Reflectance Modeling from Observation. In *Proceedings of SIGGRAPH 97*, pages 379–388, August 1997.

[Sato99] I. SATO, Y. SATO, AND K. IKEUCHI. Illumination Distribution from Shadows. In *IEEE Conf. CVPR*, volume 1, pages 306–312, 1999.

[Schaufler98] G. SCHAUFLER. Per-Object Image Warping with Layered Impostors. In *9th Eurographics Rendering Workshop*, pages 145–156, June 1998.

[Schirmacher99] H. SCHIRMACHER, W. HEIDRICH, M. RUBICK, D. SCHIRON, AND H.-P. SEIDEL. Image-Based BRDF Reconstruction. In *Proc. of the 4th VMV Conference*, pages 285–292, November 1999.

[Schirmacher00] HARTMUT SCHIRMACHER, WOLFGANG HEIDRICH, AND HANS-PETER SEIDEL. High-Quality Interactive Lumigraph Rendering Through Warping. In *Graphics Interface*, pages 87–94, 2000.

[Schlick94] C. SCHLICK. An Inexpensive BRDF Mdoel for Physically Based Rendering. In *Eurographics'94*, pages 149–162, sept 1994.

[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.

[Scopigno02] ROBERTO SCOPIGNO, CARLOS ANDUJAR, MICHAEL GOESELE, AND HENDRIK P. A. LENSCH. 3D Data Acquisition. In *Tutorial Notes of the Eurographics 2002*, pages 1–71, Saarbrücken, Germany, 2002. The Eurographics Association.

[Scott01] W. SCOTT, G. ROTH, AND J.-F. RIVEST. View planning with a registration constraint. In *3DIM01*, pages 127–134, 2001.

[Segal92] MARK SEGAL, CARL KOROBKIN, ROLF VAN WIDENFELT, JIM FORAN, AND PAUL HAEBERLI. "Fast Shadow and Lighting Effects Using Texture Mapping". *Computer Graphics (SIGGRAPH '92 Proceedings)*, 26(2):249–252, July 1992.

[Shade98] J. SHADE, S. GORTLER, L. HE, AND R. SZELISKI. Layered Depth Images. In *Proceedings SIGGRAPH*, pages 231–242, July 1998.

[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.

[Shum99] HEUNG-YEUNG SHUM AND LI-WEI HE. Rendering with Concentric Mosaics. In *Proceedings of SIGGRAPH 99*, Computer Graphics Proceedings, Annual Conference Series, pages 299–306, August 1999.

[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] FRANÇOIS X. SILLION AND CLAUDE PUECH. *Radiosity and Global Illumination*. Morgan Kaufmann Publishers, San Francisco, 1994.

[Sloan02] PETER-PIKE SLOAN, JAN KAUTZ, AND JOHN SNYDER. Precomputed Radiance Transfer for Real-Time Rendering in Dynamic, Low-Frequency Lighting Environments. *ACM Transactions on Graphics*, 21(3):527–536, July 2002. ISSN 0730-0301 (Proceedings of ACM SIGGRAPH 2002).

[Smith87] A. SMITH. Planar 2-Pass Texture Mapping and Warping. In *Proceedings SIGGRAPH*, pages 263–272, July 1987.

[Sorkine02] OLGA SORKINE, DANIEL COHEN-OR, RONY GOLDENTHAL, AND DANI LISCHINSKI. Bounded-distortion Piecewise Mesh Parameterization. In *IEEE Visualization*, 2002.

[Stürzlinger99] WOLFGANG STÜRZLINGER. Imaging all Visible Surfaces. In I. Scott MacKenzie and James Stewart, editors, *Graphics Interface '99*, pages 115–122, June 1999.

[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.

[Tanaka01] N. TANAKA AND S. TOMINAGA. Measurement of Surface Reflection Properties. In *Proceedings of the Ninth Color Imaging Conference: Color Science and Engineering: Systems, Technologies, Applications*, pages 52–57. IS&T, 2001.

[Tarabanis95] K. TARABANIS, R. TSAI, AND P.K. ALLEN. The MVP sensor planning system for robotic tasks. *IEEE Transactions on Robotics and Automation*, 11(1):72–85, 1995.

[Tarbox95] G. H. TARBOX AND S. N. GOTTSCHLICH. Planning for Complete Sensor Coverage. *Computer Vision and Image Understanding*, 61(1):84–111, 1995.

[Tarini03] MARCO TARINI, HENDRIK P.A. LENSCH, MICHAEL GOESELE, AND HANS-PETER SEIDEL. 3D Acquisition of Mirroring Objects. Research report, Max-Planck-Institut für Informatik, Munich, Germany, 2003.

[Teo97] P. TEO, E. SIMONCELLI, AND D. HEEGER. Efficient Linear Re-rendering for Interactive Lighting Design. Technical Report STAN-CS-TN-97-60, Stanford University, 1997.

[Tominaga01] S. TOMINAGA, T. MATSUMOTO, AND N. TANAKA. 3D Recording and Rendering for Art Paintings. In *Proceedings of the Ninth Color Imaging Conference: Color Science and Engineering: Systems, Technologies, Applications*, pages 337–341. IS&T, 2001.

[Tominaga02] S. TOMINAGA AND TANAKA N. Measuring and Rendering Art Paintings Using an RGB Camera. In *Eurographics 2002 Short Presentations*, pages 299–306, 2002.

[Torrance67] K. TORRANCE AND E. SPARROW. Theory for Off-Specular Reflection from Roughened Surfaces. *Journal of Optical Society of America*, 57(9):1105 1114, 1967.

[Triggs95] B. TRIGGS AND C. LAUGIER. Automatic Camera Placement for Robot Vision Tasks. In *icra*, 1995.

[Tsai87a] R. TSAI. A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4), Aug 1987.

[Tsai87b] R.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.

[Tumblin93] JACK TUMBLIN AND HOLLY RUSHMEIER. Tone Reproduction for Realistic Images. *IEEE Computer Graphics and Applications*, 13(6):42–48, 1993.

[Turk94] G. TURK AND M. LEVOY. Zippered Polygon Meshes from Range Images. In *Proceedings SIGGRAPH*, pages 311–318, July 1994.

[v. Helmholtz25] H. V. HELMHOLTZ. *Treatise on Physiological Optics*. Dover, New York, 1925.

[Vázquez02] P.P. VÁZQUEZ, M. FEIXAS, M. SBERT, AND W. HEIDRICH. Image-Based Modeling Using Viewpoint Entropy. In *Computer Graphics International*, 2002.

[Wallner] DAWN WALLNER. Building ICC Profiles – the Mechanics and Engineering. available at http://www.color.org/iccprofiles.html.

[Walter97] B. WALTER, G. ALPPAY, E. LAFORTUNE, S. FERNANDEZ, AND D. GREENBERG. Fitting Virtual Lights for Non-Diffuse Walkthroughs. In *Proc. SIGGRAPH*, pages 45–48, August 1997.

[Ward Larson92] G. WARD LARSON. Measuring and Modeling Anisotropic Reflection. In *Proc. SIGGRAPH*, pages 265–272, July 1992.

[Ward94] GREG WARD. A Contrast-Based Scalefactor for Luminance Display. In P.S. Heckbert, editor, *Graphics Gems IV*, pages 415–421. Academic Press, 1994.

[Westin92] S. WESTIN, J. ARVO, AND K. TORRANCE. Predicting Reflectance Functions From Complex Surfaces. In *Proc. SIGGRAPH*, pages 255–264, July 1992.

[Wexler02] Y. WEXLER, A. FITZGIBBON, AND A. ZISSERMAN. Image-based Environment Matting. In S. Debevec, P. Gibson, editor, *Proc. EuroGraphics Workshop on Rendering*, pages 289–299, Pisa, Italy, June 26–28 2002.

[Whaite97] P. WHAITE AND F.P. FERRIE. Autonomous Exploration: Driven by Uncertainty. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 193–205, 1997.

[Wheeler98] M. WHEELER, Y. SATO, AND K. IKEUCHI. Consensus surfaces for modeling 3D objects from multiple range images. In *International Conference on Computer Vision*, pages 917–924, 1998.

[Wong97] T.-T. WONG, P.-A. HENG, S.-H. OR, AND W.-Y. NG. Image-based Rendering with Controllable Illumination. In

Eighth Eurographics Workshop on Rendering, pages 13–22, June 1997.

[Wood00] D. WOOD, D. AZUMA, K. ALDINGER, B. CURLESS, T. DUCHAMP, D. SALESIN, AND W. STUETZLE. Surface Light Fields for 3D Photography. In *Proc. SIGGRAPH*, pages 287–296, July 2000.

[Woodham81] R. WOODHAM. Analysing Smages of Curved Surfaces. *Artificial Intelligence*, 17:117–140, 1981.

[Yu98] YIZHOU YU AND JITENDRA MALIK. Recovering Photometric Properties of Architectural Scenes from Photographs. In *Proceedings of SIGGRAPH 98*, 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.

[Zhang94] Z. ZHANG. Iterative Point Matching for Registration of Free-form Curves and Surfaces. *International Journal of Computer Vision*, 13(2):119–142, 1994.

[Zhang99a] RUO ZHANG, PING-SING TSAI, JAMES CRYER, AND MUBARAK SHAH. Shape from Shading: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):690–706, August 1999.

[Zhang99b] Z. ZHANG. Flexible Camera Calibration By Viewing a Plane From Unknown Orientations. In *Int. Conf. on Computer Vision*, pages 666–673, September 1999.

[Zhang99c] ZHENGYOU ZHANG. A Flexible New Technique for Camera Calibration. Technical Report MSR-TR-98-71, Microsoft Research, 1999. Updated version of March 25, 1999.

[Zongker99] DOUGLAS E. ZONGKER, DAWN M. WERNER, BRIAN CURLESS, AND DAVID H. SALESIN. Environment Matting and Compositing. In *Proceedings of SIGGRAPH 99*, Computer Graphics Proceedings, Annual Conference Series, pages 205–214, August 1999.